



Hak cipta dan penggunaan kembali:

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

Copyright and reuse:

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

BAB III

METODOLOGI DAN PERANCANGAN

3.1 Metode Penelitian

Penelitian aplikasi penjadwalan dengan menggunakan algoritma *particle swarm optimization* ini melalui beberapa langkah yaitu, sebagai berikut.

1. Studi Literatur

Penelitian diawali dengan melakukan studi kepustakaan dari hasil penelitian yang dilakukan sebelumnya oleh orang lain dalam bentuk jurnal-jurnal dan artikel yang terdapat pada media cetak maupun media internet. Studi yang dilakukan terkait dengan aplikasi untuk penjadwalan, teori-teori terkait, dan algoritma *particle swarm optimization*.

2. Pengumpulan Kebutuhan pengguna (*user requirements*)

Tahap selanjutnya ialah melakukan pengumpulan data mengenai hal-hal yang dibutuhkan oleh pengguna, konsep dan cara kerja penjadwalan yang digunakan oleh Universitas Multimedia Nusantara. Pengumpulan data dilakukan dengan melakukan wawancara kepada pihak akademik, khususnya bagian yang bertugas dalam hal penjadwalan.

3. Analisis dan Perancangan

Setelah kebutuhan pengguna didapatkan, maka selanjutnya dilakukan analisis dan perancangan terhadap aplikasi yang dibangun, meliputi analisis hasil pemahaman berdasarkan studi literatur dan data kebutuhan pengguna, sistem yang akan dibuat, serta perancangan alur kerja aplikasi dalam bentuk *system*

flow, *flowchart*, *entity relationship diagram* (ERD), dan rancangan tampilan antar muka (*user interface*).

4. Implementasi

Pada tahap ini, dilakukan implementasi *coding* berdasarkan hasil rancangan yang telah dibuat sebelumnya. *Coding* dilakukan dengan menggunakan aplikasi Visual Studio 2008, bahasa pemrograman C#, serta *database* MySQL.

5. Uji Coba (*Testing*) dan evaluasi

Melakukan uji coba terhadap aplikasi yang telah dibuat. Selanjutnya dilakukan evaluasi terhadap hasil uji coba tersebut apakah jadwal yang dihasilkan sesuai dengan yang diharapkan.

3.2 Analisis

Melalui hasil wawancara dengan staff penjadwalan Badan Administrasi Akademis dan Kemahasiswaan (BAAK) Universitas Multimedia Nusantara, didapatkan beberapa jenis *constraints* yang tidak boleh dilanggar. *Constraints* yang dimaksud ialah

1. Tidak boleh ada satu kelas yang dipakai pada saat yang bersamaan untuk mata kuliah yang berbeda
2. Tidak boleh ada satu dosen yang mengajar pada dua kelas yang berbeda pada saat yang bersamaan

3. Kelas yang sedang dipakai oleh mata kuliah tertentu tidak boleh dipakai lagi oleh mata kuliah lainnya selama durasi mata kuliah yang bersangkutan belum selesai.
4. Dosen yang sedang mengajar mata kuliah tertentu, tidak dapat mengajar kelas lainnya selama durasi mata kuliah yang bersangkutan belum selesai.

Berdasarkan hasil analisa yang dilakukan, didapatkan beberapa skenario pengujian atau kondisi yang harus diujikan terhadap sistem yang dibuat. Adapun skenario pengujian atau kondisi yang harus diujikan adalah

1. Bagaimana sistem menghasilkan penjadwalan apabila terdapat kelas yang bentrok jadwal penggunaannya
2. Bagaimana sistem menghasilkan jadwal apabila terdapat dosen yang bentrok jadwal mengajarnya
3. Bagaimana sistem akan mengatur jadwal penggunaan kelas untuk mata kuliah yang memerlukan ruangan khusus
4. Bagaimana sistem akan memposisikan jadwal mengajar antara tiap-tiap kelas yang diajar oleh seorang dosen dan penempatan kelasnya, agar sebisa mungkin berada pada kelas yang sama atau berdekatan posisinya, sehingga dosen yang bersangkutan bisa lebih nyaman
5. Bagaimana sistem menjadwalkan mata kuliah yang diampu oleh lebih dari satu orang dosen.
6. Bagaimana sistem akan mengatur, apabila jadwal mengajar dosen yang bersangkutan lebih atau kurang dari yang diperlukan.

3.3 Perancangan Sistem

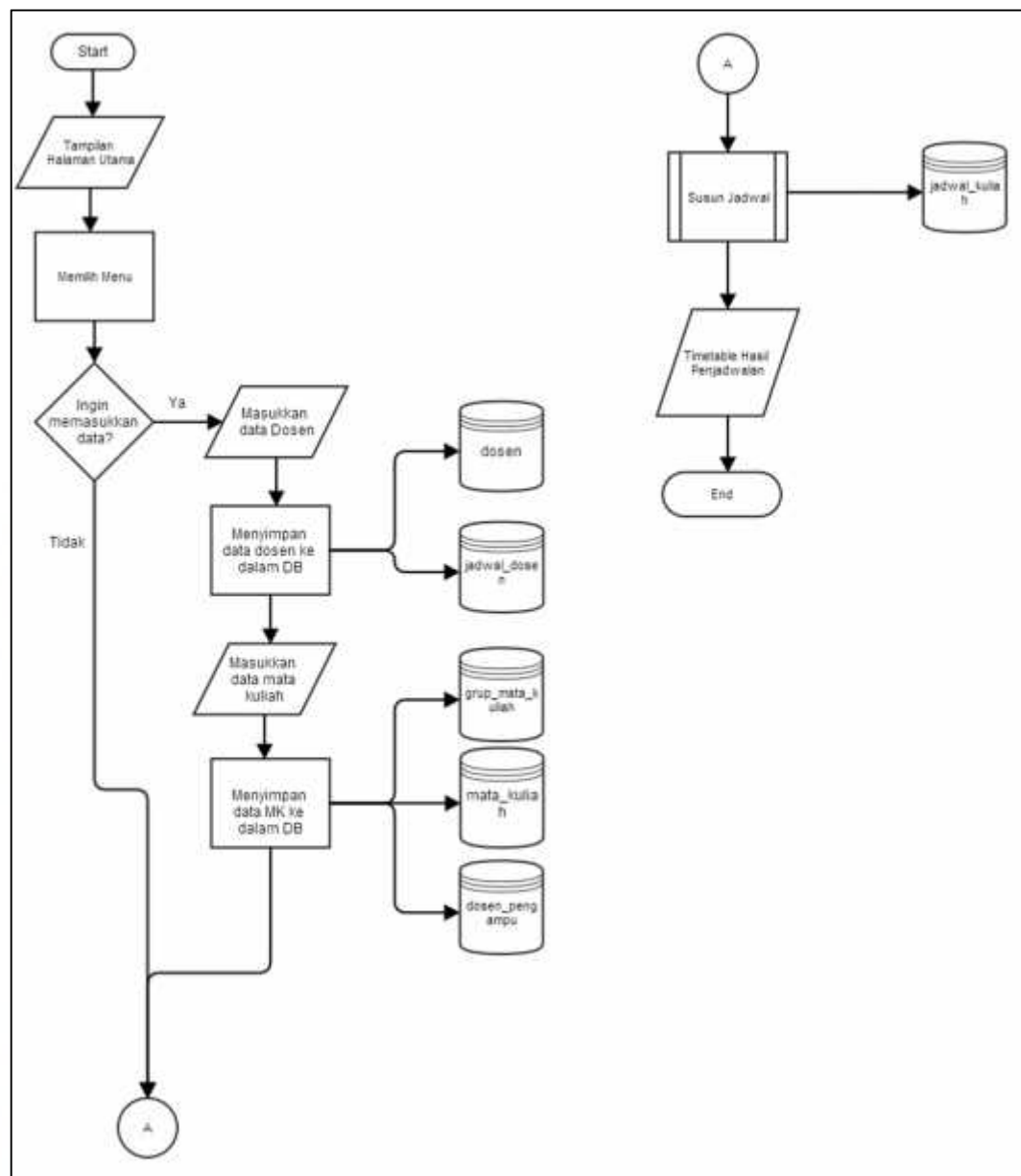
Perancangan sistem dimulai dengan pembuatan diagram arus langkah penyelesaian permasalahan yang dihadapi, dilanjutkan dengan perancangan tabel-tabel pada *database* untuk mendukung keperluan penyimpanan data yang diperlukan dan dihasilkan oleh sistem.

3.3.1 System Flow

Sytem flow adalah sebuah bagan yang menggambarkan arus atau aliran langkah-langkah penyelesaian suatu permasalahan.

A. System Flow Sistem Penyusunan Jadwal

Sistem ini memiliki beberapa langkah seperti yang digambarkan pada gambar 3.1. Dimana, sistem ketika dijalankan, akan menampilkan tampilan utama kepada pengguna. Ketika data yang diperlukan belum ada, pengguna akan diminta untuk mengisi terlebih dahulu data yang diperlukan oleh sistem.



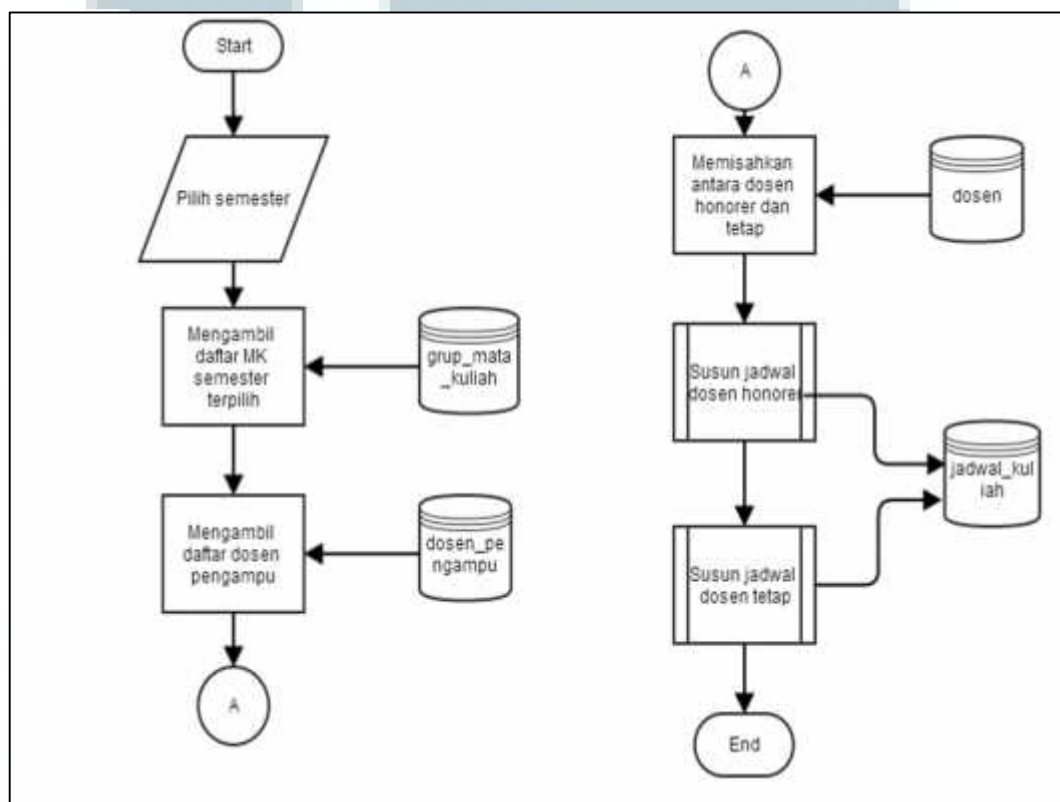
Gambar 3.1 System Flow Penyusunan Jadwal

B. System Flow Subproses Susun Jadwal

Pada subproses susun jadwal, pengguna akan terlebih dahulu diminta untuk memilih semester yang disusun jadwalnya yaitu, semester ganjil/genap. Secara *default* pilihan semester yang dipilih adalah semester ganjil. Setelah itu

dilanjutkan dengan mengambil daftar mata kuliah untuk semester terpilih, beserta dengan mengambil daftar dosen pengampu mata kuliah yang bersangkutan.

Ketika daftar sudah selesai diambil, maka sistem secara otomatis akan memilah antara dosen yang berstatus dosen honorer dan dosen tetap dengan mengacu pada data dosen dalam tabel dosen, untuk kemudian dilanjutkan dengan menyusun jadwal dosen honorer terlebih dahulu, baru dosen tetap.



Gambar 3.2 System Flow Subprocess Susun Jadwal

C. System Flow Subproses Penyusunan Jadwal Dosen Honorar

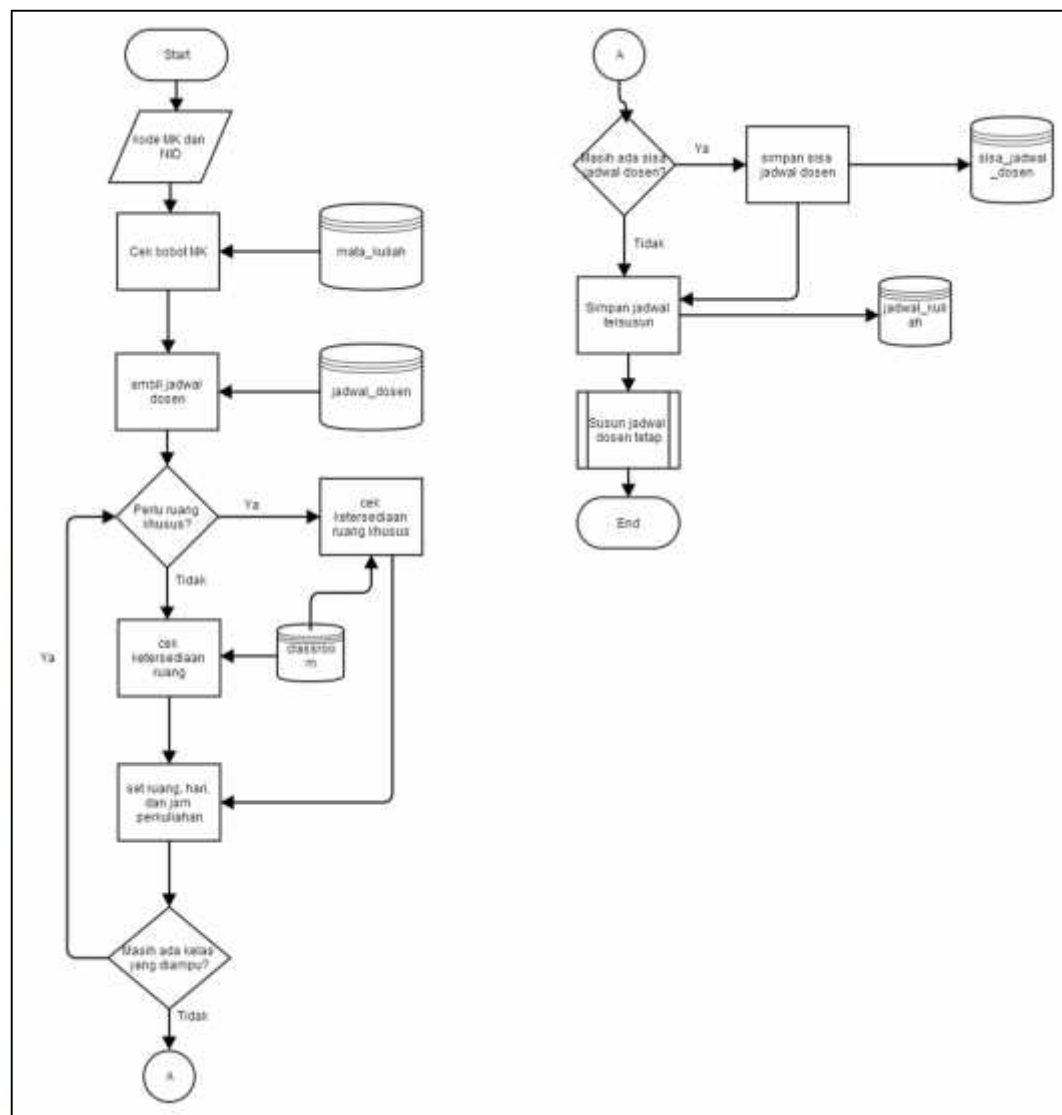
Pada penelitian ini, penyusunan jadwal dosen honorer tidak menggunakan algoritma *particle swarm optimization*. Hal ini disebabkan karena jadwal dosen honorer yang lebih terbatas dibandingkan jadwal dosen tetap yang lebih fleksibel, akan membuatnya sulit dijadwalkan dengan menggunakan algoritma *particle*

swarm optimization. Sehingga dengan demikian, subproses penyusunan jadwal dosen honorer dan dosen tetap memiliki logika algoritma yang berbeda.

Pada penelitian ini, dikarenakan jadwal dosen honorer yang lebih terbatas pula, maka penyusunan jadwal dosen honorer dijadwalkan terlebih dahulu, baru kemudian penyusunan jadwal dosen tetap dilakukan.

Langkah dimulai dengan pengecekan bobot mata kuliah yang diampu untuk kemudian dipakai sebagai pertimbangan durasi mengajar dosen untuk mata kuliah yang bersangkutan. Kemudian, sistem akan mengambil jadwal ketersediaan mengajar dosen honorer. Selanjutnya, sistem akan mengecek apakah mata kuliah yang akan diselenggarakan memerlukan ruangan khusus atau tidak. Baik memerlukan ruangan khusus atau tidak, sistem akan mengecek ketersediaan ruangan terlebih dahulu sesuai dengan jadwal ketersediaan mengajar dosen. Data yang terdapat pada *database classroom* pada diagram, terlihat tidak menerima *input* karena data yang terdapat didalam *database classroom* tidak di-*input*-kan pada sistem ini, melainkan dari sistem lain, sehingga dapat kita asumsikan bahwa *database classroom* pasti sudah tersedia data yang diperlukan sistem untuk melakukan pengecekan ketersediaan ruangan.

Sistem akan terus mengulang proses selama dosen masih memiliki kelas yang harus diajar. Setelah semua kelas yang diampu oleh dosen yang bersangkutan selesai dijadwalkan, sistem akan mengecek apakah dosen yang bersangkutan masih memiliki sisa jadwal mengajar atau tidak. Jika ya, akan dimasukan ke dalam *database*, dilanjutkan dengan menyimpan daftar jadwal yang sudah tersusun ke dalam *database*.



Gambar 3.3 System Flow Susun Jadwal Dosen Honorer

D. Flowchart Subproses Penyusunan Jadwal Dosen tetap

Penyusunan jadwal dosen tetap dengan menggunakan algoritma *particle swarm optimization* dimulai dengan menginisialisasikan $c1$ (*particle learning rates*) sebesar 1.5 dan $c2$ (*swarm learning rates*) dengan nilai 1.5 juga, serta w (*inertia weight*) dengan nilai 0.5.

Nilai yang digunakan dipilih berdasarkan studi yang dilakukan pada beberapa jurnal dan penelitian lainnya, serta percobaan yang dilakukan

sebelumnya, untuk menetapkan bahwa nilai $c1$, $c2$ dan w yang dipilih menghasilkan nilai yang lebih baik dibanding nilai yang lainnya untuk ketiga variabel tersebut. Kemudian nilai iterasi awal yang ditulis k ditetapkan dengan nilai 0, karena ini adalah iterasi pertama.

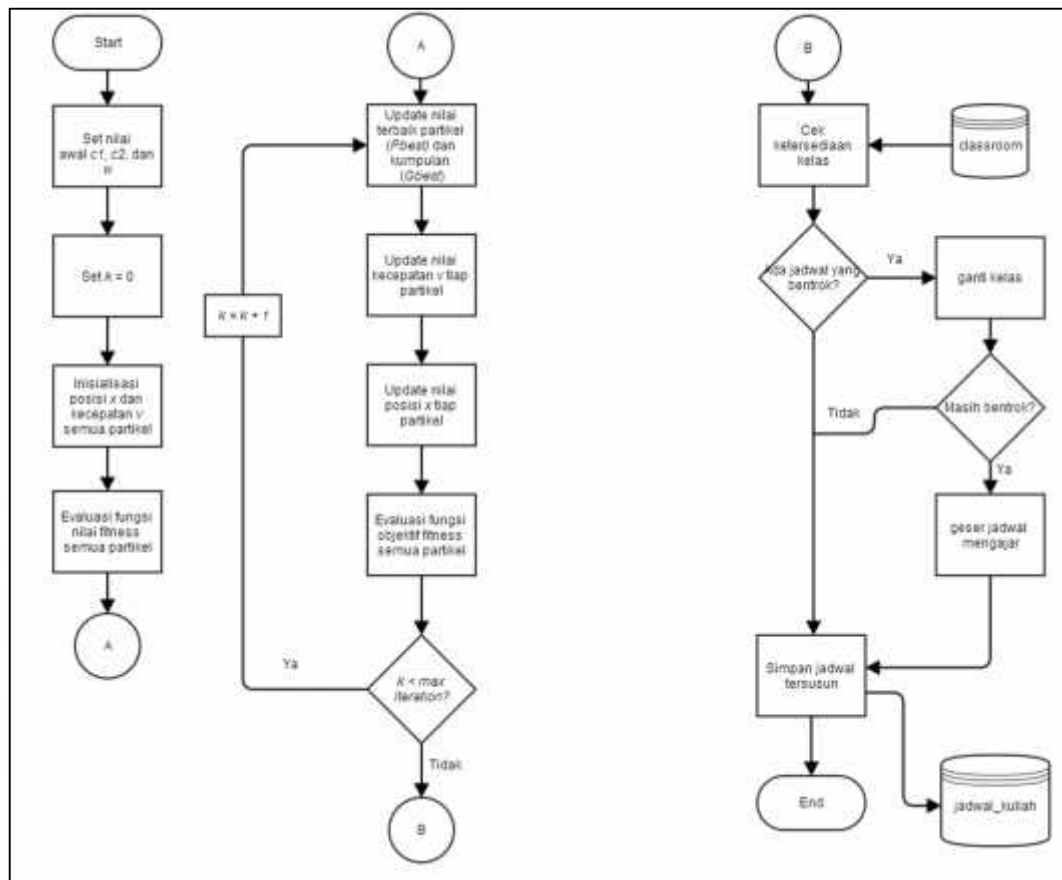
Kemudian secara acak, akan diinisialisasikan posisi tiap-tiap partikel dengan *range* 0-53. Dilanjutkan dengan memberikan nilai 0 sebagai nilai awal kecepatan tiap-tiap partikel. Dari posisi yang telah dihasilkan secara acak, akan dipilih posisi terbaik partikel ($pBest$). Pada iterasi pertama ini, nilai posisi terbaik *swarm* ($gBest$) masih sama dengan nilai posisi terbaik partikel.

Dengan menggunakan rumus 2.3 dan 2.4 yang telah dijelaskan sebelumnya, pembaharuan nilai posisi dan kecepatan partikel dilakukan.

Hasil posisi tiap partikel yang baru akan dievaluasi kembali untuk menentukan nilai posisi yang terbaik. Jika kondisi berhenti (nilai iterasi maksimal) tercapai, maka proses pengulangan akan berhenti. Jika belum, proses akan mengevaluasi kembali nilai terbaik posisi partikel dan *swarm*.

Posisi akhir (posisi yang dianggap terbaik) yang telah didapatkan, akan ditempatkan sesuai dengan slot yang telah tersedia.

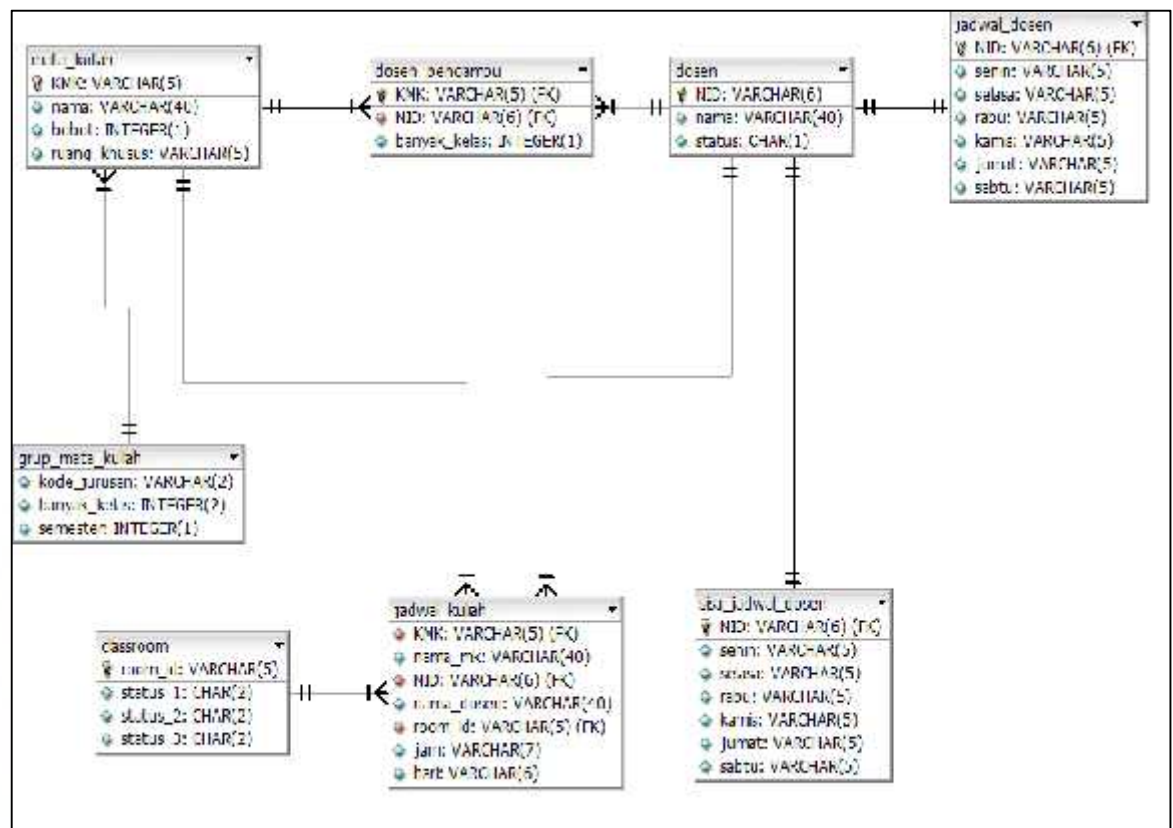
Sama seperti pada gambar 3.3, pada gambar 3.4 terdapat tabel *classroom* yang tidak menerima *input* dari sistem ini, melainkan dari sistem yang lain diluar sistem penjadwalan ini.



Gambar 3.4 Flowchart Susun Jadwal Dosen Tetap

3.3.2 Penggunaan Database

Data-data yang dimasukkan oleh pengguna akan disimpan dalam *database* yang telah dirancang sedemikian rupa, untuk kemudian diakses kembali oleh sistem ketika sistem membutuhkan informasi terkait, guna menyusun jadwal. Hubungan antar tabel dapat dilihat seperti pada gambar 3.5.



Gambar 3.5 Diagram Relasi Entitas

Terdapat delapan tabel yang menyimpan informasi yang dapat diperlukan oleh pengguna sistem. Beberapa tabel yang terdapat pada *database*, ialah

1. Tabel Informasi Dosen

Nama Tabel : dosen
 Fungsi : menyimpan informasi dosen
 Primary Key : NID

Tabel 3.1 Struktur tabel Dosen

Nama field	Type Data	Keterangan
NID	VARCHAR(6)	Nomor Induk Dosen
Nama	VARCHAR(40)	Nama Dosen
Status	CHAR(1)	Status Dosen Pengajar (Tetap / Tidak Tetap)

Setiap *record* pada tabel dosen ini, mewakilkan satu dosen pengajar, dengan *field* NID sebagai *primary key*. Pada *field* status, dosen diberi status A (dosen tetap) atau B (dosen tidak tetap).

2. Tabel Informasi Jadwal Dosen

Nama Tabel : jadwal_dosen
 Fungsi : menyimpan informasi jadwal ketersediaan mengajar dosen
 Primary Key : -
 Foreign Key : NID

Tabel 3.2 Struktur tabel jadwal_dosen

Nama field	Type Data	Keterangan
NID	VARCHAR(6)	Nomor Induk Dosen
Senin	VARCHAR(5)	Ketersediaan mengajar hari Senin
Selasa	VARCHAR(5)	Ketersediaan mengajar hari Selasa
Rabu	VARCHAR(5)	Ketersediaan mengajar hari Rabu
Kamis	VARCHAR(5)	Ketersediaan mengajar hari Kamis
Jumat	VARCHAR(5)	Ketersediaan mengajar hari Jumat
Sabtu	VARCHAR(5)	Ketersediaan mengajar hari Sabtu

Setiap *record* pada tabel ini, mewakili ketersediaan mengajar dosen yang bersangkutan pada satu minggu perkuliahan.

Waktu mengajar diisi dengan format waktu 24 jam. Misalkan pada hari Senin diisi dengan 9-11 dan Selasa diisi dengan 13-17, maka yang dimaksud ialah dosen tersebut pada hari Senin memiliki waktu ketersediaan mengajar pukul 09.00 hingga pukul 11.00, sementara pada hari Selasa pukul 01.00 siang hingga 05.00 sore.

3. Tabel Informasi Mata Kuliah

Nama Tabel : mata_Kuliah
 Fungsi : menyimpan informasi mata kuliah

Primary Key : KMK

Tabel 3.3 Struktur tabel mata_kuliah

Nama field	Tipe Data	Keterangan
KMK	VARCHAR(5)	Kode Mata Kuliah
Nama	VARCHAR(40)	Nama Mata Kuliah
Bobot	INT(1)	Bobot SKS Mata Kuliah
Ruang_khusus	VARCHAR(5)	Memerlukan ruangan khusus / tidak

Setiap *record* pada tabel mata_kuliah, mewakili satu mata kuliah. *Field* ruang_khusus berisi keterangan apakah mata kuliah memerlukan ruangan khusus pada penyelenggaraannya atau tidak. Bila diperlukan, maka *field* akan diisi dengan nomor ruangan khusus yang diperlukan. Apabila tidak diperlukan, maka diisi dengan keterangan 'tidak'.

4. Tabel Informasi Dosen Pengampu Mata Kuliah

Nama Tabel : dosen_pengampu

Fungsi : menyimpan informasi daftar dosen pengampu suatu mata kuliah

Primary Key : -

Foreign Key : NID, KMK

Tabel 3.4 Struktur tabel dosen_pengampu

Nama field	Tipe Data	Keterangan
KMK	VARCHAR(5)	Kode Mata Kuliah
NID	VARCHAR(6)	Nomor Induk Dosen
Banyak_kelas	INT(1)	Banyaknya kelas yang dipegang

Setiap *record* mewakili hubungan antara dosen pengampu mata kuliah, dengan mata kuliah yang bersangkutan. *Field* banyak_kelas, berisikan informasi

jumlah banyaknya kelas yang dipegang dosen pengampu tersebut untuk satu mata kuliah.

5. Tabel Informasi Grup Mata Kuliah

Nama Tabel	: grup_mata_kuliah
Fungsi	: menyimpan informasi banyaknya kelas yang dibuka per mata kuliah dalam tiap jurusan
Primary Key	: -
Foreign Key	: KMK

Tabel 3.5 Struktur tabel grup_mata_kuliah

Nama field	Tipe Data	Keterangan
KMK	VARCHAR(5)	Kode Mata Kuliah
Kode_jurusan	VARCHAR(2)	Kode Jurusan
Banyak_kelas	INT(2)	Banyaknya kelas

Setiap *record* mewakili hubungan antara jurusan (program studi) di Universitas Multimedia Nusantara dengan mata kuliah yang bersangkutan. Pada tabel grup_mata_kuliah, *field* banyak kelas, menunjukkan berapa banyak jumlah kelas yang akan dibuka untuk mata kuliah tersebut. Tujuan dari tabel grup_mata_kuliah adalah agar total kelas yang akan dibuka per semester, menjadi lebih jelas.

6. Tabel Jadwal Mata Kuliah Tersusun

Nama Tabel	: jadwal_kuliah
Fungsi	: menyimpan informasi jadwal mata kuliah yang tersusun
Primary Key	: -
Foreign Key	: KMK, NID

Tabel 3.6 Struktur tabel jadwal_kuliah

Nama field	Type Data	Keterangan
KMK	VARCHAR(5)	Kode Mata Kuliah
Nama_mk	VARCHAR(40)	Nama Mata Kuliah
NID	VARCHAR(6)	Nomor Induk Dosen
Nama_dosen	VARCHAR(40)	Nama Dosen
Room_id	VARCHAR(5)	Kode Ruangan
Jam	VARCHAR(7)	Jam Mata Kuliah diselenggarakan
Hari	VARCHAR(6)	Hari Mata Kuliah diselenggarakan

Tabel 3.6 berfungsi untuk mencatat hasil mata kuliah yang telah terjadwal, baik bagi dosen honorer maupun dosen tetap.

7. Tabel Informasi Sisa Jadwal Ketersediaan Mengajar Dosen

Nama Tabel : sisa_jadwal_dosen

Fungsi : menyimpan informasi sisa jadwal ketersediaan mengajar dosen

Primary Key : -

Foreign Key : NID

Tabel 3.7 Struktur tabel sisa_jadwal_dosen

Nama field	Type Data	Keterangan
NID	VARCHAR(6)	Nomor Induk Dosen
Senin	VARCHAR(5)	Sisa ketersediaan mengajar hari Senin
Selasa	VARCHAR(5)	Sisa ketersediaan mengajar hari Selasa
Rabu	VARCHAR(5)	Sisa ketersediaan mengajar hari Rabu
Kamis	VARCHAR(5)	Sisa ketersediaan mengajar hari Kamis
Jumat	VARCHAR(5)	Sisa ketersediaan mengajar hari Jumat
Sabtu	VARCHAR(5)	Sisa ketersediaan mengajar hari Sabtu

Fungsi dari tabel 3.7 adalah mencatat sisa dari jadwal dosen yang masih ada. Sisa jadwal yang tercatat pada tabel ini, bisa disebabkan oleh beberapa hal seperti sisa jadwal yang tersisa tidak dapat digunakan untuk mengajar mata kuliah

yang lain karena durasi waktu mengajarnya tidak cukup atau waktu mengajar yang disediakan dosen memang melebihi jumlah kelas yang diampu olehnya.

Secara struktur, tabel *siswa_jadwal_dosen* memiliki struktur yang sama dengan tabel *jadwal_dosen*.

8. Tabel Informasi Ruang Kelas

Nama Tabel : classroom

Fungsi : menyimpan informasi jadwal penggunaan kelas

Primary Key : room_id

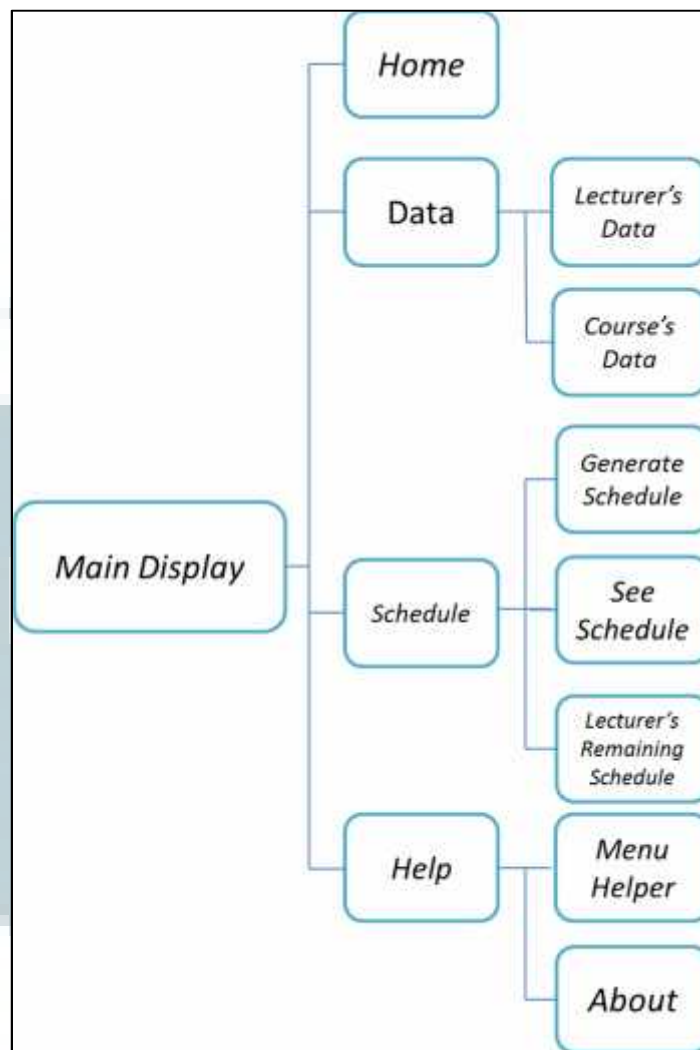
Tabel 3.8 Struktur tabel classroom

Nama field	Tipe Data	Keterangan
Room_id	VARCHAR(5)	Kode Ruang
Status_1	VARCHAR(2)	Status penggunaan ruangan kloter 1
Status_2	VARCHAR(2)	Status penggunaan ruangan kloter 2
Status_3	VARCHAR(2)	Status penggunaan ruangan kloter 3

Fungsi dari tabel 3.8 adalah menyimpan daftar ruang kelas yang terdapat pada Universitas Multimedia Nusantara, serta status penggunaannya yang terbagi atas tiga kloter penggunaan, pagi, siang dan sore.

3.4 Struktur Menu

Struktur navigasi dari aplikasi penyusun jadwal mata kuliah ini, dapat digambarkan dalam bagan gambar 3.6.



Gambar 3.6 Struktur Menu

Dari tampilan utama, aplikasi terbagi atas 4 (empat) menu utama, yaitu *Home*, *Data*, *Schedule* dan *Help*.

Menu *home* hanya berfungsi untuk membawa pengguna kembali kepada tampilan utama.

Menu *Data* terbagi kembali menjadi 2 bagian, yaitu *Lecturer's Data* dan *Course's Data*. Pada menu *lecturer's data*, pengguna akan bisa melihat daftar dosen yang terdapat pada *database*, pengguna juga diberikan beberapa fasilitas seperti fasilitas pencarian dan penyortiran. Pada menu ini, pengguna juga dapat

melakukan tambah data dosen baru, mengedit data dosen yang sudah ada, serta menghapus data dosen.

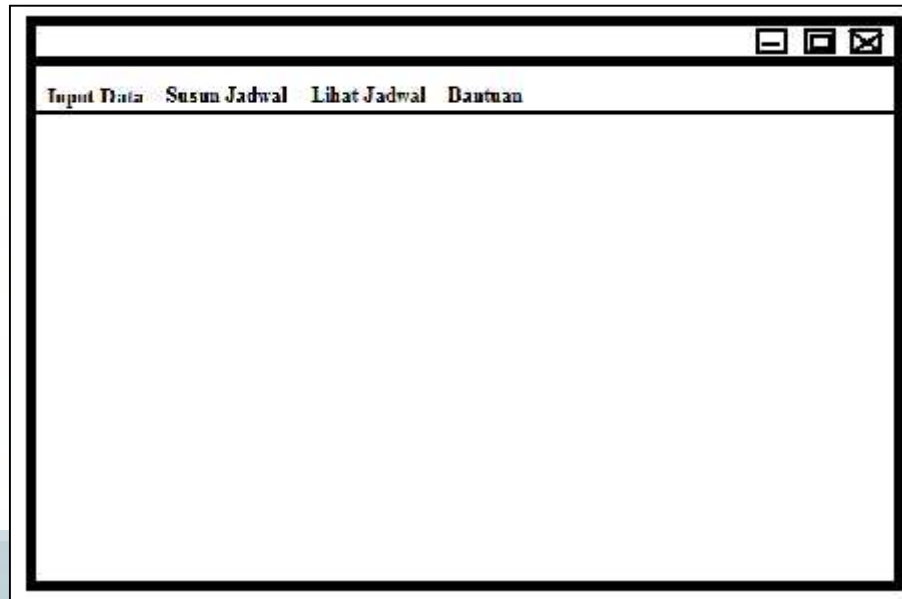
Sementara pada menu *course's data*, memiliki fungsi yang hampir serupa dengan menu *lecturer's data*, hanya saja, data yang ditampilkan adalah data mata kuliah.

Menu *schedule*, terbagi atas 3 (tiga) bagian yang memungkinkan pengguna untuk menyusun jadwal (menu *Generate Schedule*), melihat jadwal yang telah tersusun (menu *See Schedule*), serta melihat daftar dosen yang belum mendapatkan jadwal (menu *Lecturer's Remaining Schedule*). Pada sub-menu *Generate Schedule* inilah, pengguna dapat memerintahkan sistem untuk menyusun jadwal, setelah sebelumnya memilih semester mana yang akan disusun jadwalnya.

Menu terakhir, ialah menu *Help*. Terbagi atas 2 (dua) menu. *Menu Helper*, berisikan daftar keterangan serta petunjuk mengenai menu-menu yang terdapat pada aplikasi ini, sementara *About* berisikan informasi singkat mengenai aplikasi dan pengembang.

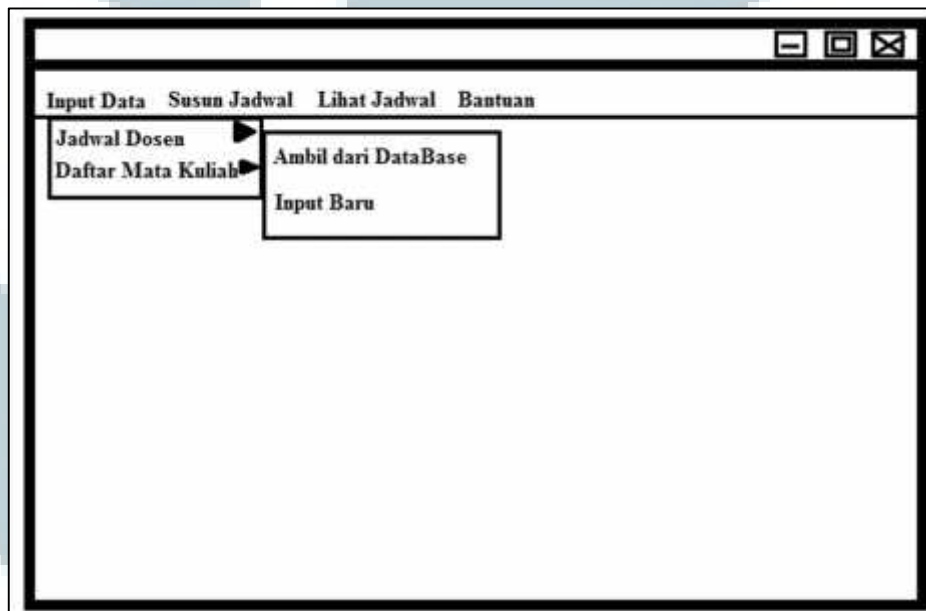
3.5 Desain Antarmuka

Sketsa desain antarmuka, dibuat sebagian acuan utama dalam pembuatan tampilan antarmuka (*user interface*) yang berperan sebagai pengakomodasi interaksi pengguna dengan sistem. Berikut adalah penjelasan mengenai sketsa desain awal tampilan antarmuka aplikasi.



Gambar 3.7 Tampilan Utama Aplikasi

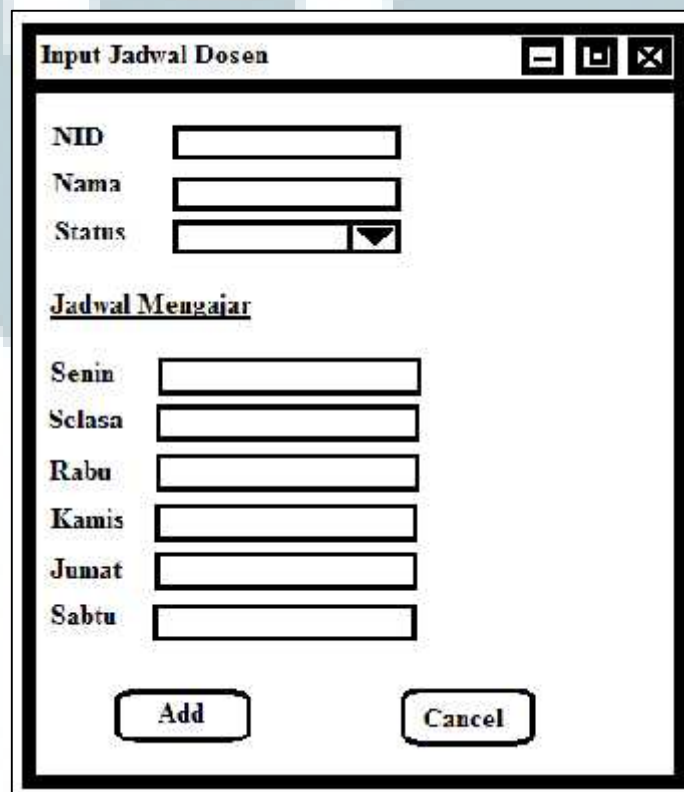
Gambar 3.7 merupakan sketsa tampilan utama aplikasi ini. Seperti yang telah dijelaskan dalam struktur menu diatas, tampilan utama memiliki empat menu utama yang memiliki fungsionalitasnya masing-masing.



Gambar 3.8 Tampilan Sub-Menu

Ketika pengguna mengklik sebuah menu, apabila menu tersebut tidak memiliki submenu, maka menu akan langsung membuka tampilan antarmuka dimana pengguna dapat melakukan interaksi dengan sistem. Tetapi, apabila menu tersebut memiliki submenu, maka menu tersebut akan membuka tampilan submenunya, hingga ke submenu terakhir. Seperti pada gambar 3.8, menu *input data* memiliki submenu paling banyak, yaitu hingga 2 level kedalaman.

Apabila menu *jadwal dosen*, *input* baru yang dipilih, maka aplikasi akan membuka tampilan seperti pada gambar 3.9.



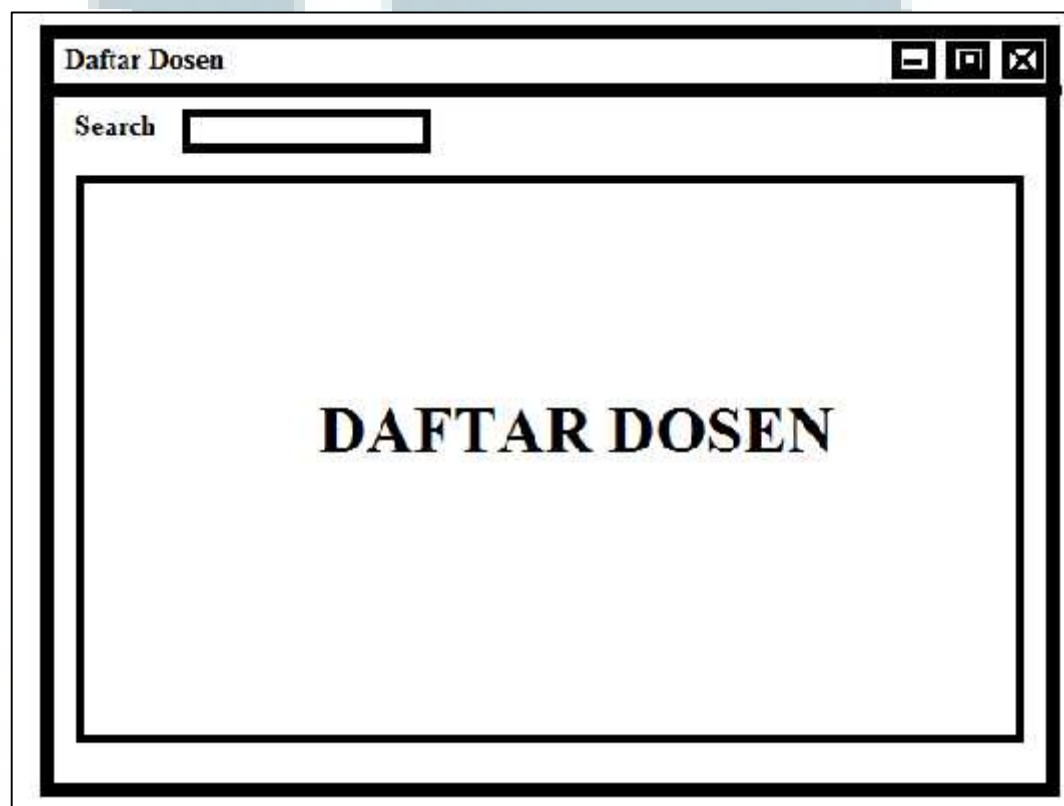
The image shows a software window titled "Input Jadwal Dosen". Inside the window, there are three input fields: "NID", "Nama", and "Status" (which is a dropdown menu). Below these fields is a section titled "Jadwal Mengajar" which contains six input fields corresponding to the days of the week: "Senin", "Selasa", "Rabu", "Kamis", "Jumat", and "Sabtu". At the bottom of the window, there are two buttons: "Add" and "Cancel".

Gambar 3.9 Input Baru Jadwal Dosen

Pada menu ini, pengguna pertama akan dapat memasukkan data ketersediaan mengajar bagi dosen yang baru. Status dosen dibuat dalam bentuk

dropdown dimana pengguna dapat memilih satu antara dua status dosen yaitu, tetap atau honorer (tidak tetap).

Ketika pengguna selesai memasukkan data yang diperlukan, maka pengguna tinggal mengklik *button add* dan sistem akan menyimpan data tersebut kedalam *database*. Sementara *button cancel* akan menutup tampilan menu *input* baru ini tanpa menyimpan data kedalam *database*.



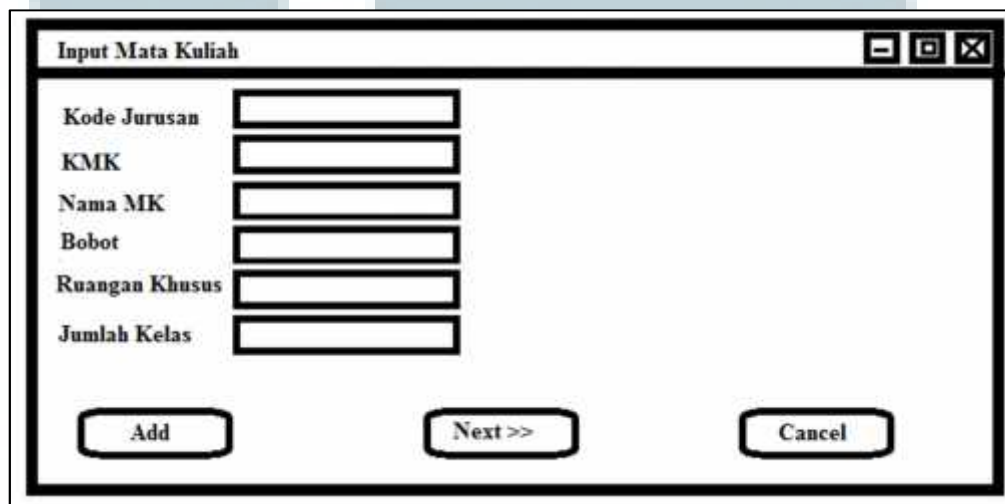
Gambar 3.10 Tampilan Daftar Dosen

Ketika pengguna memilih untuk mengambil data dari *database*, maka aplikasi akan menampilkan daftar dosen yang ada pada *database* seperti pada gambar 3.10.

Tersedia juga fasilitas pencarian berdasarkan nama, untuk memudahkan pengguna didalam mencari dosen. Jika pengguna melakukan *double* klik pada sebuah nama dosen, maka tampilan detail dari dosen tersebut akan muncul.

Pada tampilan ini, pengguna dapat mengubah detail dari data-data yang telah ada pada *database*. Tampilan detail ini, mirip dengan tampilan *input* baru data dosen, hanya saja sudah terisi akan data-data yang telah dimasukan sebelumnya.

Kemudian pada menu *input* jadwal baru mata kuliah pengguna akan diberikan tampilan pertama seperti gambar 3.11

The image shows a software window titled "Input Mata Kuliah". Inside the window, there are six labels on the left side, each followed by a text input field: "Kode Jurusan", "KMK", "Nama MK", "Bobot", "Ruangan Khusus", and "Jumlah Kelas". At the bottom of the window, there are three buttons: "Add", "Next >>", and "Cancel". The window has a standard Windows-style title bar with minimize, maximize, and close buttons.

Gambar 3.11 Tampilan *Input* Mata Kuliah Baru (1)

Pada gambar 3.11 diatas, pengguna akan diminta untuk memasukkan data-data umum sebuah mata kuliah. Setelah semua data dimasukkan, pengguna dapat menekan tombol *next>>* untuk melanjutkan ke halaman berikutnya seperti yang tampak pada gambar 3.12.

The screenshot shows a window titled "Input Mata Kuliah". It contains two text input fields labeled "NID" and "Banyak Kelas". Between these fields are two buttons labeled ">>" and "<<". To the right of these fields is a large rectangular area labeled "Daftar Dosen yang Akan Mengajar Mata Kuliah". At the bottom of the window are three buttons: "Add", "<< Prev", and "Cancel".

Gambar 3.12 Tampilan *Input* Mata Kuliah Baru (2)

Pada tampilan ini, pengguna diminta untuk mengisi dosen-dosen yang akan mengajar pada mata kuliah yang bersangkutan, termasuk jumlah kelas yang dipegang oleh masing-masing dosen pada mata kuliah yang bersangkutan.

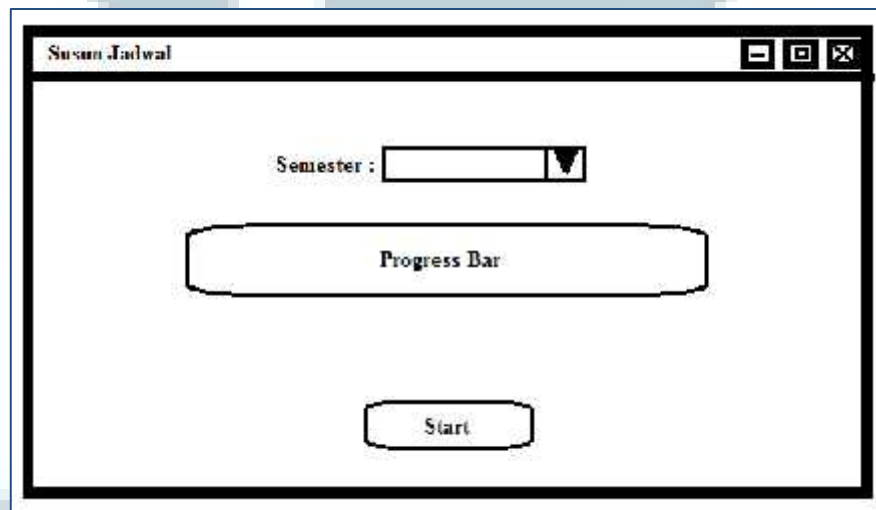
The screenshot shows a window titled "Daftar Mata Kuliah". It features a "Search" input field at the top. Below this is a large rectangular area labeled "Daftar Mata Kuliah". Underneath this area is a section titled "Dosen Pengampu" (underlined), which contains a rectangular box labeled "Daftar Dosen Pengampu MK".

Gambar 3.13 Tampilan Daftar Mata Kuliah

Gambar 3.13 diatas, adalah tampilan daftar mata kuliah yang telah terdapat di dalam *database*. Tampilan tersebut akan muncul, jika pengguna memilih menu *input* mata kuliah dengan mengambil data dari *database*. Ketika pengguna mengklik salah satu mata kuliah yang ada pada mata kuliah, maka secara otomatis data dosen pengampu mata kuliah akan muncul dibawah dalam bentuk tabel, menunjukkan daftar dosen yang mengajar bagi mata kuliah tersebut.

Terdapat juga sebuah fasilitas pencarian berdasarkan nama mata kuliah, yang dapat digunakan oleh pengguna di dalam membantu mempermudah pencarian sebuah mata kuliah tertentu.

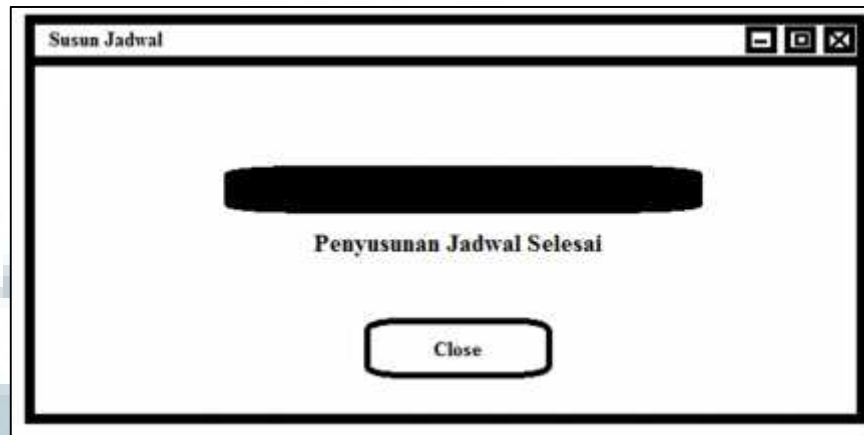
Menu Susun Jadwal tampil sederhana dengan sebuah *dropdown* menu yang memungkinkan pengguna memilih semester yang akan disusun jadwalnya, sebuah *progress bar* dan sebuah tombol untuk memulai proses penyusunan jadwal.



Gambar 3.14 Tampilan Menu Susun Jadwal

Jika pengguna telah menekan tombol *start*, maka yang bisa dilakukan pengguna adalah menunggu hingga proses penyusunan jadwal selesai. Jika proses

penyusunan jadwal telah selesai, maka tampilan antar muka menu Susun Jadwal akan berubah seperti yang terlihat pada gambar 3.15.



Gambar 3.15 Tampilan Susun Jadwal yang Telah Selesai

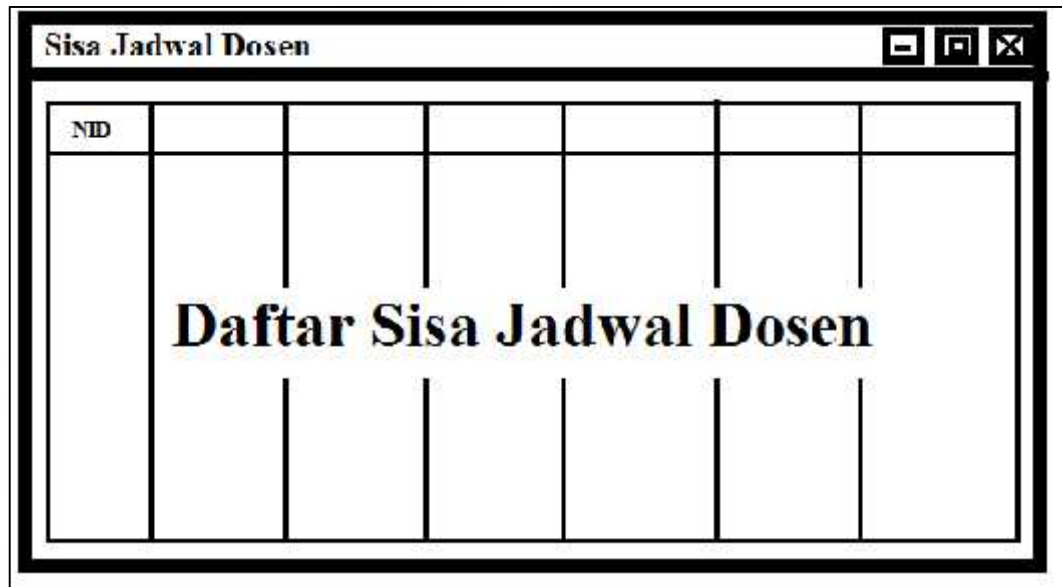
Untuk melihat jadwal yang telah selesai disusun, maka pengguna tinggal memilih menu lihat jadwal, jadwal yang telah tersusun.

 A screenshot of a software window titled "Jadwal Mata Kuliah". The window has a standard Windows-style title bar. The main content area contains a table with 5 columns. The first row is the header row with the following text: "Mata Kuliah", "Dosen", "Ruang", "Jam KBM", and "Hari". The remaining rows in the table are empty. In the center of the table, the text "DAFTAR JADWAL YANG TERSUSUN" is displayed in a large, bold, black font.

Mata Kuliah	Dosen	Ruang	Jam KBM	Hari
DAFTAR JADWAL YANG TERSUSUN				

Gambar 3.16 Tampilan Jadwal yang Telah Tersusun

Pada menu lihat jadwal, juga terdapat pilihan untuk melihat sisa jadwal dosen yang masih ada.



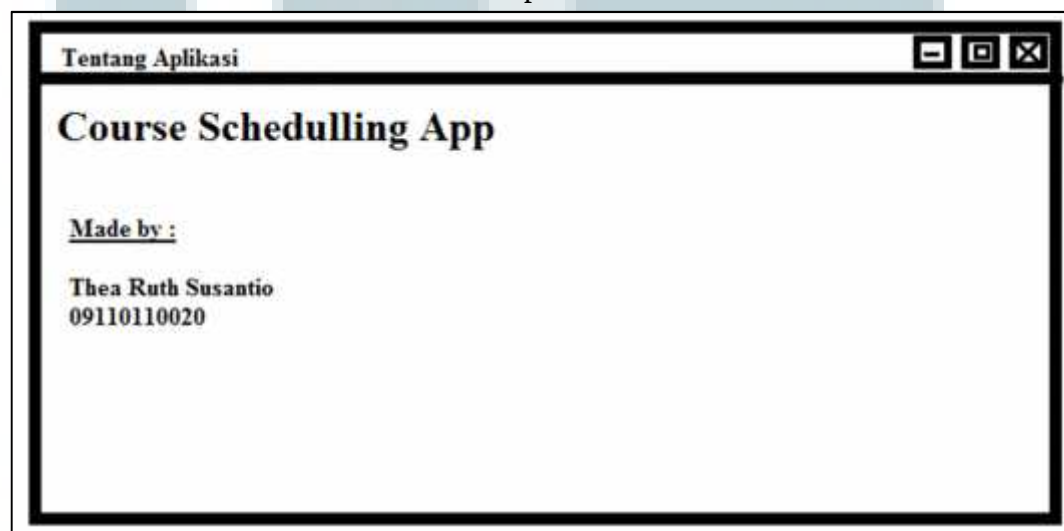
NID						

Gambar 3.17 Tampilan Daftar Sisa Jadwal Dosen

Menu terakhir ialah menu bantuan. Pada menu ini terdapat dua *sub*-menu yang dapat dipilih pengguna yaitu Bantuan Menu yang berisi informasi dan penjelasan mengenai menu-menu yang terdapat pada aplikasi, serta Tentang Aplikasi yang berisi informasi singkat mengenai pembuat aplikasi. Bentuk rancangan keduanya, dapat dilihat pada gambar 3.18 dan 3.19.



Gambar 3.18 Tampilan Bantuan Menu



Gambar 3.19 Tampilan Tentang Aplikasi

UMN