

BAB III

ANALISIS SISTEM DAN PERANCANGAN

Pada bab ini dijelaskan tentang analisis dan perancangan dari aplikasi yang akan dibuat, yaitu Rancang Bangun Aplikasi Penjadwalan Kuliah di Institut Bisnis dan Informatika Stikom Surabaya dengan Menggunakan Algoritma Genetika.

3.1 Analisis Sistem

Pada tahap analisis sistem terdapat proses mengidentifikasi serta menganalisis masalah sehingga dapat merencanakan suatu rancangan untuk menyelesaikan permasalahan yang ada. Tahapan analisis sistem terdiri dari identifikasi dan analisis permasalahan yang terkait dengan penelitian yang akan dilakukan, dan identifikasi kebutuhan perangkat lunak.

3.1.1 Tahapan Komunikasi

Pada tahap komunikasi, dilakukan proses observasi dan wawancara. Proses observasi dilakukan dengan cara mengamati secara langsung ke Bagian AAK yang bertujuan untuk mengetahui informasi tentang proses penjadwalan kuliah. Pada proses wawancara dilakukan dengan cara melakukan proses tanya jawab kepada Kepala Bagian AAK yang berfungsi untuk mencocokkan data dan informasi dari hasil observasi. Selain itu proses wawancara juga berfungsi untuk menanyakan beberapa hal yang tidak didapat dari hasil observasi. Setelah melakukan observasi dan wawancara langsung dengan pihak yang bersangkutan

maka dapat disusun analisis bisnis, analisis kebutuhan pengguna dan analisis kebutuhan fungsional

Pada tahap komunikasi, dilakukan dengan cara wawancara atau tanya jawab kepada bagian AAK Stikom Surabaya untuk mencari informasi. Kegiatan ini dilakukan secara berulang-ulang sampai beberapa kali untuk mendapatkan informasi tentang proses penjadwalan matakuliah yang dilakukan di Stikom Surabaya. Macam-macam wawancara yang dilakukan yaitu pertanyaan ruang lingkup penjadwalan, dan pertanyaan proses pembuatan jadwal mata kuliah yang dilakukan selama ini.

Hasil dari wawancara tersebut menghasilkan informasi diantaranya syarat-syarat dalam menyusun jadwal perkuliahan, faktor-faktor yang mempengaruhi dalam penyusunan jadwal mata kuliah, permasalahan dan solusi yang dibutuhkan oleh bagian AAK Stikom Surabaya. Hasil dari proses wawancara ini berupa informasi secara teknis kebutuhan aplikasi yang akan dibangun, maka dapat disusun analisis bisnis, analisis kebutuhan pengguna dan analisis kebutuhan fungsional

A Analisis Bisnis

Setelah melakukan tahapan komunikasi untuk selanjutnya dilakukan analisis bisnis meliputi identifikasi masalah, identifikasi pengguna identifikasi data, identifikasi fungsi.

1. Identifikasi Masalah

Setelah melakukan wawancara langsung dengan pihak AAK Stikom Surabaya, terdapat permasalahan yang muncul yaitu mengenai proses membuat jadwal kuliah di Stikom Surabaya. Pada saat akan memasuki semester ajaran baru

bagian AAK Stikom Surabaya harus membuat dan menyusun jadwal kuliah yang yang harus disiapkan sebelum perwalian berlangsung. Proses ini dimulai dari bagian AAK Stikom menanyakan mata kuliah yang akan diselenggarakan pada setiap prodi, kemudian prodi menentukan mata kuliah yang akan diselenggarakan pada semester yang akan dijadwalkan dan memberikannya ke bagian AAK. Bagian AAK menerima informasi mata kuliah yang akan diselenggarakan dari masing-masing prodi, bagian AAK akan menentukan jumlah yang kelas yang akan dibuka pada setiap mata kuliah dengan berdasarkan data perkiraan jumlah peserta mata kuliah setiap mata kuliah. Bagian AAK menyusun daftar mata kuliah dengan berdasarkan data yang ada yaitu seperti data kompetensi dosen, beban dosen mengajar, syarat dan kebijakan mata kuliah. Kemudian dengan menggunakan data tersebut, maka akan disusun jadwal kuliah sampai menghasilkan jadwal perkuliahan yang terbaik. Gambar 3.1 menunjukkan proses membuat jadwal yang dilakukan bagian AAK dan prodi.

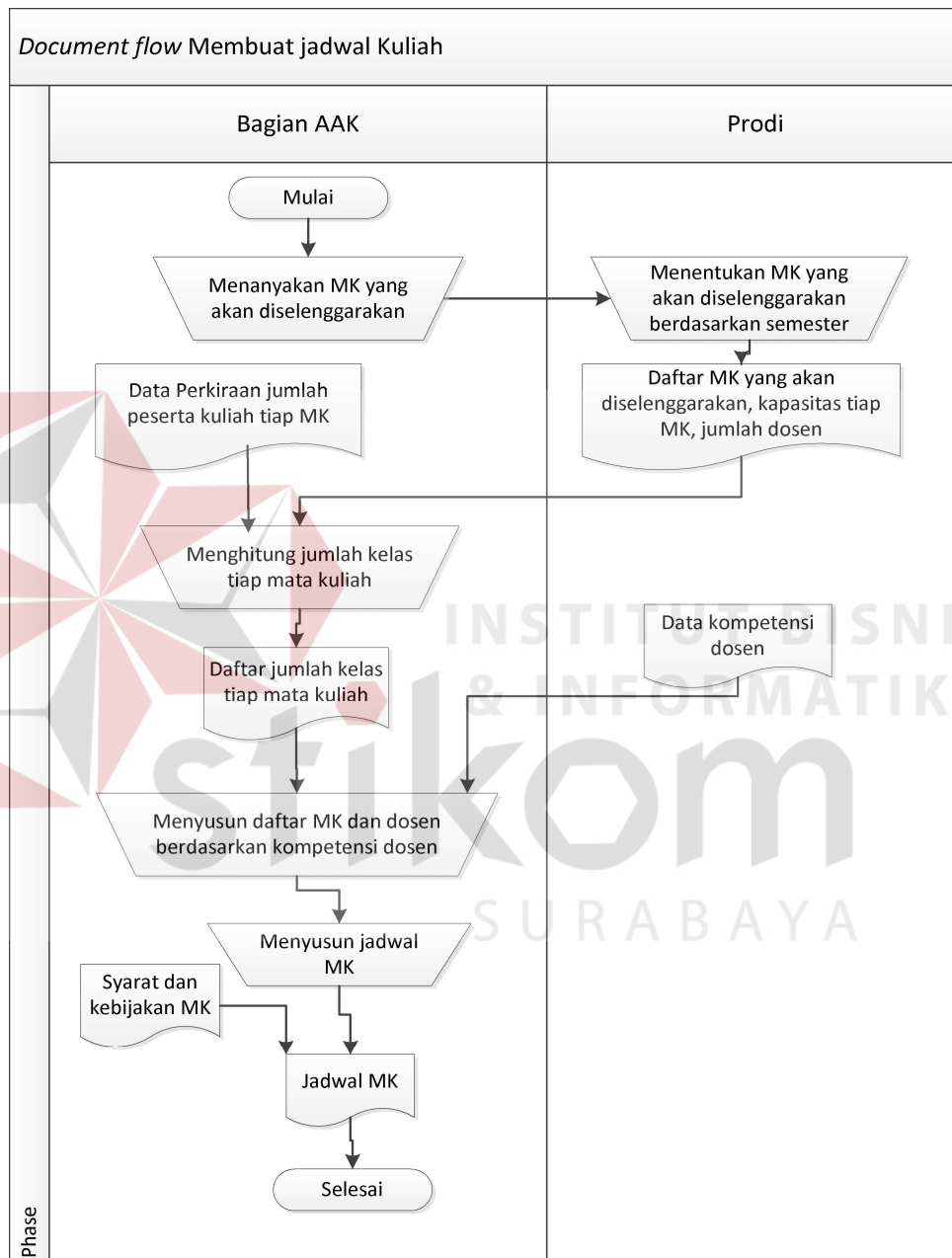
2. Identifikasi Pengguna

Berdasarkan hasil wawancara dengan bagian AAK Stikom Surabaya yang *user* yang terlibat melakukan proses pembuatan jadwal yaitu bagian AAK dan kepala program studi.

3. Identifikasi Data

Setelah dilakukan proses identifikasi permasalahan dan pengguna, maka dapat dilakukan identifikasi data. Pada proses pembuatan dan penyusunan jadwal matakuliah memerlukan data sebagai berikut: Data MK yang diselenggarakan, data perkiraan jumlah peserta MK, data kapasitas setiap MK, data sebaran

waktu yang digunakan ,data kompetensi dosen, data beban dosen , dan data ruang digunakan.



Gambar 3.1 *Document flow* proses membuat jadwal

4. Identifikasi Fungsi

Setelah dilakukan proses identifikasi permasalahan, pengguna dan data, maka dapat diidentifikasi fungsi dari proses penjadwalan matakuliah sebagai berikut: menentukan matakuliah yang diselenggarakan, menentukan kompetensi dosen, menentukan ruangan yang digunakan, menentukan sebaran waktu, memproses jadwal, dan menghasilkan laporan jadwal kuliah dalam satu semester

B Analisis Kebutuhan Pengguna

Berdasarkan hasil wawancara dengan pengguna yang bersangkutan dengan sistem, maka dapat dibuat kebutuhan pengguna. Analisis kebutuhan pengguna berfungsi untuk mengetahui kebutuhan dari masing-masing pengguna yang berhubungan langsung dengan sistem yang dibuat dapat sesuai dengan apa yang diminta oleh pengguna yang bersangkutan dengan sistem. Pengguna tersebut terbagi menjadi *role* yang berbeda, yaitu sebagai admin sekaligus memproses jadwal matakuliah (Kepala Bagian AAK) dan pengguna yang menentukan informasi data penjadwalan yang akan dibuat (Kepala Program Studi). Dari *role* yang berbeda tersebut maka memiliki fungsi yang berbeda sehingga kebutuhan data dan informasi yang dihasilkan pada setiap *user*. Untuk penerapan pada aplikasi dapat dilihat kebutuhan pengguna dalam pembuatan jadwal matakuliah sebagai berikut:

1. Kepala Bagian AAK

Tabel 3.1 Kebutuhan Pengguna Kepala Bagian AAK

Kebutuhan Fungsi	Kebutuhan Data	Kebutuhan Informasi
Menentukan ruangan yang digunakan	Data ruang yang sudah tersedia	Informasi data ruangan yang akan digunakan
Menentukan sebaran waktu	1. Data hari 2. Data jam	Informasi sebaran waktu yang akan dijadwalkan
Memproses pecah kelas	1. Data matakuliah diselenggarakan 2. Data kapasitas setiap matakuliah	Informasi jumlah kuliah yang akan dijadwalkan
Memproses jadwal kuliah	1. Data kuliah 2. Data kompetensi dosen 3. Data ruang dipilih 4. Data sebaran waktu 5. Data jumlah prediksi peserta setiap matakuliah	Informasi jadwal matakuliah
Laporan jadwal kuliah dalam satu semester	Data jadwal dalam seminggu	Laporan jadwal matakuliah yang terbaik

2. Ketua Program Studi

Tabel 3.2 Kebutuhan Pengguna Ketua Program Studi

Kebutuhan Fungsi	Kebutuhan Data	Kebutuhan Informasi
Menentukan matakuliah yang akan diselenggarakan	Data matakuliah	Informasi data matakuliah setiap prodi yang akan diselenggarakan dalam satu semester
Menentukan kompetensi dosen	1. Data dosen/karyawan 2. Data matakuliah	Informasi data kompetensi dosen setiap prodi
Menentukan kapasitas setiap matakuliah	1. Data matakuliah 2. Data kapasitas matakuliah	Informasi data kapasitas setiap matakuliah

C Analisis Kebutuhan Fungsional

Pada tahap kebutuhan fungsional digunakan untuk mengimplementasikan seluruh fungsi yang didapatkan dari hasil analisis kebutuhan pengguna yang terjadi saat ini. Fungsi- fungsi tersebut dapat dibagi menjadi delapan fungsi yang meliputi sebagai berikut :

1. Fungsi menentukan matakuliah yang akan diselenggarakan

Tabel 3.3 *Software Requirement* Menentukan Matakuliah Diselenggarakan

Nama Fungsi	Fungsi Menentukan Matakuliah yang Diselenggarakan	
Stakeholder	Ketua Program Studi	
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk menentukan matakuliah apa saja yang akan diselenggarakan oleh Ketua Program Studi.	
Kondisi Awal	1. Data matakuliah	
Alur Normal	Aksi Stakeholder	Respon Sistem
	Otentifikasi Login Stakeholder	
	Pengguna memasukkan <i>username</i> dan <i>password</i>	a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Ketua Program Studi b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa "Username/ Password Anda Salah" c. Sistem akan menampilkan "login berhasil" jika <i>username</i> dan <i>password</i> yang di inputkan benar. d. Jika Ketua Program Studi berhasil <i>login</i> , maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Ketua Program Studi
	Aksi Stakeholder	Respon Sistem
	Menentukan Matakuliah yang diselenggarakan	
1. Pengguna memilih menu matakuliah untuk menentukan matakuliah yang sudah tersedia	Sistem akan menampilkan halaman inputan untuk memilih semester akan dijadwalkan yang kemudian diisi oleh pengguna sebagai tahap awal penentuan matakuliah yang diselenggarakan dan selanjutnya	

Alur Normal	Aksi Stakeholder	Respon Sistem
	Menentukan Matakuliah yang diselenggarakan	
		sistem akan menampilkan halaman data matakuliah sesuai semester yang dipilih.
	2. Pengguna memilih matakuliah yang akan diselenggarakan	Sistem menyimpan data tersebut dan jika kolom yang diisikan sesuai dan benar, maka sistem akan menampilkan “Data berhasil disimpan”.
Kondisi Akhir	Fungsi ini menyimpan data matakuliah yang diselenggarakan	

2. Fungsi menentukan kompetensi dosen

Tabel 3.4 *Software Requirement* Menentukan Kompetensi Dosen

Nama Fungsi	Fungsi Menentukan Kompetensi dosen	
Stakeholder	Ketua Program Studi	
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk menentukan kompetensi setiap dosen oleh Ketua Program Studi.	
Kondisi Awal	1. Data matakuliah 2. Data dosen	
Alur Normal	Aksi Stakeholder	Respon Sistem
	Otentifikasi Login Stakeholder	
	Pengguna memasukkan <i>username</i> dan <i>password</i>	a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Ketua Program Studi b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa “Username/ Password Anda Salah” c. Sistem akan menampilkan “login berhasil” jika <i>username</i> dan <i>password</i> yang di inputkan benar. d. Jika Ketua Program Studi berhasil <i>login</i> , maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Ketua Program Studi

Alur Normal	Aksi <i>Stakeholder</i>	Respon Sistem
	Menentukan Kompetensi dosen	
	1. Pengguna memilih menu dosen untuk menentukan kompetensi setiap dosen.	Sistem akan menampilkan halaman inputan untuk memilih dosen berdasarkan kompetensi yang dimilikinya dan selanjutnya sistem akan menampilkan halaman data matakuliah beserta dengan dosen yang berkompeten dibidangnya.
	2. Pengguna memilih matakuliah yang akan ditentukan dosennya	Sistem menyimpan data tersebut dan jika kolom yang diisikan sesuai dan benar, maka sistem akan menampilkan “Data berhasil disimpan”.
Kondisi Akhir	Fungsi ini menyimpan data kompetensi dosen	

3. Fungsi menentukan kapasitas setiap matakuliah

Tabel 3.5 *Software Requirement* Menentukan Kapasitas Setiap Matakuliah

Nama Fungsi	Fungsi Menentukan Kapasitas Setiap Kelas Matakuliah	
Stakeholder	Ketua Program Studi	
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk menentukan jumlah kapasitas setiap kelas per matakuliah oleh Ketua Program Studi.	
Kondisi Awal	1. Data matakuliah yang diselenggarakan 2. Data kapasitas matakuliah	
Alur Normal	Aksi <i>Stakeholder</i>	Respon Sistem
	Otentifikasi Login <i>Stakeholder</i>	
	Pengguna memasukkan <i>username</i> dan <i>password</i>	a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Ketua Program Studi. b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa “Username/Password Anda Salah” c. Sistem akan menampilkan “login berhasil” jika <i>username</i> dan <i>password</i> yang di inputkan benar. d. Jika Ketua Program Studi berhasil <i>login</i> , maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Ketua Program Studi

Alur Normal	Aksi Stakeholder	Respon Sistem
	Menentukan Kapasitas Setiap Kelas Matakuliah	
	1. Pengguna memilih menu dosen untuk menentukan kompetensi setiap dosen.	Sistem akan menampilkan halaman data matakuliah yang diselenggarakan dan masukan untuk memasukkan data kapasitas
	2. Pengguna memilih matakuliah yang akan ditentukan kapasitas	Sistem menyimpan data tersebut dan jika kolom yang diisikan sesuai dan benar, maka sistem akan menampilkan “Data berhasil disimpan”.
Kondisi Akhir	Fungsi ini menyimpan data kapasitas setiap matakuliah yang diselenggarakan	

4. Fungsi menentukan ruang yang digunakan

Tabel 3.6 *Software Requirement* Menentukan Ruang yang digunakan

Nama Fungsi	Fungsi Menentukan Ruang yang Digunakan	
Stakeholder	Kepala Bagian AAK	
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk menentukan ruangan mana saja yang akan digunakan oleh Kepala Bagian AAK.	
Kondisi Awal	1. Data ruang	
Alur Normal	Aksi Stakeholder	Respon Sistem
	Otentifikasi Login Stakeholder	
	Pengguna memasukkan <i>username</i> dan <i>password</i>	<ul style="list-style-type: none"> a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Kepala Bagian AAK b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa “Username/ Password Anda Salah” c. Sistem akan menampilkan “login berhasil” jika <i>username</i> dan <i>password</i> yang di inputkan benar. d. Jika Kepala Bagian AAK berhasil <i>login</i>, maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Kepala Bagian AAK

	Aksi Stakeholder	Respon Sistem
	Menentukan Ruang yang Digunakan	
Alur Normal	1. Pengguna memilih menu ruangan untuk menentukan ruangan yang digunakan	Sistem akan menampilkan halaman inputan untuk memilih ruangan yang akan digunakan dan selanjutnya sistem akan menampilkan halaman data ruangan yang tersedia.
	2. Pengguna memilih ruangan yang akan digunakan	Sistem menyimpan data tersebut dan jika kolom yang diisi sesuai dan benar, maka sistem akan menampilkan "Data berhasil disimpan".
Kondisi Akhir	Fungsi ini menyimpan data ruangan yang digunakan	

5. Fungsi menentukan sebaran waktu

Tabel 3.7 Software Requirement Menentukan Sebaran Waktu

Nama Fungsi	Fungsi Menentukan Sebaran Waktu	
Stakeholder	Kepala Bagian AAK	
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk menentukan waktu yang akan digunakan untuk penjadwalan oleh Kepala Bagian AAK.	
Kondisi Awal	1. Data hari 2. Data jam	
Alur Normal	Aksi Stakeholder	Respon Sistem
	Otentifikasi Login Stakeholder	
	Pengguna memasukkan <i>username</i> dan <i>password</i>	a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Kepala Bagian AAK b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa "Username/ Password Anda Salah" c. Sistem akan menampilkan "login berhasil" jika <i>username</i> dan <i>password</i> yang di inputkan benar. d. Jika Kepala Bagian AAK berhasil <i>login</i> , maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Kepala Bagian AAK

Alur Normal	Aksi <i>Stakeholder</i>	Respon Sistem
	Menentukan Sebaran Waktu	
	1. Pengguna memilih menu waktu untuk menentukan waktu yang akan digunakan untuk penjadwalan	Sistem akan menampilkan halaman inputan untuk memilih kombinasi hari dan jam sesuai dengan kebijakan yang sudah ditentukan
	2. Pengguna memilih hari dan jam sesuai kebijakan	Sistem menyimpan data tersebut dan jika kolom yang diisikan sesuai dan benar, maka sistem akan menampilkan “Data berhasil disimpan”.
Kondisi Akhir	Fungsi ini menyimpan data sebaran waktu	

6. Fungsi memproses pecah kelas

Tabel 3.8 *Software Requirement* Memproses Pecah Kelas

Nama Fungsi	Fungsi Memproses Pecah Kelas	
Stakeholder	Kepala Bagian AAK	
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk memproses pecah kelas setiap matakuliah oleh Kepala Bagian AAK.	
Kondisi Awal	1. Data matakuliah diselenggarakan 2. Data jumlah peserta setiap matakuliah	
Alur Normal	Aksi <i>Stakeholder</i>	Respon Sistem
	Otentifikasi Login <i>Stakeholder</i>	
	Pengguna memasukkan <i>username</i> dan <i>password</i>	<ol style="list-style-type: none"> a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Kepala Bagian AAK b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa “Username/ Password Anda Salah” c. Sistem akan menampilkan “login berhasil” jika <i>username</i> dan <i>password</i> yang di inputkan benar. d. Jika Kepala Bagian AAK berhasil <i>login</i>, maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Kepala Bagian AAK

	Aksi Stakeholder	Respon Sistem
Alur Normal	Memproses Pecah Kelas	
	1. Pengguna memilih menu proses pecah kelas	Sistem akan menampilkan halaman inputan untuk memproses pecah kelas berdasarkan jumlah peserta dan kapasitas setiap matakuliah.
	2. Pengguna memilih tombol pecah kelas	Sistem akan memproses pecah kelas dengan membagi jumlah peserta setiap matakuliah dengan kapasitas setiap matakuliah dan menyimpan data tersebut, jika sudah sesuai dan benar, maka sistem akan menampilkan “Data berhasil disimpan”.
Kondisi Akhir	Fungsi ini memproses pecah kelas setiap matakuliah menjadi beberapa kelas kuliah. Berdasarkan jumlah peserta dan jumlah kapasitas setiap matakuliah	

7. Fungsi memproses jadwal

Tabel 3.9 *Software Requirement* Memproses Jadwal Menggunakan Algoritma Genetika

Nama Fungsi	Fungsi Memproses Jadwal	
Stakeholder	Kepala Bagian AAK	
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk memproses jadwal matakuliah oleh Kepala Bagian AAK.	
Kondisi Awal	<ol style="list-style-type: none"> 1. Data mk diselenggarakan 2. Data kompetensi dosen 3. Data ruangan digunakan 4. Data sebaran waktu 	
Alur Normal	Aksi Stakeholder	Respon Sistem
	Otentifikasi Login Stakeholder	
	Pengguna memasukkan <i>username</i> dan <i>password</i>	<ol style="list-style-type: none"> a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Kepala Bagian AAK b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa “Username/ Password Anda Salah” c. Sistem akan menampilkan “login berhasil” jika <i>username</i> dan <i>password</i> yang di inputkan benar.

Alur Normal	Aksi Stakeholder	Respon Sistem
	Otentifikasi Login Stakeholder	
		d. Jika Kepala Bagian AAK berhasil <i>login</i> , maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Kepala Bagian AAK
	Aksi Stakeholder	Respon Sistem
	Memproses Jadwal	
	1. Pengguna memilih menu proses jadwal	Sistem akan menampilkan halaman inputan untuk memproses jadwal kuliah berdasarkan data yang sudah ada.
	2. Pengguna memilih <i>generate</i> jadwal	Sistem akan memproses jadwal menggunakan algoritma genetika dan menyimpan data tersebut, jika sudah sesuai dan benar, maka sistem akan menampilkan “Data berhasil disimpan”.
3. Pengguna memilih tombol Lihat Jadwal	Sistem akan menampilkan hasil jadwal yang telah diproses dengan menggunakan algoritma genetika. Dan untuk selanjutnya menentukan dosen untuk setiap kelas yang sudah dijadwalkan, berdasarkan data kompetensi dosen yang sudah ada.	
Kondisi Akhir	Fungsi ini memproses jadwal, menentukan dosen setiap kelas dan menyimpan hasil jadwal	

8. Fungsi menghasilkan laporan

Tabel 3.10 *Software Requirement* Menghasilkan Laporan

Nama Fungsi	Fungsi Menghasilkan Laporan / Jadwal
Stakeholder	Kepala Bagian AAK
Deskripsi	Fungsi ini merupakan kegiatan yang dilakukan untuk melihat laporan hasil penjadwalan kuliah yang dilakukan
Kondisi Awal	<ol style="list-style-type: none"> 1. Data mk diselenggarakan 2. Data kompetensi dosen 3. Data ruangan digunakan 4. Data sebaran waktu 5. Data jadwal

Alur Normal	Aksi Stakeholder	Respon Sistem
		Otentifikasi Login Stakeholder
	Pengguna memasukkan <i>username</i> dan <i>password</i>	a. Sistem akan melakukan pengecekan <i>username</i> dan <i>password</i> Kepala Bagian AAK b. Jika <i>username</i> dan <i>password</i> tidak sesuai, maka sistem akan menampilkan informasi berupa "Username/ Password Anda Salah" c. Sistem akan menampilkan "login berhasil" jika <i>username</i> dan <i>password</i> yang di inputkan benar. d. Jika Kepala Bagian AAK berhasil <i>login</i> , maka sistem akan memberikan hak akses <i>login</i> dan menu sebagai Kepala Bagian AAK
	Aksi Stakeholder	Respon Sistem
	Menghasilkan Laporan / Jadwal	
	1. Pengguna memilih menu lihat jadwal.	Sistem akan menampilkan menu laporan jadwal dalam satu minggu untuk satu semester
	2. Pengguna memilih menu <i>export to excel</i> .	Sistem akan menampilkan laporan hasil jadwal dalam satu minggu untuk satu semester dalam bentuk format <i>Microsoft excel</i> .
Alur Normal		
Kondisi Akhir	Fungsi ini menghasilkan laporan terkait dengan hasil jadwal kuliah yang sudah diproses.	

3.1.2 Perencanaan Kebutuhan Sistem

Perangkat lunak dan perangkat keras yang dibutuhkan akan dijelaskan lebih rinci dan lebih mengarah kepada hal-hal teknis. Kebutuhan ini dibedakan menjadi dua yaitu untuk komputer *server* dan komputer *client*, untuk lebih jelasnya dapat dilihat pada tabel 3.11.

Berikut *user*/pengguna dari perangkat lunak:

1. *User1* adalah Bagian AAK, merupakan pengguna yang memproses pembuatan jadwal mata kuliah.
2. *User2* adalah Bagian Prodi, merupakan pengguna yang menentukan informasi data penjadwalan yang akan dibuat.

Tabel 3.11 Kebutuhan Aplikasi

No	Analisis Kebutuhan	Perangkat yang dibutuhkan
1.	Kebutuhan Perangkat Lunak	Kebutuhan ntuk <i>server</i> - Sistem Operasi server (Windows server) - (Apache (XAMPP) - Web Browser (Mozilla Firefox, Google Chrome) - Database (Oracle XE) - Microsoft Office Excel Kebutuhan untuk <i>client</i> - Sistem operasi (windows 7) - Web Browser (Mozilla Firefox, Google Chrome)
2.	Kebutuhan Perangkat Keras	- <i>Personal Computer (PC)</i> untuk <i>server</i> dengan <i>requirement Pentium Dual Core 2.2 GHz</i> dan dilengkapi dengan <i>Local Area Network (LAN) card</i> . - Modem penghubung <i>internet</i> yang dihubungkan ke komputer admin atau <i>router</i> , sehingga dapat dihubungkan ke komputer <i>client</i> . - Untuk penghubung dari komputer ke komputer dibutuhkan kabel jaringan atau kabel UTP yang telah terpasang RJ-45. - Untuk komputer <i>client</i> dapat menggunakan <i>notebook</i> ataupun <i>personal computer (PC)</i> dengan <i>requirement Pentium Dual Core 2.2 GHz</i> dan dilengkapi dengan hub <i>Local Area Network (LAN)</i> yang aktif

Dari hasil analisis diatas maka dapat diketahui kebutuhan aplikasi yang mendukung adalah sebagai berikut :

1. Perangkat lunak berbasis web
2. Database menggunakan Oracle
3. Perangkat lunak mampu melakukan proses penginputan, pengeditan, penyimpanan, pencarian, dan pencetakan data
4. Memiliki *interface* yang *user friendly*
5. Menjamin keamanan data yang tersimpan

3.2 Perancangan Sistem (*Modelling*)

Berdasarkan analisis sistem dari permasalahan yang ada, selanjutnya akan dibuatkan desain dari sistem tersebut. Tujuan dari desain sistem ini adalah untuk membuat kerangka dasar dalam melakukan implementasi ke sistem yang akan dibuat. Perancangan sistem ini terdiri dari perancangan proses, perancangan data, perancangan *interface* dan perancangan pengujian.

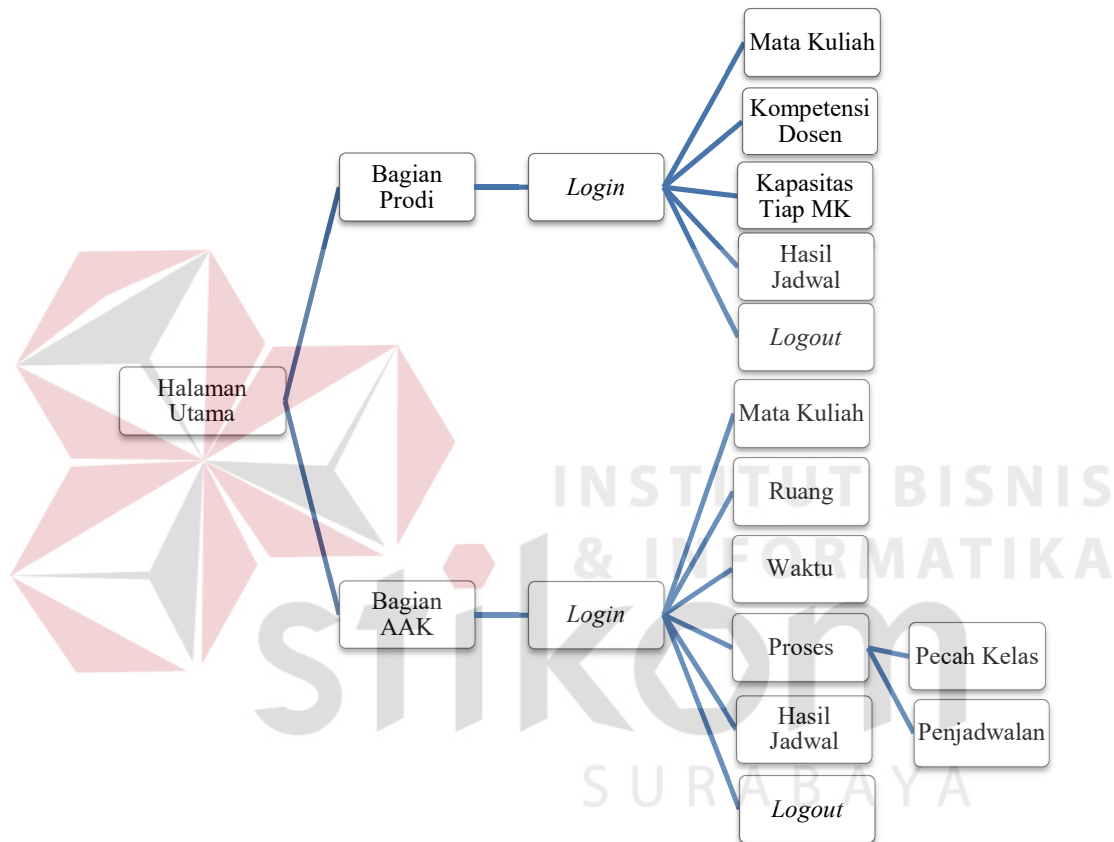
3.2.1 Perancangan Proses

Pada tahap perancangan proses akan dibahas setiap proses yang telah dibuat untuk merancang dan membangun sistem yang dibuat. Tahapan perancangan proses ini terdiri dari beberapa tahapan proses yaitu : *sitemap*, *context diagram*, diagram jenjang dan *data flow diagram*.

A *Sitemap*

Sitemap adalah salah satu alat bantu untuk mempermudah dalam pengenalan peta situs pada sebuah website. *Sitemap* akan membantu mempermudah menjelaskan aplikasi penjadwalan kuliah berbasis web pada penelitian ini. Pada aplikasi penjadwalan kuliah yang seperti pada gambar 3.2 terdiri dari dua tampilan *user*. Dalam *sitemap* sudah digambarkan bahwa masuk

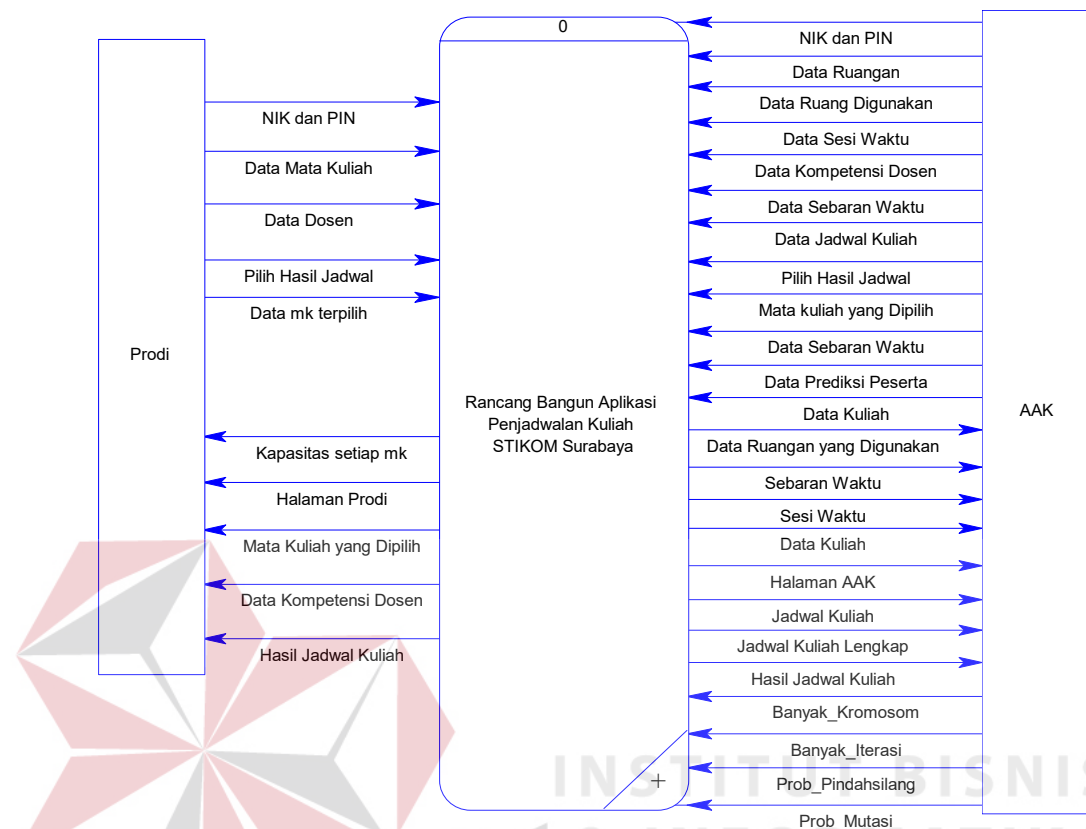
sebagai ketua prodi hanya dapat mengakses beberapa menu yaitu matakuliah disielenggarakan, kompetensi dosen, kapasitas dan hasil jadwal. Sedangkan menu yang dapat diakses oleh bagan AAK yaitu mata kuliah, Ruang, waktu proses, dan hasil jadwal. *Sitemap* aplikasi penjadwalan kuliah dapat dilihat pada Gambar 3.2.



Gambar 3.2 *Sitemap* Aplikasi Penjadwalan Kuliah

B Context Diagram

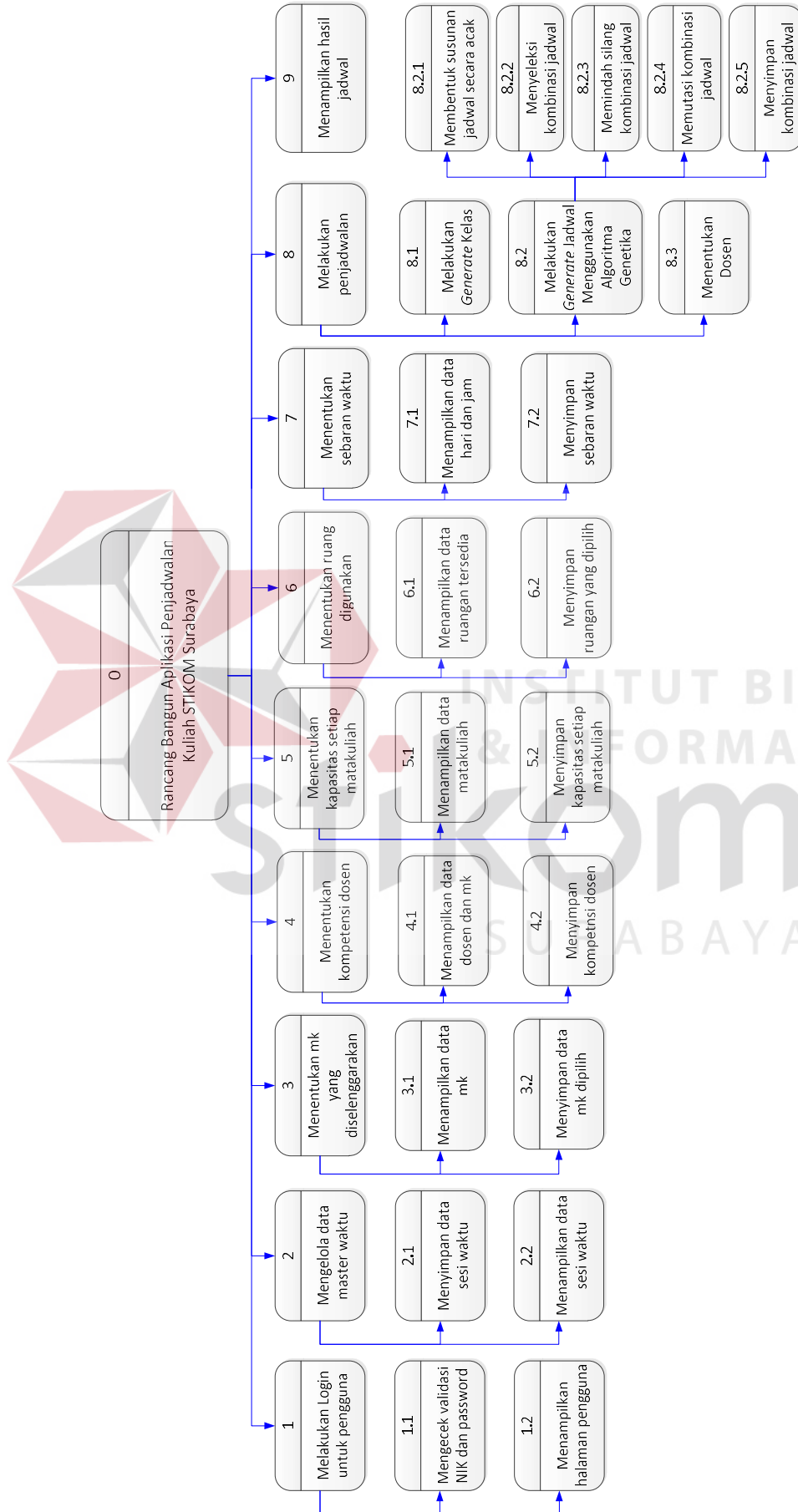
Context diagram adalah gambaran menyeluruh dari DFD. Di dalam *Context Diagram* aplikasi penjadwalan kuliah terdapat dua *External Entity*, yang terdiri atas Bagian Prodi, dan Bagian AAK. Untuk lebih jelasnya dapat melihat Gambar 3.3.



Gambar 3.3 *Contex Diagram* Aplikasi Penjadwalan Kuliah

C Diagram Jenjang

Diagram jenjang adalah sebuah bagan yang menggambarkan proses-proses yang ada dan mendukung suatu aplikasi. Diagram jenjang dari aplikasi penjadwalan kuliah terdapat delapan proses besar yaitu mengecek hak akses login, mengelola data master waktu, menentukan matakuliah yang diselenggarakan, menentukan kompetensi dosen, menentukan ruangan yang digunakan, menentukan sebaran waktu, memproses jadwal, dan mengasilkkan laporan jadwal kuliah dalam satu semester. Diagram jenjang ini digunakan sebagai pedoman dalam pembuatan *Data Flow Diagram*. Gambar 3.4 menggambarkan diagram jenjang proses aplikasi penjadwalan kuliah.

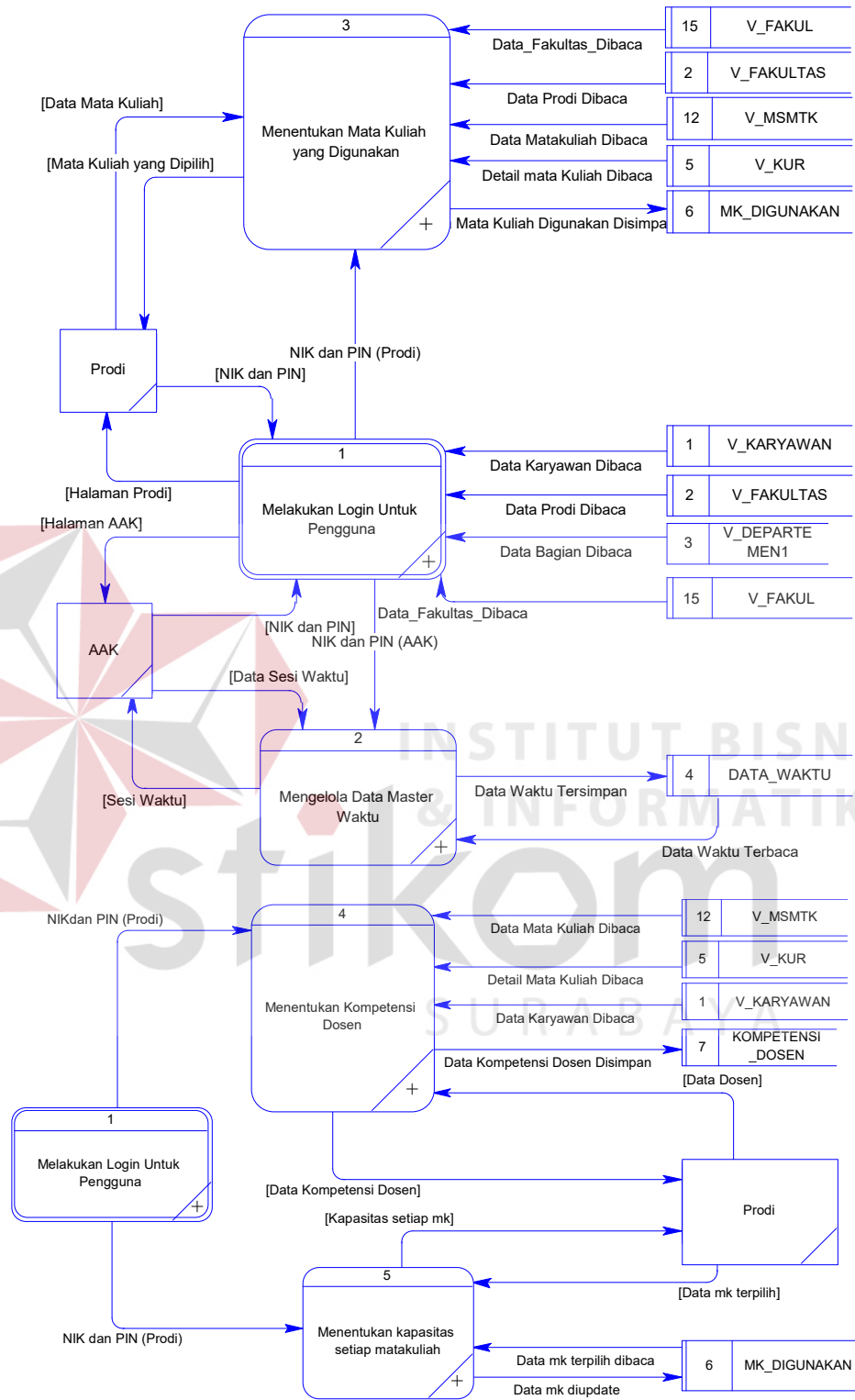


Gambar 3.4 Diagram Jenang Aplikasi penjadwalan kuliah

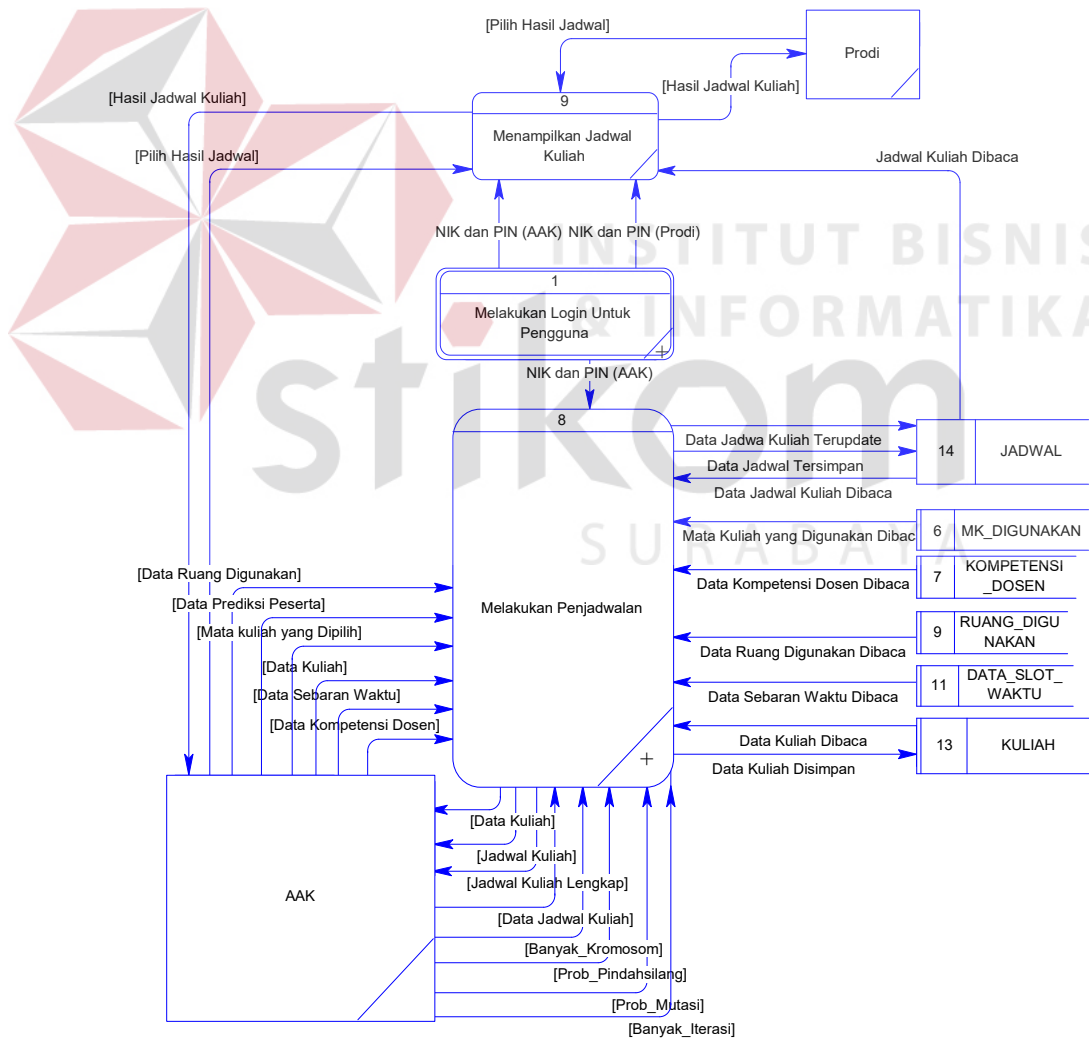
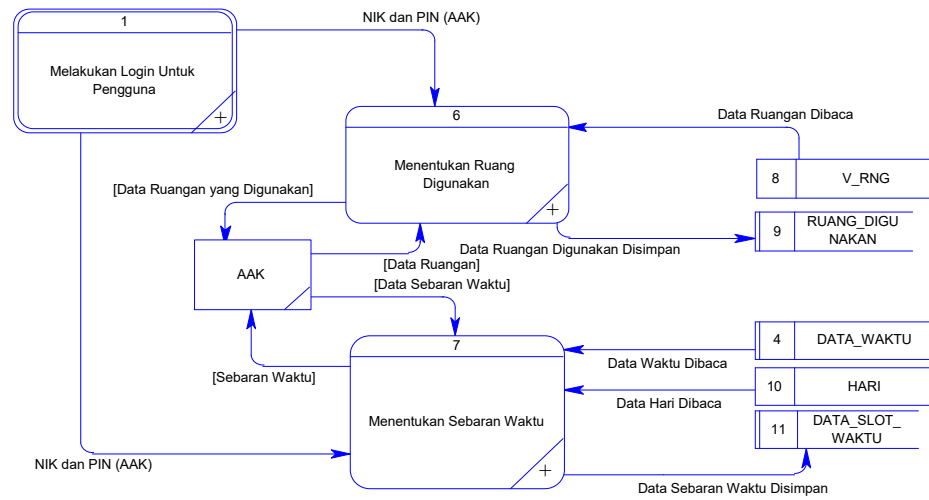
D Data Flow Diagram

Data Flow Diagram (DFD) merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured analysis and design*). DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir. Pada DFD, akan dijelaskan mengenai aliran data yang terdapat dalam sistem. Semua masukan dan keluaran dari sistem akan digambarkan dengan jelas. Pada DFD aplikasi penjadwalan kuliah ini terdapat sembilan proses yaitu melakukan *login* untuk pengguna, mengelola master waktu, menentukan mata kuliah yang diselenggarakan, menentukan kompetensi dosen, menentukan kapasitas setiap matakuliah, menentukan ruang digunakan, menentukan sebaran waktu, melakukan penjadwalan, dan menampilkan hasil jadwal.

Pada DFD level 0 aplikasi penjadwalan kuliah terdapat sembilan proses yaitu melakukan login untuk pengguna, mengelola data master waktu, menentukan mata kuliah yang diselenggarakan, menentukan kompetensi dosen, menentukan ruang digunakan, menentukan sebaran waktu, melakukan penjadwalan, dan menampilkan hasil jadwal. Tabel yang terlihat pada DFD level 0 adalah karyawan, prodi, bagian, data waktu, mata kuliah, mata kuliah diselenggarakan, kompetensi dosen, ruangan, ruang digunakan, hari, slot waktu, prediksi peserta, kuliah, dan jadwal kuliah. Gambar 3.5 menampilkan DFD level 0 aplikasi penjadwalan kuliah.

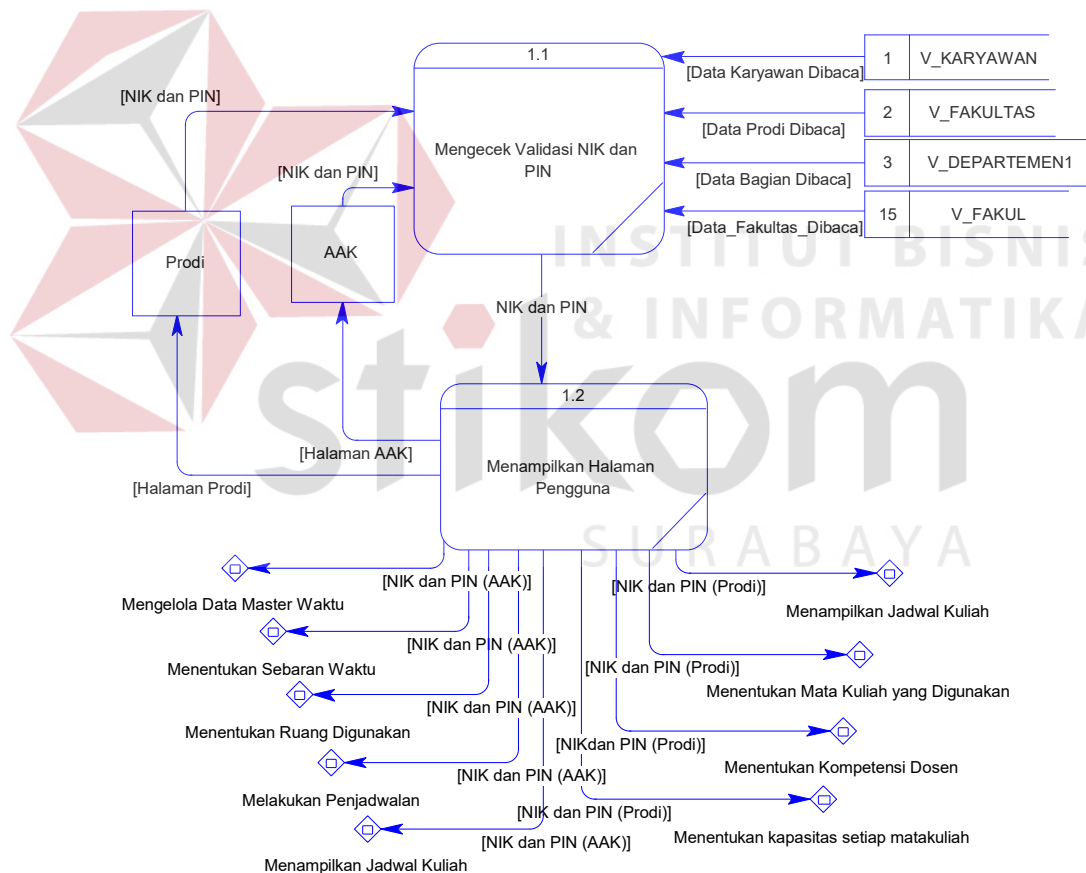


Gambar 3.5 DFD Level 0



Gambar 3.5 DFD Level 0

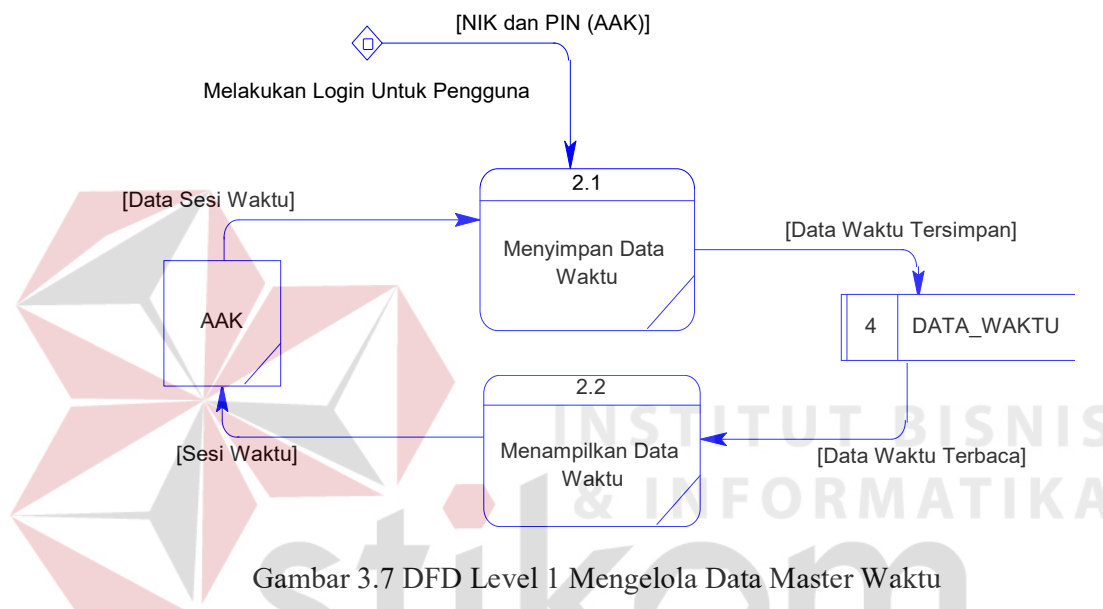
Pada DFD level 1 melakukan login untuk pengguna merupakan hasil *decompose* dari DFD level 0. DFD level 1 melakukan login untuk pengguna terdapat dua sub proses yaitu mengecek validasi nik dan pin, dan menampilkan halaman pengguna. Sub proses mengecek validasi nik dan pin berfungsi untuk mengecek login setiap pengguna yang menggunakan aplikasi penjadwalan kuliah. Sub proses menampilkan halaman pengguna berfungsi untuk menampilkan halaman sesuai dengan login pengguna. Gambar 3.6 menggambarkan DFD level 1 Melakukan Login untuk Pengguna.



Gambar 3.6 DFD Level 1 Melakukan Login untuk Pengguna

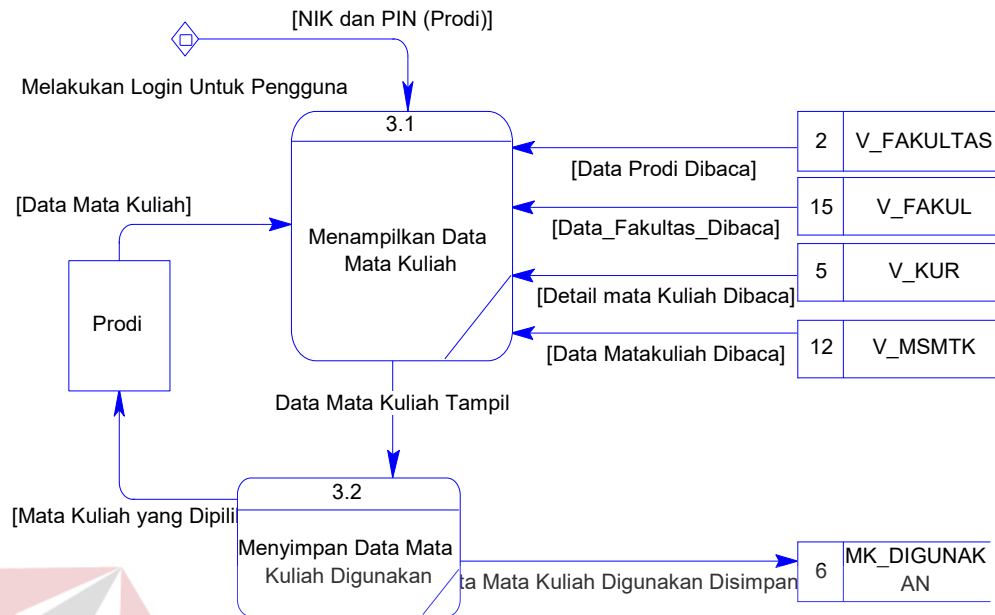
Pada DFD level 1 mengelola data master waktu merupakan hasil *decompose* dari DFD level 0. DFD level 1 mengelola data master waktu terdapat

dua sub proses yaitu menyimpan data waktu, dan menampilkan data waktu. Sub proses menyimpan data waktu berfungsi untuk menyimpan data sesi waktu yang akan digunakan aplikasi penjadwalan kuliah. Sub proses menampilkan data waktu berfungsi untuk data sesi waktu yang akan digunakan untuk aplikasi penjadwalan kuliah. Gambar 0.7 menggambarkan DFD Level 1 Mengelola Data Master Waktu.



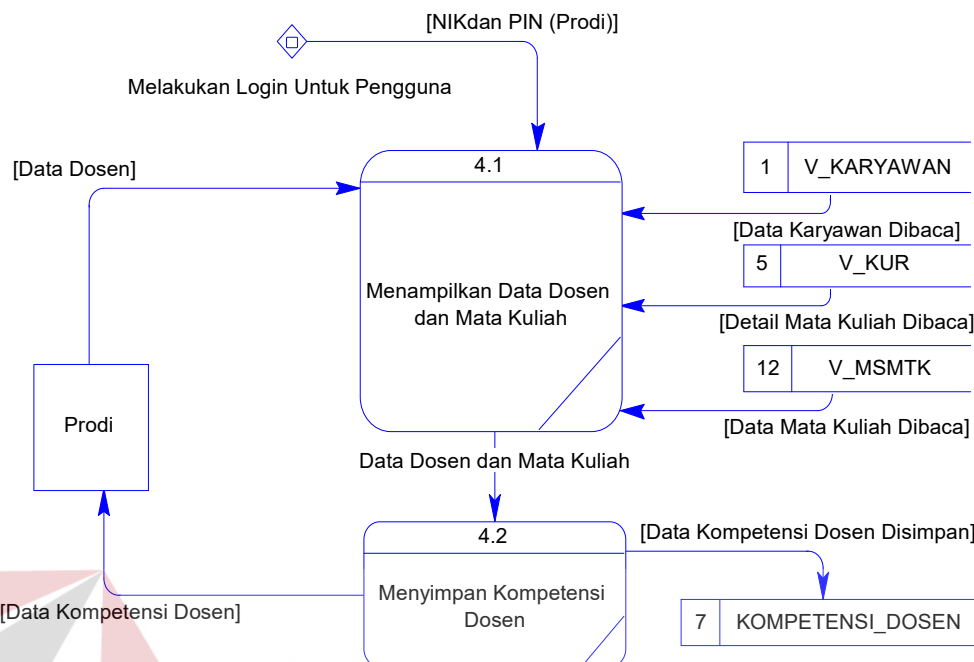
Gambar 3.7 DFD Level 1 Mengelola Data Master Waktu

Pada DFD level 1 menentukan mata kuliah yang diselenggarakan merupakan hasil *decompose* dari DFD level 0. DFD level 1 menentukan mata kuliah yang diselenggarakan terdapat dua sub proses yaitu menampilkan data mata kuliah, dan menyimpan data mata kuliah yang diselenggarakan. Sub proses menampilkan data mata kuliah berfungsi untuk menampilkan semua data mata kuliah yang sudah ada. Sub proses menyimpan data mata kuliah yang diselenggarakan berfungsi untuk menyimpan data mata kuliah yang telah dipilih untuk digunakan dalam penjadwalan kuliah. Gambar 3.8 menggambarkan DFD Level 1 Menentukan Mata Kuliah yang Diselenggarakan.



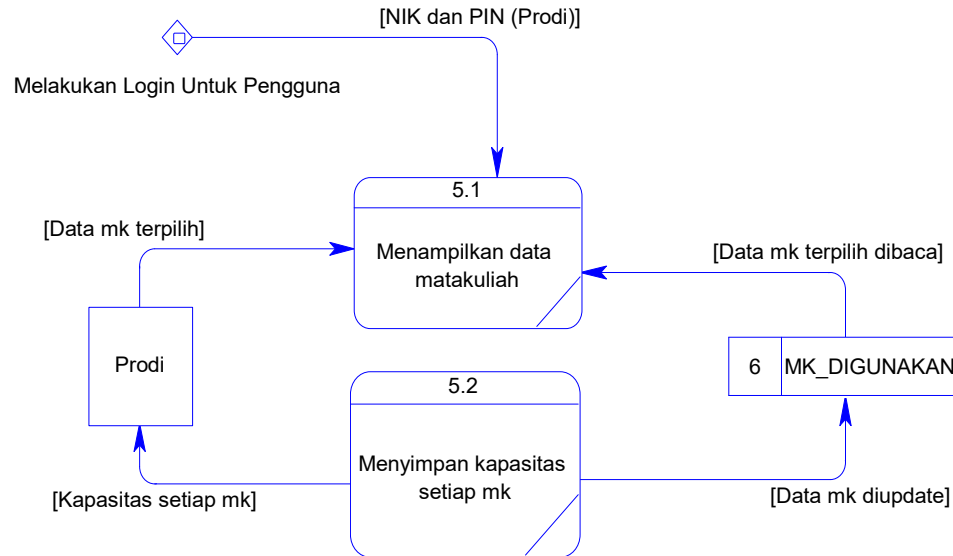
Gambar 3.8 DFD Level 1 Menentukan Mata Kuliah yang Diselenggarakan

Pada DFD level 1 menentukan kompetensi dosen merupakan hasil *decompose* dari DFD level 0. DFD level 1 menentukan kompetensi dosen terdapat dua sub proses yaitu menampilkan data dosen dan mata kuliah, dan menyimpan data kompetensi dosen. Sub proses menampilkan data dosen dan mata kuliah berfungsi untuk menampilkan semua data mata kuliah berdasarkan prodi yang telah melakukan *login* untuk dipasangkan dengan dosen sesuai dengan kompetensinya masing-masing. Sub proses menyimpan data kompetensi dosen berfungsi untuk menyimpan data kompetensi dosen yang telah dipilih untuk digunakan dalam penjadwalan kuliah. Gambar 3.9 menggambarkan DFD Level 1 Menentukan Kompetensi Dosen.

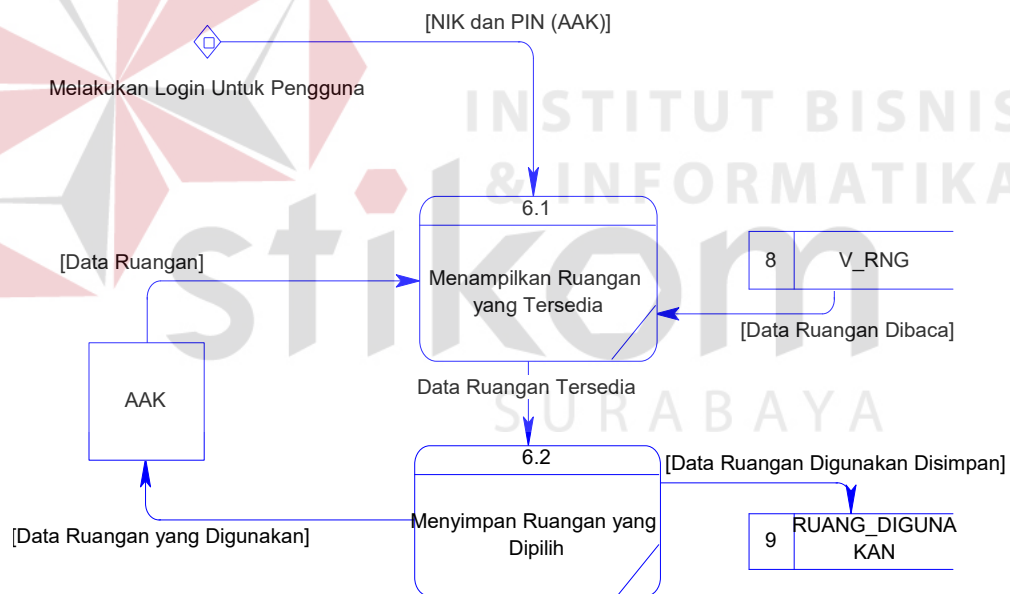


Gambar 3.9 DFD Level 1 Menentukan Kompetensi Dosen.

Pada DFD level 1 menentukan kapasitas setiap matakuliah yang diselenggarakan merupakan hasil *decompose* dari DFD level 0. DFD level 1 menentukan kapasitas setiap matakuliah yang diselenggarakan terdapat dua sub proses yaitu menampilkan data matakuliah yang diselenggarakan dan menyimpan kapasitas setiap matakuliah. Sub proses menampilkan matakuliah yang diselenggarakan berfungsi untuk menampilkan semua data matakuliah yang diselenggarakan. Sub proses menyimpan data kapasitas setiap matakuliah berfungsi untuk menyimpan data kapasitas setiap matakuliah yang diselenggarakan untuk digunakan dalam penjadwalan kuliah. Gambar 3.10 menggambarkan DFD Level 1 Menentukan Kapasitas Setiap Matakuliah.



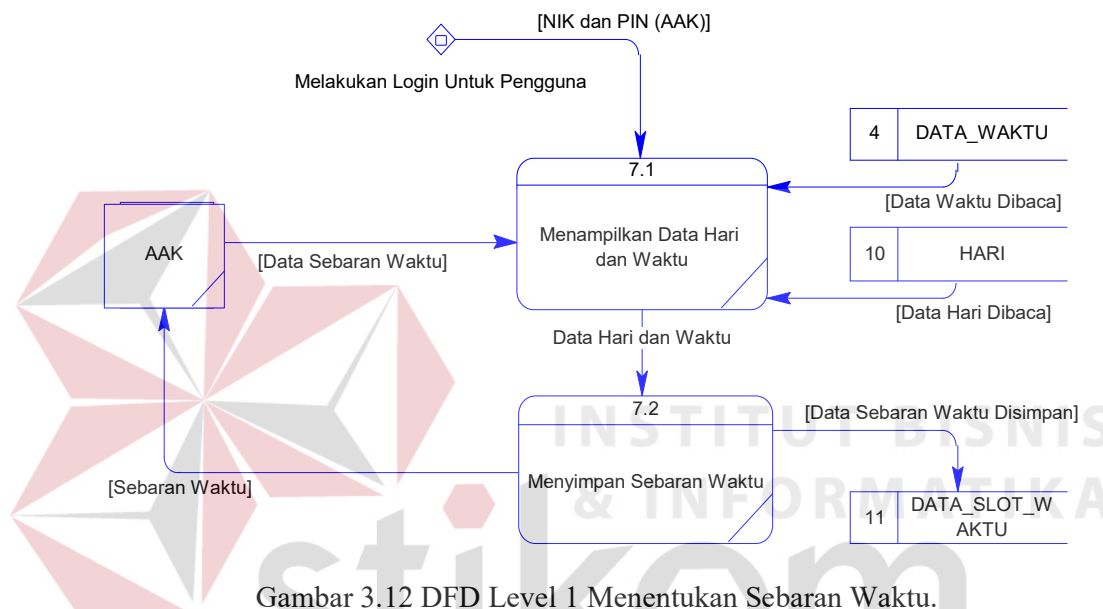
Gambar 3.10 DFD Level 1 Menentukan Kapasitas Setiap Matakuliah



Gambar 3.11 DFD Level 1 Menentukan Ruang yang Digunakan.

Pada DFD level 1 menentukan mata kuliah yang diselenggarakan merupakan hasil *decompose* dari DFD level 0. DFD level 1 menentukan ruang yang digunakan terdapat dua sub proses yaitu menampilkan data ruang yang tersedia, dan menyimpan data ruang yang digunakan. Sub proses menampilkan

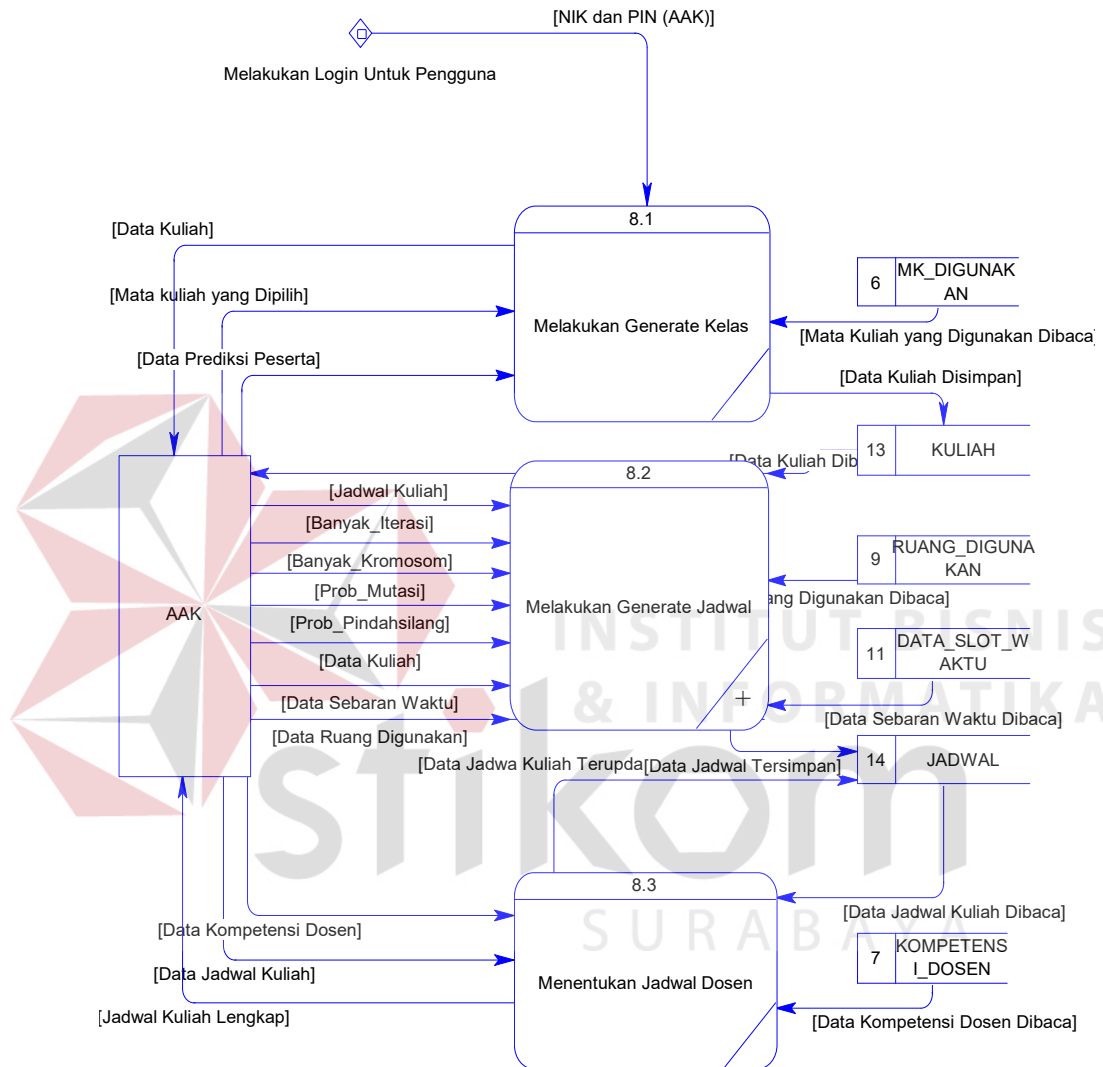
data ruangan yang tersedia berfungsi untuk menampilkan semua data ruangan yang tersedia. Sub proses menyimpan data ruangan yang digunakan berfungsi untuk menyimpan data ruangan yang telah dipilih untuk digunakan dalam penjadwalan kuliah. Gambar 3.11 menggambarkan DFD Level 1 Menentukan Ruang yang Digunakan.



Gambar 3.12 DFD Level 1 Menentukan Sebaran Waktu.

Pada DFD level 1 melakukan penjadwalan merupakan hasil *decompose* dari DFD level 0. DFD level 1 melakukan penjadwalan terdapat tiga sub proses yaitu melakukan *generate* kelas, melakukan *generate* jadwal, dan menentukan jadwal dosen. Sub proses melakukan *generate* kelas berfungsi untuk memecah menjadi beberapa kelas berdasarkan jumlah prediksi peserta setiap mata kuliah dibagi dengan kapasitas setiap kelas mata kuliah. Sub proses melakukan *generate* jadwal berfungsi untuk membuat jadwal kuliah dengan berdasarkan data yang sudah ditentukan oleh proses-proses sebelumnya. Sub proses menentukan jadwal dosen berfungsi untuk menentukan dosen siapa yang akan mengampu pada jadwal

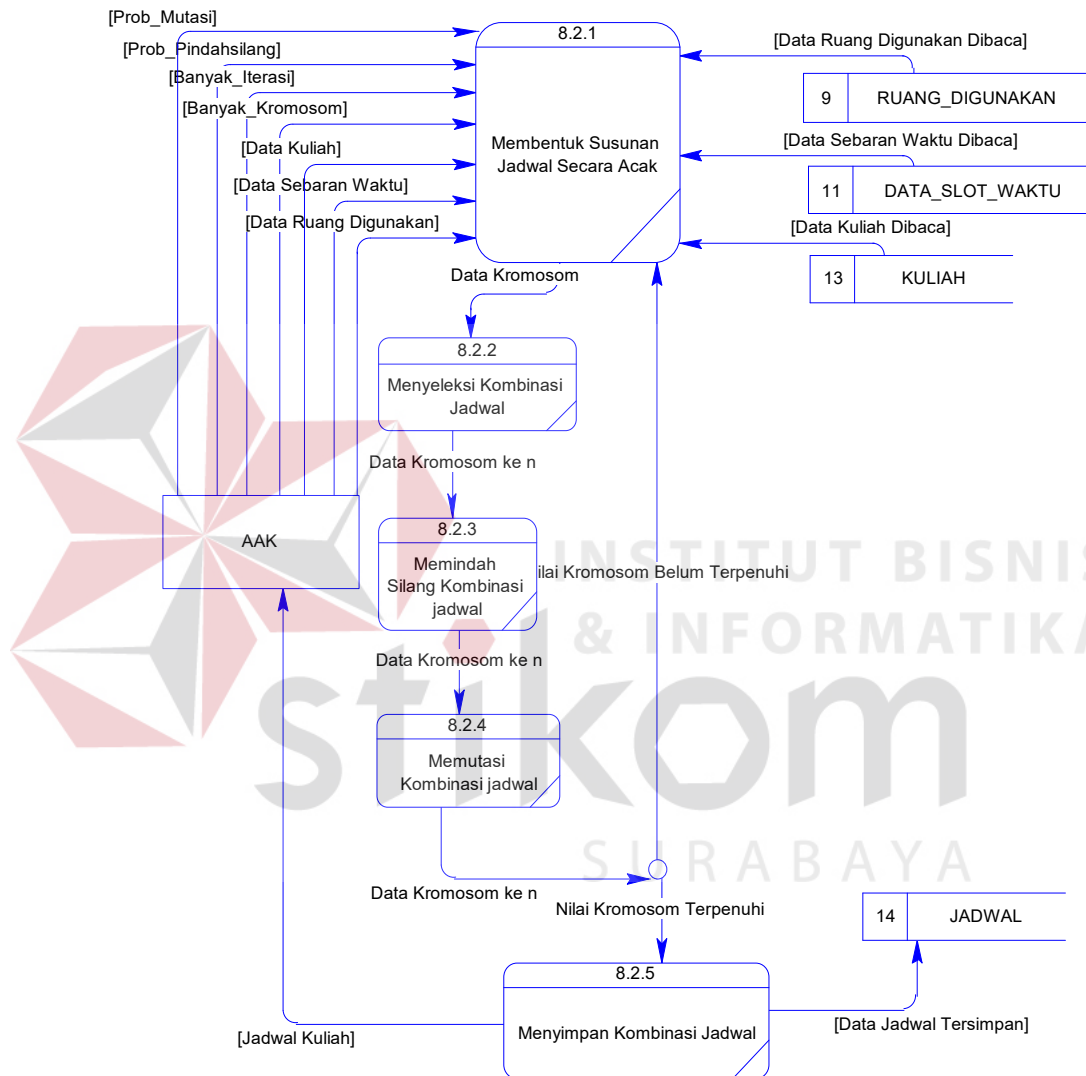
kuliah yang sudah dibuat. Gambar 3.13 menggambarkan DFD Level 1 Melakukan Penjadwalan.



Gambar 3.13 DFD Level 1 Melakukan Penjadwalan.

Pada DFD level 2 melakukan *generate* jadwal merupakan hasil *decompose* dari DFD level 1 melakukan penjadwalan. DFD level 2 melakukan *generate* jadwal merupakan proses penerapan Algoritma Genetika yang terdapat dalam aplikasi penjadwalan kuliah. Pada DFD level 2 melakukan *generate* jadwal

terdapat lima sub proses yaitu membentuk susunan jadwal secara acak, menyeleksi kombinasi jadwal, mengacak kombinasi jadwal, dan menyimpan kombinasi jadwal.



Gambar 3.14 DFD Level 2 Melakukan *Generate* Penjadwalan.

Sub proses membentuk susunan jadwal secara acak berfungsi untuk membentuk susunan kromosom dari data yang telah ada yang akan digunakan sebagai dasar untuk metode algoritma genetika. Sub proses menyeleksi kombinasi

jadwal berfungsi menyeleksi kromosom yang sudah dibuat sebelumnya untuk mencari kromosom dengan nilai yang memenuhi. Sub proses memindah silang kombinasi jadwal berfungsi untuk melanjutkan dari proses seleksi ketika nilai *fitness* kromosom masih belum terpenuhi dengan cara menukar beberapa gen-gen yang bersesuaian dari dua induk untuk menghasilkan individu baru. Sub proses memutasi kombinasi jadwal berfungsi untuk melanjutkan dari proses seleksi dan pindah silang apabila nilai *fitness* kromosom masih belum juga terpenuhi dengan cara mengganti satu atau beberapa nilai gen dari kromosom. Sehingga menghasilkan susunan kromosom baru. Jika kondisi telah memenuhi, maka algoritma genetika akan menghentikan proses pencariannya, tetapi jika belum terpenuhi maka algoritma genetika akan kembali ke membentuk susunan jadwal secara acak. Gambar 3.14 menggambarkan DFD Level 2 Melakukan *Generate* Penjadwalan.

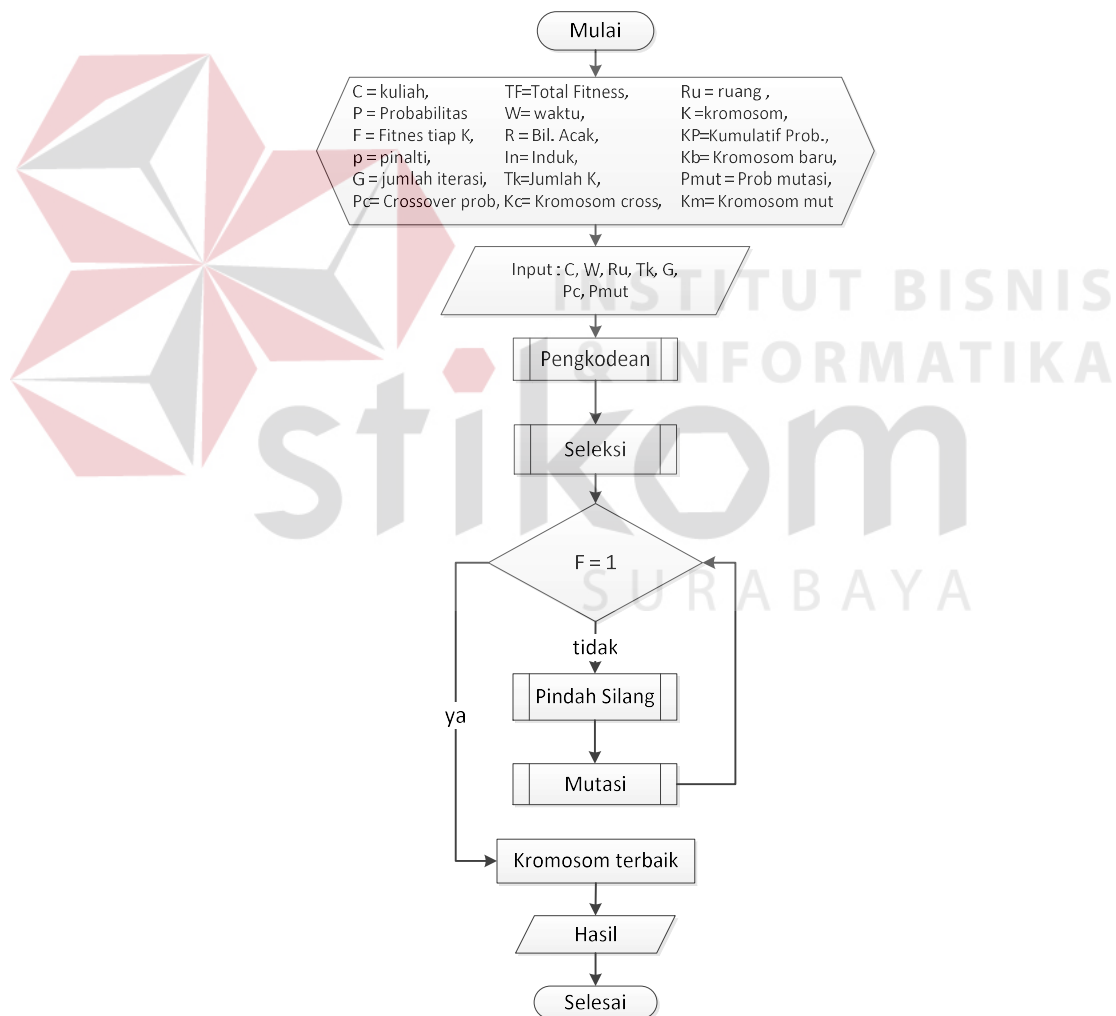
E Penerapan Algoritma Genetika

Penerapan Algoritma Genetika pada penjadwalan kuliah digambarkan pada *flowchart* program yang dapat dilihat pada gambar 3.15 *flowchart* program.

Adapun istilah yang digunakan dalam penerapan Algoritma Genetika yaitu:

1. *Allele*, sebuah nilai yang ada dalam satu susunan kombinasi jadwal
2. *Crossover* (pindah silang), proses menukar kombinas jadwal dengan kombinasi jadwal yang lain (antar susunan jadwal atau individu).
3. *Fitness*, suatu nilai yang dihasilkan dari jumlah tumpang tindih jadwal
4. Generasi, satu siklus proses atau satu iterasi dalam Algoritma Genetika.
5. *Genotype* (Gen), sebuah nilai sebagai dasar susunan kombinasi jadwal.
6. Individu, merupakan satu solusi atau susunan jadwal.

7. Kromosom, satu kesatuan nilai yang membentuk satu kombinasi jadwal.
8. Mutasi, merupakan proses menggantikan gen yang hilang dari sekumpulan susunan jadwal akibat proses seleksi.
9. Populasi, sekumpulan susunan jadwal yang yang akan diproses dan diseleksi untuk dipilih jadwal yang terbaik.
10. Seleksi, proses menyeleksi nilai *fitness* terbaik untuk mendapatkan calon jadwal yang terbaik.

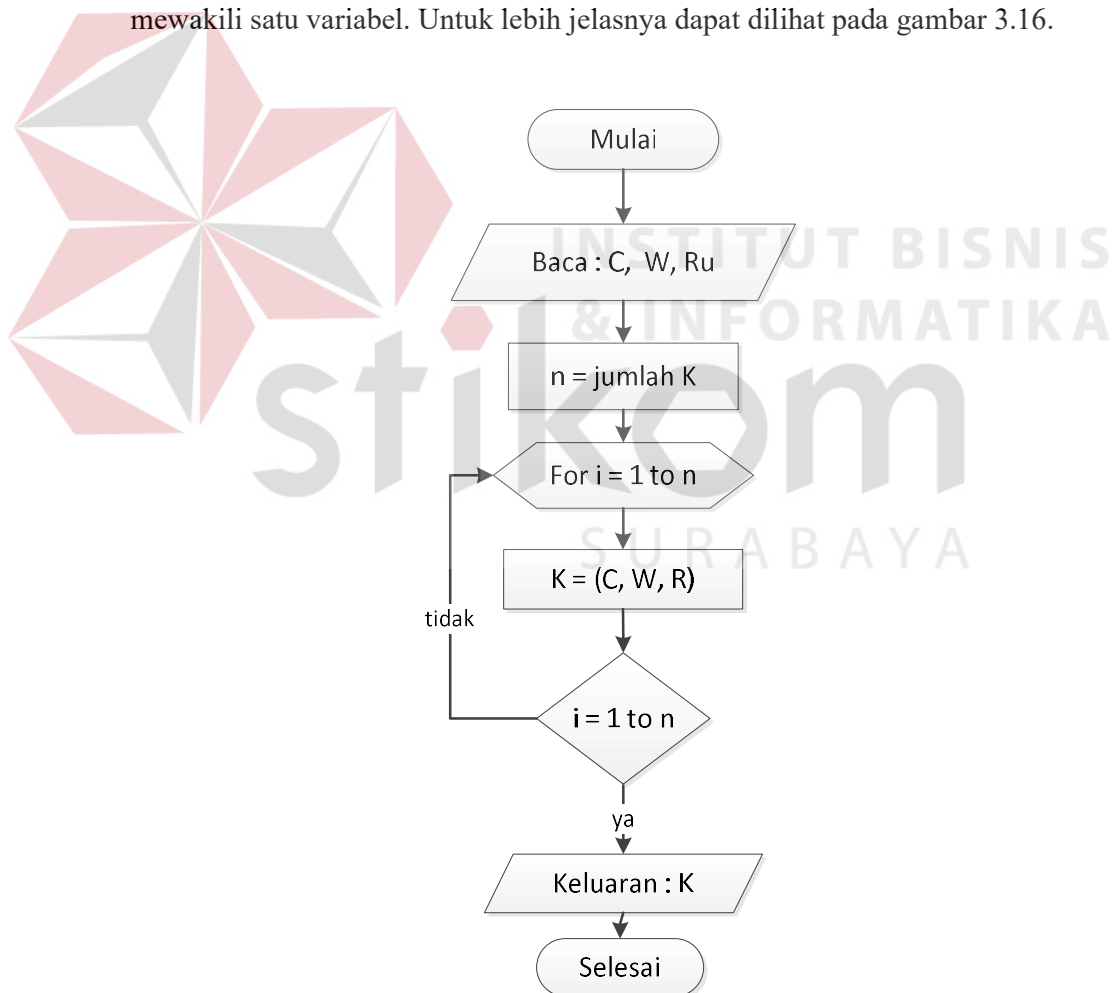


Gambar 3.15 *Flowchart* program

Proses yang harus dilakukan dalam Algoritma Genetika adalah sebagai berikut:

a. Pengodean

Sebelum memanfaatkan Algoritma Genetika dalam aplikasi penjadwalan kuliah, hal yang harus dilakukan adalah menyandikan solusi dari masalah yang diberikan ke dalam kromosom pada Algoritma Genetika. Teknik Pengodean atau membentuk Kromosom dari populasi (membentuk susunan jadwal secara acak). Teknik pengodean adalah bagaimana mengodekan gen dari kromosom, dimana gen merupakan bagian dari kromosom dan satu gen biasanya akan mewakili satu variabel. Untuk lebih jelasnya dapat dilihat pada gambar 3.16.



Gambar 3.16 *Flowchart* Pengodean pembentukan kromosom

Sebagai contoh untuk inisialisasi pembentukan kromosom, misalkan ada data mata kuliah pada tabel 3.12, data kelas kuliah pada tabel 3.13, sebaran waktu yang telah disediakan pada tabel 3.14 dan data ruangan yang digunakan pada tabel 3.15.

Tabel 3.12 mata kuliah

Kode	Nama MK	Semester
0	Penulisan Ilmiah	2
1	Desain dan Pemrograman Web	2
2	Riset Operasional	4
3	Pengembangan Sistem Informasi	6
4	Sistem Pendukung Keputusan	6
5	Pendidikan Pancasila & Kewarganegaraan	8

Tabel 3.13 kelas kuliah

Kode	MK	Semester
0	Penulisan Ilmiah	2
1	Penulisan Ilmiah	2
2	Desain dan Pemrograman Web	2
3	Desain dan Pemrograman Web	2
4	Desain dan Pemrograman Web	2
5	Riset Operasional	4
6	Riset Operasional	4
7	Riset Operasional	4
8	Pengembangan Sistem Informasi	6
9	Sistem Pendukung Keputusan	6
10	Sistem Pendukung Keputusan	6
11	Pendidikan Pancasila & Kewarganegaraan	8
12	Pendidikan Pancasila & Kewarganegaraan	8

Tabel 3.14 Sebaran waktu yang tersedia

Kode	Hari	Mulai	Akhir
0	Senin	07.30	10.00
1	Senin	10.30	13.00

Kode	Hari	Mulai	Akhir
2	Senin	13.30	16.00
3	Selasa	07.30	10.00
4	Selasa	10.30	13.00
5	Selasa	13.30	16.00
6	Rabu	07.30	10.00
7	Rabu	10.30	13.00
8	Rabu	13.30	16.00
9	Kamis	07.30	10.00
10	Kamis	10.30	13.00
11	Kamis	13.30	16.00
12	Jumat	07.30	10.00
13	Jumat	13.30	16.00

Tabel 3.15 Ruangan yang digunakan

Kode	Nama Ruang
0	M301
1	M302

Pembentukan kromosom yang digunakan dalam penelitian ini direpresentasikan dalam susunan gen yang terdiri atas tiga nilai atau *allele*, susunan gen tersebut akan di jelaskan sebagai berikut :

1. Nilai pertama adalah kode kuliah yang akan dijadwalkan.
2. Nilai kedua adalah kode sebaran waktu yang akan digunakan.
3. Nilai ketiga adalah kode ruangan yang akan digunakan

Diasumsikan dalam satu populasi yang terbentuk dari kromosom berjumlah sesuai dengan jumlah kuliah yang akan dijadwalkan dan masing masing kromosom memiliki gen yang telah ditentukan. Penempatan kode pada setiap gen dilakukan berdasarkan suatu bilangan yang dibangkitkan secara acak.

Kromosom[0]:

([0,0,5],[1,1,2],[2,0,1],[3,0,7],[4,0,7],[5,1,11],[6,1,8],[7,0,12],[8,1,10],[9,0,12],[10,1,1],[11,0,6],[12,1,13])

Kromosom[1]:

([0,1,13],[1,1,4],[2,1,0],[3,1,10],[4,1,12],[5,0,2],[6,1,4],[7,1,13],[8,0,6],[9,1,3],[10,1,0],[11,1,4],[12,0,4])

Kromosom[2]:

([0,1,4],[1,0,13],[2,1,8],[3,1,3],[4,0,2],[5,1,11],[6,1,7],[7,1,0],[8,1,8],[9,0,6],[10,0,11],[11,1,5],[12,1,3])

Kromosom[3]:

([0,1,7],[1,1,3],[2,1,1],[3,1,12],[4,1,11],[5,0,8],[6,0,12],[7,0,5],[8,0,0],[9,0,8],[10,0,7],[11,1,5],[12,1,0])

Kromosom[4]:

([0,0,0],[1,1,10],[2,1,8],[3,1,5],[4,0,8],[5,0,3],[6,1,12],[7,1,0],[8,0,0],[9,0,12],[10,0,13],[11,1,8],[12,1,1])

Kromosom[5]:

([0,1,5],[1,1,7],[2,1,12],[3,0,8],[4,0,6],[5,1,5],[6,1,9],[7,1,3],[8,0,11],[9,1,3],[10,0,8],[11,0,11],[12,1,4])

Menghitung nilai *fitness* Nilai *fitness* ini yang nantinya akan digunakan pada tahap-tahap seleksi berikutnya. Nilai yang dihasilkan oleh fungsi *fitness* mempresentasikan seberapa banyak jumlah persyaratan yang dilanggar, untuk menghitung nilai *fitness* menggunakan persamaan 2.1

$$Fitness[0]: 1 / (1 + 2 + 0) = 0,3333333333333333$$

$$Fitness [1]: 1 / (1 + 5 + 3) = 0,1111111111111111$$

$$\text{Fitness [2]: } 1 / (1 + 2 + 2) = 0,2$$

$$\text{Fitness [3]: } 1 / (1 + 1 + 2) = 0,25$$

$$\text{Fitness [4]: } 1 / (1 + 2 + 2) = 0,2$$

$$\text{Fitness [5]: } 1 / (1 + 4 + 3) = 0,125$$

$$\text{Total Fitness: } 1,21944444444444$$

b. Seleksi

Melakukan proses seleksi, seleksi digunakan untuk memilih individu-individu mana saja yang akan dipilih untuk proses kawin silang dan mutasi. Metode

seleksi yang akan digunakan disini yaitu seleksi dengan mesin *roulette*.

Metode seleksi mesin roulette ini merupakan metode yang paling sederhana dan sering juga dikenal dengan nama *stochastic sampling with replacement*.

Untuk lebih jelasnya dapat dilihat pada gambar 3.17. Langkah pertama dalam proses seleksi ini adalah dengan menghitung total nilai *fitness* semua kromosom dengan menggunakan persamaan 2.1.

Langkah selanjutnya adalah menghitung probabilitas setiap kromosom yaitu dengan cara membagi nilai *fitness* setiap kromosom dengan total nilai *fitness*, seperti pada tabel 3.16.

Tabel 3.16 Probabilitas setiap kromosom

Kromosom	Probabilitas
0	$0.33 / 1.21 = 0.27$
1	$0.11 / 1.21 = 0.09$
2	$0.2 / 1.21 = 0.16$
3	$0.25 / 1.21 = 0.2$
4	$0.2 / 1.21 = 0.16$
5	$0.125 / 1.21 = 0.1$
Total Probabilitas	1

Maka susunan kromosom populasi baru hasil seleksi adalah :

Kromosom[0]:

([0,0,5],[1,1,2],[2,0,1],[3,0,7],[4,0,7],[5,1,11],[6,1,8],[7,0,12],[8,1,10],[9,0,12],[10,1,1],[11,0,6],[12,1,13])

Kromosom[1]:

([0,1,7],[1,1,3],[2,1,1],[3,1,12],[4,1,11],[5,0,8],[6,0,12],[7,0,5],[8,0,0],[9,0,8],[10,0,7],[11,1,5],[12,1,0])

Kromosom[2]:

([0,0,5],[1,1,2],[2,0,1],[3,0,7],[4,0,7],[5,1,11],[6,1,8],[7,0,12],[8,1,10],[9,0,12],[10,1,1],[11,0,6],[12,1,13])

Kromosom[3]:

([0,1,13],[1,1,4],[2,1,0],[3,1,10],[4,1,12],[5,0,2],[6,1,4],[7,1,13],[8,0,6],[9,1,3],[10,1,0],[11,1,4],[12,0,4])

Kromosom[4]:

([0,1,7],[1,1,3],[2,1,1],[3,1,12],[4,1,11],[5,0,8],[6,0,12],[7,0,5],[8,0,0],[9,0,8],[10,0,7],[11,1,5],[12,1,0])

Kromosom[5]:

([0,1,13],[1,1,4],[2,1,0],[3,1,10],[4,1,12],[5,0,2],[6,1,4],[7,1,13],[8,0,6],[9,1,3],[10,1,0],[11,1,4],[12,0,4])

Fitness [0]: $1 / (1 + 2 + 0) = 0,333333333333333$

Fitness [1]: $1 / (1 + 2 + 0) = 0,333333333333333$

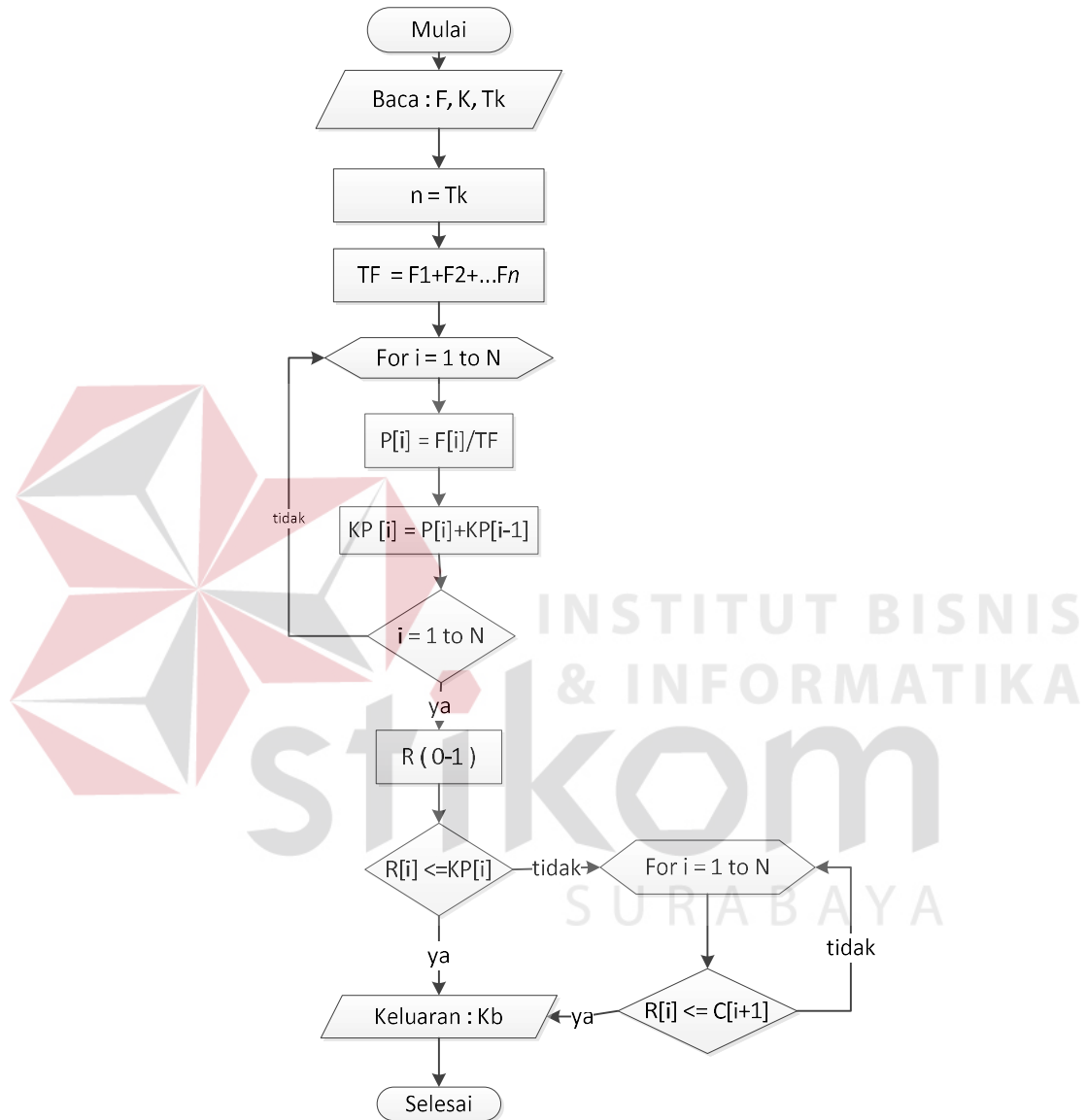
Fitness [2]: $1 / (1 + 2 + 0) = 0,333333333333333$

Fitness [3]: $1 / (1 + 2 + 0) = 0,333333333333333$

Fitness [4]: $1 / (1 + 2 + 0) = 0,333333333333333$

$Fitness [5]: 1 / (1 + 2 + 0) = 0,3333333333333333$

Total $Fitness: 2$

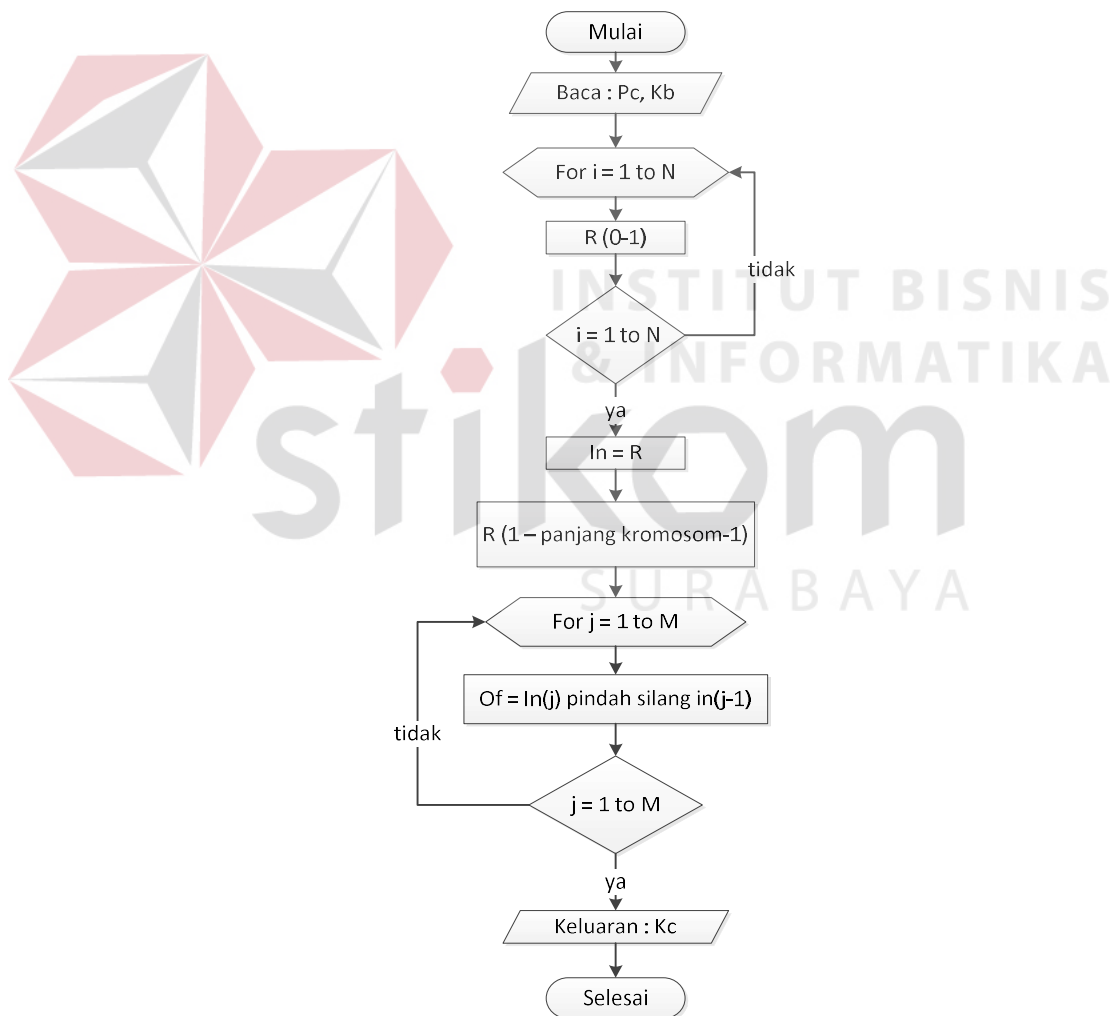


Gambar 3.17 *Flowchart* Seleksi

c. Pindah Silang

Apabila kondisi nilai *fitness* masih belum terpenuhi selanjutnya adalah Melakukan pindah silang (*crossover*). Pindah silang disini adalah melakukan

operasi (pertukaran, aritmatika) pada gen-gen yang bersesuaian dari dua induk untuk menghasilkan individu baru. Proses pindah silang dilakukan pada setiap individu dengan probabilitas *crossover* yang ditentukan. Pindah silang bisa dilakukan hanya jika suatu bilangan acak yang dibangkitkan untuk kromosom kurang dari probabilitas pindah silang yang ditentukan. Probabilitas pindah silang umumnya ditentukan mendekati nilai 1 (Suyanto, 2005), misalnya probabilitas pindah silang dalam perhitungan ini ditentukan sebesar 0,5.



Gambar 3.18 *Flowchart* Pindah Silang

Metode pindah silang yang digunakan adalah pindah silang satu titik potong (*one-point crossover*). Suatu titik potong dipilih secara acak, kemudian bagian pertama kromosom induk 1 digabungkan dengan bagian kedua dari kromosom induk 2. Bilangan acak yang dibangkitkan untuk menentukan posisi titik potong adalah $[1-N]$ dimana N adalah banyaknya jumlah gen dalam satu kromosom. Bilangan acak dibangkitkan sebanyak kromosom yang mengalami pindah silang.

Hasil pindah silang antar kromosom tersebut adalah :

Induk1 [0] : Induk2 [0]

Induk1 [1] : Induk2 [2]

Induk1 [2] : Induk2 [3]

Induk1 [3] : Induk2 [4]

Induk1 [4] : Induk2 [5]

Kromosom[0]:

([0,0,5],[1,1,2],[2,0,1],[3,0,7],[4,0,7],[5,1,11],[6,1,8],[7,0,12],[8,1,10],[9,0,12],[10,1,1],[11,0,6],[12,1,13])

Kromosom[1]:

([0,1,7],[1,1,3],[2,1,1],[3,1,12],[4,1,11],[5,0,8],[6,0,12],[7,0,5],[8,0,0],[9,0,8],[10,0,7],[11,1,5],[12,1,0])

Kromosom[2]:

([0,0,5],[1,1,2],[2,0,1],[3,0,7],[4,0,7],[5,1,11],[6,1,8],[7,0,12],[8,1,10],[9,0,12],[10,1,1],[11,1,4],[12,0,4])

Kromosom[3]:

([0,1,13],[1,1,4],[2,1,0],[3,1,10],[4,1,12],[5,0,2],[6,1,4],[7,1,13],[8,0,0],[9,0,8],[10,0,7],[11,1,5],[12,1,0])

Kromosom[4]:

([0,1,7],[1,1,3],[2,1,1],[3,1,12],[4,1,11],[5,0,8],[6,0,12],[7,0,5],[8,0,6],[9,1,3],[10,1,0],[11,1,4],[12,0,4])

Kromosom[5]:

([0,1,13],[1,1,4],[2,1,0],[3,1,10],[4,1,12],[5,0,2],[6,1,4],[7,1,13],[8,0,6],[9,1,3],[10,1,0],[11,0,6],[12,1,13])

Fitness [0]: $1 / (1 + 2 + 0) = 0,3333333333333333$

Fitness [1]: $1 / (1 + 2 + 0) = 0,3333333333333333$

Fitness [2]: $1 / (1 + 2 + 0) = 0,3333333333333333$

Fitness [3]: $1 / (1 + 3 + 2) = 0,1666666666666667$

Fitness [4]: $1 / (1 + 1 + 1) = 0,3333333333333333$

Fitness [5]: $1 / (1 + 6 + 2) = 0,1111111111111111$

Total *Fitness*: 1.611111111111111

d. Mutasi

Melakukan proses mutasi gen, operator ini berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi. Kromosom anak dimutasi dengan menambahkan nilai random yang sangat kecil (ukuran langkah mutasi), dengan probabilitas yang rendah. Tetapi bila probabilitas mutasi ini terlalu besar, maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dengan induknya.

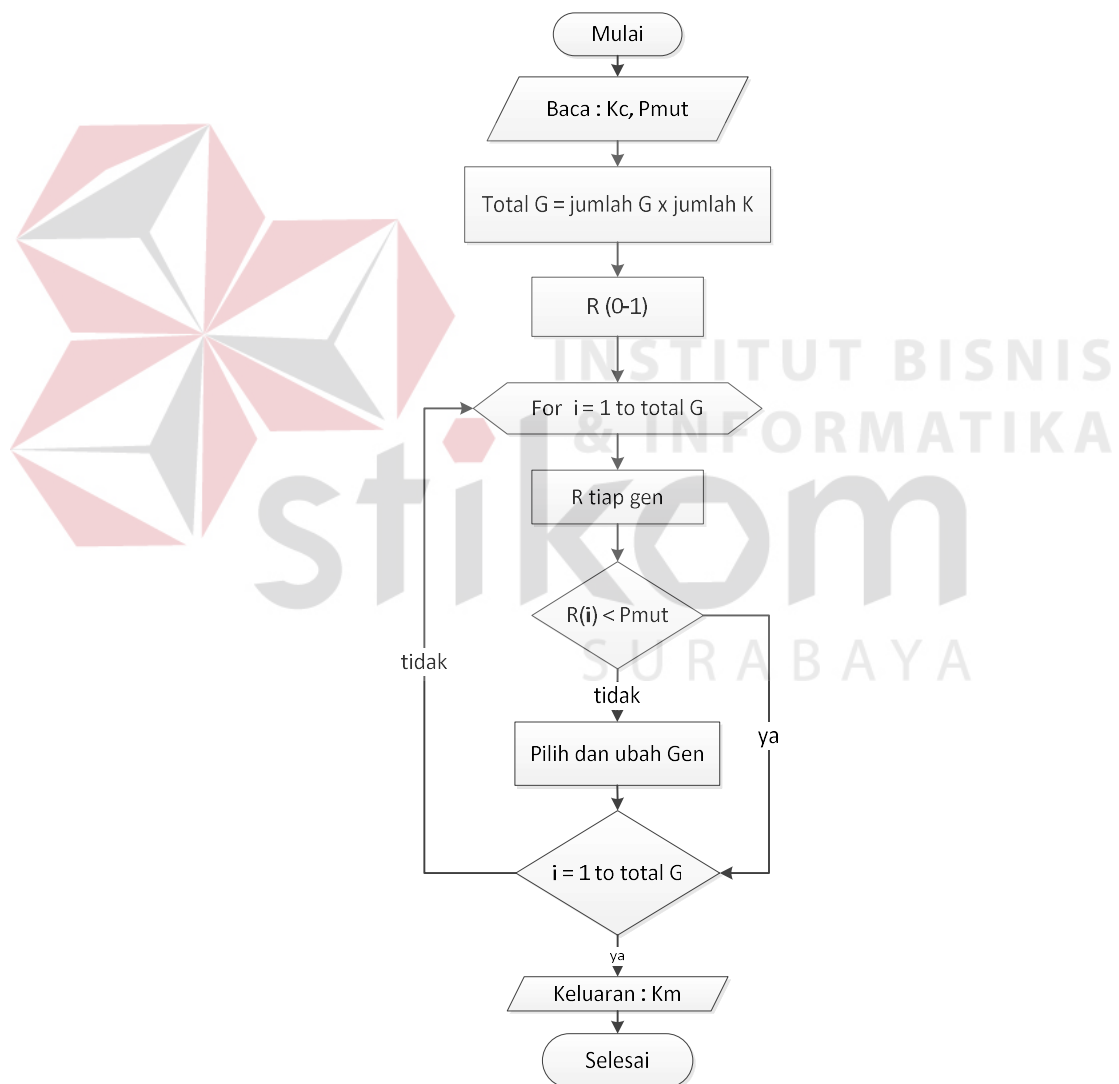
Untuk mendapatkan posisi gen yang akan dimutasi maka perlu dihitung jumlah total gen dalam satu populasi yaitu

Total gen = jumlah gen dalam satu kromosom X jumlah kromosom yang ada.

Total gen = $13 \times 6 = 78$

Probabilitas mutasi ditetapkan 0,1 maka mutasi yang terjadi adalah :

Mutasi = $0,1 \times 78 = 7,8 = 8$



Gambar 3.19 Flowchart Mutasi

Bangkitkan bilangan acak sebanyak 8. Yaitu 1 - 78 untuk menentukan posisi gen yang akan dimutasi. Sehingga akan menghasilkan susunan kromosom baru sebagai berikut :

Kromosom[0]:

((0,0,5],[1,1,2],[2,0,1],[3,0,7],[4,0,7],[5,1,11],[6,1,8],[7,0,12],[8,1,10],[9,0,12],[10,1,1],[11,0,6],[12,1,13])

Kromosom[1]:

((0,1,7],[1,1,3],[2,1,1],[3,1,12],[4,1,11],[5,0,8],[6,0,12],[7,0,5],[8,0,0],[9,0,8],[10,0,7],[11,1,5],[12,1,0])

Kromosom[2]:

((0,0,5],[1,1,2],[2,0,1],[3,0,7],[4,0,7],[5,1,11],[6,1,8],[7,0,12],[8,1,10],[9,0,12],[10,1,1],[11,1,4],[12,0,4])

Kromosom[3]:

((0,1,13],[1,1,4],[2,1,0],[3,1,10],[4,1,12],[5,0,2],[6,1,4],[7,1,13],[8,0,0],[9,0,8],[10,0,7],[11,1,5],[12,1,0])

Kromosom[4]:

((0,1,7],[1,1,3],[2,1,1],[3,1,12],[4,1,11],[5,0,8],[6,0,12],[7,0,5],[8,0,6],[9,1,3],[10,1,0],[11,1,4],[12,0,4])

Kromosom[5]:

((0,1,13],[1,1,4],[2,1,0],[3,1,10],[4,1,12],[5,0,2],[6,1,4],[7,1,13],[8,0,6],[9,1,3],[10,1,0],[11,0,6],[12,1,13])

Fitness[0]: $1 / (1 + 2 + 0) = 0,33333333333333$

Fitness [1]: $1 / (1 + 2 + 0) = 0,33333333333333$

Fitness [2]: $1 / (1 + 2 + 0) = 0,33333333333333$

$$\text{Fitness [3]: } 1 / (1 + 3 + 2) = 0,166666666666667$$

$$\text{Fitness [4]: } 1 / (1 + 1 + 1) = 0,333333333333333$$

$$\text{Fitness [5]: } 1 / (1 + 6 + 2) = 0,111111111111111$$

$$\text{Total Fitness : } 1,61111111111111$$

Jika kondisi telah memenuhi, maka algoritma genetika akan menghentikan proses pencariannya, tetapi jika belum terpenuhi maka algoritma genetika akan kembali ke proses seleksi. Dalam kondisi diatas kondisi masih belum terpenuhi. Nilai *fitness* terpenuhi apabila mencapai nilai 1. Untuk lebih jelasnya dapat dilihat pada gambar 3.19. Dalam aplikasi penjadwalan yang dibuat disediakan *inputan* untuk memasukkan jumlah iterasi sebanyak yang diinginkan. Apabila kondisi sebelum sebanyak iterasi yang diinginkan dan sudah menemukan kondisi yang terpenuhi, maka proses akan berhenti dan menampilkan hasil kromosom yang sudah memenuhi tersebut. Tetapi apabila kondisi nilai *fitness* belum memenuhi, maka proses pencarian akan berhenti sebanyak iterasi yang diinginkan dan akan menampilkan hasil yang paling terbaik. Semakin besar nilai *fitness* maka semakin sedikit jumlah jadwal yang tumpang tindih. Begitu juga dengan sebaliknya semakin kecil nilai *fitness* semakin banyak jumlah jadwal yang tumpang tindih

3.2.2 Perancangan Data

Pada tahap perancangan data akan dibahas *entity relationship diagram* dan struktur tabel yang akan digunakan pada aplikasi penjadwalan kuliah. Dalam membangun aplikasi penjadwalan ini menggunakan beberapa tabel yang sudah ada, sehingga beberapa tabel bersifat *read only*. Berikut tabel-tabel yang dibutuhkan oleh sistem yang terdiri atas tabel V_KARYAWAN, tabel

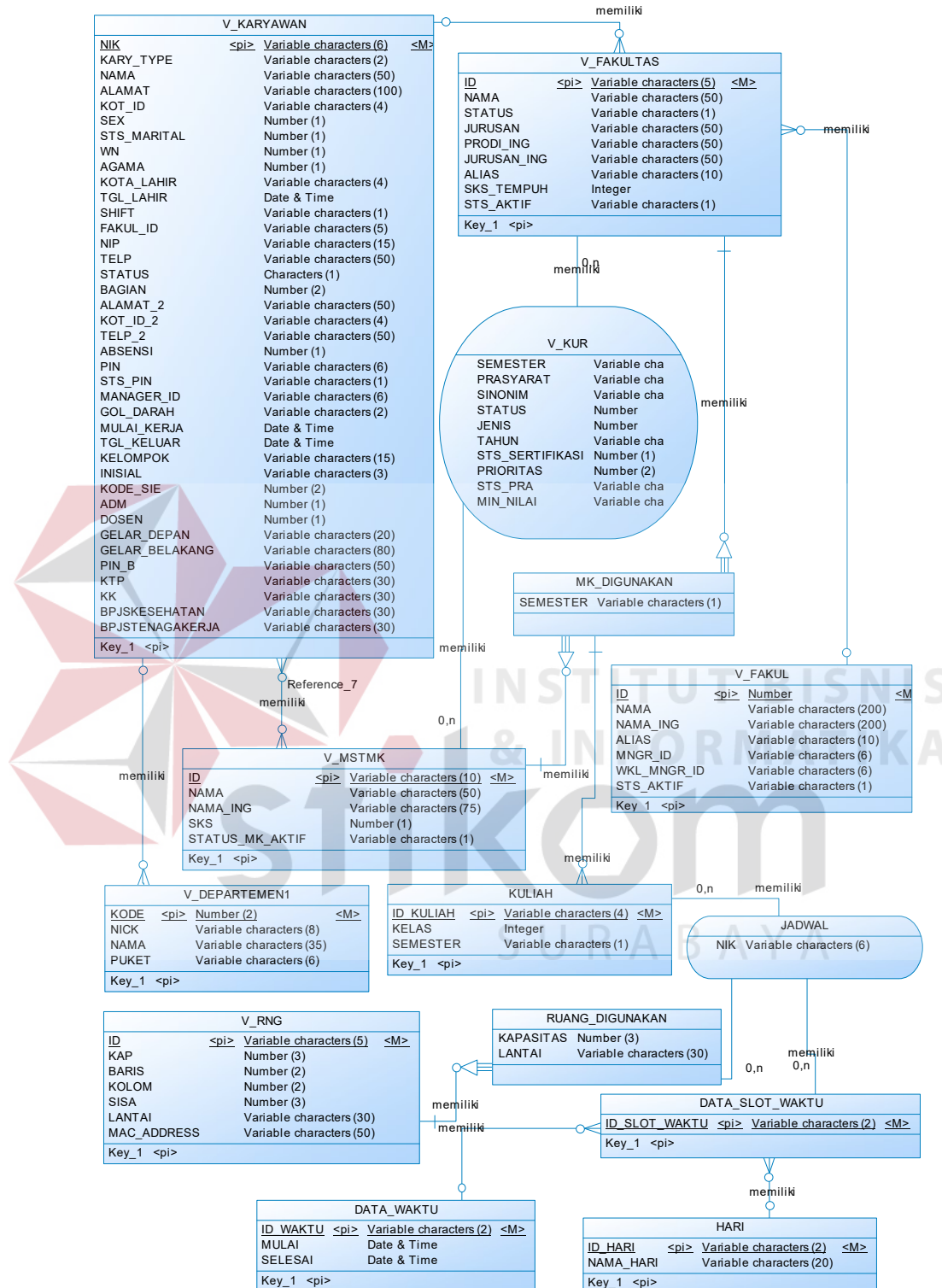
V_FAKULTAS, tabel V_FAKUL, tabel V_MSTMK, tabel V_KUR, tabel V_DEPARTMENT1, tabel V_RNG dan tabel HARI.

A Entity Relationship Diagram

Perancangan basis data atau yang lebih dikenal dengan *entity relationship diagram* (ERD) merupakan representasi model basis data yang berasal dari gambaran rancangan DFD. ERD terbagi menjadi dua bagian, yaitu *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM). Adapun kedua jenis basis data tersebut akan dijelaskan sebagai berikut:

Conceptual Data Model dari aplikasi penjadwalan kuliah ini terdapat 14 entitas, yang terdiri atas tabel V_KARYAWAN, tabel V_FAKULTAS, tabel V_FAKUL, tabel V_MSTMK, tabel V_KUR, tabel V_DEPARTMENT1, tabel V_RNG, tabel MK_DIGUNAKAN, tabel KULIAH, tabel JADWAL, tabel RUANG_DIGUNAKAN, tabel DATA_WAKTU, tabel HARI, dan tabel DATA_SLOT_WAKTU. CDM aplikasi penjadwalan kuliah dapat dilihat pada Gambar 3.20.

Physical data model dari aplikasi penjadwalan kuliah terdapat 15 tabel, yang terdiri atas tabel V_KARYAWAN, tabel V_FAKULTAS, tabel V_FAKUL, tabel V_MSTMK, tabel V_KUR, tabel V_DEPARTMENT1, tabel V_RNG, tabel MK_DIGUNAKAN, tabel KOMPETENSI_DOSEN, tabel KULIAH, tabel JADWAL, tabel RUANG_DIGUNAKAN, tabel DATA_WAKTU, tabel HARI, dan tabel DATA_SLOT_WAKTU. PDM aplikasi penjadwalan kuliah dapat dilihat pada Gambar 3.21.



Gambar 3.20 CDM Aplikasi Penjadwalan Kuliah

B Struktur Tabel

Berdasarkan *Physical Data Model* (PDM) yang telah dirancang sebelumnya, pada aplikasi penjadwalan kuliah ini terdapat 15 tabel. Struktur tabel pada aplikasi penjadwalan kuliah ini adalah sebagai berikut :

a. Tabel V_KARYAWAN

Primary Key : NIK

Foreign Key : -

Fungsi : membaca data karyawan atau dosen

Tabel 3.17 Struktur Tabel V_KARYAWAN

No.	Field	Type Data	Length	Keterangan
1	NIK	Varchar	6	<i>Primary Key</i>
2	KARY_TYPE	Varchar	2	
3	NAMA	Varchar	50	
4	ALAMAT	Varchar	100	
5	KOT_ID	Varchar	4	
6	SEX	Number	1	
7	STS_MARITAL	Number	1	
8	WN	Number	1	
9	AGAMA	Number	1	
10	KOTA_LAHIR	Varchar	4	
11	TGL_LAHIR	Date		
12	SHIFT	Varchar	1	
13	FAKUL_ID	Varchar	5	
14	NIP	Varchar	15	
15	TELP	Varchar	50	
16	STATUS	Char	1	
17	BAGIAN	Number	2	
18	ALAMAT_2	Varchar	50	
19	KOT_ID_2	Varchar	4	
20	TELP_2	Varchar	50	
21	ABSENSI	Number	1	
22	PIN	Varchar	6	
23	STS_PIN	Varchar	1	
24	MANAGER_ID	Varchar	6	

No.	Field	Type Data	Length	Keterangan
25	GOL_DARAH	Varchar	2	
26	MULAI_KERJA	Date		
27	TGL_KELUAR	Date		
28	KELOMPOK	Varchar	15	
29	INISIAL	Varchar	3	
30	KODE_SIE	Number	2	
31	ADM	Number	1	
32	DOSEN	Number	1	
33	GELAR_DEPAN	Varchar	20	
34	GELAR_BELAKANG	Varchar	80	
35	PIN_B	Varchar	50	
36	KTP	Varchar	30	
37	KK	Varchar	30	
38	BPJSKESEHATAN	Varchar	30	
39	BPJSTENAGAKERJA	Varchar	30	

b. Tabel V_FAKUL

Primary Key : ID

Foreign Key : -

Fungsi : membaca data fakultas

Tabel 3.18 Struktur Tabel V_FAKUL

No.	Field	Type Data	Length	Keterangan
1	ID	Number		<i>Primary Key</i>
2	NAMA	Varchar	200	
3	NAMA_ING	Varchar	200	
4	ALIAS	Varchar	10	
5	MNGR_ID	Varchar	6	
6	WKL_MNGR_ID	Varchar	6	
7	STS_AKTIF	Varchar	1	

c. Tabel V_FAKULTAS

Primary Key : ID

Foreign Key : MNGR_ID dan ID_FAKULTAS

Fungsi : membaca data program studi

Tabel 3.19 Struktur Tabel V_FAKULTAS

No.	Field	Type Data	Length	Keterangan
1	ID	Varchar	5	Primary Key
2	NAMA	Varchar	50	
3	STATUS	Varchar	1	
4	JURUSAN	Varchar	50	
5	PRODI_ING	Varchar	50	
6	JURUSAN_ING	Varchar	50	
7	MNGR_ID	Varchar	6	Foreign Key
8	ALIAS	Varchar	10	
9	SKS_TEMPUH	Integer		
10	STS_AKTIF	Varchar	1	
11	ID_FAKULTAS	Number		Foreign Key

d. Tabel V_MSTMK

Primary Key : ID

Foreign Key :-

Fungsi : membaca data mata kuliah

Tabel 3.20 Struktur Tabel V_MSTMK

No.	Field	Type Data	Length	Keterangan
1	ID	Varchar	10	Primary Key
2	NAMA	Varchar	50	
3	NAMA_ING	Varchar	75	
4	SKS	Number	1	
5	STATUS_MK_AKTIF	Varchar	1	

e. Tabel V_KUR

Primary Key : PRODI dan ID

Foreign Key : PRODI dan ID

Fungsi : membaca data detail mata kuliah

Tabel 3.21 Struktur Tabel V_KUR

No.	Field	Type Data	Length	Keterangan
1	PRODI	Varchar	5	Primary key, Foreign key
2	ID	Varchar	10	Primary key, Foreign key
3	SEMESTER	Varchar	1	
4	PRASYARAT	Varchar	100	
5	SINONIM	Varchar	40	
6	STATUS	Number		
7	JENIS	Number		
8	TAHUN	Varchar	4	
9	STS_SERTIFIKASI	Number	1	
10	PRIORITAS	Number	2	
11	STS_PRA	Varchar	1	
12	MIN_NILAI	Varchar	2	
13	KOMPETENSI	Varchar	2	
14	JENIS_WAJIB	Varchar	2	

f. Tabel V_DEPARTMENT1

Primary Key : KODE

Foreign Key : -

Fungsi : membaca data bagian

Tabel 3.22 Struktur Tabel V_DEPARTMENT1

No.	Field	Type Data	Length	Keterangan
1	KODE	Number	2	Primary Key
2	NICK	Varchar	8	
3	NAMA	Varchar	35	
4	MANAGER_ID	Varchar	6	
5	PUKET	Varchar	6	

g. Tabel V_RNG

Primary Key : ID

Foreign Key : -

Fungsi : membaca tabel ruang

Tabel 3.23 Struktur Tabel V_RNG

No.	Field	Type Data	Length	Keterangan
1	ID	Varchar	5	Primary Key
2	KAP	Number	3	
3	BARIS	Number	2	
4	KOLOM	Number	2	
5	SISA	Number	3	
6	LANTAI	Varchar	30	
7	MAC_ADDRESS	Varchar	50	

h. Tabel MK_DIGUNAKAN

Primary Key : ID_PRODI dan ID_MATAKULIAH

Foreign Key : ID_PRODI dan ID_MATAKULIAH

Fungsi : menyimpan data mata kuliah yang akan digunakan

Tabel 3.24 Struktur Tabel MK_DIGUNAKAN

No.	Field	Type Data	Length	Keterangan
1	ID_PRODI	Varchar	5	Primary key, Foreign key
2	ID_MATAKULIAH	Varchar	10	Primary key, Foreign key
3	SEMESTER	Varchar	1	
3	SEMESTER	Number	3	

i. Tabel KOMPETENSI_DOSEN

Primary Key : ID_MATAKULIAH dan NIK

Foreign Key : ID_MATAKULIAH dan NIK

Fungsi : menyimpan data kompetensi dosen

Tabel 3.25 Struktur Tabel KOMPETENSI_DOSEN

No.	Field	Type Data	Length	Keterangan
1	ID_MATAKULIAH	Varchar	10	Primary key, Foreign key
2	NIK	Varchar	6	Primary key, Foreign key

j. Tabel RUANG_DIGUNAKAN

Primary Key : ID_RUANG

Foreign Key : ID_RUANG

Fungsi : menyimpan data ruangan yang akan digunakan

Tabel 3.26 Struktur Tabel RUANG_DIGUNAKAN

No.	Field	Type Data	Length	Keterangan
1	ID_RUANG	Varchar	5	Primary key, Foreign key
2	KAPASITAS	Number	3	
3	LANTAI	Varchar	30	

k. Tabel HARI

Primary Key : ID_HARI

Foreign Key : -

Fungsi : membaca data hari

Tabel 3.27 Struktur Tabel HARI

No.	Field	Type Data	Length	Keterangan
1	ID_HARI	Varchar	2	Primary Key
2	NAMA_HARI	Varchar	20	

l. Tabel DATA_WAKTU

Primary Key : ID_WAKTU

Foreign Key : -

Fungsi : menyimpan data sesi waktu

Tabel 3.28 Struktur Tabel DATA_WAKTU

No.	Field	Type Data	Length	Keterangan
1	ID_WAKTU	Varchar	2	Primary Key
2	MULAI	Date		
3	SELESAI	Date		

m. Tabel DATA_SLOT_WAKTU

Primary Key : ID_SLOT_WAKTU

Foreign Key : ID_HARI dan ID_WAKTU

Fungsi : menyimpan data waktu yang digunakan untuk penjadwalan

Tabel 3.29 Struktur Tabel DATA_SLOT_WAKTU

No.	Field	Type Data	Length	Keterangan
1	ID_SLOT_WAKTU	Varchar	2	Primary Key
2	ID_HARI	Varchar	2	Foreign key
3	ID_WAKTU	Varchar	2	Foreign key

n. Tabel KULIAH

Primary Key : ID_KULIAH

Foreign Key : ID_PRODI dan ID_MATAKULIAH

Fungsi : menyimpan data mata kuliah yang telah dibagi

Tabel 3.30 Struktur Tabel KULIAH

No.	Field	Type Data	Length	Keterangan
1	ID_KULIAH	Varchar	4	Primary Key
2	ID_PRODI	Varchar	5	Foreign key
3	ID_MATAKULIAH	Varchar	10	Foreign key
4	NO_URUT	Integer		
5	SEMESTER	Varchar	1	

o. Tabel JADWAL

Primary Key : ID_KULIAH, ID_RUANG dan ID_SLOT_WAKTU

Foreign Key : ID_KULIAH, ID_RUANG dan ID_SLOT_WAKTU

Fungsi : menyimpan data jadwal kuliah

Tabel 3.31 Struktur Tabel JADWAL

No.	Field	Type Data	Length	Keterangan
1	ID_KULIAH	Varchar	4	Primary key, Foreign key
2	ID_RUANG	Varchar	5	Primary key, Foreign key
3	ID_SLOT_WAKTU	Varchar	2	Primary key, Foreign key
4	NIK	Varchar	6	

3.2.3 Perancangan *Interface*

Rancangan *interface* digunakan untuk memberikan gambaran antarmuka sistem yang akan dibuat. Berikut ini adalah rancangan antarmuka dari aplikasi penjadwalan kuliah di institut bisnis dan informatika stikom surabaya dengan menggunakan algoritma genetika.

a. Rancangan Fungsi Menentukan MK yang Diselenggarakan

Fungsi menentukan mata kuliah yang diselenggarakan berfungsi untuk menentukan mata kuliah apa saja yang akan diselenggarakan pada semester yang akan dijadwalkan. Pengguna halaman ini adalah Ketua Prodi. Pada halaman ini akan muncul pilihan semester yang akan dijadwalkan. Dan kemudian akan menampilkan semua daftar mata kuliah berdasarkan semester yang dipilih. Gambar 3.22 menggambarkan Rancangan Halaman Fungsi Menentukan MK yang Diselenggarakan dan gambar 3.23 menggambarkan Rancangan Halaman Data MK yang Diselenggarakan.

Penjadwalan Kuliah

Home

Mata Kuliah

Kompetensi Dosen

Kapasitas

Hasil Jadwal

Pilih semester yang akan dijadwalkan Program Studi xxxxxxxx

Semester :

Selanjutnya

Gambar 3.22 Rancangan Halaman Menentukan MK yang Digunakan

Penjadwalan Kuliah

Home

Mata Kuliah

Kompetensi Dosen

Kapasitas

Hasil Jadwal

Mata Kuliah diselenggarakan Program Studi xxxxxxxx

Tambah MK

No	Nama MK	semester	Aksi
1	MK1	xx	Hapus
2	MK2	xx	Hapus
3	MK3	xx	Hapus
4	MK4	xx	Hapus
5	MK5	xx	Hapus
6	MK6	xx	Hapus
7	MK7	xx	Hapus
8	MK8	xx	Hapus
9	MK9	xx	Hapus
10	MK10	xx	Hapus

Gambar 3.23 Rancangan Halaman Data MK yang Diselenggarakan.

Pada fungsi menentukan matakuliah yang diselenggarakan terdapat tampilan tambah matakuliah lain yang akan diselenggarakan bertujuan untuk menambah data matakuliah lain yang masih belum masuk ke dalam daftar untuk diselenggarakan. Pengguna dapat memilih dan menambah beberapa matakuliah lagi dengan menyentang matakuliah yang dipilih, lalu

menyimpannya ke dalam daftar matakuliah yang diselenggarakan. Pengguna halaman ini adalah Ketua Prodi. Untuk memilihnya dengan cara mencentang nama mata kuliah yang akan ditambahkan. Gambar 3.24 menggambarkan Rancangan Tampilan Tambah MK Diselenggarakan.

Pilih	Kode MK	Nama MK	semester
<input checked="" type="checkbox"/>	1	MK1	xx
<input type="checkbox"/>	2	MK2	xx
<input checked="" type="checkbox"/>	3	MK3	xx
<input type="checkbox"/>	4	MK4	xx
<input checked="" type="checkbox"/>	5	MK5	xx
<input checked="" type="checkbox"/>	6	MK6	xx
<input type="checkbox"/>	7	MK7	xx
<input checked="" type="checkbox"/>	8	MK8	xx
<input type="checkbox"/>	9	MK9	xx
<input checked="" type="checkbox"/>	10	MK10	xx
<input checked="" type="checkbox"/>	11	MK11	xx
<input type="checkbox"/>	12	MK12	xx

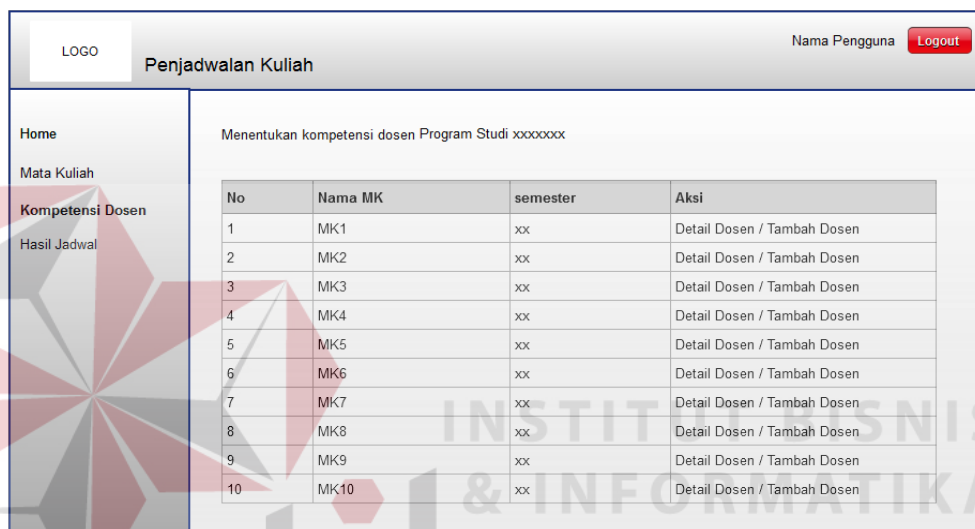
Simpan Batal

Gambar 3.24 Rancangan Tampilan Tambah MK Diselenggarakan.

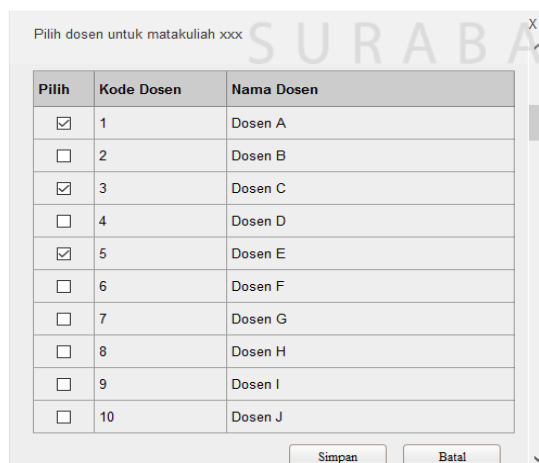
b. Rancangan Fungsi Menentukan Kompetensi Dosen

Fungsi menentukan kompetensi dosen berfungsi untuk menentukan kompetensi dosen berdasarkan mata kuliah setiap prodi. Pengguna halaman ini adalah Ketua Prodi. Untuk menentukan kompetensi dosen dengan memilih mata kuliah yang dimiliki oleh setiap prodi, dan dilanjutkan dengan memilih tambah dosen. Gambar 3.25 menggambarkan Rancangan Halaman Menentukan Kompetensi Dosen.

Melalui halaman menentukan kompetensi dosen, pengguna dapat menambahkan kompetensi dosen dan melihat detail kompetensi dosen setiap mata kuliah yang tersedia. Rancangan tampilan tambah dosen digambarkan pada gambar 3.26 dan rancangan tampilan detail dosen digambarkan pada gambar 3.27



Gambar 3.25 Rancangan Halaman Menentukan Kompetensi Dosen



Gambar 3.26 Rancangan Tampilan Tambah Dosen.

Dosen Untuk matakuliah xxx		
No	Nama	Aksi
1	MK1	Hapus
2	MK2	Hapus
3	MK3	Hapus
4	MK4	Hapus

Gambar 3.27 Rancangan Tampilan Detail Dosen.

c. Rancangan Fungsi Menentukan Kapasitas Setiap Matakuliah

Fungsi menentukan kapasitas setiap matakuliah berfungsi untuk menentukan kapasitas setiap matakuliah berdasarkan mata kuliah setiap prodi yang diselenggarakan. Pengguna halaman ini adalah Ketua Prodi. Untuk menentukan kapasitas setiap matakuliah dengan memilih mata kuliah yang ada oleh setiap prodi, dan dilanjutkan dengan menambahkan jumlah kapasitas setiap matakuliah. Gambar 3.28 menggambarkan Rancangan Halaman Menentukan jumlah kapasitas setiap matakuliah.

LOGO
Penjadwalan Kuliah
Nama Pengguna [Logout](#)

Home

Mata Kuliah

Kompetensi Dosen

Kapasitas

Hasil Jadwal

Tentukan Kapasitas Setiap Matakuliah Program Studi xxxxxxx

[Simpan](#)

No	Nama MK	semester	Kapasitas
1	MK1	xx	<input type="text" value="0"/>
2	MK2	xx	<input type="text" value="0"/>
3	MK3	xx	<input type="text" value="0"/>
4	MK4	xx	<input type="text" value="0"/>
5	MK5	xx	<input type="text" value="0"/>
6	MK6	xx	<input type="text" value="0"/>
7	MK7	xx	<input type="text" value="0"/>
8	MK8	xx	<input type="text" value="0"/>
9	MK9	xx	<input type="text" value="0"/>
10	MK10	xx	<input type="text" value="0"/>

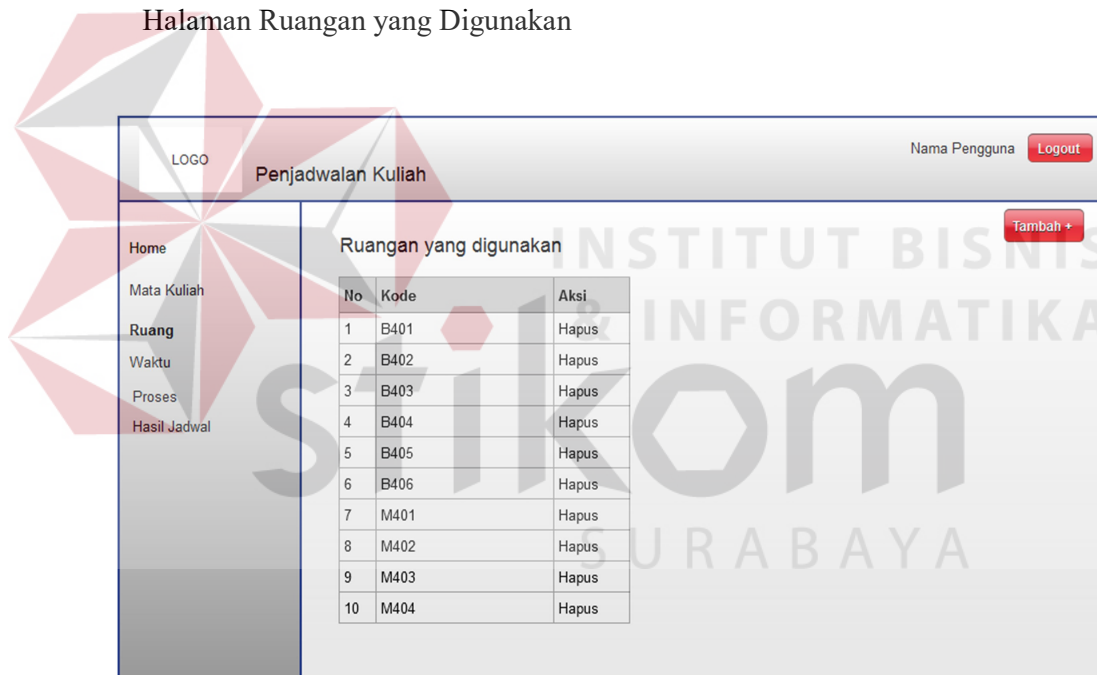
Gambar 3.28 Rancangan Halaman Menentukan Kapasitas Setiap Matakuliah

Melalui halaman menentukan kapasitas setiap matakuliah, pengguna dapat menambahkan kapasitas setiap mata kuliah yang tersedia.

d. Rancangan Fungsi Ruangan yang Digunakan

Halaman ruangan yang digunakan berfungsi untuk melihat data ruangan mana saja yang akan digunakan. Pengguna halaman ini adalah Kepala Bagian AAK. Pada halaman ini akan muncul tombol tambah, yang berfungsi untuk menampilkan data ruangan yang tersedia. Dan kemudian akan menampilkan semua daftar ruangan yang tersedia. Gambar 3.29 menggambarkan Rancangan

Halaman Ruangan yang Digunakan



Gambar 3.29 Rancangan Halaman Ruangan yang Digunakan

Dalam halaman ruangan digunakan terdapat tampilan tambah ruangan yang berfungsi untuk menambah ruangan yang akan digunakan dengan cara mencentang ruangan yang dipilih kemudian klik simpan. Gambar 3.30 menggambarkan Rancangan Tampilan Tambah Ruangan yang Digunakan.

Pilih ruangan yang digunakan

No	Kode Ruang	Lantai	Pilih
1	B401	4	<input checked="" type="checkbox"/>
2	B402	4	<input type="checkbox"/>
3	B403	4	<input type="checkbox"/>
4	B404	4	<input checked="" type="checkbox"/>
5	B405	4	<input type="checkbox"/>
6	B406	4	<input checked="" type="checkbox"/>
7	M401	4	<input type="checkbox"/>
8	M402	4	<input type="checkbox"/>
9	M403	4	<input type="checkbox"/>
10	M404	4	<input checked="" type="checkbox"/>

Simpan Batal

Gambar 3.30 Rancangan Tampilan Tambah Ruangan yang Digunakan

e. Rancangan Fungsi Menentukan Sebaran Waktu

Halaman sebaran waktu berfungsi untuk menyimpan dan menampilkan data pembagian sebaran waktu yang akan digunakan untuk penjadwalan. Pengguna halaman ini adalah Kepala Bagian AAK. Gambar 3.31 menggambarkan Rancangan Halaman Sebaran Waktu.

Penjadwalan Kuliah

LOGO Nama Pengguna Logout

Home
Mata Kuliah
Ruang
Waktu
Sesi Jam
Sebaran Waktu
Proses
Hasil Jadwal

Sebaran Waktu

Hari Combo Box
Jam Combo Box

Simpan Batal

No	Hari	Sesi	Aksi
1	xx	xx	Ubah
2	xx	xx	Ubah
3	xx	xx	Ubah

Gambar 3.31 Rancangan Halaman Sebaran Waktu

f. Rancangan Fungsi Pecah Kelas

Halaman pecah kelas berfungsi untuk membagi kelas setiap mata kuliah berdasarkan dengan jumlah peserta yang ada. Pengguna halaman ini adalah Kepala Bagian AAK. Gambar 3.32 menggambarkan Rancangan Halaman *Generate Kelas*.

The screenshot shows a web application interface for 'Penjadwalan Kuliah'. It features a header with a logo, the title 'Penjadwalan Kuliah', and a user login area with 'Nama Pengguna' and a 'Logout' button. A sidebar on the left contains navigation links: Home, Mata Kuliah, Ruang, Waktu, Proses, Pecah Kelas, Proses Jadwal, and Hasil Jadwal. The main content area is titled 'Data Mata Kuliah yang Diselenggarakan' and includes a 'Generate Kelas' button. Below this is a table with 8 rows of course data.

No	Kode MK	Nama MK	SKS	Semester	Prodi	Jumlah Peserta
1	xx	xx	xx	xx	xx	xx
2	xx	xx	xx	xx	xx	xx
3	xx	xx	xx	xx	xx	xx
4	xx	xx	xx	xx	xx	xx
5	xx	xx	xx	xx	xx	xx
6	xx	xx	xx	xx	xx	xx
7	xx	xx	xx	xx	xx	xx
8	xx	xx	xx	xx	xx	xx

Gambar 3.32 Rancangan Halaman Pecah Kelas

The screenshot shows a similar web application interface for 'Penjadwalan Kuliah'. It features a header with a logo, the title 'Penjadwalan Kuliah', and a user login area with 'Nama Pengguna' and a 'Logout' button. A sidebar on the left contains navigation links: Home, Mata Kuliah, Ruang, Waktu, Proses, Pecah Kelas, Proses Jadwal, and Hasil Jadwal. The main content area is titled 'Data Mata Kuliah yang Diselenggarakan'. Below this is a table with 8 rows of course data, including an additional 'Kelas' column.

No	Kode MK	Nama MK	Kelas	SKS	Semester	Prodi
1	xx	xx	xx	xx	xx	xx
2	xx	xx	xx	xx	xx	xx
3	xx	xx	xx	xx	xx	xx
4	xx	xx	xx	xx	xx	xx
5	xx	xx	xx	xx	xx	xx
6	xx	xx	xx	xx	xx	xx
7	xx	xx	xx	xx	xx	xx
8	xx	xx	xx	xx	xx	xx

Gambar 3.33 Rancangan Halaman Kelas Mata Kuliah

Pada halaman pecah kelas setelah memproses pecah kelas maka sistem akan menampilkan data kelas mata kuliah yang telah dibagi berdasarkan jumlah peserta kuliah setiap matakuliah. Gambar 3.33 menggambarkan Rancangan Halaman Kelas Matakuliah.

g. Rancangan Fungsi Proses Penjadwalan

Halaman proses penjadwalan berfungsi untuk memproses *generate* jadwal dengan masukan jumlah kromosom yang dibangkitkan dan masukan maksimal generasi. Tombol *generate* jadwal berfungsi untuk memproses jadwal dalam satu semester berdasarkan masukan dan data yang sebelumnya sudah ditentukan. Pengguna halaman ini adalah Kepala Bagian AAK. Gambar 3.34 menggambarkan Rancangan Proses Penjadwalan.

Gambar 3.34 Rancangan Halaman Proses Penjadwalan

h. Rancangan Fungsi Laporan Jadwal Kuliah dalam Satu Semester

Halaman laporan jadwal kuliah berfungsi untuk menampilkan hasil dari proses pembuatan jadwal yang dilakukan pada tahap sebelumnya. Pada halaman ini juga dapat menentukan dosen yang ada berdasarkan kompetensi masing-

masing. Pengguna halaman ini adalah Kepala Bagian AAK. Gambar 3.35 menggambarkan Rancangan Halaman Menentukan Jadwal Dosen.

Penjadwalan Kuliah

Home

Mata Kuliah

Ruang

Waktu

Proses

Hasil Jadwal

Jadwal Kuliah Semester xxx

Export Ms. Excel

No	Nama MK	Kelas	Semester	Hari	Mulai	Selesai	Ruang	Nama Dosen
1	MK1	xx	xx	xxx	xx	xx	xx	Combo Box
2	MK2	xx	xx	xxx	xx	xx	xx	Combo Box
3	MK3	xx	xx	xxx	xx	xx	xx	Combo Box
4	MK4	xx	xx	xxx	xx	xx	xx	Combo Box
5	MK5	xx	xx	xxx	xx	xx	xx	Combo Box
6	MK6	xx	xx	xxx	xx	xx	xx	Combo Box
8	MK8	xx	xx	xxx	xx	xx	xx	Combo Box
9	MK9	xx	xx	xxx	xx	xx	xx	Combo Box
10	MK10	xx	xx	xxx	xx	xx	xx	Combo Box

Gambar 3.35 Rancangan Halaman Menentukan Jadwal Dosen

3.2.4 Perancangan Pengujian dan Evaluasi Aplikasi

Setelah melakukan perancangan dan desain aplikasi penjadwalan kuliah, maka tahap selanjutnya adalah melakukan perencanaan uji coba aplikasi yang akan dilakukan setelah aplikasi selesai dibangun. Pengujian aplikasi dilakukan dengan cara melakukan berbagai percobaan terhadap aplikasi penjadwalan kuliah di Institut Bisnis dan Informatika Stikom Surabaya. Fungsi dari pengujian sistem adalah membuktikan bahwa aplikasi telah berjalan sesuai dengan tujuan dan fungsionalitasnya. Pengujian sistem ini menggunakan metode *Black Box Testing*. Berikut ini adalah perancangan uji coba pada aplikasi penjadwalan kuliah.

a. Rancangan Uji Coba Fungsi Menentukan MK yang Diselenggarakan

Rancangan uji coba fungsi menentukan mata kuliah yang diselenggarakan bertujuan untuk mengetahui kesesuaian fungsi terhadap keluaran yang

dihasilkan oleh fungsi mata kuliah yang diselenggarakan. Rancangan uji coba fungsi menentukan mata kuliah yang diselenggarakan dapat dilihat pada tabel 3.32.

Tabel 3.32 Rancangan Uji Coba Fungsi Menentukan Matakuliah yang Diselenggarakan

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu mata kuliah	Klik menu mata kuliah	Tampilan halaman pilih semester
2	Mengetahui respon aplikasi tidak memilih semester yang dijadwalkan	Data semester	Tampilan halaman data mata kuliah berdasarkan semester yang dipilih

b. Rancangan Uji Coba Fungsi Menentukan Kompetensi Dosen

Rancangan uji coba fungsi menentukan kompetensi dosen bertujuan untuk menentukan kompetensi dosen. Rancangan uji coba fungsi menentukan kompetensi dosen dapat dilihat pada Tabel 3.33.

Tabel 3.33 Rancangan Uji Coba Fungsi Menentukan Kompetensi Dosen

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu kompetensi dosen	Klik menu kompetensi dosen	Tampilan informasi data mata kuliah setiap prodi
2	Mengetahui respon fungsi setelah memilih detail dosen	Klik detail dosen	Tampilan informasi detail dosen setiap mata kuliah
3	Mengetahui respon fungsi setelah memilih tambah dosen	Klik tambah	Informasi data dosen yang tersedia
4	Mengetahui respon fungsi jika data dosen disimpan	Data Dosen	Tampilan informasi data berhasil disimpan

c. Rancangan Uji Coba Fungsi Menentukan Kapasitas Setiap Matakuliah

Rancangan uji coba fungsi menentukan kapasitas setiap matakuliah bertujuan untuk menentukan jumlah kapasitas setiap satu kelas matakuliah yang diselenggarakan. Rancangan uji coba fungsi menentukan kapasitas setiap matakuliah dapat dilihat pada Tabel 3.34.

Tabel 3.34 Rancangan Uji Coba Fungsi Menentukan Kapasitas Setiap Kelas Matakuliah

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu kapasitas	Klik menu kapasitas	Tampilan informasi data mata kuliah yang diselenggarakan
2	Mengetahui respon fungsi setelah memasukkan data kapasitas setiap matakuliah	Masukkan jumlah kapsitas setiap matakuliah dan klik simpan	Tampilan informasi data matakuliah yang diselenggarakan disertai dengan jumlah kapasitas setiap kelasnya

d. Rancangan Uji Coba Fungsi Ruangan yang Digunakan

Rancangan uji coba fungsi ruangan yang digunakan bertujuan untuk menentukan ruangan yang akan digunakan. Rancangan uji coba fungsi Ruangan yang Digunakan dapat dilihat pada Tabel 3.35.

Tabel 3.35 Rancangan Uji Coba Fungsi Ruangan yang Digunakan

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu ruang	Klik menu ruang	Tampilan halaman ruangan yang digunakan
2	Mengetahui respon fungsi memilih ruangan yang	Pilih tambah ruang, tentukan	Tampilan informasi data ruangan yang

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
	digunakan dan disimpan	ruangan yang digunakan dan selanjutnya disimpan	tersedia dan tampilan informasi berhasil disimpan

e. Rancangan Uji Coba Fungsi Sebaran Waktu

Rancangan uji coba fungsi ruangan yang digunakan bertujuan untuk menentukan ruangan yang akan digunakan. Rancangan uji coba fungsi Sebaran Waktu dapat dilihat pada Tabel 3.36.

Tabel 3.36 Rancangan Uji Coba Fungsi Sebaran Waktu

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu sebaran waktu	Klik menu sebaran waktu	Tampilan halaman <i>form</i> sebaran waktu
2	Mengetahui respon fungsi jika data sebaran waktu dimasukkan dan disimpan	Masukan data sebaran waktu	Tampilan informasi data berhasil disimpan dan informasi data sebaran waktu yang disimpan

f. Rancangan Uji Coba Fungsi Memproses Pecah Kelas

Rancangan uji coba fungsi memproses pecah kelas bertujuan untuk menentukan jumlah kelas kuliah yang akan dijadwalkan. Rancangan uji coba fungsi pecah kelas dapat dilihat pada Tabel 3.37.

Tabel 3.37 Rancangan Uji Coba Fungsi Memproses Pecah Kelas

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu pecah kelas	Klik menu pecah kelas	Tampilan informasi data mata kuliah diselenggarakan
2	Mengetahui respon fungsi pecah kelas dan disimpan	Masukan data prediksi jumlah peserta setiap matakuliah, klik tombol pecah kelas	Tampilan informasi data berhasil disimpan dan informasi data kelas kuliah yang akan dijadwalkan

g. Rancangan Uji Coba Fungsi Memproses Jadwal Kuliah

Rancangan uji coba fungsi memproses jadwal kuliah bertujuan untuk memproses jadwal kuliah untuk satu semester berdasarkan dengan data yang sudah ada sebelumnya. Rancangan uji coba fungsi memproses jadwal kuliah dapat dilihat pada Tabel 3.38.

Tabel 3.38 Rancangan Uji Coba Fungsi Memproses Jadwal Kuliah

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu proses penjadwalan	Klik menu Proses Penjadwalan	Tampilan halaman proses penjadwalan
2	Mengetahui respon halaman jika menekan tombol <i>generate</i> jadwal	Klik tombol <i>generate</i> jadwal	Tampilan informasi data berhasil diproses dan disimpan

h. Rancangan Uji Coba Fungsi Hasil Jadwal

Rancangan uji coba Fungsi Hasil Jadwal bertujuan untuk melihat hasil jadwal yang sudah jadi. Rancangan uji coba fungsi hasil jadwal dapat dilihat pada Tabel 3.39.

Tabel 3.39 Rancangan Uji Coba Halaman Hasil Jadwal

No	Tujuan	Skenario Pengujian	Hasil Yang Diharapkan
1	Mengetahui respon aplikasi setelah proses memilih menu lihat jadwal	Klik menu lihat jadwal	Tampilan halaman jadwal kuliah yang sudah jadi
2	Mengetahui respon aplikasi setelah proses memilih menu lihat jadwal jika jadwal belum tersedia	Klik menu lihat jadwal	Informasi jadwal belum tersedia

