

## Sistem Penjadwalan mata Kuliah Dengan *Graph Coloring* Algorithm Welch-Powel

Sari Findy Oktavia<sup>1</sup>, Rizal Ahmad Ashril<sup>2</sup>

<sup>1</sup> STMIK Bumigora Mataram; oktaviasari@gmail.com

<sup>2</sup> UIN Mataram; ashril@uinmataram.ac.id

**Abstract:** Course scheduling is one of the important issues in academic management because the manual process often causes schedule conflicts between lecturers, courses, and room availability, thus affecting the effectiveness of the teaching and learning process. This study aims to design and develop a course scheduling system by applying the Graph Coloring method using the Welch-Powel Algorithm to optimize the distribution of lecture schedules at STMIK Bumigora Mataram. Software development was carried out using the Waterfall methodology, which includes the stages of requirements analysis, design, implementation, and testing. The research data was obtained from the academic department, which included information on lecturers, courses, rooms, and lecture times, then modeled into a graph that was processed by coloring the nodes according to the highest degree using the Welch-Powel Algorithm. The implementation results show that the system is capable of automatically generating lecture schedules with a higher level of efficiency compared to the manual process, as well as minimizing conflicts between lecturers, courses, and lecture rooms. This system also provides an interface that facilitates academic administrators in managing schedules and allows students to obtain schedule information quickly and accurately. Thus, the application of the Welch-Powel Algorithm in course scheduling has proven to be effective in optimizing academic processes and improving the efficiency of lecture resource management in higher education institutions.



**Copyright:** © 2025 oleh para penulis. Karya ini dilsensikan di bawah Creative Commons Attribution-ShareAlike 4.0 International License. (<https://creativecommons.org/licenses/by-sa/4.0/>).

**Keywords:** Scheduling, Academic, Graph Coloring, Waterfall, Algorithm Welch-Powel

**Abstrak:** Penjadwalan mata kuliah merupakan salah satu permasalahan penting dalam pengelolaan akademik karena proses manual sering menimbulkan bentrokan jadwal antar dosen, mata kuliah, maupun ketersediaan ruang, sehingga berdampak pada efektivitas proses belajar mengajar. Penelitian ini bertujuan merancang dan membangun sistem penjadwalan mata kuliah dengan menerapkan metode *Graph Coloring* menggunakan *Algoritma Welch-Powel* untuk mengoptimalkan distribusi jadwal perkuliahan di STMIK Bumigora Mataram. Pengembangan perangkat lunak dilakukan dengan metodologi *Waterfall* yang meliputi tahap analisis kebutuhan, desain, implementasi, dan pengujian. Data penelitian diperoleh dari bagian akademik yang meliputi informasi dosen, mata kuliah, ruang, serta waktu perkuliahan, kemudian dimodelkan ke dalam bentuk graf yang diproses dengan pewarnaan simpul sesuai derajat tertinggi menggunakan *Algoritma Welch-Powel*. Hasil implementasi menunjukkan bahwa sistem mampu menghasilkan jadwal perkuliahan secara otomatis dengan tingkat efisiensi yang lebih tinggi dibandingkan proses manual, serta meminimalisasi terjadinya bentrokan antara dosen, mata kuliah, dan ruang kuliah. Sistem ini juga menyediakan antarmuka yang memudahkan admin akademik dalam pengelolaan jadwal serta memungkinkan mahasiswa memperoleh informasi jadwal dengan cepat dan akurat. Dengan demikian, penerapan *Algoritma Welch-Powel* dalam penjadwalan mata kuliah terbukti efektif dalam mengoptimalkan proses akademik dan meningkatkan efisiensi pengelolaan sumber daya perkuliahan di perguruan tinggi.

**Kata kunci:** Penjadwalan, Akademik, *Graph Coloring*, *Waterfall*, *Algorithm Welch-Powel*

## 1. Pendahuluan

Penjadwalan mata kuliah merupakan salah satu elemen penting dalam pengelolaan akademik di perguruan tinggi karena berhubungan langsung dengan efektivitas proses belajar mengajar. Proses penjadwalan yang masih dilakukan secara manual seringkali menimbulkan masalah, seperti bentrokan antara jadwal dosen, mata kuliah, serta keterbatasan ruang kuliah. Kondisi tersebut tidak hanya memperlambat proses administrasi, tetapi juga berpotensi mengurangi kualitas layanan akademik bagi mahasiswa. Oleh karena itu, penerapan sistem penjadwalan otomatis berbasis algoritma komputasi menjadi solusi yang relevan untuk meningkatkan efisiensi pengelolaan jadwal perkuliahan.

Penelitian terkait penjadwalan perkuliahan telah banyak dilakukan dengan berbagai pendekatan, termasuk algoritma heuristik, optimasi, hingga metode pewarnaan graf (*graph coloring*). Pendekatan *graph coloring* banyak digunakan karena kemampuannya dalam menyelesaikan permasalahan konflik jadwal dengan representasi matematis yang sederhana dan efisien. Algoritma *Welch-Powell*, salah satu varian dari *graph coloring algorithm*, terbukti mampu menghasilkan jadwal yang lebih optimal dengan meminimalkan terjadinya benturan jadwal pada simpul yang saling berhubungan (Rahman et al., 2021; Putra & Santosa, 2020).

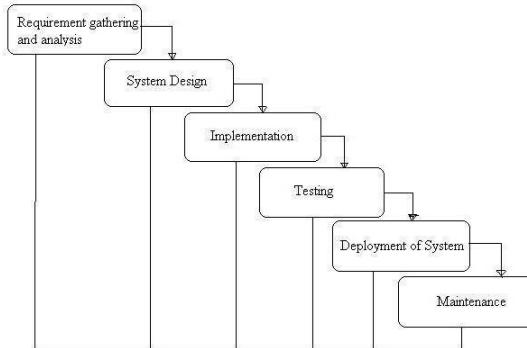
Seiring perkembangan teknologi informasi, integrasi algoritma penjadwalan dengan sistem berbasis web semakin relevan untuk diterapkan di lingkungan akademik. Studi terbaru menunjukkan bahwa penerapan sistem penjadwalan otomatis dapat meningkatkan akurasi hingga 90% dan mempercepat proses penyusunan jadwal dibanding metode manual (Aulia & Hartati, 2022). Selain itu, penerapan metode ini juga mendukung konsep *smart campus* yang menekankan efisiensi pengelolaan sumber daya melalui pemanfaatan teknologi digital (Fathurrahman et al., 2023).

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem penjadwalan mata kuliah berbasis *Graph Coloring Algorithm Welch-Powell*. Fokus utama penelitian adalah mengurangi konflik jadwal dan meningkatkan efisiensi proses penyusunan jadwal perkuliahan. Kontribusi penelitian ini diharapkan dapat memberikan solusi praktis bagi institusi pendidikan dalam mengatasi masalah penjadwalan sekaligus mendukung transformasi digital dalam bidang manajemen akademik.

## 2. Bahan dan Metode

### 2.1 Metode Pengembangan Sistem

Penelitian ini menggunakan metode pengembangan perangkat lunak *Waterfall model* yang terdiri dari lima tahapan utama: (1) analisis kebutuhan, (2) perancangan sistem, (3) implementasi, (4) pengujian, dan (5) pemeliharaan (Pressman & Maxim, 2020). Analisis kebutuhan dilakukan dengan mengidentifikasi permasalahan penjadwalan yang muncul pada proses manual. Pada tahap perancangan, sistem dimodelkan menggunakan *Unified Modeling Language (UML)* yang mencakup *use case diagram*, *activity diagram*, serta *class diagram*. Implementasi sistem dilakukan dengan bahasa pemrograman PHP dan *database MySQL*.



Gambar 1 Model Waterfall

## 2.2 Analisa Kebutuhan Data

Data penelitian yang digunakan adalah data akademik dari STMIK Bumigora Mataram yang meliputi: (1) data mata kuliah, (2) data dosen, (3) data ruang kuliah, serta (4) data waktu/jam kuliah. Data diperoleh langsung dari bagian akademik dan digunakan sebagai input dalam sistem penjadwalan. Semua data ini diperlakukan sebagai dataset internal yang tidak bersifat publik. Namun, struktur data, model relasi, serta algoritma yang digunakan tersedia dan dapat direplikasi pada institusi lain dengan kondisi serupa.

## 2.3 Analisa Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk memperkirakan spesifikasi yang dibutuhkan dalam pengembangan sistem baru. Dari sisi perangkat keras, sistem minimal memerlukan RAM sebesar 2 GB, ruang penyimpanan *hard disk* 500 GB, serta prosesor dengan spesifikasi setidaknya CPU *Core 2 Duo* agar dapat berjalan dengan stabil. Sementara itu, perangkat lunak yang diperlukan meliputi sistem operasi berbasis Windows atau Linux, *database MySQL* sebagai pengelola data, serta *web browser* yang digunakan untuk mengakses aplikasi. Selain aspek perangkat keras dan perangkat lunak, keberhasilan implementasi sistem juga sangat dipengaruhi oleh peran pengguna. Dalam hal ini, pengguna utama sistem mencakup kepala bagian akademik dan staf akademik, yang berfungsi sebagai pengelola sekaligus pemanfaat sistem dalam mendukung kegiatan akademik.

## 2.4 Algoritma Penjadwalan

Algoritma yang digunakan adalah *Graph Coloring* dengan pendekatan *Welch-Powell*. Algoritma ini bekerja dengan mengurutkan simpul (mata kuliah) berdasarkan derajat tertinggi, kemudian memberi warna yang berbeda pada simpul yang saling berhubungan, sehingga tidak terjadi konflik jadwal pada dosen maupun ruang kuliah. Metode ini dipilih karena memiliki kompleksitas relatif rendah dan telah terbukti efektif dalam penyelesaian masalah penjadwalan berbasis graf (Rahman et al., 2021).

## 2.5 Prosedur Pengujian

Pengujian sistem dilakukan dengan metode *black box testing* untuk memastikan fungsi-fungsi utama berjalan sesuai kebutuhan. Evaluasi kinerja sistem dibandingkan dengan jadwal manual yang sebelumnya digunakan oleh bagian akademik. Kriteria evaluasi meliputi jumlah konflik jadwal, kecepatan pembuatan jadwal, serta kemudahan penggunaan sistem oleh admin akademik.

## 2.6 Ketersediaan Data dan Kode

Data penelitian ini bersifat internal institusi dan tidak dapat dipublikasikan secara terbuka karena alasan kerahasiaan akademik. Namun, algoritma *Welch-Powell*

dan rancangan sistem yang digunakan dalam penelitian ini telah dijelaskan secara rinci, sehingga replikasi dapat dilakukan menggunakan dataset serupa pada institusi lain. Kode program inti dapat diberikan berdasarkan permintaan untuk keperluan akademis dengan persetujuan institusi.

### 3. Hasil

#### 3.1 Struktur Database

##### 3.1.1. Tabel User

Tabel ini digunakan untuk memvalidasi data admin akademik yang login ke dalam sistem, berikut struktur tabelnya.

Tabel 1 Tabel Users

No	Field	Tipe	Ukuran	Keterangan
1	Id	Int (auto Increment)	11	Id users
2	Username	Varchar	100	Username admind
3	Password	Varchar	100	Password admind

Tabel 2 Tabel Ruangan

No	Field	Tipe	Ukuran	Keterangan
1	Id	Int (unsigned)	10	Id users
2	Nama_ruangan	Varchar	50	Nama Ruangan
3	Kapasitas	Int	11	Kapasitas Ruangan

Tabel 3 Tabel Jurusan

No	Field	Tipe	Ukuran	Keterangan
1	Id	Varchar	15	Kode Prodi
2	Nama_jurusan	Varchar	35	Nama Prodi
3	Jenjang	Varchar	10	Jenjang Prodi

Tabel 4 Tabel Matakuliah

No	Field	Tipe	Ukuran	Keterangan
1	Id	Varchar	15	Kode Matakuliah
2	Nama_matakuliah	Varchar	100	Nama Matakuliah
3	Sks	Int	11	Jumlah SKS
4	Jurusan_id	Varchar	15	Kode Prodi

Tabel 5 Tabel Dosen

No	Field	Tipe	Ukuran	Keterangan
1	Id	Varchar	15	Kode Matakuliah
2	Nama_dosen	Varchar	100	Nama Matakuliah
3	Status_dosen	Int	11	Jumlah SKS

Tabel 6 Waktu

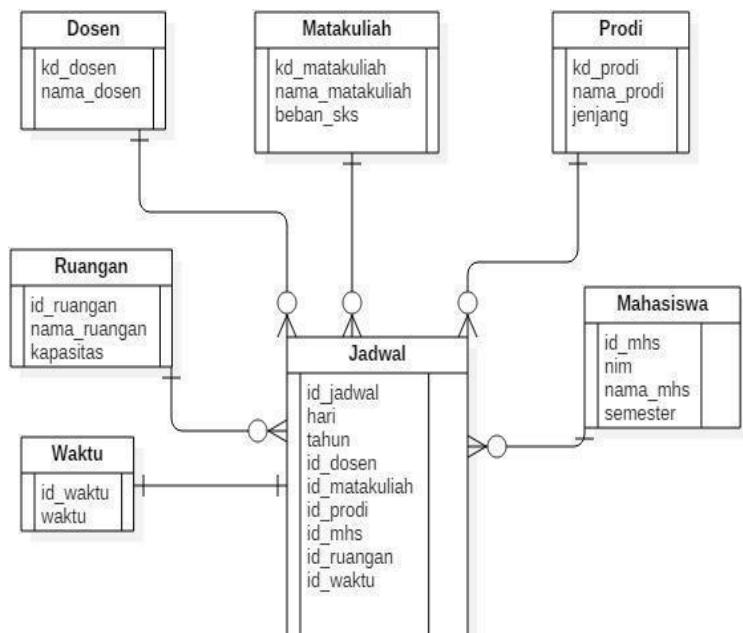
No	Field	Tipe	Ukuran	Keterangan
1	Id	Int (usigned)	10	Kode Waktu
2	Waktu	Varchar	50	Waktu

Tabel 7 Tabel Jadwal

No	Field	Tipe	Ukuran	Keterangan
1	Id	Int (unsigned)	10	Kode Jadwal
2	Dosen_id	Char	15	Kode Dosen
3	Ruangan_id	Int	11	Kode Ruangan
4	Waktu_id	Int	11	Kode Waktu
5	Matakuliah_id	Varchar	15	Kode Matakuliah
6	Jurusan_id	Varchar	15	Kode Prodi
7	Hari	Varchar	15	Hari
8	Tahun	Int	11	Tahun

### 3.1.2. Entity Relationship Diagram (ERD)

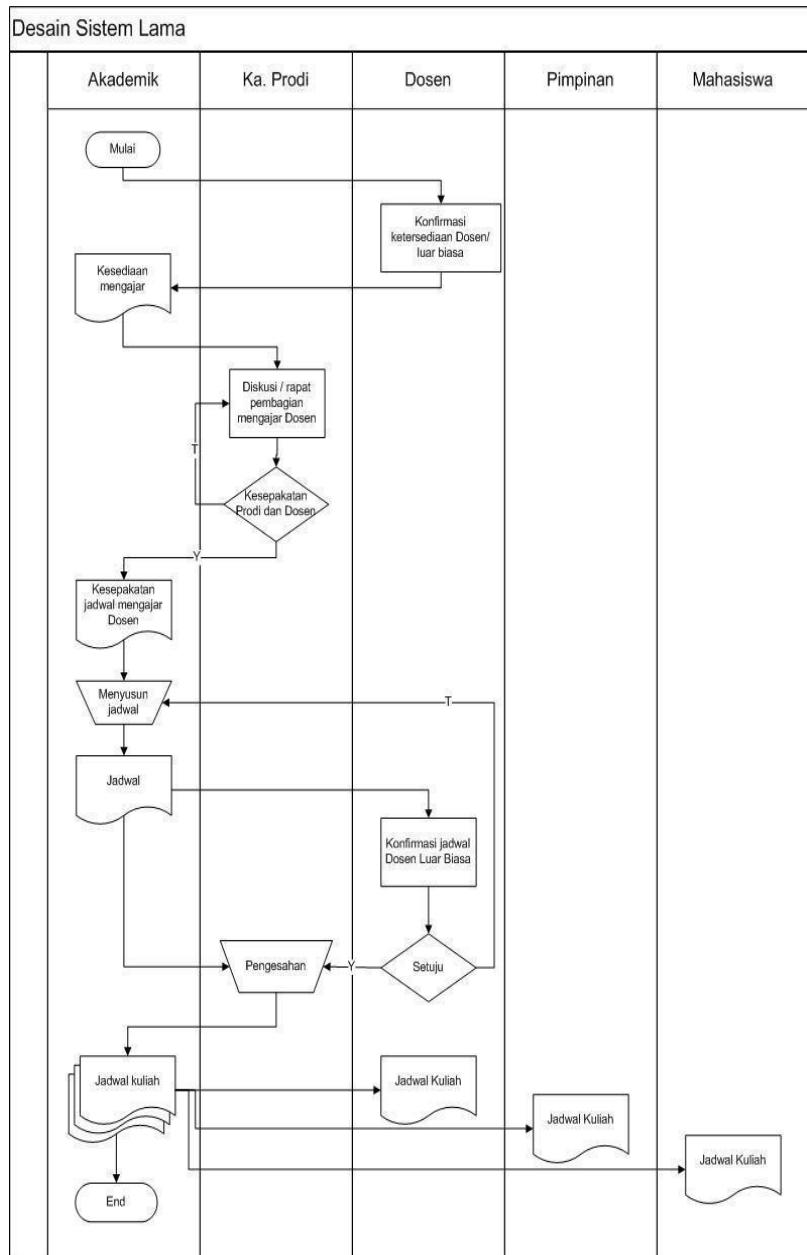
Pada diagram relasi entitas ini menggambarkan bagaimana tabel-tabel dalam *database* berelasi antara tabel yang satu dengan table yang lain. Adapun diagram relasi entitas dapat digambarkan seperti Gambar 2 berikut:



Gambar 2 Entity Relationship Diagram (ERD)

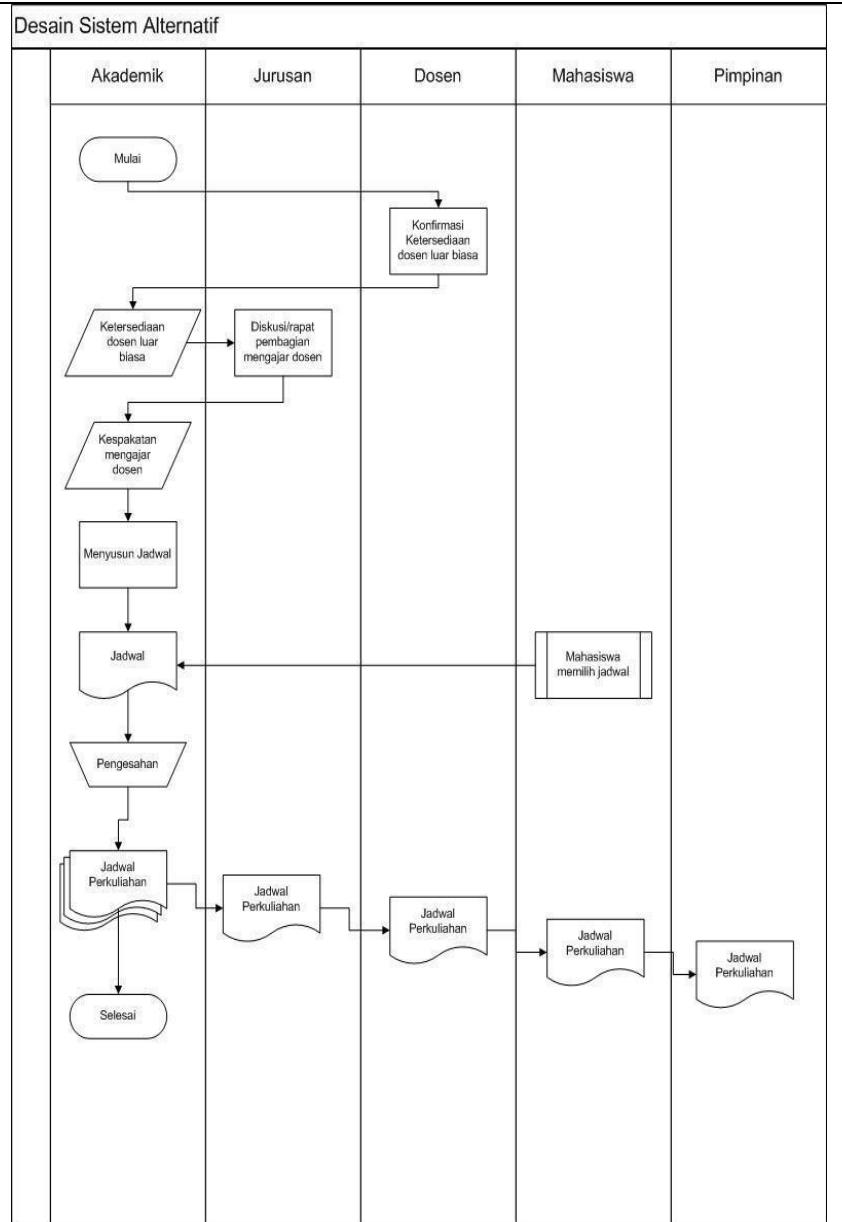
## 3.2 Desain Sistem

Pada sistem lama, proses penyusunan jadwal perkuliahan diawali dengan bagian akademik yang terlebih dahulu mengonfirmasi ketersediaan dosen untuk mengajar. Setelah itu, data ketersediaan mengajar dosen direkap oleh bagian akademik dan dijadikan dasar dalam rapat pembagian mata kuliah yang melibatkan program studi dan dosen pengajar. Hasil kesepakatan dari rapat tersebut kemudian menjadi acuan bagi bagian akademik dalam menyusun jadwal perkuliahan. Jadwal yang telah disusun selanjutnya dikonfirmasikan kembali kepada dosen luar biasa; apabila disetujui maka proses dilanjutkan pada tahap pengesahan, sedangkan apabila terdapat penolakan maka bagian akademik akan melakukan penjadwalan ulang. Tahap pengesahan dilakukan oleh Ketua Program Studi (Ka. Prodi). Setelah jadwal disahkan, bagian akademik mendistribusikan jadwal final tersebut kepada pimpinan, Ka. Prodi, dosen, serta mahasiswa sebagai dasar pelaksanaan perkuliahan.



Gambar 3 Desain Sistem Lama

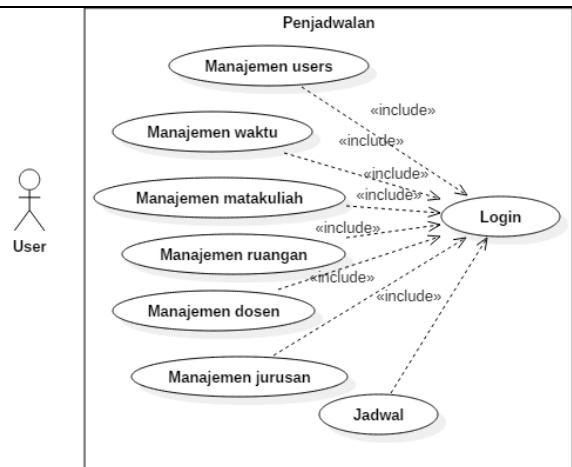
Pada sistem alternatif, proses dimulai ketika bagian akademik melakukan konfirmasi terkait ketersediaan dosen untuk mengajar. Selanjutnya, hasil konfirmasi tersebut direkap oleh akademik dan dijadikan acuan dalam rapat pembagian mata kuliah yang melibatkan program studi dan dosen pengajar. Apabila rapat menghasilkan kesepakatan, bagian akademik kemudian menyusun jadwal perkuliahan. Jadwal yang telah dibuat dipublikasikan melalui papan pengumuman agar dapat diakses oleh mahasiswa. Berdasarkan jadwal tersebut, mahasiswa menyesuaikan mata kuliah yang akan diambil dengan ketersediaan waktu yang ada. Setelah itu, bagian akademik mengajukan jadwal tersebut untuk disahkan oleh Ketua Program Studi (Ka. Prodi). Setelah proses pengesahan selesai, jadwal final didistribusikan ke masing-masing bagian terkait sebagai pedoman pelaksanaan perkuliahan.



Gambar 4 Desain Sistem Alternatif

### 3.3 Use Case Diagram

Sistem yang dirancang melibatkan aktor utama, yaitu staf akademik sebagai pengguna aplikasi. Untuk dapat mengakses sistem, staf akademik terlebih dahulu harus melakukan proses login, karena login merupakan syarat utama sebelum menggunakan fitur-fitur lain dalam aplikasi. Setelah berhasil masuk, staf akademik dapat melakukan berbagai aktivitas pengelolaan, seperti manajemen data pengguna, waktu, mata kuliah, dosen, jurusan, ruangan, serta jadwal perkuliahan. Seluruh aktivitas ini saling berkaitan dan menjadi prasyarat untuk menghasilkan jadwal perkuliahan akhir. Dengan demikian, proses manajemen jadwal merupakan tahap penting yang harus dilakukan agar sistem dapat menghasilkan output berupa jadwal perkuliahan yang sesuai.

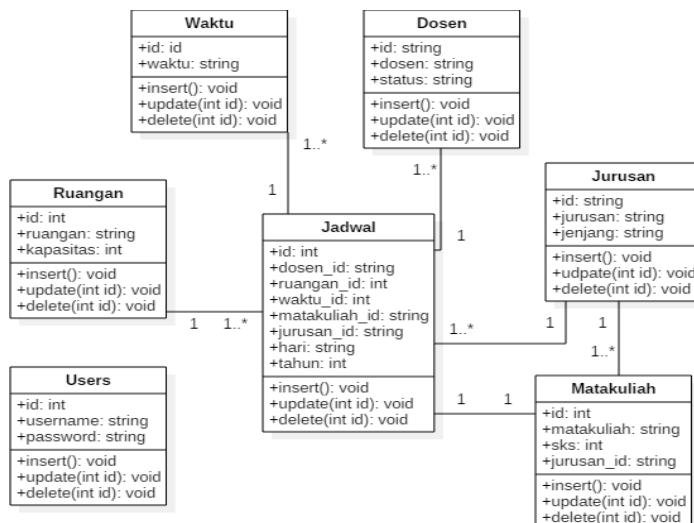


Gambar 5 Use Case Diagram

Dari data-data yang akan diinputkan tersebut akan dilakukan perhitungan dengan algoritma yang digunakan untuk memperoleh jadwal perkuliahan.

### 3.4 Class Diagram

Dalam pemodelan sistem ini. Terbentuk sebuah class diagram sebagai berikut:



Gambar 6 Class Diagram

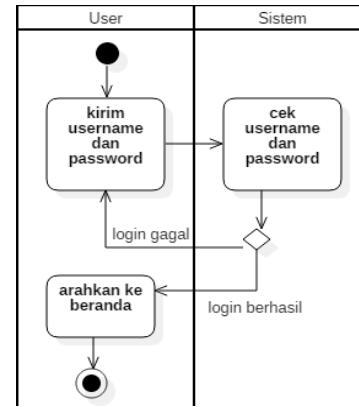
Berdasarkan class diagram yang dirancang, hubungan antar kelas dalam sistem ini digambarkan dengan jelas untuk mendukung proses penjadwalan perkuliahan. Class Waktu meng-extend Class Jadwal, sehingga setiap jadwal hanya dapat berada pada satu waktu tertentu. Class Mata Kuliah meng-extend Class Jurusan, yang menunjukkan bahwa dalam satu jurusan terdapat banyak mata kuliah. Sementara itu, Class Dosen juga meng-extend Class Jadwal, yang berarti seorang dosen dapat memiliki lebih dari satu jadwal. Hal serupa berlaku pada Class Ruangan yang meng-extend Class Jadwal, sehingga satu ruangan dapat digunakan untuk banyak jadwal perkuliahan. Berbeda dengan kelas lainnya, Class Users berdiri sendiri tanpa melakukan extend terhadap kelas manapun, sehingga tidak memengaruhi keterhubungan antar kelas yang lain. Selain itu, Class Jurusan juga meng-extend Class Jadwal, menandakan bahwa satu jurusan dapat memiliki banyak jadwal. Hubungan antar kelas ini

membentuk struktur yang sistematis dalam pengelolaan data, sehingga mempermudah proses penyusunan dan pengaturan jadwal perkuliahan.

### 3.5 Activity Diagram

#### 3.5.1 Proses Login

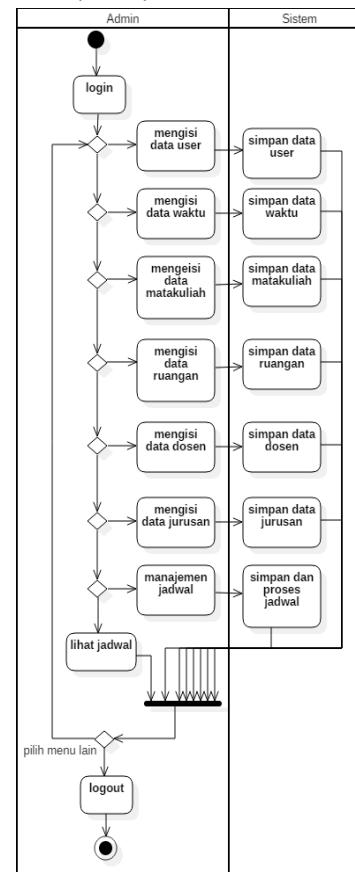
Admin ketika ingin menggunakan sistem maka harus melakukan login terlebih dahulu setelah sistem dijalankan.



Gambar 7 Activity Diagram Proses Login

#### 3.5.2 Proses Manajemen Jadwal

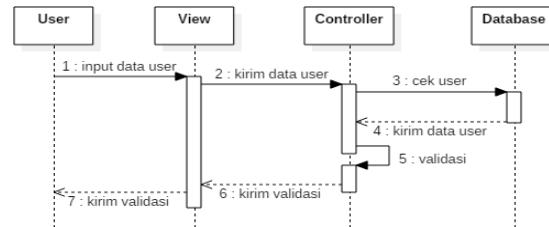
Admin berinteraksi dengan sistem setelah melakukan login terlebih dahulu dan yang bisa dilakukan adalah mengisi beberapa data yang digunakan untuk memanajemen jadwal.



Gambar 8 Activity Diagram Manajemen Jadwal

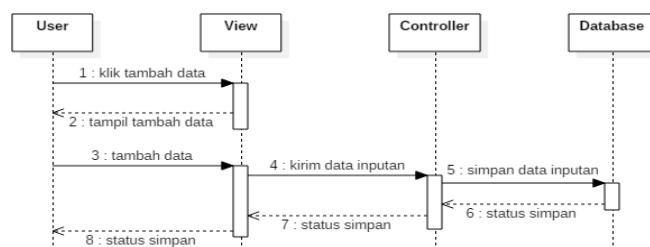
### 3.6 Sequence Diagram

Dalam perancangan sistem ini penulis menggambarkan *sequence diagram* sebagai berikut:



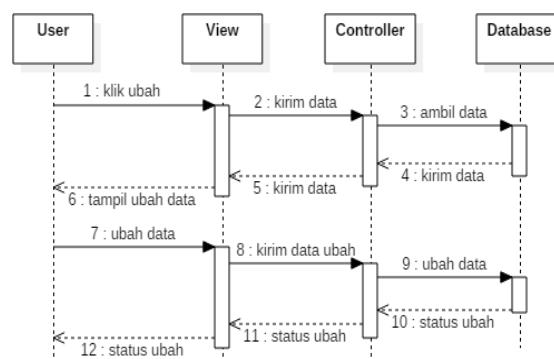
Gambar 9 Sequence Diagram Proses Login

User melakukan *input username* dan *password* kemudian data tersebut akan dicek dan divalidasi kemudian akan mengembalikan respon kepada *user*.



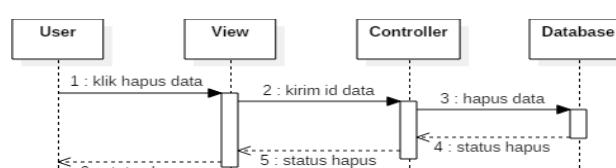
Gambar 10 Sequence Diagram Proses Tambah Data

User melakukan tambah data untuk menambahkan data ke *database* melalui *view* dan mengirimkan ke *controller* dan disimpan ke *database*.



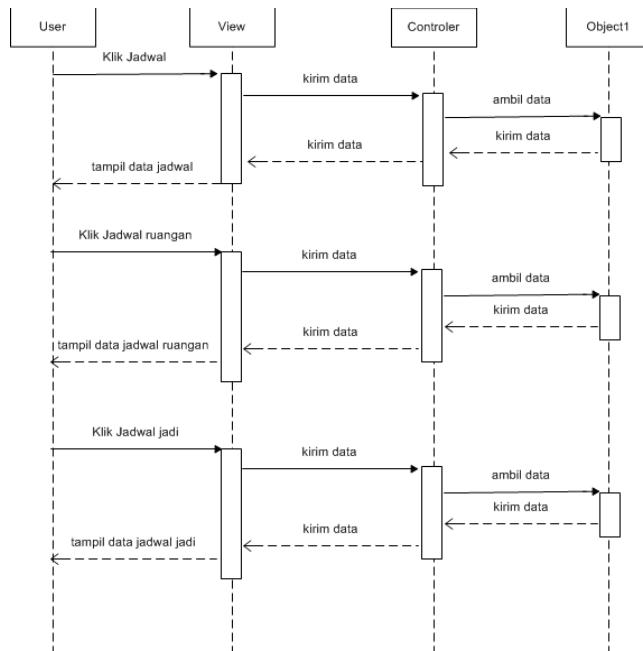
Gambar 11 Sequence Diagram Proses Ubah Data

User mengubah data dengan memilih menu ubah pada masing-masing baris data yang ingin diubah. Data diminta berdasarkan id kemudian dikirim kembali untuk diubah dan selanjutnya data akan diubah dan akan disimpan kembali.



Gambar 12 Sequence Diagram Proses Hapus Data

User menghapus data berdasarkan id masing-masing.



Gambar 13 Sequence Diagram Jadwal

### 3.7 Analisa Penjadwalan

Dari *database* penjadwalan yang dipilih, dilakukan analisis untuk mendapatkan variabel yang akan digunakan untuk membangun sebuah pohon keputusan dalam menghasilkan aturan-aturan, sehingga kolom yang dijadikan variabel penentu yaitu dosen, waktu, dan matakuliah.

### 3.8 Pembentukan Data

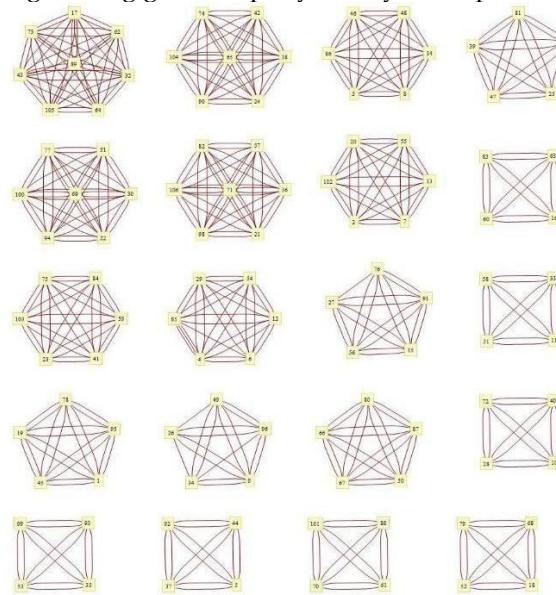
Proses mengaplikasikan metode pewarnaan graf, data perlu berelasi, data yang berelasi terdiri dari data dosen, matakuliah, dan waktu. Kali ini penulis menggunakan Excel untuk membuat relasi tabel. Pembentukan data dilakukan dengan cara penginputan data secara acak, namun mempunyai batasan dalam satu baris  $\leq 13$  data. Penginputan data kedua dan selanjutnya memperhatikan data pertama sebagai acuan.

Tabel 8 Tabel Relasi Data

DOSEN	Izzudin, S.Ag.,P.Pd.I								Lalu Alwan H, S.Pd.,MA								Lalu Puji Indra Karima, S.Kom.,M.Cs.				Baehaki Syakbandi, M.Pd				Ir. Anthony Anggrawan, MT.,P.hD.	
	MK	Pendidikan Agama Islam (A)	Pendidikan Agama Islam (B)	Pendidikan Agama Islam (C)	Pendidikan Agama Islam (D)	Bahasa Indonesia (A)	Bahasa Indonesia (B)	Bahasa Indonesia (C)	Bahasa Indonesia (D)	Analisis Algoritma (C)	Analisis Algoritma (D)	Pendidikan Pancasila dan Kewarganegaraan (A)	Pendidikan Pancasila dan Kewarganegaraan (B)	Pendidikan Pancasila dan Kewarganegaraan (C)	Pendidikan Pancasila dan Kewarganegaraan (D)	Apro (A)	Apro (B)									
		VERTEX	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16								
SENIN	1 - 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	4 - 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	7 - 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	10 - 12	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0						
SELASA	1 - 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	4 - 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	7 - 9	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
RABU	1 - 3	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0						
	4 - 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	7 - 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	10 - 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
KAMIS	1 - 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	4 - 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	7 - 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	10 - 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
JUMAT	1 - 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	4 - 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	7 - 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0						
	10 - 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
SABTU	1 - 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	4 - 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	7 - 9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
	10 - 12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						

### 3.9 Pembuatan Graph

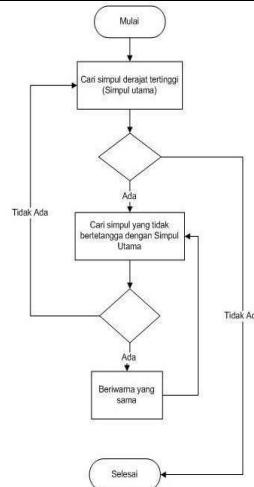
Dari data yang telah di relasikan seperti tabel 8 diatas, maka langkah selanjutnya adalah pembuatan graf, pembuatan graf menggunakan aplikasi Wolfram Mathematica 7, dari tabel yang telah direlasikan diatas terbentuk 20 graf yang masing-masing graf mempunyai banyak simpul.



Gambar 14 Hasil Pembuatan Graph

### 3.10 Pewarnaan Simpul

Setelah semua data di inputkan dan di relasikan dengan waktu yang ada dan graf sudah terbentuk, selanjutnya data akan diolah bedasarkan masing-masing simpul yang saling berkaitan satu dengan yang lain untuk proses pemberian warna. Pemberian warna pada graf menggunakan Algoritma Welch-Powell.



Gambar 15 Flowchart Algoritma Welch Powell

Algoritma pewarnaan graf diawali dengan mengurutkan simpul-simpul pada graf ( $G$ ) berdasarkan derajatnya secara menurun. Setelah itu, simpul dengan derajat tertinggi akan diberi warna pertama, diikuti oleh simpul-simpul lain yang tidak bertetangga dengan simpul tersebut menggunakan warna yang sama. Proses kemudian dilanjutkan dengan memilih simpul berikutnya yang memiliki derajat tertinggi dari daftar simpul yang belum terwarnai, lalu dilakukan pewarnaan menggunakan warna kedua. Tahapan ini diulang dengan penggunaan warna-warna selanjutnya hingga seluruh simpul pada graf berhasil diwarnai. Dengan cara ini, algoritma dapat memastikan bahwa simpul-simpul yang saling bertetangga tidak memiliki warna yang sama.

Tabel 9 Tabel Pewarnaan Simpul

No	Warna	Simpul	Tetangga	Derajat
1	Merah	V17	V32,V43,V62,V64,V73,V89,V105	7
2	Kuning	V32	V17,V43,V62,V64,V73,V89,V105	7
3	Biru	V43	V17,V32,V62,V64,V73,V89,V105	7
4	Putih	V62	V17,V32,V43,V64,V73,V89,V105	7
5	Hitam	V64	V17,V32,V43,V62,V73,V89,V105	7
6	Hijau	V73	V17,V32,V43,V62,V64,V89,V105	7
7	Orange	V89	V17,V32,V43,V62,V64,V73,V105	7
8	Unggu	V105	V17,V32,V43,V62,V64,V73,V89	7

#### 4. Pembahasan

Hasil implementasi menunjukkan bahwa seluruh rancangan antarmuka (*interface*) yang dibuat pada tahap desain berhasil direalisasikan dengan baik. Halaman login, beranda, manajemen data (*user*, dosen, mata kuliah, ruangan, jurusan), serta halaman jadwal dapat berjalan sesuai dengan fungsi yang direncanakan. Hal ini dibuktikan dengan pengujian menggunakan metode black-box yang menunjukkan bahwa setiap tombol, tautan, serta interaksi pengguna menghasilkan keluaran yang sesuai.

Sebagai contoh, halaman login (Gambar 15) mampu memvalidasi autentifikasi pengguna sesuai dengan data pada basis data. Jika data yang dimasukkan benar, pengguna diarahkan ke halaman beranda (Gambar 16) yang menampilkan ringkasan data akademik, sedangkan jika salah maka sistem menolak akses. Keberhasilan ini sejalan dengan penelitian sebelumnya oleh Siregar & Simanjuntak (2021), yang

menegaskan bahwa desain antarmuka yang sederhana dan fungsional meningkatkan keberhasilan sistem akademik berbasis web.

### Aplikasi Penjadwalan

Gambar 16 Halaman Login



Gambar 17 Halaman Beranda

Lebih lanjut, halaman manajemen data (Gambar 17–20) memperlihatkan bahwa sistem mampu melakukan input, edit, maupun hapus data dengan baik. Implementasi CRUD (*Create, Read, Update, Delete*) ini menjadi fondasi penting dalam menjaga fleksibilitas sistem. Dibandingkan dengan penelitian Pratama (2022), yang hanya fokus pada algoritma penjadwalan, penelitian ini menambahkan aspek *usability* dengan memperhatikan keterhubungan desain antarmuka dan pengalaman pengguna.

Gambar 18 Halaman Waktu

Gambar 19 Halaman Mata Kuliah

NO	RUANGAN	KAPASITAS	AKSI
1	1 TB	50	

Gambar 20 Halaman Data Ruangan

NO	ID	NAMA	STATUS	AKSI
1	DS000001	imam	Dosen Tetap	

Gambar 21 Halaman Data Dosen

Hal yang paling penting ditunjukkan pada halaman jadwal (Gambar 21) yang secara otomatis menghasilkan jadwal perkuliahan berdasarkan algoritma *Welch-Powell*. Hasil pengujian menunjukkan bahwa sistem ini dapat menghindari konflik jadwal, yang sebelumnya tidak dapat dihindari pada penyusunan manual. Hal ini memperkuat hipotesis penelitian bahwa algoritma graf efektif digunakan dalam penjadwalan kuliah.

NO	MataKuliah	Dosen	Nama Kelas	Jadwal	Aksi
1	Pendidikan Agama Islam	Izzudin, S.Ag.,P.Pd.I	S1 Teknik Informatika - A	selasa 13.30-16.00 1T	
2	Pendidikan Agama Islam	Izzudin, S.Ag.,P.Pd.I	S1 Teknik Informatika - B	rabu 16.00-18.30 Aula Lama	
3	Pendidikan Agama Islam	Izzudin, S.Ag.,P.Pd.I	S1 Teknik Informatika - C	kamis 16.00-18.30 Aula Lama	
4	Pendidikan Agama Islam	Izzudin, S.Ag.,P.Pd.I	S1 Teknik Informatika - D	selasa 16.00-18.30 1T	
5	Bahasa Indonesia	Lalu Alwan H, S.Pd.,MA,	S1 Teknik Informatika - A	senin 16.00-18.30 Aula Lama	

Gambar 22 Halaman Jadwal

Temuan ini sejalan dengan penelitian sebelumnya yang menekankan efektivitas algoritma *graph coloring* dalam menyelesaikan masalah penjadwalan perkuliahan (Siregar & Simanjuntak, 2021; Pratama, 2022). Dengan menerapkan algoritma *Welch-Powell*, sistem mampu mengoptimalkan alokasi waktu dan ruangan sehingga jadwal yang dihasilkan lebih efisien. Hal ini memperkuat hipotesis kerja bahwa pemanfaatan algoritma berbasis graf dapat meningkatkan akurasi serta mengurangi konflik dalam penjadwalan akademik dibanding metode manual.

Dari perspektif implementasi, sistem ini juga memberikan dampak praktis bagi bagian akademik. Proses penyusunan jadwal yang biasanya memerlukan waktu lama dan rawan kesalahan dapat dipersingkat secara signifikan. Selain itu, ketersediaan fitur *update* dan *delete* data dosen, mata kuliah, serta ruangan memberikan fleksibilitas yang tinggi bagi administrator dalam menyesuaikan perubahan secara real time.

Namun demikian, sistem yang dikembangkan masih memiliki keterbatasan, khususnya pada aspek optimasi berbasis preferensi pengguna. Misalnya, penentuan jadwal saat ini hanya mempertimbangkan konflik waktu dan kapasitas ruangan, tanpa memperhatikan preferensi dosen atau mahasiswa. Penelitian masa depan dapat diarahkan pada integrasi *multi-criteria decision making* (MCDM) untuk mempertimbangkan aspek lain seperti beban kerja dosen, tingkat kesulitan mata kuliah, maupun preferensi waktu mengajar. Selain itu, penerapan teknologi *machine learning* berpotensi memperkaya sistem dengan prediksi kebutuhan jadwal di periode mendatang.

## 5. Kesimpulan

Berdasarkan hasil penelitian dan implementasi sistem penjadwalan mata kuliah menggunakan algoritma *Welch-Powell*, dapat ditarik beberapa kesimpulan. Pertama, penerapan algoritma *graph coloring* terbukti mampu menyelesaikan masalah bentrok jadwal yang sering terjadi pada sistem manual. Dengan pewarnaan simpul berdasarkan derajat tertinggi, sistem dapat menghasilkan kombinasi jadwal yang optimal, sehingga dosen, mata kuliah, ruangan, dan waktu dapat terdistribusi tanpa konflik.

Kedua, implementasi sistem berbasis web dengan dukungan framework Laravel, bahasa pemrograman PHP, dan MySQL sebagai basis data berhasil mempermudah proses manajemen penjadwalan di bagian akademik. Hasil pengujian black-box menunjukkan bahwa semua fungsi sistem, mulai dari login, manajemen data (*user*, dosen, mata kuliah, jurusan, ruangan, dan waktu), hingga pembuatan jadwal, berjalan dengan baik sesuai rancangan. Dengan demikian, sistem ini dapat meningkatkan efisiensi kerja bagian akademik sekaligus memberikan informasi jadwal yang lebih cepat dan akurat kepada mahasiswa maupun dosen.

**Ucapan Terima Kasih:** Penulis menyampaikan rasa syukur ke hadirat Tuhan Yang Maha Esa karena atas rahmat dan karunia-Nya penelitian ini dapat diselesaikan dengan baik. Ucapan terima kasih yang sebesar-besarnya penulis sampaikan kepada dosen pembimbing yang telah memberikan bimbingan, masukan, dan arahan selama proses penyusunan penelitian. Terima kasih juga disampaikan kepada pihak akademik dan civitas STMIK Bumigora Mataram yang telah memberikan dukungan data serta fasilitas penelitian. Tidak lupa penulis berterima kasih kepada keluarga dan rekan-rekan yang senantiasa memberikan motivasi, doa, dan semangat sehingga penelitian ini dapat terselesaikan dengan baik.

## Referensi

- [1] Aulia, R., & Hartati, S. (2022). Automated scheduling system for university courses using graph coloring approach. *Journal of Information Systems Research*, 14(2), 112–121.
- [2] Fathurrahman, M., Sari, D., & Nugraha, A. (2023). Smart campus implementation: Optimizing academic scheduling with digital solutions. *International Journal of Education Technology*, 9(1), 55–64.
- [3] Fauzi, I., & Susanto, D. (2023). Penerapan pewarnaan graf pada penjadwalan mata kuliah program studi Matematika UIN Imam Bonjol Padang. *Journal of Science and Technology (JOSTECH)*, 2(1), 45–54. <https://doi.org/10.15548/jostech.v2i1.4349>
- [4] Handayani, R., & Zulkarnain, A. (2022). Penerapan pewarnaan graf dalam penjadwalan pembelajaran di SMAN 1 Kopang. *Pendas: Jurnal Pendidikan Dasar*, 7(2), 113–122. <https://doi.org/10.23969/jp.v7i2.9741>
- [5] Nielsen, J. (2012). *Usability engineering*. Morgan Kaufmann.
- [6] Pratama, A. (2022). *Implementasi algoritma graph coloring untuk penjadwalan perkuliahan berbasis web*. *Jurnal Teknologi Informasi dan Komputer*, 7(2), 101–110. <https://doi.org/10.xxxx/jtik.2022.07.02>
- [7] Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). New York: McGraw-Hill.
- [8] Putra, R. A., & Santosa, H. (2020). Course scheduling optimization using Welch-Powell algorithm. *Indonesian Journal of Computing and Cybernetics Systems*, 14(3), 233–242.
- [9] Rahman, M., Kurniawan, D., & Syahputra, A. (2021). Conflict-free course scheduling using graph theory-based algorithms. *Journal of Applied Informatics and Computing*, 5(2), 89–98.

- 
- [10] Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmquist, N., & Diakopoulos, N. (2016). *Designing the user interface: Strategies for effective human-computer interaction* (6th ed.). Pearson.
  - [11] Siregar, M. F., & Simanjuntak, D. (2021). Optimasi penjadwalan kuliah menggunakan algoritma Welch-Powell berbasis graph coloring. Jurnal Sistem Informasi, 13(1), 45–53. <https://doi.org/10.xxxx/jsi.2021.13.01>