

Simulasi Penerapan Algoritma Pewarnaan Graf Pada Sistem Penjadwalan Kuliah (Studi Kasus : STMIK Sumedang)

Fathoni Mahardika

Program Studi Sistem Informasi

STMIK Sumedang, Jl. Angkrek Situ No. 19, Sumedang, 45323 Indonesia

email : fathoni@stmik-sumedang.ac.id

ABSTRACT

Pembuatan Jadwal di STMIK Sumedang masih menggunakan Ms.Excel sehingga waktu pembuatan jadwal menjadi cukup lama, berkisar antara 3 minggu, dan sering menemukan jadwal yang bentrok. Dengan adanya system baru ini diharapkan kadap meminimalisir masalah tersebut. Untuk memudahkan admin (BAAK) dalam pengelolaan data dan melihat adanya jadwal bentrok system baru ini memberi tanda field berwarna sehingga penggunaan waktu dalam penyusunan jadwal dapat lebih efektif dan efisien. Metode yang digunakan dalam penelitian ini menggunakan model waterfall. Dengan Analisis dan Perancangan menggunakan UML dan menggunakan bahasa pemrograman PHP dengan MySQL Sebagai databasenya serta Algoritma Pewarnaan Graf untuk mewarnai Jadwal yang bentrok. Penelitian ini membuat Simulasi Penerapan Algoritma Pewarnaan Graf pada penjadwalan Perkuliahan STMIK Sumedang yang dikelola oleh admin (BAAK) dan dosen diberi hak akses untuk mengelola data permintaan dosen serta melihat laporan. Penerapan algoritma pewarnaan graf dapat digunakan sebagai penanda jadwal bentrok, sehingga memudahkan admin untuk mengatur atau memperbaharui jadwal bentrok, dan system menyediakan tempat untuk dosen mengisi data permintaan untuk tidak mengajar di hari dan jam tertentu.

Kata Kunci : Algoritma Pewarnaan Graf, Simulasi, Penjadwalan Perkuliahan, PHP, MySQL

1. Introduction

Nana Syaodih S. menyebutkan bahwa sesungguhnya sejak dahulu teknologi telah ada atau manusia telah memakai teknologi. Bila manusia pada zaman dahulu memecahkan kemiri dengan batu atau memetik buah dengan galah, sebenarnya mereka telah memakai teknologi, yakni teknologi simpel. Namun Iskandar Alisyahbana (1980 : 1) merumuskan lebih jelas serta lengkap perihal definisi teknologi yakni langkah lakukan suatu hal untuk mencukupi keperluan manusia dengan pertolongan alat serta akal hingga seakan-akan memperpanjang, memperkuat, atau membuat lebih ampuh bagian tubuh, panca indera, serta otak manusia. Menurut beliau, teknologi sudah dikenal manusia sejak jutaan tahun yang dikarenakan dorongan untuk hidup yang lebih nyaman, lebih makmur serta lebih sejahtera. Sejak awal peradaban sesungguhnya sudah ada teknologi, walau arti teknologi belum dipakai. Arti teknologi datang dari techne atau langkah serta logos atau pengetahuan. Pengertian teknologi sendiri yaitu langkah melakukan suatu hal untuk mencukupi keperluan manusia dengan pertolongan akal serta alat.

Menurut Richard Weiner (1996) teknologi adalah pemerosesan, pengelolaan dan penyebaran data oleh kombinasi komputer dan telekomunikasi. Dari beberapa definisi diatas, dapat disimpulkan bahwa teknologi semakin berkembang. Dari mulai teknologi memecahkan kemiri menggunakan batu sampai dengan teknologi komputer yang dapat membantu manusia untuk memproses, mengolah dan menyebarkan data menggunakan kombinasi komputer dan telekomunikasi.

Pada buku Introduction to The Design and Analysis of Algorithms yang ditulis oleh Anany Levitin pada tahun 2003 menyebutkan bahwa Algoritma adalah deretan instruksi yang jelas untuk memecahkan

masalah, yaitu untuk memperoleh keluaran yang diinginkan dari suatu masukan dalam jumlah waktu yang terbatas. Oleh karena itu penulis ingin menerapkan algoritma dalam memecahkan masalah penjadwalan agar dapat menghasilkan keluaran yang diinginkan dalam waktu yang singkat untuk meminimalisir kesalahan dari pembuatan jadwal secara manual.

Dalam pembuatan sistem penjadwalan perkuliahan pada STMIK Sumedang ada beberapa hal yang harus diperhatikan diantaranya Tidak boleh ada jadwal yang bentrok antara waktu, kelas, dosen dan ruangan; Adanya permintaan dosen tidak dapat mengajar di hari dan jam tertentu.

Untuk lebih mengoptimalkan penyelesaian masalah tersebut, ada beberapa algoritma yang masing-masing dari algoritma tersebut memiliki kekurangan dan kelebihan. Namun pada kesempatan ini penulis akan menggunakan algoritma pewarnaan graf. Dengan penerapan algoritma pewarnaan graf pada penjadwalan perkuliahan akan terbantu sistem yang optimal dan waktu proses akan lebih sedikit sehingga penyelesaian pembuatan jadwal dapat dilakukan secara cepat dan tepat dibanding dengan proses manual.

Berdasarkan latar belakang di atas maka ruang lingkup bahasan penelitian dapat diuraikan diantaranya bagaimana cara penerapan algoritma pewarnaan graf pada sistem penjadwalan perkuliahan di STMIK Sumedang untuk menandai jadwal bentrok?; bagaimana cara membuat sistem penjadwalan perkuliahan. Dari permasalahan tersebut, tujuan penelitian ini untuk mengetahui cara penerapan algoritma pewarnaan graf dalam sistem penjadwalan perkuliahan di STMIK Sumedang, mengetahui cara membuat sistem penjadwalan perkuliahan dan terciptanya suatu sistem penjadwalan perkuliahan yang akan memberikan kemudahan pembuatan jadwal. Dari hasil penelitian ini, diharapkan dapat bermanfaat dan berguna bagi semua pihak. Penulis mengaharapkan penelitian tersebut memiliki kontribusi untuk membantu pekerjaan dosen STMIK Sumedang dalam pembuatan jadwal perkuliahan, proses pembuatan jadwal perkuliahan tidak memakan waktu hingga 3 minggu seperti pada pembuatan secara manual, mengoptimalkan pembuatan jadwal dan terbangunnya sistem penjadwalan perkuliahan yang berbasis komputer.

2. Research Method

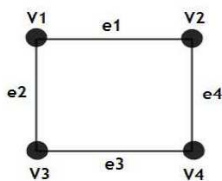
2.1. Model *Waterfall* / Air Terjun.

Disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

2.2. *Library Research*

- a. Menurut Pusat Bahasa Depdiknas (2005) simulasi adalah satu metode pelatihan yang memperagakan sesuatu dalam bentuk tiruan yang mirip dengan keadaan sesungguhnya; simulasi : penggambaran suatu sistem atau proses dengan peragaan memakai model statistik atau pemeran.
- b. Algoritma Graf adalah representasi grafik dalam berbagai masalah yang berkaitan dengan sekumpulan obyek-obyek diskret yang digambarkan sebagai simpul (vertex), dan relasi-relasi biner antar obyek-obyek diskret tersebut yang digambarkan sebagai rusuk (edge). Contoh masalah yang dapat disajikan dalam bentuk graf antara lain peta jalan raya yang menghubungkan satu lokasi dengan lokasi yang lain, skema pertandingan sistem gugur suatu kejuaraan, flowchart (diagram alir) suatu program, dsb. Ketika memodelkan sebuah masalah sebagai suatu graf, seringkali dibutuhkan penambahan informasi lain pada vertex atau edge dalam graf tersebut. Contohnya, pada graf yang menggambarkan jalan raya antar kota, perlu penambahan nama vertex unruk mengidentifikasikan kota, dan bilangan pada edge untuk menunjukkan jarak antar kota.

Pada gambar 1 di bawah ini, dapat diketahui hubungan antara vertex melalui edge-edge yang terhubung. Berdasarkan edges yang terbentuk, maka V1 terhubung dengan V2 dan V3, karena itu V2 dan V3 disebut sebagai tetangga dari V1. Sedangkan V4 tidak bertetangga pada V1.

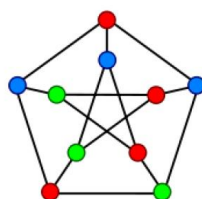


Gambar 1. Graf Sederhana

Pewarnaan graf (graph coloring) adalah pemberian warna pada vertex sedemikian rupa sehingga tidak ada dua buah vertex yang adjacent (terhubung langsung) memiliki warna yang sama dan penggunaan warna diusahakan sesedikit mungkin. Ada 3 macam pewarnaan graf :

a. Pewarnaan Simpul

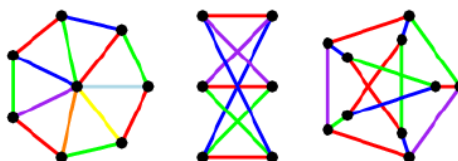
Pewarnaan simpul (vertex coloring) adalah member warna pada simpul-simpul suatu graf sedemikian sehingga tidak ada dua simpul bertetangga mempunyai warna yang sama.



Gambar 2. Contoh Pewarnaan Simpul

b. Pewarnaan Sisi

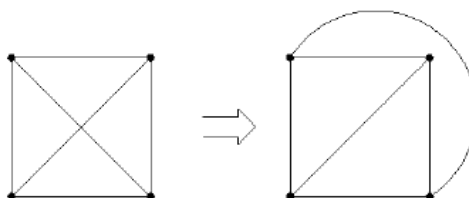
Pewarnaan sisi (edge coloring) adalah memberi warnaberbeda pada sisi yang bertetangga sehingga tidak ada dua sisi yang bertetangga mempunyai warna yang sama.



Gambar 3. Contoh Pewarnaan Sisi

c. Pewarnaan Bidang

Pewarnaan bidang adalah memberi warna pada bidang sehingga tidak ada bidang yang bertetangga mempunyai warna yang sama. Pewarnaan bidang hanya bisa dilakukan dengan membuat graf tersebut menjadi graf planar terlebih dahulu. Graf planar adalah graf yang dapat digambarkan pada bidang datar dengan sisi-sisi yang tidak saling memotong (bersilangan), seperti yang ditunjukkan gambar di bawah ini.



Gambar 4. Contoh Pewarnaan Planar

d. Setelah terbentuk graf planar, lalu memberikan warna berbeda untuk setiap bidang yang berdekatan. Dan jumlah warna yang digunakan harus sedikit mungkin.



Gambar 5. Contoh Pewarnaan Bidang

Dalam pewarnaan graf, jumlah warna yang digunakan untuk mewarnai simpul, sisi, maupun bidang diusahakan sesedikit mungkin. Jumlah warna minimum yang dapat digunakan tersebut disebut bilangan kromatik graf G , disimbolkan dengan $\chi(G)$. Suatu graf G yang mempunyai bilangan kromatis k dilambangkan dengan $\chi(G) = k$.

2.3. Langkah Pengembangan Model

Langkah-langkah pengembangan model merupakan penjelasan dari langkah-langkah sistematis yang akan dilakukan dalam pengembangan model. Berdasarkan uraian metodologi pengembangan model di atas, berikut adalah langkah-langkah yang peneliti lakukan dalam Sistem Penjadwalan Perkuliahan di STMIK Sumedang :

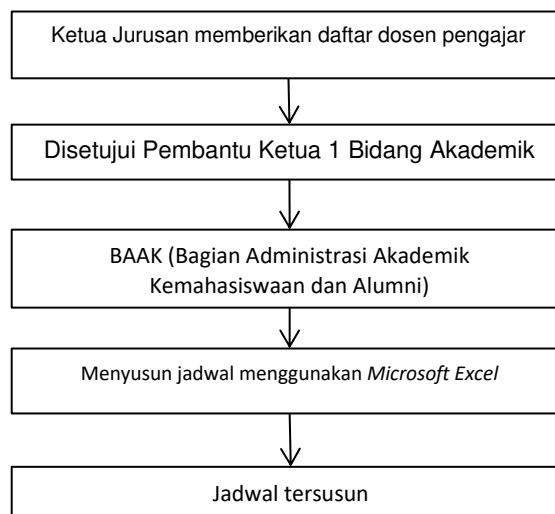
1. Perencanaan penelitian

Peneliti melakukan persiapan dan perencanaan penelitian untuk pembuatan sistem, diantaranya :

 - a. Peneliti melakukan identifikasi dan merumuskan masalah,
 - b. Menentukan tujuan dan manfaat dari penelitian, salah satunya yaitu dengan mengetahui bagaimana cara memfasilitasi petugas dalam membuat jadwal sehingga lebih memudahkan petugas dalam pembuatan jadwal perkuliahan.
 - c. Selanjutnya, peneliti menentukan metodologi penelitian yang akan digunakan dalam pembentukan model, menyiapkan alat dan bahan penelitian, dan menentukan jadwal penelitian.
2. Melakukan analisis terhadap kebutuhan, kegunaan dan batasan serta perancangan software dalam pembuatan sistem penjadwalan perkuliahan.
 - a. Kebutuhan software diantaranya :
 - 1) Data mata kuliah,
 - 2) Data semester,
 - 3) Data ruangan,
 - 4) Data dosen yang mengajar,
 - 5) Data permintaan dosen,
 - 6) Data waktu
 - 7) Prosedur yang sedang berjalan,
 - 8) Contoh jadwal yang telah dibuat.
 - b. Kegunaan software :

Kegunaan software adalah untuk membuat jadwal perkuliahan pada semua jurusan yang ada pada STMIK Sumedang.

- c. Batasan software :
 - 1) Software yang dibuat hanya untuk penjadwalan kegiatan belajar mengajar.
 - 2) Penerapan algoritma pewarnaan graf diimplementasikan menggunakan pendekatan pewarnaan bidang dengan memberikan warna terhadap hasil masukan dari data jadwal untuk menampilkan warna biru apabila terdapat bentrok jadwal.
- d. Perancangan aplikasi Penjadwalan Perkuliahan di STMIK Sumedang, diantaranya :
 - 1) Merancang tabel pada database yang dibutuhkan untuk penyusunan aplikasi,
 - 2) Merancang use case diagram, class diagram, sequence diagram, activity diagram, statechart diagram, dan deployment diagram.
- 3. Merancang tampilan web, penulis membuat desain web yang akan di buat pada aplikasi penjadwalan perkuliahan, rancangan desainnya adalah :
 - a. Rancangan Interface Login
 - b. Rancangan Interface Halaman Home Admin
 - c. Struktur Menu Admin
 - d. Rancangan Interface Halaman Home Dosen
 - e. Struktur Menu Dosen
- 4. Membuat kode disini adalah mengimplementasikan struktur diagram yang telah kita design menjadi form di dalam aplikasi yang kita terjemahkan menggunakan bahasa pemrograman PHP.
- 5. Pengujian sistem yang dilakukan pada aplikasi penjadwalan perkuliahan menggunakan pengujian validasi model yang berdasarkan hasil dari uji coba sistem. Proses ini bertujuan untuk mengetahui apabila ada kesalahan agar dapat diperbaiki.
- 6. Pemeliharaan sistem dilakukan agar sistem dapat berjalan dengan baik.
- 7. Formulasi Sistem Yang Dikembangkan dibuat dengan cara meneliti tentang sistem yang sedang berjalan sehingga dapat diusulkan sebuah sistem baru sebagai tahap pengembangan dari sistem sebelumnya. Prosedur Sistem yang sedang berjalan



Gambar 6. Prosedur Sistem yang Sedang Berjalan

- 8. Evaluasi sistem yang berjalan pada sistem penjadwalan yang sedang berjalan di STMIK Sumedang dilakukan menggunakan microsoft excel. Masalah yang dihadapi saat ini adalah pada proses penyusunan jadwal apabila disusun dengan tidak teliti antara ruangan, dosen, semester dan waktu akan ditemukan jadwal yang bentrok. Selain itu, untuk mengalokasikan ruang dan waktu, data akan disimpan dalam sheet yang berbeda. Tidak adanya hubungan atau relasi data antara satu data dengan yang lainnya membuat proses penyusunan jadwal menjadi tidak efektif.

9. Sistem yang dikembangkan

Sistem yang dikembangkan yaitu sistem dengan relasi data, sehingga data ruang, dosen, waktu dan semester dapat berhubungan satu sama lain melalui data jadwal untuk menghasilkan output jadwal yang tidak bentrok, karena jadwal bentrok akan mudah ditemukan ketika penyusunan jadwal dengan penerapan algoritma pewarnaan graf.

3. Result and Analysis

Tahap pertama yang dilakukan adalah analisis, dari hasil analisis yang penulis lakukan melalui observasi dan wawancara ke tempat penelitian maka kebutuhan untuk sistem pengolahan data untuk pembuatan jadwal adalah sebagai berikut :

1. Sistem

- a. Membutuhkan sistem untuk dua level akses yaitu Admin dan Dosen.
- b. Pengolahan data untuk data user, dosen, data permintaan dosen, data mata kuliah, data ruang kuliah, data semester dan data waktu.
- c. Penyusunan jadwal.
- d. Laporan dari penjadwalan.

2. Interface

- a. Halaman login sebagai fasilitas keamanan dan pembatas pengguna
- b. Dua halaman utama dengan menu yang berbeda untuk Admin dan Dosen.
- c. Halaman untuk mengolah data user
- d. Halaman untuk mengolah data permintaan dosen
- e. Halaman untuk mengolah data dosen
- f. Halaman untuk mengolah data mata kuliah
- g. Halaman untuk mengolah data ruang kuliah
- h. Halaman untuk mengolah data waktu
- i. Halaman untuk mengolah data semester
- j. Halaman untuk menyusun jadwal
- k. Halaman untuk menampilkan laporan

3. Struktur data (*table database*)

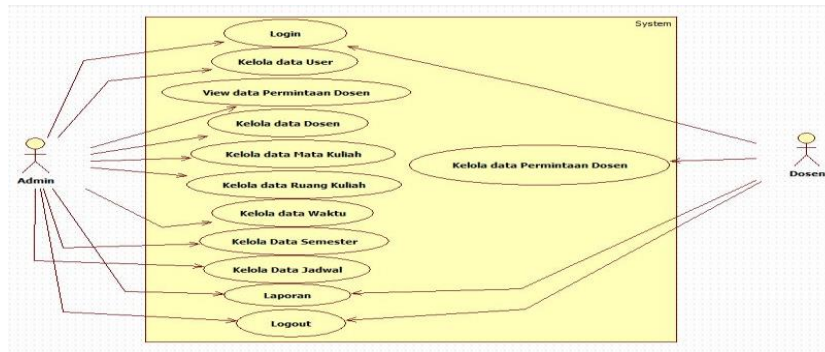
- a. Tabel user
- b. Tabel permintaan dosen
- c. Tabel dosen
- d. Tabel mata kuliah
- e. Tabel ruang kuliah
- f. Tabel waktu
- g. Tabel semester
- h. Tabel jadwal

Pada tahap ini desain disesuaikan dengan hasil analisis maka ada tiga desain yang dilakukan yaitu, desain sistem, desain interface dan desain struktur data atau tabel apa saja yang akan digunakan, maka uraian dari setiap tahapan desain. Desain sistem ini merupakan alur kerja / tahapan dari pengoperasian sistem yang dapat dilakukan oleh *User*. Untuk desain sistem, penulis menggunakan pendekatan UML (*Unified Modeling Language*). Pendekatan UML tersebut akan menggambarkan sistem secara keseluruhan dengan enam diagram yaitu diagram *usecase*, diagram *class*, diagram *statechart*, diagram *activity*, diagram *sequence* dan diagram *deployment*. Adapun beberapa diagram tersebut adalah seperti berikut :

1. *Usecase Diagram*

Usecase adalah bagian tingkat tinggi dari fungsionalitas yang disediakan oleh sistem. Dengan kata lain, *Usecase* menggambarkan bagaimana seseorang menggunakan system.

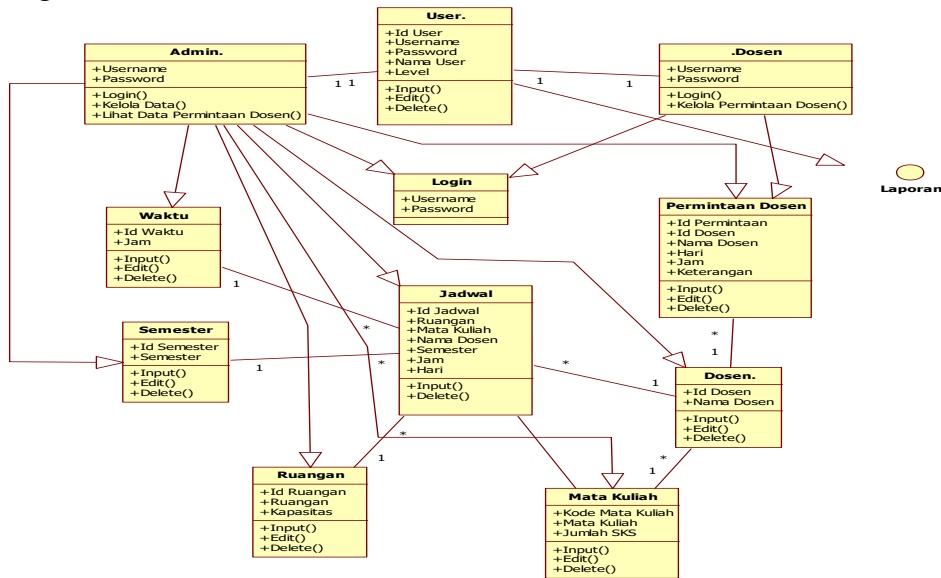
Berikut ini adalah *usecae* diagram dalam sistem Penjadwalan Perkuliahan :



Gambar 7. Usecase Diagram

Dari gambar diagram *usecase* diatas dapat dijelaskan bahwa ada dua entitas dalam yang terlibat secara langsung dengan sistem yaitu Admin dan Dosen. Untuk memulai menggunakan sistem, *User* melakukan login terlebih dahulu, dimana setiap *User* memiliki batasan sistem tersendiri, untuk Dosen hanya mengakses menu permintaan dosen dan laporan. Sedangkan untuk Admin memiliki hak akses penuh dapat mengolah data *User*, data dosen, data mata kuliah, data ruang kuliah, data waktu, data semester, dan data jadwal. Kecuali untuk permintaan dosen, Admin hanya diberi hak untuk melihat data yang telah di input dosen.

2. Class Diagram



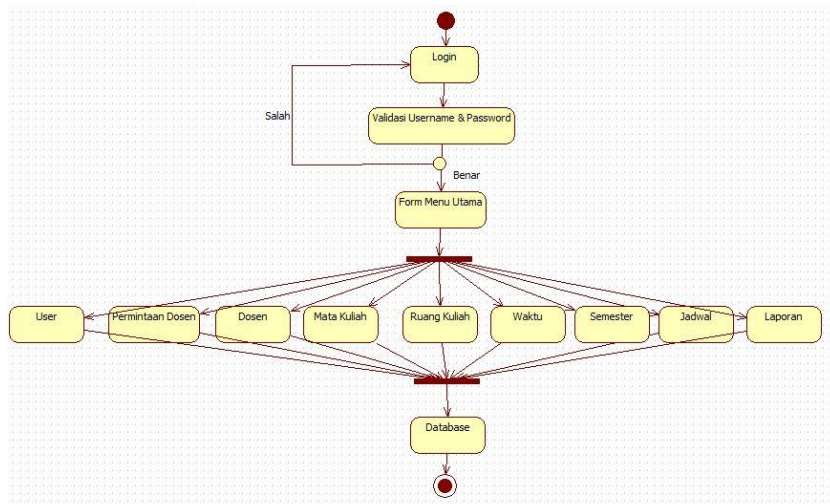
Gambar 8. Class Diagram

Class diagram diatas menggambarkan kelas-kelas dalam sistem penjadwalan perkuliahan dan hubungannya antar satu dengan yang lain. Hubungan antar kelas dijelaskan dengan tanda panah yang memiliki keterangan 1-* atau 1-1. 1-* artinya 1 ke banyak, dan 1-1 artinya 1 ke satu. Serta ada atribut dari setiap sistem dan operasi apa saja yang dapat dilakukan, ada 3 operasi yang disebutkan yaitu *Input*, *Edit*, *Delete*.

3. Statechart Diagram

Statechart diagram merupakan model perilaku yang dinamis dari *class* secara individual maupun beberapa bentuk obyek. Semua itu menunjukkan deretan dari *state* yang dilakukan obyek melalui event yang menyebabkan sebuah transisi dari satu ke aktifitas yang lain, dan beberapa aksi yang menghasilkan dari satu *state* atau aktifitas yang berubah.

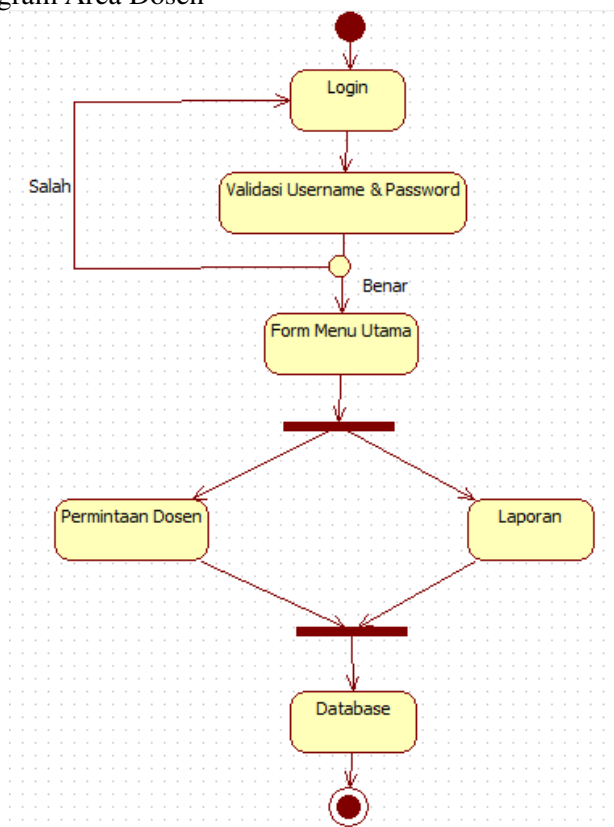
a. *Statechart* Diagram Area Admin



Gambar 9. Statechart Diagram Area Admin

Pada gambar *statechart* diatas dapat dijelaskan, Admin memulai sistem dengan melakukan *login* terlebih dengan memasukkan *username* dan *password* jika *username* dan *password* benar maka akan masuk ke Form menu utama yang akan menampilkan semua menu yang dapat di pilih oleh Admin. Namun jika *username* dan *password* salah maka akan kembali ke menu login.

b. Statechart Diagram Area Dosen



Gambar 10. Statechart Diagram Area Dosen

Sama halnya dengan *statechart* diagram area admin, dosen memulai sistem dengan melakukan *login*. Setelah validasi *login* berhasil maka akan tampil menu utama.

4. Desain Tabel

Berikut ini adalah desain tabel pada sistem Penjadwalan Perkuliahan :

a. Tabel user

Tabel 1. Struktur Tabel user

Nama Field	Tipe Data	Ukuran	Status
id_user	Int		Primary Key
username	varchar	15	
password	varchar	10	
nama_user	varchar	25	
level	enum	'Admin','Dosen'	

b. Tabel permintaan_dosen

Tabel 2. Struktur Tabel permntaan_dosen

Nama Field	Tipe Data	Ukuran	Status
id_permintaan	int		Primary Key
id_dosen	varchar	10	
nama_dosen	varchar	25	
hari	enum	'Senin','Selasa','Rabu','Kamis','Jumat','Sabtu'	
jam	varchar	20	
keterangan	varchar	50	

c. Tabel dosen

Tabel 3. Struktur Tabel dosen

Nama Field	Tipe Data	Ukuran	Status
id_dosen	varchar	5	Primary Key
nama_dosen	varchar	25	

d. Tabel ruang

Tabel 4. Struktur Tabel ruang

Nama Field	Tipe Data	Ukuran	Status
id_ruang	int		Primary Key
ruang	varchar	15	
kapasitas	varchar	5	

e. Tabel matakuliah

Tabel 5. Struktur Tabel matakuliah

Nama Field	Tipe Data	Ukuran	Status
kode_mk	varchar	10	Primary Key
nama_mk	varchar	25	
jumlah_sks	varchar	1	

f. Tabel waktu

Tabel 6. Struktur Tabel waktu

Nama Field	Tipe Data	Ukuran	Status
id_waktu	int		Primary Key
jam	varchar	20	

g. Tabel semester

Tabel 7. Struktur Tabel semester

Nama Field	Tipe Data	Ukuran	Status
id_smt	int		Primary Key
smt	varchar	7	

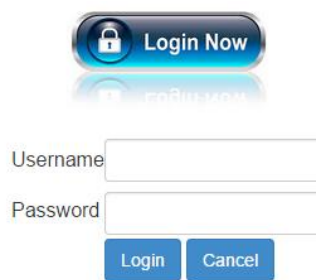
h. Tabel jadwal

Tabel 8. Struktur Tabel jadwal

Nama Field	Tipe Data	Ukuran	Status
id_jadwal	varchar	10	Primary Key
ruang	varchar	15	
nama_mk	varchar	25	
nama_dosen	varchar	25	
smt	varchar	7	
jam	varchar	20	
hari	enum	Senin,Selasa,Rabu, Kamis,Jumat,Sabtu	

5. Implementasi Sistem

a. Tampilan Halaman Login



Gambar 11. Tampilan Halaman login

Tampilan halaman login merupakan tampilan awal dalam sistem penjadwalan perkuliahan STMIK Sumedang. Sebagai tahap awal diharuskan untuk melakukan *login* agar dapat masuk ke dalam sistem, tahap ini menentukan level akses yang diberikan sistem yaitu level untuk Dosen atau Admin. Level tersebut dibedakan dengan mengisi *username* dan *password* pada kotak yang telah disediakan lalu menekan *button login*, sedangkan *button cancel* untuk membatalkan login.

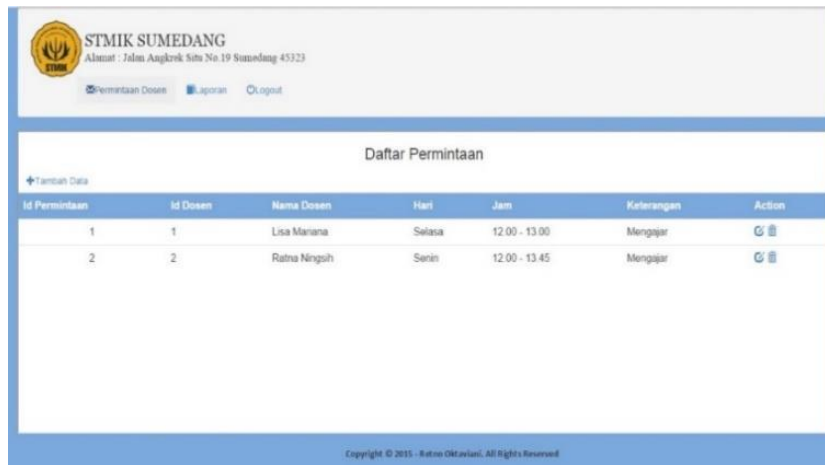
b. Halaman *Home* Dosen



Gambar 12. Tampilan Halaman *Home* Dosen

Tampilan halaman *home* dosen merupakan tampilan awal setelah *login* yang dilakukan oleh dosen melalui hak akses yang diberikan pada saat *login*.

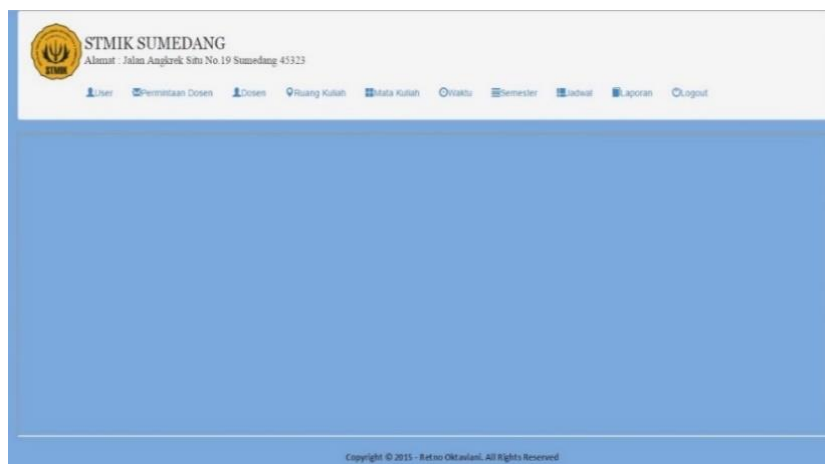
c. Halaman Utama Permintaan Dosen pada Home Dosen



Gambar 13. Tampilan Halaman Utama Permintaan Dosen pada *Home* Dosen

Tampilan halaman utama permintaan dosen di atas merupakan halaman yang hanya dapat diakses oleh dosen saja. Pada halaman tersebut terdapat daftar permintaan berupa tabel yang dapat ditambah dengan memilih menu tambah data, dosen dapat memperbarui data dengan memilih *action* edit dan menghapus data dengan memilih *action* hapus. Dengan proses tambah, edit dan hapus, data yang telah diproses akan tersimpan dan tampil dalam tabel.

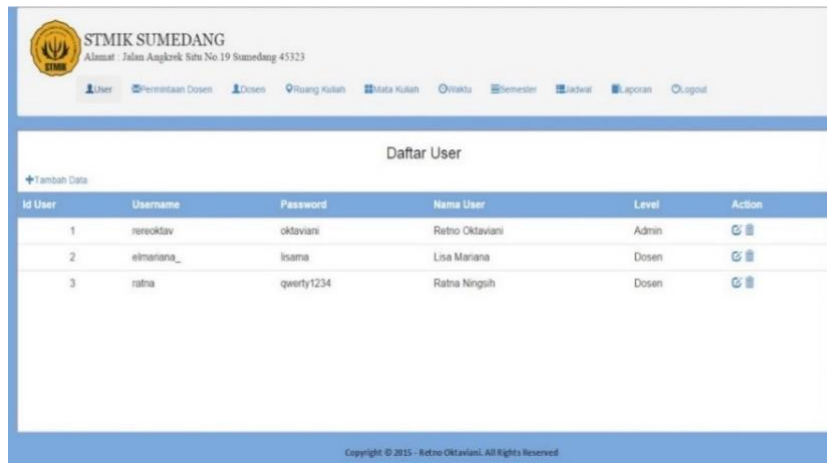
d. Halaman *Home* Admin



Gambar 14. Tampilan Halaman *Home* Admin

Tampilan halaman *home* admin merupakan tampilan awal setelah *login* yang dilakukan oleh admin melalui hak akses yang diberikan pada saat *login*. Pada halaman ini terdapat menu user, permintaan dosen, dosen, ruang kuliah, mata kuliah, waktu, semester, jadwal, laporan, dan logout. Semua data tersebut dapat dikelola oleh admin kecuali data permintaan dosen admin hanya diberi hak akses untuk melihat data permintaan dosen tersebut.

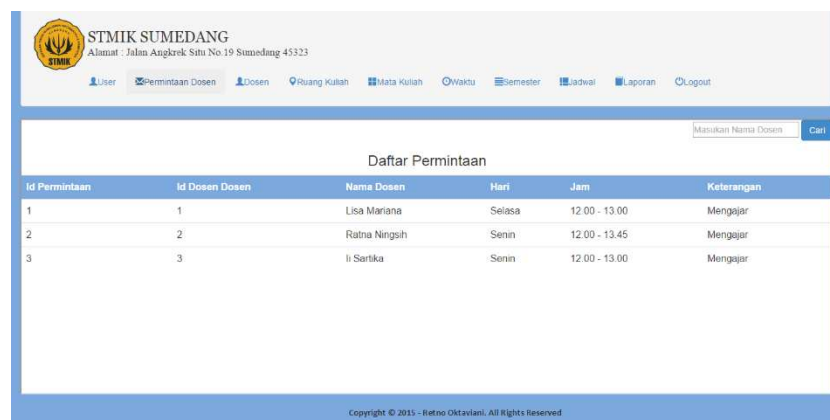
e. Halaman Utama User



Gambar 15. Tampilan Halaman Utama User

Tampilan halaman utama user di atas merupakan halaman yang hanya dapat diakses oleh admin saja. Pada halaman tersebut terdapat daftar user berupa tabel yang dapat ditambah dengan memilih menu tambah data, admin dapat memperbarui data dengan memilih *action* edit dan menghapus data dengan memilih *action* hapus. Dengan proses tambah, edit dan hapus, data yang telah diproses akan tersimpan dan tampil dalam tabel.

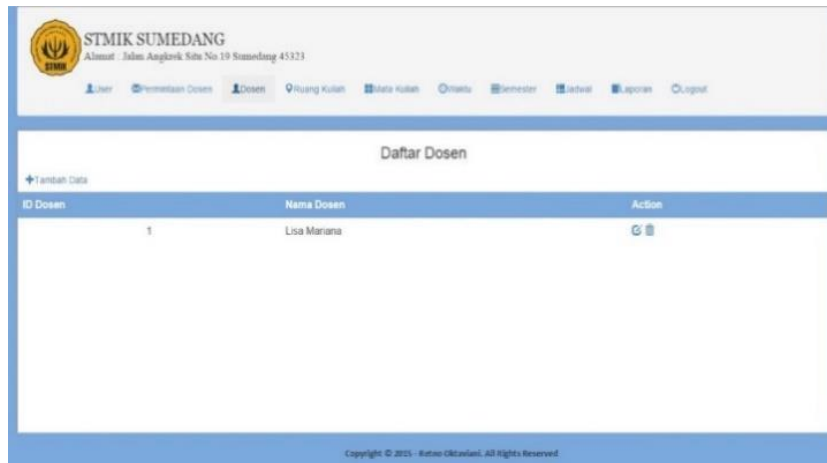
f. Halaman Utama Permintaan Dosen



Gambar 16. Tampilan Halaman Utama Permintaan Dosen

Tampilan halaman utama permintaan dosen di atas merupakan halaman yang hanya dapat diakses oleh admin saja. Pada halaman tersebut terdapat daftar permintaan berupa tabel yang telah ditambah atau diedit oleh dosen. Admin dapat mencari daftar permintaan dengan memasukkan nama dosen yang akan dicari pada kotak isian yang telah disediakan dan menekan tombol cari untuk mengeksekusi perintah sehingga data yang dicari dapat ditampilkan.

g. Halaman Utama Dosen



Gambar 17. Tampilan Halaman Utama Dosen

Tampilan halaman utama dosen di atas merupakan halaman yang hanya dapat diakses oleh admin saja. Pada halaman tersebut terdapat daftar dosen berupa tabel yang dapat ditambah dengan memilih menu tambah data, admin dapat memperbarui data dengan memilih *action* edit dan menghapus data dengan memilih *action* hapus. Dengan proses tambah, edit dan hapus, data yang telah diproses akan tersimpan dan tampil dalam tabel.

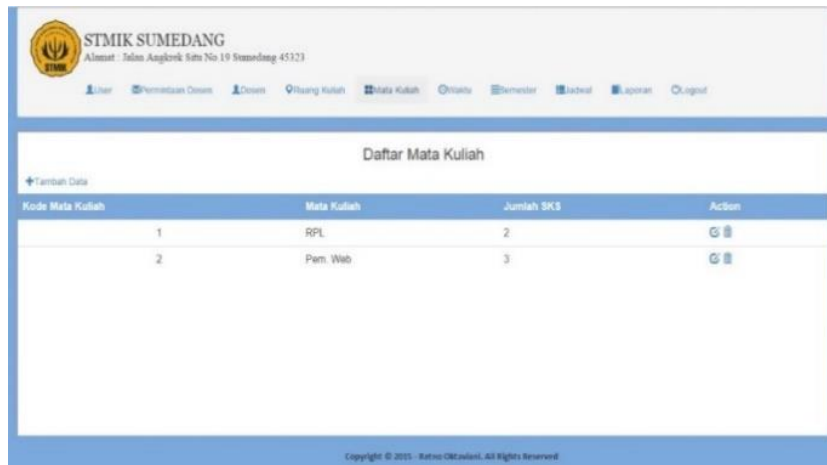
h. Halaman Utama Ruang Kuliah



Gambar 18. Tampilan Halaman Utama Ruang Kuliah

Tampilan halaman utama ruang kuliah di atas merupakan halaman yang hanya dapat diakses oleh admin saja. Pada halaman tersebut terdapat daftar ruangan berupa tabel yang dapat ditambah dengan memilih menu tambah data, admin dapat memperbarui data dengan memilih *action* edit dan menghapus data dengan memilih *action* hapus. Dengan proses tambah, edit dan hapus, data yang telah diproses akan tersimpan dan tampil dalam tabel.

i. Halaman Utama Mata Kuliah



Gambar 19. Tampilan Halaman Utama Mata Kuliah

Tampilan halaman utama mata kuliah di atas merupakan halaman yang hanya dapat diakses oleh admin saja. Pada halaman tersebut terdapat daftar mata kuliah berupa tabel yang dapat ditambah dengan memilih menu tambah data, admin dapat memperbarui data dengan memilih *action* edit dan menghapus data dengan memilih *action* hapus. Dengan proses tambah, edit dan hapus, data yang telah diproses akan tersimpan dan tampil dalam tabel.

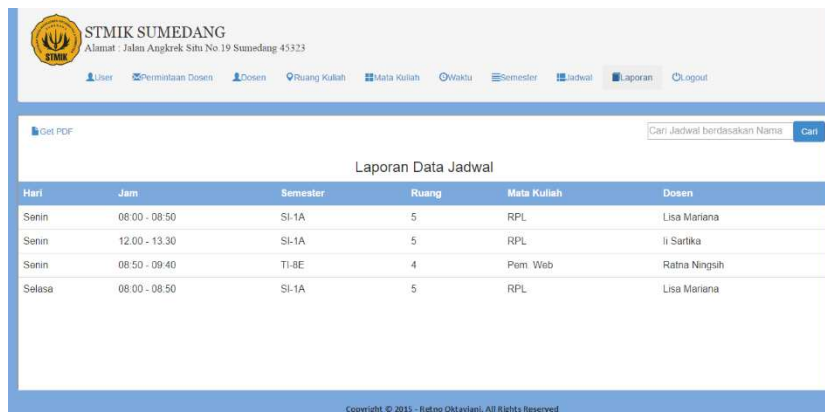
j. Halaman Utama Jadwal



Gambar 20. Tampilan Halaman Utama Jadwal

Tampilan halaman utama jadwal di atas merupakan halaman yang hanya dapat diakses oleh admin saja. Pada halaman tersebut terdapat daftar jadwal berupa tabel, isi tabel berwarna biru dikarenakan jadwal bentrok, disini penulis menerapkan algoritma pewarnaan graf. Data dari tabel tersebut dapat ditambah dengan memilih menu tambah data, admin juga dapat memperbarui data dengan memilih *action* edit dan menghapus data dengan memilih *action* hapus untuk mengoptimalkan data agar tidak terjadi bentrok jadwal. Dengan proses tambah, edit dan hapus, data yang telah diproses akan tersimpan dan tampil dalam tabel.

k. Halaman Laporan



Gambar 21. Tampilan Halaman Utama Laporan

Tampilan halaman utama laporan di atas merupakan halaman yang dapat diakses oleh admin dan dosen. Pada halaman tersebut terdapat data jadwal yang telah tersusun berdasarkan hari. Data tersebut dapat difilter untuk menampilkan data yang dikehendaki melalui proses pencarian dengan mengetikkan nama dosen pada form isian dan menekan tombol cari. Sedangkan menu Get PDF untuk menampilkan data berupa PDF.

4. Conclusion

Setelah melakukan analisis dan implementasi sistem yang dilanjutkan dengan pengujian sistem, maka dari hasil implementasi dan pengujian tersebut dapat ditarik beberapa kesimpulan diantaranya Sistem Penjadwalan Perkuliahan memudahkan dalam pengolahan data, tersedia tempat untuk dosen mengisi permintaan agar tidak mengajar di hari dan jam tertentu dan dapat dirubah secara langsung oleh dosen tersebut sebelum jadwal tersusun, memudahkan dalam melihat data jadwal yang bentrok, membantu Sekolah Tinggi Manajemen Informatika dan Komputer Sumedang dalam penyusunan jadwal perkuliahan menjadi lebih cepat dan algoritma pewarnaan graf dapat diterapkan pada sistem penjadwalan perkuliahan untuk menandai jadwal yang bentrok.

Adapun yang dapat penulis sampaikan yang berhubungan dengan sistem penjadwalan di STMIC Sumedang diantaranya penjadwalan ini masih harus dikembangkan karena belum mencapai tahap yang sempurna dan masih dalam tahap pengembangan dengan demikian diharapkan untuk kedepannya menjadi lebih baik, Perlu adanya pemeliharaan terhadap sistem secara berkala sehingga sistem dapat tetap berjalan dan untuk memudahkan proses pengembangan. Perlu adanya hubungan antara sistem akademik agar mahasiswa dapat terhubung langsung dengan sistem penjadwalan perkuliahan untuk memilih jadwal yang akan dilaksanakan oleh mahasiswa tersebut ketika mengisi KRS

References

[1]. A.S. Rosa., M. Salahudin. (2013). Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung : Informatika Bandung.
 [2]. Kadir, Abdul. 2003. Pengenalan Sistem Informasi. Yogyakarta: CV Andi Offset.
 [3]. Presman, S. Roger. Ph.D.2010. Perangkat Lunak Rekayasan. Yogyakarta: CV Andi Offset.
 [4]. Budi. 2011. Database Menggunakan MySQL. Bandung : Informatika Bandung.
 [5]. Simarmata, Janner. 2010. Rekayasa Perangkat Lunak. Yogyakarta : CV Andi Offset.
 [6]. Sommerville, Ian. 2011. Software Engineering (Rekayasa Perangkat Lunak). Jakarta : Erlangga.
 [7]. Komputer, Wahana. 2006. Menguasai Pemrograman Web dengan PHP 5, Yogyakarta : Penerbit Andi.
 [8]. Ashari, Fadhy. 2012. Rekayasa Perangkat Lunak Menggunakan UML. Yogyakarta : Andi Yogyakarta.