

IF-ITB-WEB-DOC-2009

**DOKUMEN PEMBANGUNAN PERANGKAT LUNAK**

**SISTEM INFORMASI AKADEMIK – PERWALIAN**

**Dokumen Revisi Berdasarkan Review Kelompok Lain**

Dipersiapkan oleh:

Faisal Ibrahim Hadiputra 13509048

Rido Ramadan 13509049

Alif Finandhita 23511034

Muhammad Ghufro M. 23512066

Ni Made Satvika Iswari 23512077

Program Studi Magister Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

# Daftar Isi

1	Kebutuhan Perangkat Lunak .....	5
1.1	Deskripsi Umum Sistem .....	5
1.2	Batasan Sistem .....	6
1.3	Model Use Case .....	7
1.3.1	Definisi Aktor.....	7
1.3.2	Definisi Use Case .....	8
1.3.3	Skenario Use Case.....	10
1.3.3.1	Skenario Use Case Menawarkan Mata Kuliah.....	10
1.3.3.1.1	Activity Diagram Skenario Use Case Menawarkan Mata Kuliah .....	10
1.3.3.2	Skenario Use Case Entri Mata Kuliah .....	11
1.3.3.2.1	Activity Diagram Use Case Entri Mata Kuliah .....	13
1.3.3.3	Skenario Use Case Daftar PRS .....	15
1.3.3.3.1	Activity Diagram Use Case Daftar PRS.....	15
1.3.3.4	Skenario Use Case Melihat Status Mahasiswa .....	16
1.3.3.4.1	Activity Diagram Use Case Melihat Status Mahasiswa .....	17
1.3.3.5	Skenario Use Case Melihat Entri Mata Kuliah.....	17
1.3.3.5.1	Activity Diagram Use Case Melihat Entri Mata Kuliah.....	18
1.3.3.6	Skenario Use Case Memberikan Persetujuan .....	18
1.3.3.6.1	Activity Diagram Use Case Memberikan Persetujuan .....	19
1.3.3.7	Skenario Use Case Mencetak KSM.....	20
1.3.3.7.1	Activity Diagram Use Case Mencetak KSM.....	21
1.3.3.8	Skenario Use Case Mencetak DPK.....	22
1.3.3.8.1	Activity Diagram Use Case Mencetak DPK .....	22
2	Model Perancangan Arsitektural .....	24
3	Model Perancangan Level Komponen .....	25
3.1	Realisasi Use Case Tahap Perancangan .....	25
3.1.1	Use Case Menawarkan Mata Kuliah.....	25
3.1.1.1	Identifikasi Kelas Perancangan.....	25
3.1.1.2	Sequence Diagram .....	25
3.1.1.2.1	Menawarkan Mata Kuliah .....	25
3.1.1.2.2	Menawarkan Mata Kuliah Tetapi Tidak Disimpan .....	26
3.1.1.3	Diagram Kelas Perancangan .....	26
3.1.2	Use Case Entri Mata Kuliah.....	27
3.1.2.1	Identifikasi Kelas Perancangan.....	27
3.1.2.2	Sequence Diagram .....	27
3.1.2.2.1	Entri Mata Kuliah .....	27
3.1.2.2.2	Memilih Jurusan Lain.....	29
3.1.2.2.3	Kembali Memilih Mata Kuliah dari Jurusan Lain.....	30
3.1.2.2.4	Kembali Mengambil Mata Kuliah Setelah Simpan .....	31
3.1.2.2.5	Hapus Mata Kuliah.....	32
3.1.2.3	Diagram Kelas Perancangan .....	33
3.1.3	Use Case Daftar PRS.....	34
3.1.3.1	Identifikasi Kelas Perancangan.....	34
3.1.3.2	Sequence Diagram .....	34

3.1.3.2.1	Daftar PRS tanpa mengubah Rencana Studi .....	34
3.1.3.2.2	Daftar PRS dengan mengubah Rencana Studi.....	35
3.1.3.3	Diagram Kelas Perancangan .....	35
3.1.4	Use Case Melihat Status Mahasiswa.....	35
3.1.4.1	Identifikasi Kelas Perancangan .....	35
3.1.4.2	Sequence Diagram .....	36
3.1.4.2.1	Melihat Status Mahasiswa .....	36
3.1.4.2.2	Melihat Status Mahasiswa sebelum Log In .....	36
3.1.4.3	Diagram Kelas Perancangan .....	37
3.1.5	Use Case Melihat Entri Mata Kuliah .....	37
3.1.5.1	Identifikasi Kelas Perancangan .....	37
3.1.5.2	Sequence Diagram .....	37
3.1.5.2.1	Melihat Entri Mata Kuliah.....	37
3.1.5.3	Diagram Kelas Perancangan .....	39
3.1.6	Use Case Memberikan Persetujuan.....	39
3.1.6.1	Identifikasi Kelas Perancangan .....	39
3.1.6.2	Sequence Diagram .....	39
3.1.6.2.1	Memberikan Keputusan Menyetujui .....	40
3.1.6.2.2	Memberikan Keputusan Menolak .....	40
3.1.6.3	Diagram Kelas Perancangan .....	41
3.1.7	Use Case Mencetak KSM .....	41
3.1.7.1	Identifikasi Kelas Perancangan .....	41
3.1.7.2	Sequence Diagram .....	41
3.1.7.2.1	Mencetak Seluruh KSM .....	42
3.1.7.2.2	Mencetak Sebagian KSM.....	42
3.1.7.3	Diagram Kelas Perancangan .....	43
3.1.8	Use Case Mencetak DPK .....	43
3.1.8.1	Identifikasi Kelas Perancangan .....	43
3.1.8.2	Sequence Diagram .....	43
3.1.8.2.1	Mencetak Seluruh DPK.....	43
3.1.8.2.2	Mencetak Sebagian DPK.....	44
3.1.8.3	Diagram Kelas Perancangan .....	45
3.2	Diagram Kelas Keseluruhan .....	45
3.3	Rincian Kelas .....	47
3.3.1	Kelas ProdiData.....	47
3.3.2	Kelas KSMData.....	47
3.3.3	Kelas KuliahData .....	48
3.3.4	Kelas MahasiswaData .....	48
3.3.5	Kelas StatusController.....	49
3.3.6	Kelas StatusInterface.....	50
3.3.7	Kelas LoginInterface .....	50
3.3.8	Kelas TawarMakulInterface .....	50
3.3.9	Kelas TawarMakulController.....	51
3.3.10	Kelas EntryInterface.....	52
3.3.11	Kelas EntryController .....	52
3.3.12	Kelas PRSInterface .....	53
3.3.13	Kelas PRSController .....	53
3.3.14	Kelas PersetujuanInterface .....	54
3.3.15	Kelas PersetujuanController.....	54

3.3.16	Kelas CetakKSMInterface.....	55
3.3.17	Kelas CetakKSMController .....	56
3.3.18	Kelas CetakDPKInterface .....	56
3.3.19	Kelas CetakDPKController .....	57
3.3.20	Kelas DepKeuInterface .....	58
3.4	State Chart Diagram.....	59
3.4.1	Kelas EntryController .....	59
3.4.2	Kelas PersetujuanController.....	60
4	Model Perancangan Basis Data .....	61
5	Model Perancangan Antarmuka/Navigasi.....	63
5.1	Antarmuka Homepage .....	63
5.2	Antarmuka Entri FRS.....	63
5.3	Antarmuka Pendaftaran PRS.....	64
5.4	Antarmuka PRS Sukses.....	64
5.5	Antarmuka Status Mahasiswa .....	65
5.6	Antarmuka List Mahasiswa oleh Dosen Wali.....	65
5.7	Antarmuka Status Mahasiswa oleh Dosen Wali .....	66
6	Tracebility Use Case .....	66
7	Tanggapan Hasil Review.....	67

# 1 Kebutuhan Perangkat Lunak

## 1.1 Deskripsi Umum Sistem

Sistem Informasi Akademik adalah sebuah aplikasi untuk mendukung kegiatan administrasi perkuliahan seperti perwalian, pemilihan mata kuliah, pencetakan KSM, dan lain sebagainya. Sistem ini dibuat sebagai solusi untuk mempermudah kegiatan administrasi perkuliahan di sebuah institusi pendidikan.

Sistem ini memiliki fitur utama yaitu fitur pendaftaran studi mahasiswa. Proses ini diawali dengan penawaran mata kuliah di awal semester oleh Kurikulum Management. Kurikulum Management akan memberikan status untuk setiap mata kuliah yang ada, agar mahasiswa dapat menentukan mata kuliah yang akan diambil pada semester yang bersangkutan. Adapun data status mata kuliah yang dimasukkan berupa data kode mata kuliah, nama mata kuliah, jumlah sks, kelas dibuka, kelas ditutup, jadwal, dosen, dsb.

Setelah status mata kuliah ditetapkan, proses dilanjutkan dengan pemilihan mata kuliah oleh mahasiswa. Mahasiswa dapat memilih mata kuliah yang akan diambil pada semester yang bersangkutan. Mata kuliah dapat dipilih berdasarkan tabel mata kuliah yang dibuka per program studi. Setelah memilih mata kuliah, mahasiswa dapat melihat mata kuliah apa saja yang dipilih dan jumlah sks total, yang kemudian akan diteruskan ke dosen wali.

Proses selanjutnya adalah *review* oleh dosen wali. Dosen wali dapat melihat daftar mahasiswa, baik yang telah mengentri mata kuliah maupun yang belum. Dosen wali dapat menyetujui atau menolak rencana studi yang dimasukkan oleh mahasiswa. Jika disetujui, maka KSM mahasiswa dapat dicetak. Jika rencana studi ditolak, maka mahasiswa dapat mengentri kembali rencana studinya untuk di-*review* kembali oleh dosen wali.

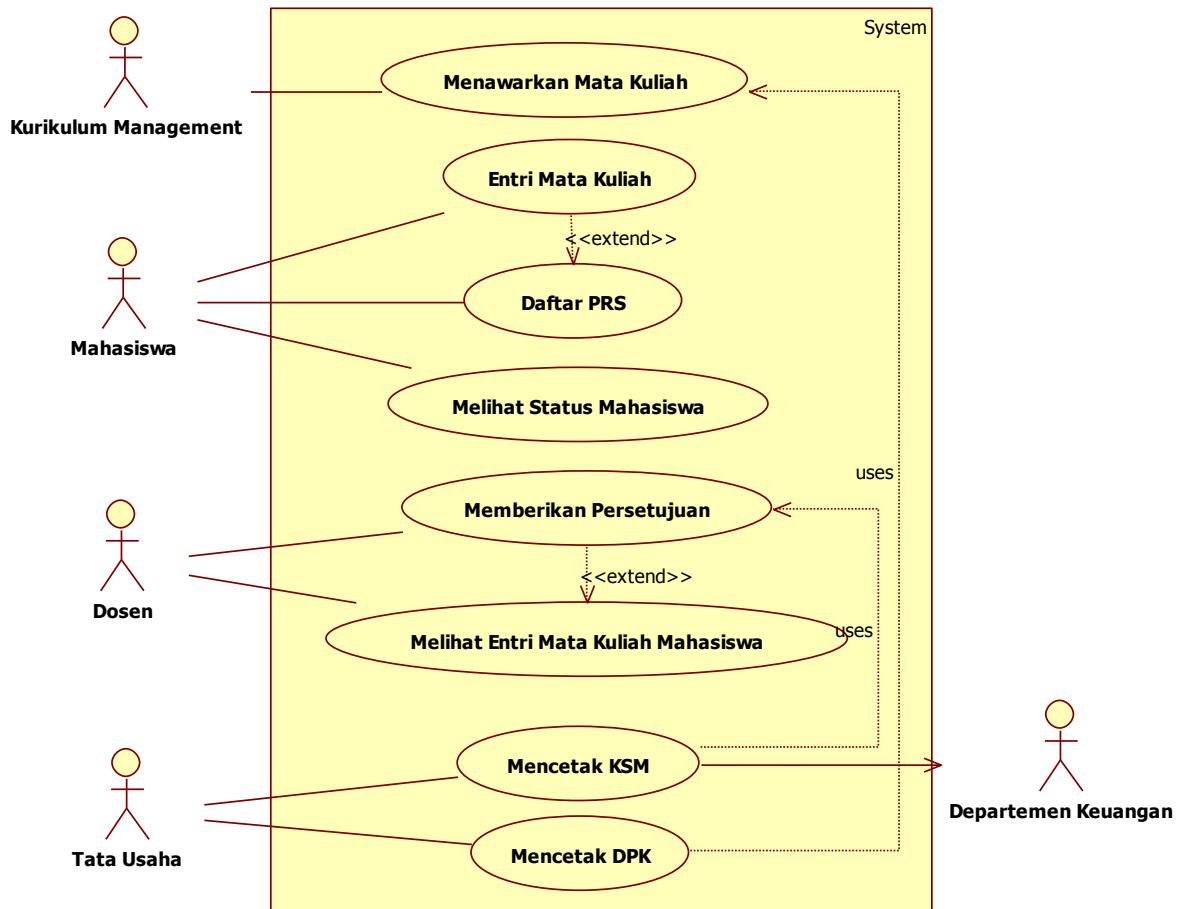
Setelah batas waktu perwalian berakhir, proses pendaftaran mata kuliah (perwalian) akan diakhiri dengan pencetakan KSM yang berisi daftar mata kuliah yang diambil dan DPK yang berisi daftar peserta kelas pada mata kuliah tertentu. Syarat pencetakan KSM adalah rencana studi telah disetujui oleh dosen wali dan telah membayar biaya kuliah untuk semester tersebut. Rencana studi yang ditolak oleh dosen wali atau belum membayar mata kuliah tidak akan diproses dan mahasiswa yang bersangkutan dianggap tidak mendaftar untuk semester tersebut.

## 1.2 Batasan Sistem

Batasan yang digunakan dalam pengembangan sistem ini antara lain:

1. Sistem ini hanya melayani proses pendaftaran mata kuliah dan perwalian secara umum. Untuk data tertentu, seperti data pembayaran mahasiswa dari Departemen Keuangan, dihasilkan dari sistem lain yang diakses melalui suatu kelas interface.
2. Sistem perwalian ini aktif pada saat FRS dan PRS.
3. Untuk mengakses sistem ini, pengguna telah login dengan akun dan hak akses tersendiri.
4. Sistem dikembangkan berbasis web dengan menggunakan bahasa PHP dan DBMS berupa MySQL.
5. Sistem dapat melakukan pencetakan KSM dengan syarat rencana studi telah disetujui dosen wali dan telah membayar SPP.
6. Sistem dapat melakukan pencetakan DPK dengan syarat kuliah dibuka untuk semester yang bersangkutan.

### 1.3 Model Use Case



Gambar 1-1 Use Case Diagram

#### 1.3.1 Definisi Aktor

No	Aktor	Deskripsi
1	Kurikulum Management	Aktor dengan <i>role</i> ini memiliki wewenang untuk memasukan daftar mata kuliah yang ditawarkan
2	Mahasiswa	Aktor dengan <i>role</i> ini memiliki wewenang untuk melakukan pengisian rencana studi
3	Dosen	Aktor dengan <i>role</i> ini memiliki wewenang untuk menyetujui/menolak rencana studi mahasiswa
4	Tata Usaha	Aktor dengan <i>role</i> ini memiliki wewenang untuk melakukan pencetakan KSM

### 1.3.2 Definisi Use Case

No	Use Case	Deskripsi
1	Menawarkan Mata Kuliah	<p><b>Kode Use Case:</b> UC-01</p> <p><b>Deskripsi Singkat:</b> Sistem menyediakan menu penawaran mata kuliah untuk setiap program studi, sistem dapat menyimpan status mata kuliah (dibuka/ditutup, dosen, jadwal, dan sks) yang dimasukkan oleh Kurikulum Management</p> <p><b>Aktor:</b> Kurikulum Management</p> <p><b>Trigger:</b> Aktor memilih menu “Menawarkan Mata Kuliah”</p> <p><b>Prekondisi:</b> Halaman Menu Utama untuk Kurikulum Management sudah ditampilkan, database telah terupdate dengan data mata kuliah terbaru</p> <p><b>Pascakondisi:</b> Sistem menyimpan status setiap mata kuliah pada semester tersebut.</p>
2	Entri Mata Kuliah	<p><b>Kode Use Case:</b> UC-02</p> <p><b>Deskripsi singkat:</b> Mahasiswa memilih mata kuliah untuk mengisi rencana studi baik FRS maupun PRS.</p> <p><b>Aktor:</b> Mahasiswa</p> <p><b>Trigger:</b> Aktor memilih menu “Ambil Kuliah”.</p> <p><b>Prekondisi:</b> Mahasiswa telah berada pada form perubahan rencana studi baik dari menu FRS maupun telah mendaftar dari PRS.</p> <p><b>Pascakondisi:</b> Rencana studi sementara telah dibuat dan status mahasiswa tercatat telah mengisi rencana studi untuk diajukan ke dosen wali.</p>
3	Daftar PRS	<p><b>Kode Use Case:</b> UC-03</p> <p><b>Deskripsi singkat:</b> Mahasiswa mendaftarkan diri untuk PRS.</p> <p><b>Aktor:</b> Mahasiswa</p> <p><b>Trigger:</b> Aktor memilih tombol Daftar PRS.</p> <p><b>Prekondisi:</b> Mahasiswa sudah login dan pernah mengambil FRS pada masa pendaftaran FRS.</p> <p><b>Pascakondisi:</b> Rencana studi sementara telah dibuat baik telah diubah dari rencana studi saat FRS maupun tidak diubah.</p>
4	Melihat Status Mahasiswa	<p><b>Kode Use Case:</b> UC-04</p> <p><b>Deskripsi singkat:</b> Melihat status pengambilan rencana studi pada semester yang bersangkutan.</p> <p><b>Aktor:</b> Mahasiswa</p> <p><b>Trigger:</b> Aktor mengakses URL status mahasiswa via klik hyperlink, login, atau mengetik URL langsung.</p> <p><b>Prekondisi:</b> Tidak ada.</p> <p><b>Pascakondisi:</b> Terdapat tampilan status mahasiswa pada periode akademik saat program dijalankan.</p>
5	Melihat Entri Mata Kuliah	<p><b>Kode Use Case:</b> UC-05</p> <p><b>Deskripsi singkat:</b> Aplikasi menampilkan daftar mata kuliah yang dipilih oleh mahasiswa.</p> <p><b>Aktor:</b> Dosen.</p> <p><b>Trigger:</b> Aktor memilih menu Daftar Mahasiswa Wali.</p>



		<p><b>Prekondisi:</b> Dosen telah login, dosen wali telah dipasangkan dengan mahasiswa yang sesuai, berada dalam masa FRS atau PRS, mata kuliah telah dipilih oleh mahasiswa, menu Pendaftaran dan Perwalian Mahasiswa serta menu Daftar Mahasiswa Dosen Wali telah dipilih.</p> <p><b>Pascakondisi:</b> daftar mata kuliah yang dipilih oleh mahasiswa ditampilkan di layar.</p>
6	Memberikan Persetujuan	<p><b>Kode Use Case:</b> UC-06</p> <p><b>Deskripsi singkat:</b> Aplikasi menerima keputusan rencana studi mahasiswa disetujui atau ditolak oleh dosen wali.</p> <p><b>Aktor:</b> Dosen wali.</p> <p><b>Trigger:</b> Aktor telah menjalankan use case Melihat Entri Mata Kuliah.</p> <p><b>Prekondisi:</b> Dosen wali telah login, dosen wali telah dipasangkan dengan mahasiswa yang sesuai, berada dalam masa FRS atau PRS, mata kuliah telah dipilih oleh mahasiswa, daftar mata kuliah yang dipilih mahasiswa telah ditampilkan (pascakondisi use case Melihat Entri Mata Kuliah).</p> <p><b>Pascakondisi:</b> Keputusan persetujuan disimpan dalam basis data dan notifikasi perubahan persetujuan ditampilkan ke layar.</p>
7	Mencetak KSM	<p><b>Kode Use Case:</b> UC-07</p> <p><b>Deskripsi Singkat:</b> Sistem menerima perintah untuk mencetak KSM berdasarkan data yang diperoleh dari database KSM dan sistem lain, yaitu Departemen Keuangan. KSM yang dicetak oleh sistem adalah KSM yang sudah disetujui dosen wali dan sudah membayar uang kuliah.</p> <p><b>Aktor:</b> Tata Usaha (TU)</p> <p><b>Trigger:</b> Aktor memilih menu "Mencetak KSM"</p> <p><b>Prekondisi:</b> Halaman Menu Utama untuk TU sudah ditampilkan, database KSM telah diupdate dengan data terbaru mengenai status persetujuan dosen wali.</p> <p><b>Pascakondisi:</b> KSM tercetak</p>
8	Mencetak DPK	<p><b>Kode Use Case:</b> UC-08</p> <p><b>Deskripsi Singkat:</b> Sistem menerima perintah untuk mencetak DPK berdasarkan data yang diterima dari database Kuliah dan Mahasiswa. DPK yang dicetak oleh sistem adalah DPK untuk kelas yang dibuka.</p> <p><b>Aktor:</b> Tata Usaha (TU)</p> <p><b>Trigger:</b> Aktor memilih menu "Mencetak DPK"</p> <p><b>Prekondisi:</b> Halaman Menu Utama untuk TU sudah ditampilkan, database Kuliah dan Mahasiswa telah diupdate dengan data terbaru mengenai status pembukaan kuliah.</p> <p><b>Pascakondisi:</b> DPK tercetak</p>

### 1.3.3 Skenario Use Case

#### 1.3.3.1 Skenario Use Case Menawarkan Mata Kuliah

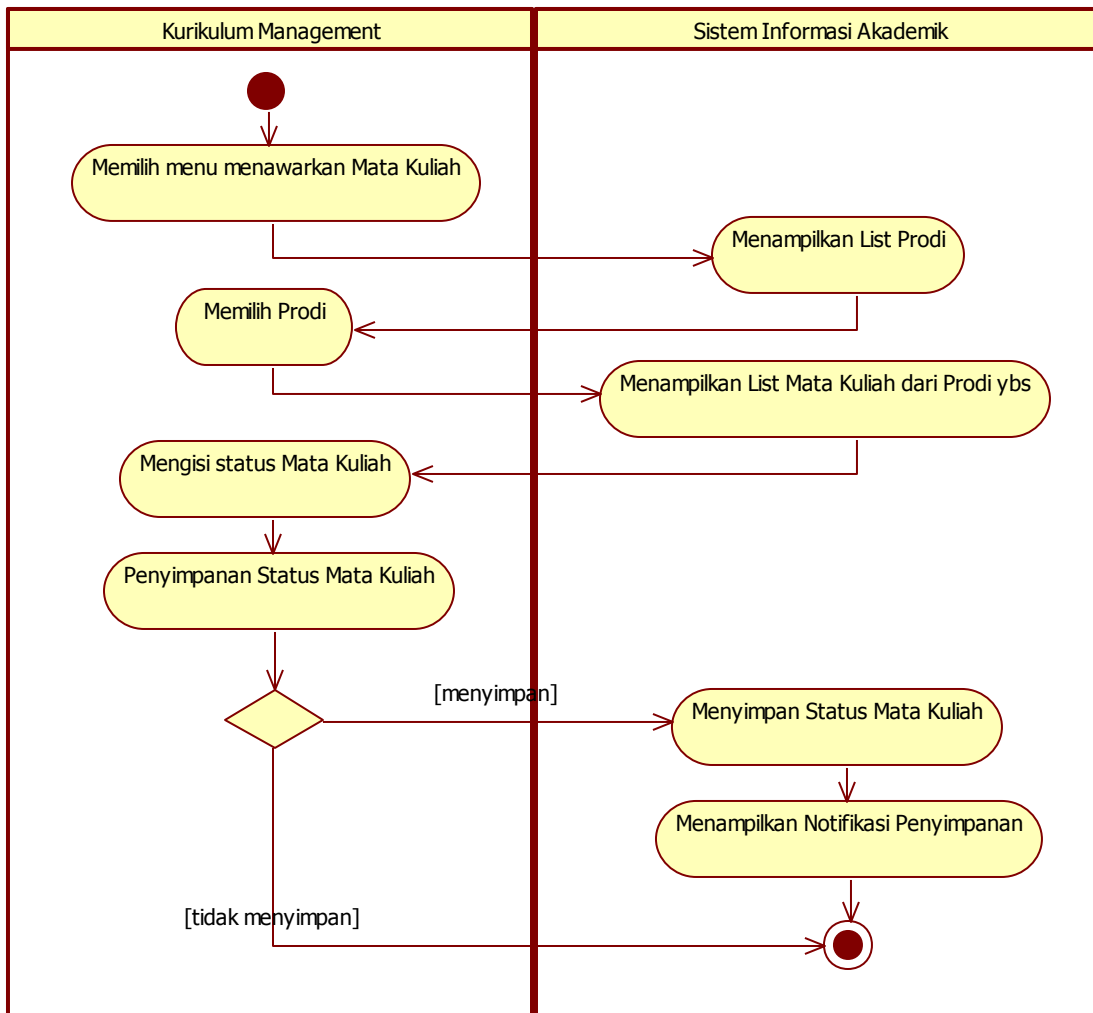
Nama Use Case: Menawarkan Mata Kuliah (UC-01)

Skenario:

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal : Menawarkan Mata Kuliah dan Disimpan SC-01-01</b>	
1. Memilih menu menawarkan mata kuliah	
	2. Menampilkan list program studi
3. Memilih program studi	
	4. Menampilkan list mata kuliah
5. Mengisi status mata kuliah (dibuka/ditutup, dosen, jadwal, sks)	
6. Menyimpan perubahan	
	7. Menyimpan status mata kuliah yang dibuka
	8. Memberikan pesan simpan berhasil
<b>Skenario Normal : Menawarkan Mata Kuliah tetapi Tidak Disimpan SC-01-02</b>	
1. Memilih menu menawarkan mata kuliah	
	2. Menampilkan list program studi
3. Memilih program studi	
	4. Menampilkan list mata kuliah
5. Mengisi status mata kuliah (dibuka/ditutup, dosen, jadwal, sks)	
6. Menutup menu	

##### 1.3.3.1.1 Activity Diagram Skenario Use Case Menawarkan Mata Kuliah

AC-01 Activity Diagram untuk Use Case UC-01 Menawarkan Mata Kuliah



Gambar 1-2 AC-01 Activity Diagram untuk Use Case UC-01 Menawarkan Mata Kuliah

### 1.3.3.2 Skenario Use Case Entri Mata Kuliah

Nama Use Case: Entri Mata Kuliah (UC-02)

Skenario:

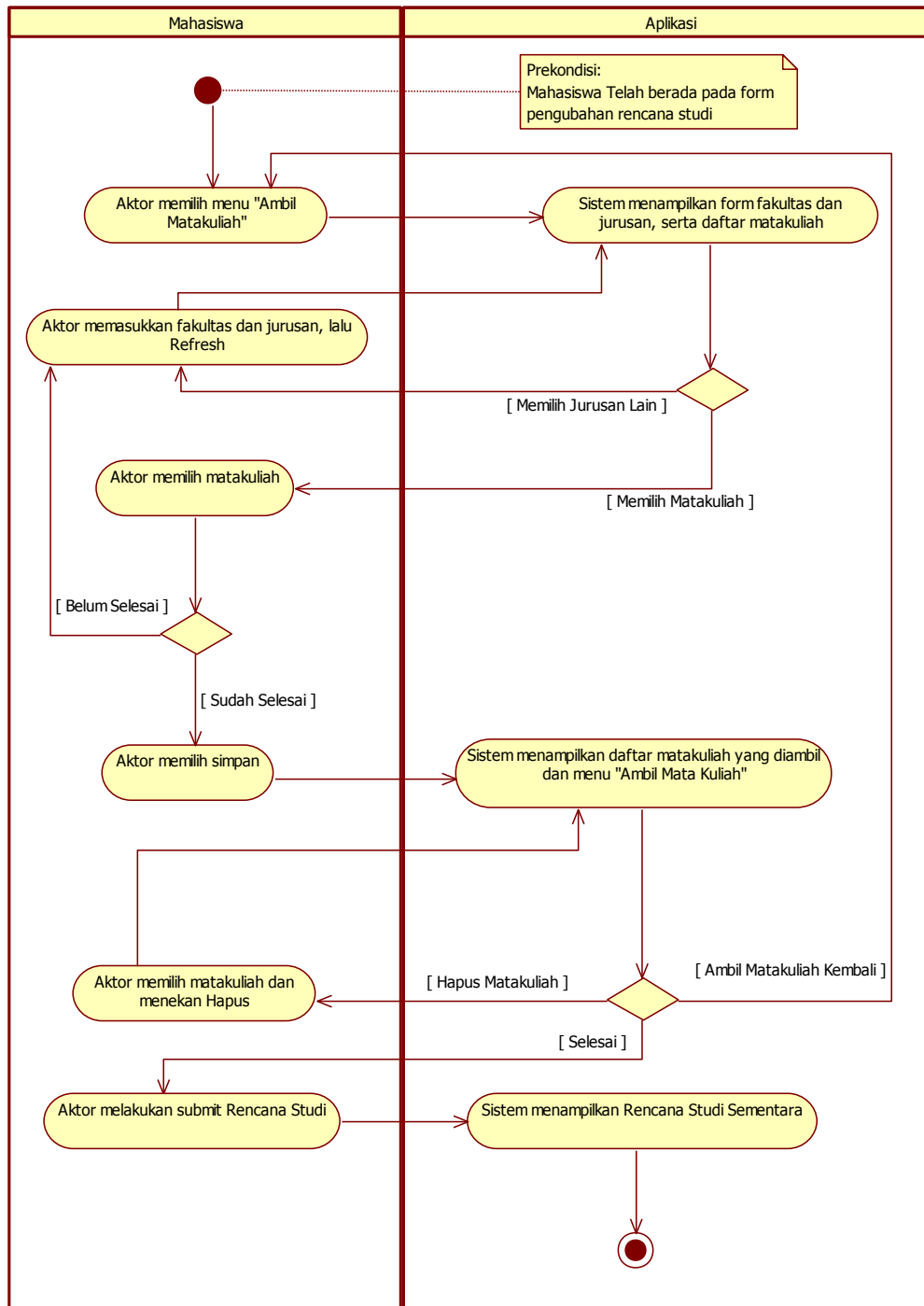
Aksi Aktor	Reaksi Sistem
<b>Skenario Normal : Entri Mata Kuliah SC-02-01</b>	
1. Memilih menu "Ambil Mata Kuliah"	
	2. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
3. Memilih Mata Kuliah	

4. Memilih "Simpan"	
	5. Menampilkan daftar matakuliah yang diambil dan menu "Ambil Mata Kuliah"
6. Melakukan submit Rencana Studi	
	7. Menampilkan Rencana Studi Sementara
<b>Skenario Alternatif 1 : Memilih Jurusan Lain SC-02-02</b>	
1. Memilih menu "Ambil Mata Kuliah"	
	2. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
3. Memasukkan fakultas dan jurusan	
4. Memilih "Refresh"	
	5. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
6. Memilih Mata Kuliah	
7. Memilih "Simpan"	
	8. Menampilkan daftar matakuliah yang diambil dan menu "Ambil Mata Kuliah"
9. Melakukan submit Rencana Studi	
	10. Menampilkan Rencana Studi Sementara
<b>Skenario Alternatif 2 : Kembali Memilih Mata Kuliah dari Jurusan Lain SC-02-03</b>	
1. Memilih menu "Ambil Mata Kuliah"	
	2. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
3. Memilih Mata Kuliah	
4. Memasukkan fakultas dan jurusan	
5. Memilih "Refresh"	
	6. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
7. Memilih Mata Kuliah	
8. Memilih "Simpan"	
	9. Menampilkan daftar matakuliah yang diambil dan menu "Ambil Mata Kuliah"
10. Melakukan submit Rencana Studi	
	11. Menampilkan Rencana Studi Sementara
<b>Skenario Alternatif 3 : Kembali Mengambil Mata Kuliah Setelah Simpan SC-02-04</b>	
1. Memilih menu "Ambil Mata Kuliah"	
	2. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
3. Memilih Mata Kuliah	
4. Memilih "Simpan"	
	5. Menampilkan daftar matakuliah yang diambil dan menu "Ambil Mata Kuliah"

6. Memilih menu “Ambil Mata Kuliah”	
	7. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
8. Memilih Mata Kuliah	
9. Memilih “Simpan”	
	10. Menampilkan daftar matakuliah yang diambil dan menu "Ambil Mata Kuliah"
11. Melakukan submit Rencana Studi	
	12. Menampilkan Rencana Studi Sementara
<b>Skenario Alternatif 4 : Hapus Mata Kuliah SC-02-05</b>	
1. Memilih menu “Ambil Mata Kuliah”	
	2. Menampilkan form fakultas dan jurusan, serta daftar matakuliah
3. Memilih Mata Kuliah	
4. Memilih “Simpan”	
	5. Menampilkan daftar matakuliah yang diambil dan menu "Ambil Mata Kuliah"
6. Memilih Mata Kuliah untuk dihapus	
7. Menekan “Hapus”	
	8. Menampilkan daftar matakuliah yang diambil dan menu "Ambil Mata Kuliah"
9. Melakukan submit Rencana Studi	
	10. Menampilkan Rencana Studi Sementara

### 1.3.3.2.1 Activity Diagram Use Case Entri Mata Kuliah

AC-02 Activity Diagram untuk Use Case UC-02 Entri Mata Kuliah



Gambar 1-3 AC-02 Activity Diagram untuk Use Case UC-02 Entri Mata Kuliah

### 1.3.3.3 Skenario Use Case Daftar PRS

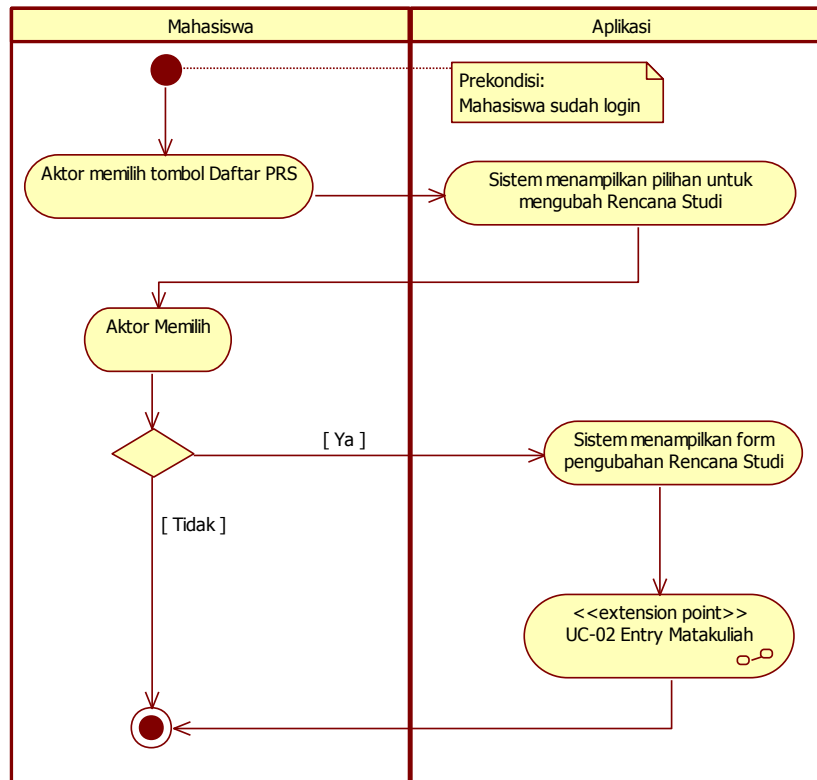
Nama Use Case: Daftar PRS (UC-03)

Skenario:

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal : Daftar PRS tanpa mengubah Rencana Studi SC-03-01</b>	
1. Memilih tombol Daftar PRS	
	2. Menampilkan pilihan untuk mengubah Rencana Studi
<b>Skenario Alternatif 1 : Daftar PRS dengan mengubah Rencana Studi SC-03-02</b>	
1. Memilih tombol Daftar PRS	
	2. Menampilkan pilihan untuk mengubah Rencana Studi
3. Menekan tombol "Ubah Rencana Studi"	
	4. <<extends>> UC-02 Entry Matakuliah

#### 1.3.3.3.1 Activity Diagram Use Case Daftar PRS

AC-03 Activity Diagram untuk Use Case UC-03 Daftar PRS



Gambar 1-4 AC-03 Activity Diagram untuk Use Case UC-03 Daftar PRS

### 1.3.3.4 Skenario Use Case Melihat Status Mahasiswa

Nama Use Case: Melihat Status Mahasiswa (UC-04)

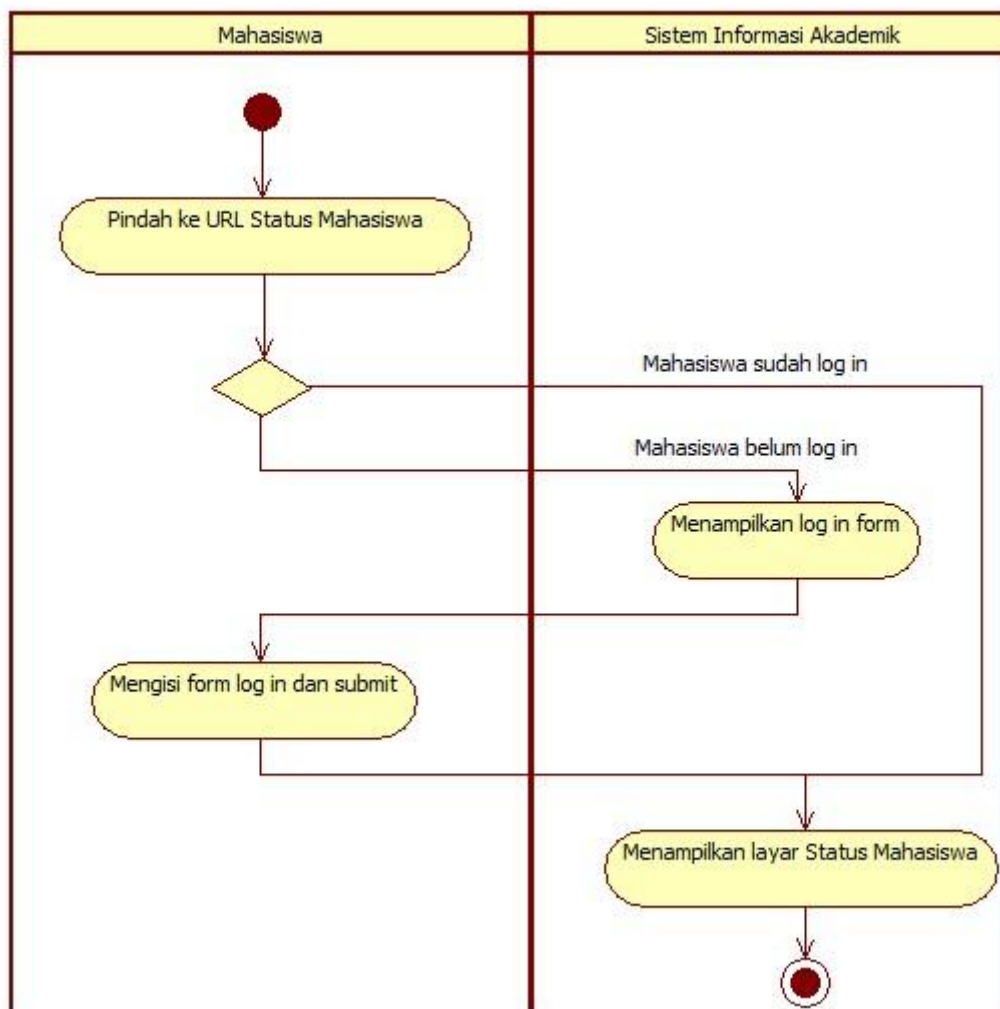
Skenario:

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal : Melihat Status Mahasiswa SC-04-01</b>	
1. Mengakses URL menuju halaman status mahasiswa	
	2. Menampilkan layar status mahasiswa
<b>Skenario Normal : Melihat Status Mahasiswa sebelum Log In (asumsi log in berhasil) SC-04-02</b>	
1. Mengakses URL menuju halaman status mahasiswa	
	2. Menampilkan layar log in
3. Mengisi form log in dan melakukan <i>submission</i>	
	4. Menampilkan layar status mahasiswa



### 1.3.3.4.1 Activity Diagram Use Case Melihat Status Mahasiswa

AC-04 Activity Diagram untuk Use Case UC-04 Melihat Status Mahasiswa



Gambar 1-5 AC-04 Activity Diagram untuk Use Case UC-04 Melihat Status Mahasiswa

### 1.3.3.5 Skenario Use Case Melihat Entri Mata Kuliah

Nama Use Case: Melihat Entri Mata Kuliah (UC-05)

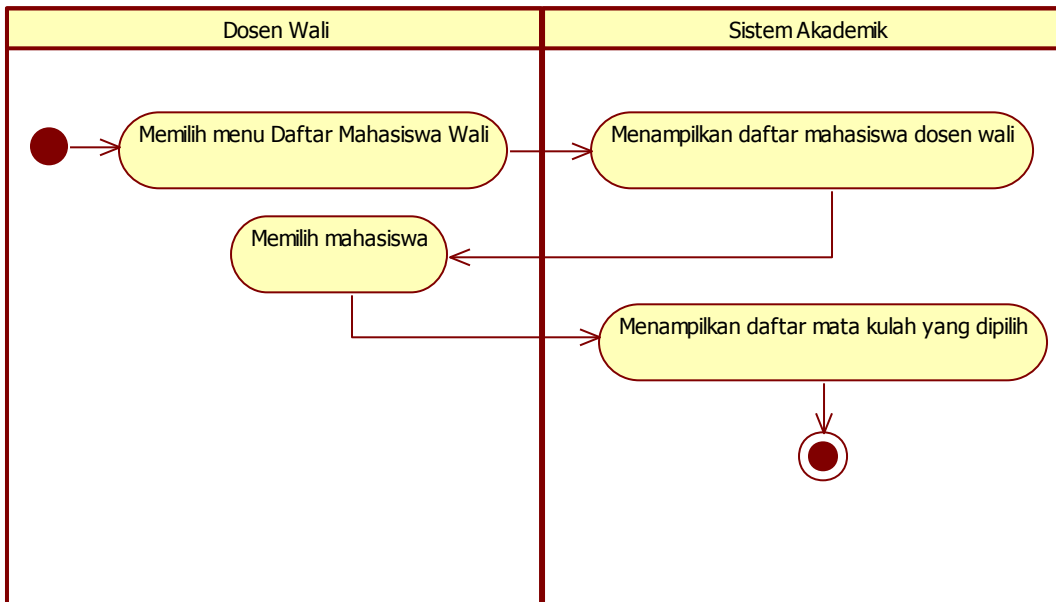
Skenario:

Aksi Aktor	Reaksi Aplikasi
<b>Skenario Normal : Melihat Entri Mata Kuliah</b>	
<b>SC-05-01</b>	
1. Memilih menu Daftar Mahasiswa Wali.	

	2. Menampilkan daftar mahasiswa dosen wali.
3. Memilih salah satu mahasiswa.	
	4. Menampilkan daftar mata kuliah yang dipilih mahasiswa yang bersangkutan.

### 1.3.3.5.1 Activity Diagram Use Case Melihat Entri Mata Kuliah

AC-05 Activity Diagram untuk Use Case UC-05 Melihat Entri Mata Kuliah



Gambar 1-6 AC-05 Activity Diagram untuk Use Case UC-05 Melihat Entri Mata Kuliah

### 1.3.3.6 Skenario Use Case Memberikan Persetujuan

Nama Use Case: Memberikan Persetujuan (UC-06)

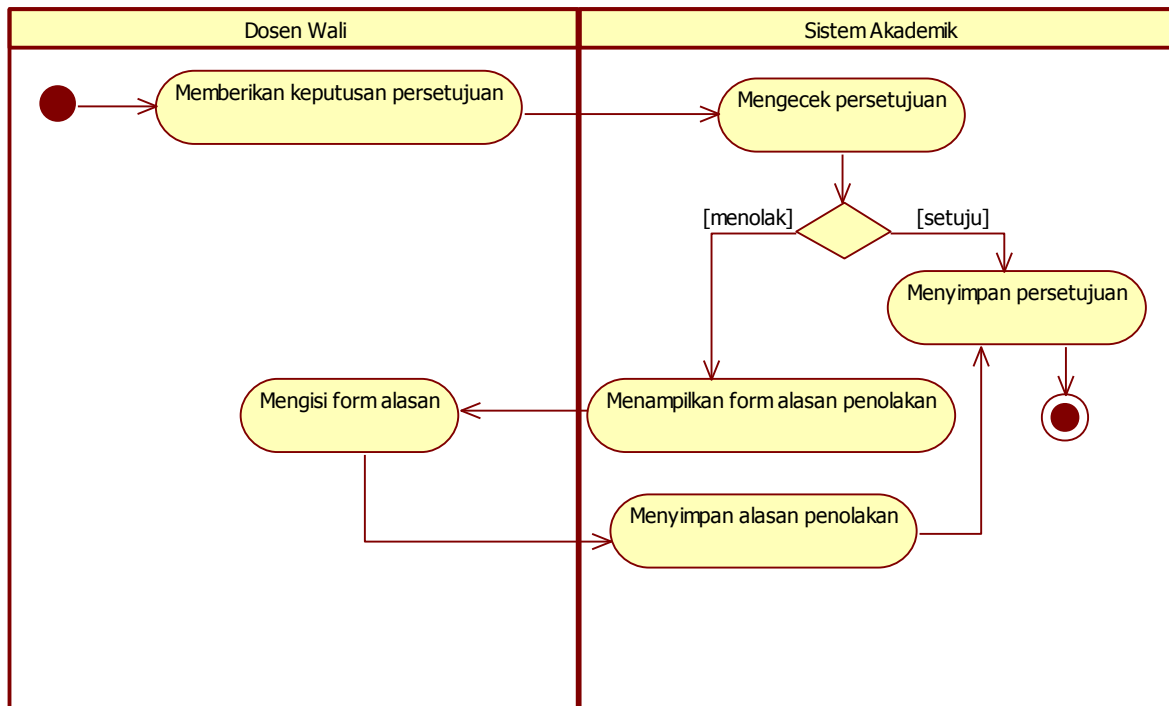
Skenario:

Aksi Aktor	Reaksi Aplikasi
<b>Skenario Normal : Memberikan Keputusan Setuju SC-06-01</b>	
1. (Menjalankan use case Melihat Entri Mata Kuliah).	
	2. Menampilkan daftar mata kuliah yang dipilih mahasiswa.
3. Memberikan keputusan persetujuan (setuju atau menolak).	

	4. Keputusan setuju, menyimpan keputusan ke dalam basis data serta menampilkan notifikasi perubahan keputusan ke layar.
<b>Skenario Alternative 1 : Memberikan Keputusan Menolak SC-06-02</b>	
	4.a. Keputusan menolak, menampilkan form alasan penolakan.
4.b. Mengisi form alasan penolakan	
	4.c. Menyimpan keputusan dan alasan ke dalam basis data serta menampilkan notifikasi perubahan keputusan ke layar.

### 1.3.3.6.1 Activity Diagram Use Case Memberikan Persetujuan

AC-06 Activity Diagram untuk Use Case UC-06 Memberikan Persetujuan



Gambar 1-7 AC-06 Activity Diagram untuk Use Case UC-06 Memberikan Persetujuan

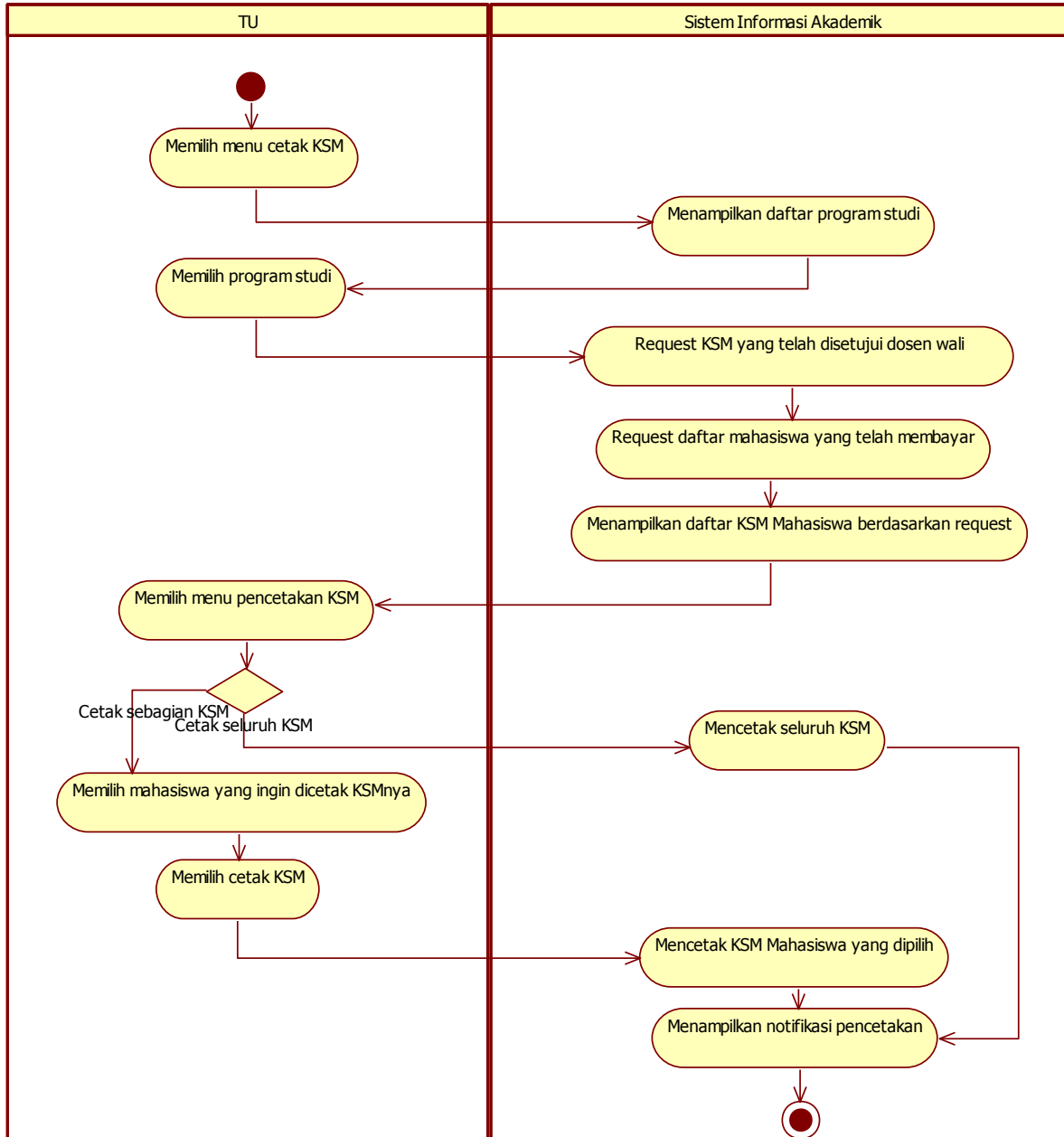
### 1.3.3.7 Skenario Use Case Mencetak KSM

Nama Use Case: Mencetak KSM (UC-07)

Skenario:

Aksi Aktor	Reaksi Sistem
<b>Skenario Normal : Mencetak KSM SC-07-01</b>	
1. Memilih menu cetak KSM	
	2. Menampilkan daftar program studi
3. Memilih program studi	
	4. Melakukan pengecekan status persetujuan dosen wali untuk program studi yang dipilih
	5. Melakukan pengecekan status keuangan mahasiswa untuk program studi yang dipilih
	6. Menampilkan daftar entri mata kuliah mahasiswa yang sudah disetujui dosen wali dan status keuangannya sudah membayar
7. Memilih tombol cetak seluruh KSM	
	8. Mencetak seluruh KSM mahasiswa yang ditampilkan
	9. Menampilkan notifikasi pencetakan
<b>Skenario Alternatif 1 : Mencetak sebagian KSM SC-07-02</b>	
1. Memilih menu cetak KSM	
	2. Menampilkan daftar program studi
3. Memilih program studi	
	4. Melakukan pengecekan status persetujuan dosen wali untuk program studi yang dipilih
	5. Melakukan pengecekan status keuangan mahasiswa untuk program studi yang dipilih
	6. Menampilkan daftar entri mata kuliah mahasiswa yang sudah disetujui dosen wali dan status keuangannya sudah membayar
7. Memilih mahasiswa yang ingin dicetak KSMnya	
8. Memilih tombol cetak KSM	
	8. Mencetak KSM mahasiswa yang dipilih
	9. Menampilkan notifikasi pencetakan

### 1.3.3.7.1 Activity Diagram Use Case Mencetak KSM



Gambar 1-8 AC-07 Activity Diagram untuk Use Case UC-07 Mencetak KSM

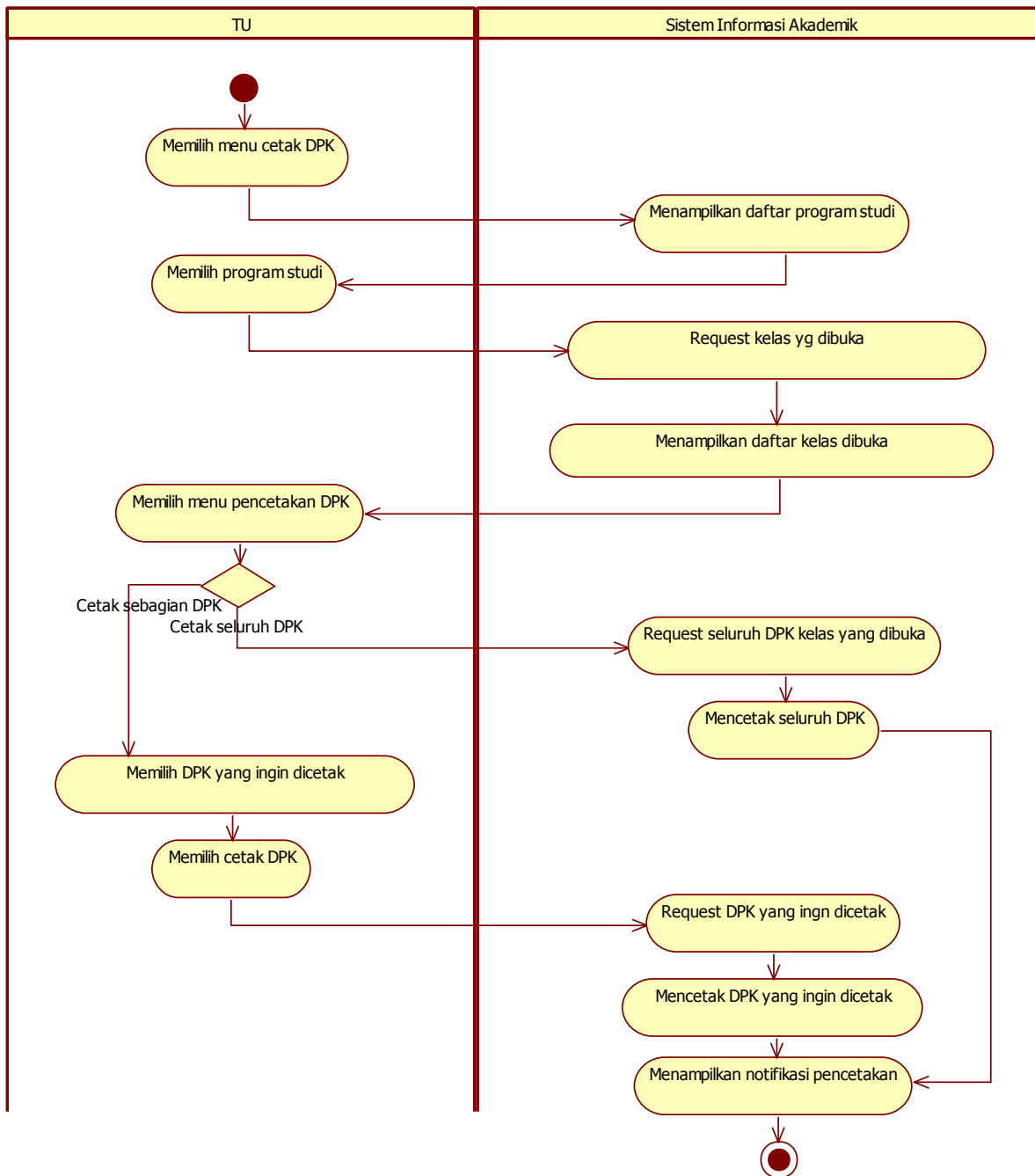
### 1.3.3.8 Skenario Use Case Mencetak DPK

Nama Use Case: Mencetak DPK (UC-08)

Skenario:

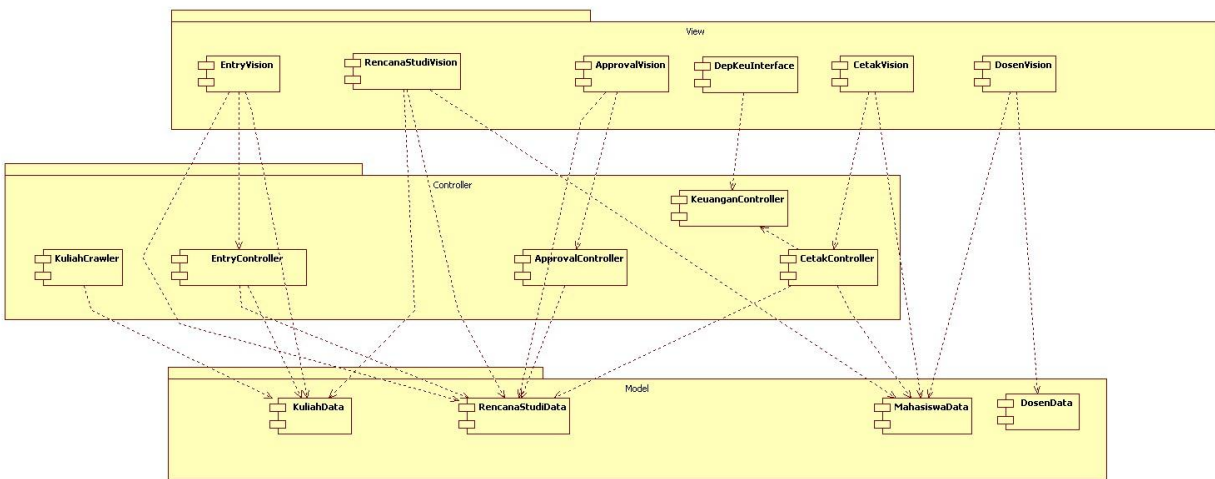
Aksi Aktor	Reaksi Sistem
<b>Skenario Normal : Mencetak DPK SC-08-01</b>	
1. Memilih menu cetak DPK	
	2. Menampilkan daftar program studi
3. Memilih program studi	
	4. Melakukan pengecekan status pembukaan mata kuliah untuk program studi yang dipilih
	5. Menampilkan daftar kuliah yang dibuka
6. Memilih tombol cetak seluruh DPK	
	7. <i>Me-request</i> seluruh DPK untuk kelas yang dibuka
	8. Mencetak seluruh DPK yang ditampilkan
	9. Menampilkan notifikasi pencetakan
<b>Skenario Alternatif 1 : Mencetak sebagian DPK SC-08-02</b>	
1. Memilih menu cetak KSM	
	2. Menampilkan daftar program studi
3. Memilih program studi	
	4. Melakukan pengecekan status pembukaan mata kuliah untuk program studi yang dipilih
	5. Menampilkan daftar kuliah yang dibuka
6. Memilih kuliah yang ingin dicetak DPKnya	
7. Memilih tombol cetak DPK	
	8. <i>Me-request</i> DPK untuk kelas yang dipilih
	8. Mencetak DPK yang dipilih
	9. Menampilkan notifikasi pencetakan

#### 1.3.3.8.1 Activity Diagram Use Case Mencetak DPK



Gambar 1-9 AC-08 Activity Diagram untuk Use Case UC-08 Mencetak DPK

## 2 Model Perancangan Arsitektural



Gambar 2-1 Model Perancangan Arsitektural

Model Perancangan Arsitektural dapat dilihat pada Gambar 2-1. Dalam perancangan arsitektur perangkat Lunak Sistem Informasi Akademik ini, digunakan *architectural style layered* dengan *architectural pattern MVC*.

Dalam hal ini, Model merupakan layer terbawah yang berhubungan langsung dengan pengaksesan basisdata, Controller bertanggung jawab dengan logika bisnis perangkat lunak termasuk komunikasi dengan sistem lain seperti DepKeu, dan View bertanggung jawab terhadap user interface.



### 3 Model Perancangan Level Komponen

#### 3.1 Realisasi Use Case Tahap Perancangan

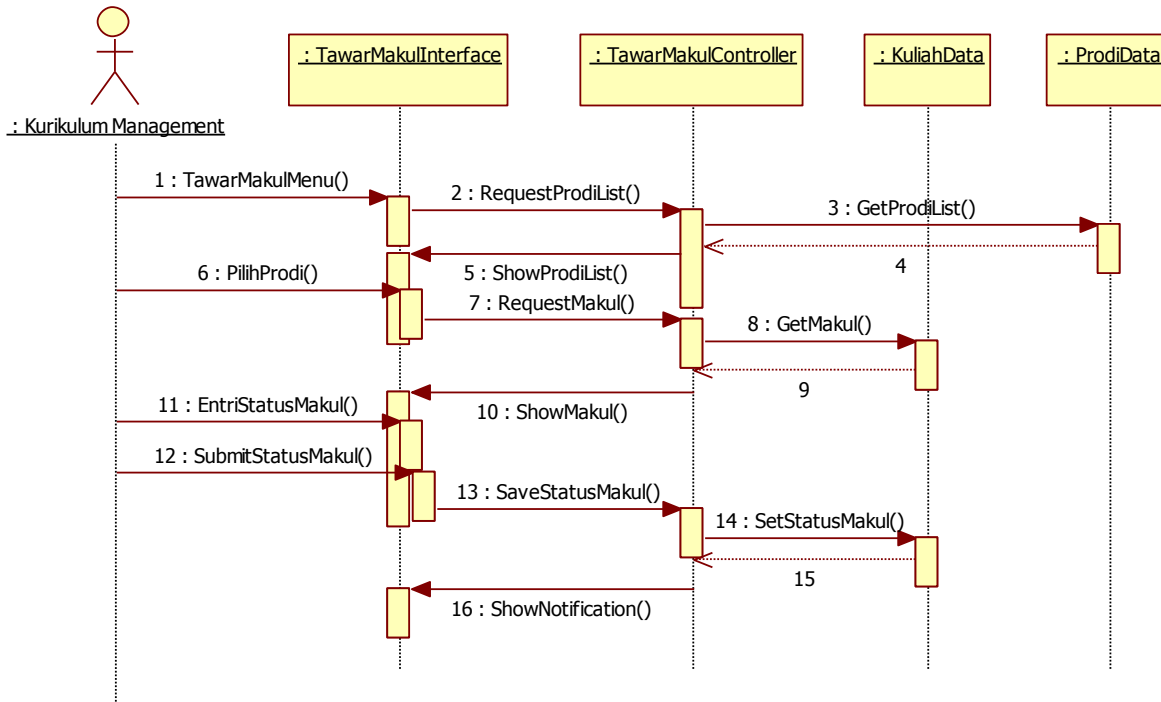
##### 3.1.1 Use Case Menawarkan Mata Kuliah

###### 3.1.1.1 Identifikasi Kelas Perancangan

No	Nama Kelas	Jenis Kelas (Interface, Control, Entity)
1.	TawarMakulInterface	Interface
2.	TawarMakulController	Control
3.	KuliahData	Entity
4.	ProdiData	Entity

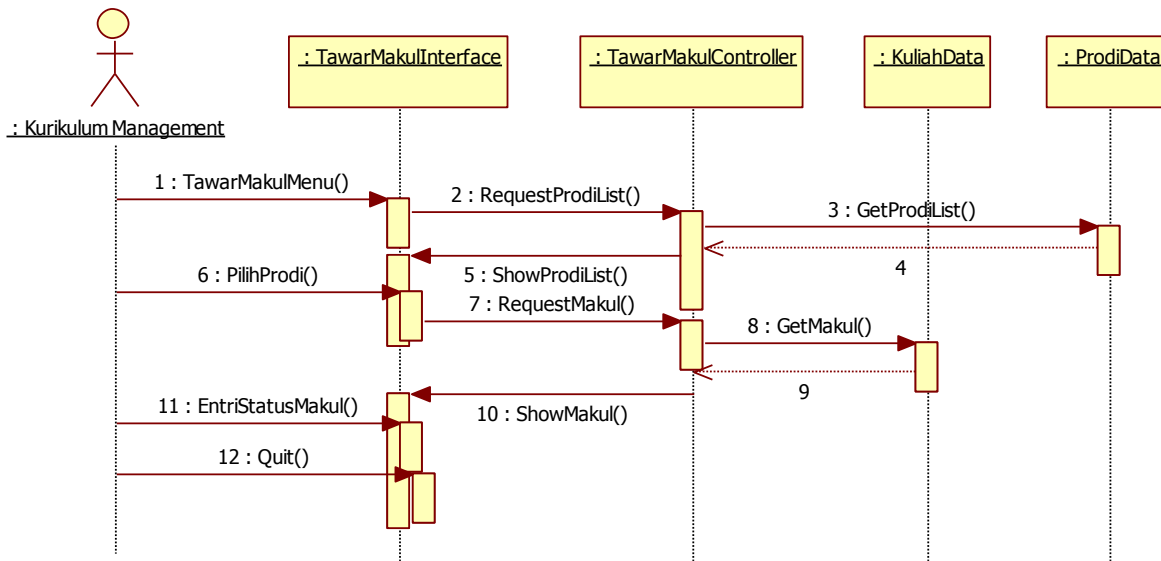
###### 3.1.1.2 Sequence Diagram

###### 3.1.1.2.1 Menawarkan Mata Kuliah



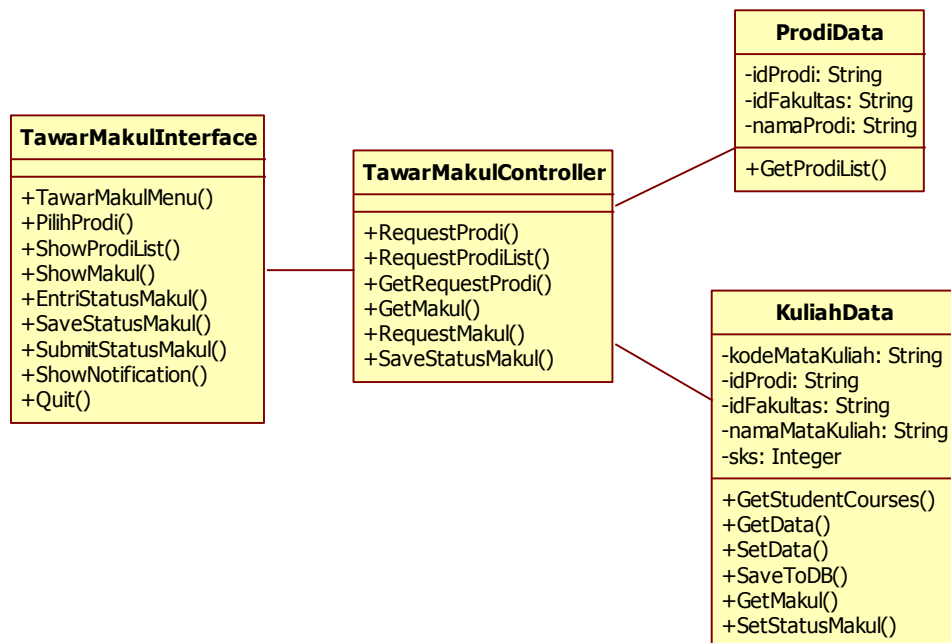
Gambar 3-1 SQ-01-01 Sequence Diagram untuk skenario SC-01-01 pada UC-01

### 3.1.1.2 Menawarkan Mata Kuliah Tetapi Tidak Disimpan



Gambar 3-2 SQ-01-02 Sequence Diagram untuk Skenario SC-01-02 pada UC-01

### 3.1.1.3 Diagram Kelas Perancangan



Gambar 3-3 CL-D-01 Diagram Kelas Perancangan untuk UC-01

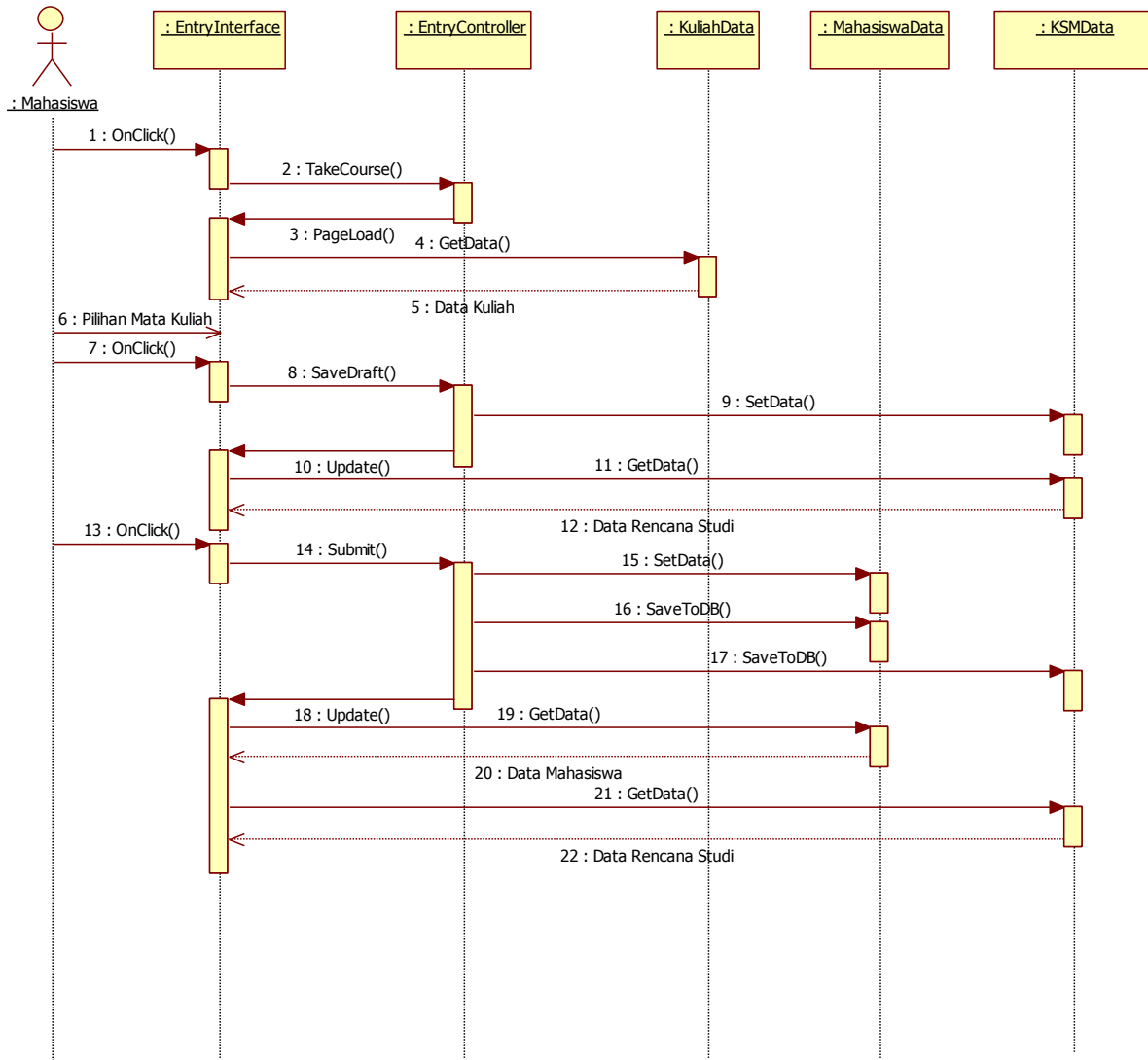
### 3.1.2 Use Case Entri Mata Kuliah

#### 3.1.2.1 Identifikasi Kelas Perancangan

<b>No</b>	<b>Nama Kelas</b>	<b>Jenis Kelas (Interface, Control, Entity)</b>
1.	EntryController	Control
2.	EntryInterface	Interface
3.	KuliahData	Entity
4.	MahasiswaData	Entity
5.	KSMDData	Entity

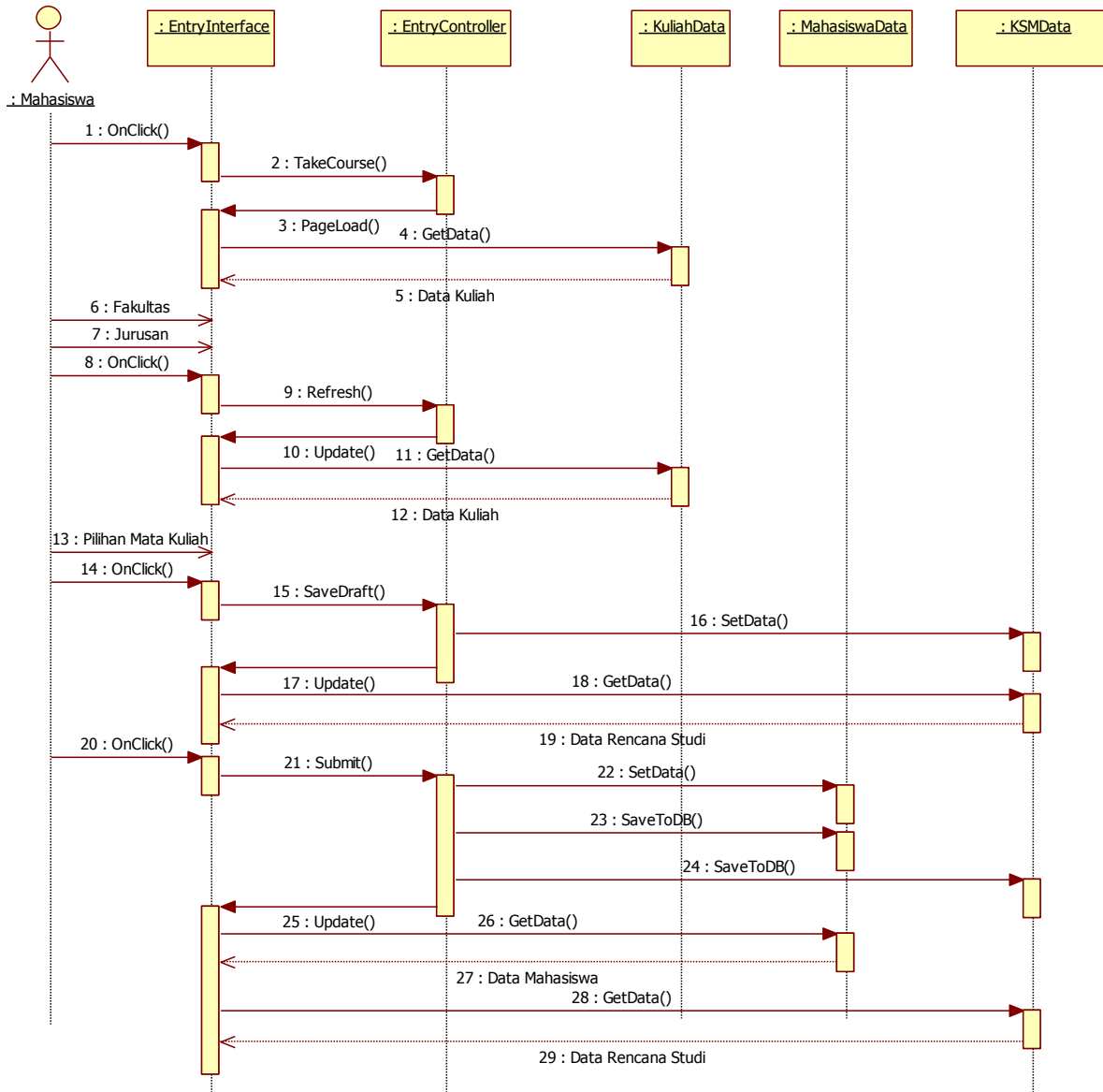
#### 3.1.2.2 Sequence Diagram

##### 3.1.2.2.1 Entri Mata Kuliah



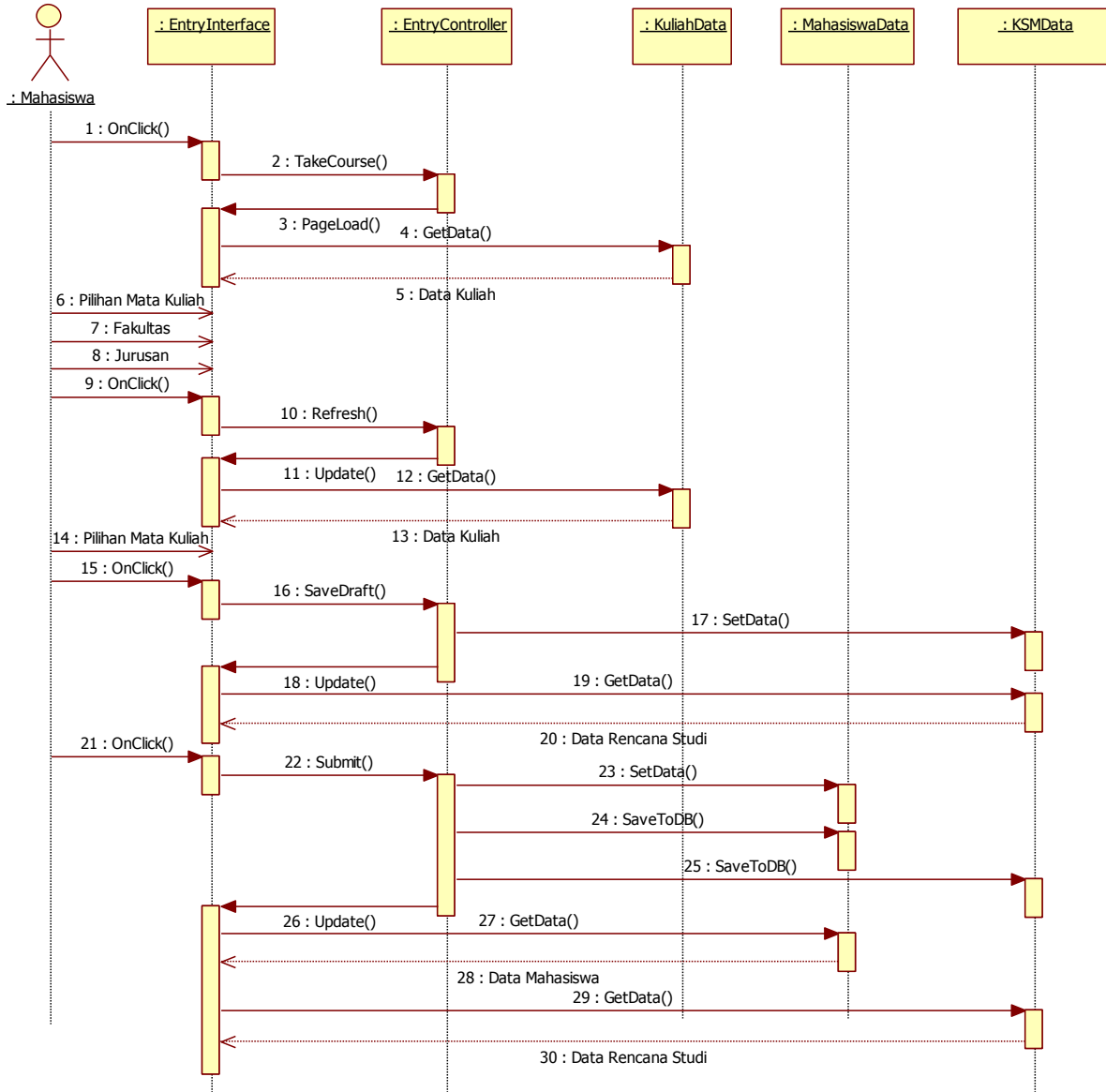
Gambar 3-4 SQ-02-01 Sequence Diagram untuk skenario SC-02-01 pada UC-02

### 3.1.2.2.2 Memilih Jurusan Lain



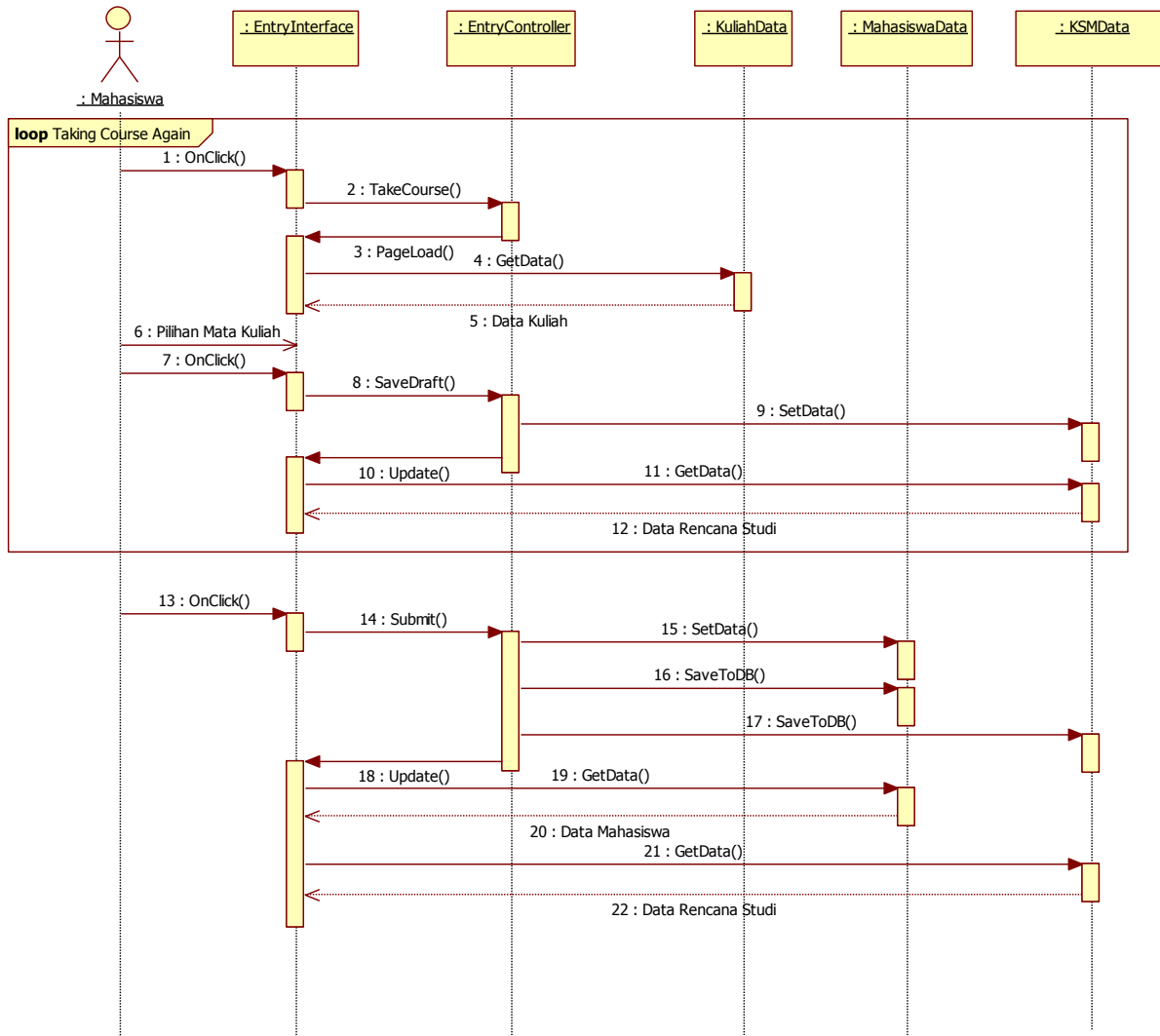
Gambar 3-5 SQ-02-02 Sequence Diagram untuk Skenario SC-02-02 pada UC-02

### 3.1.2.2.3 Kembali Memilih Mata Kuliah dari Jurusan Lain



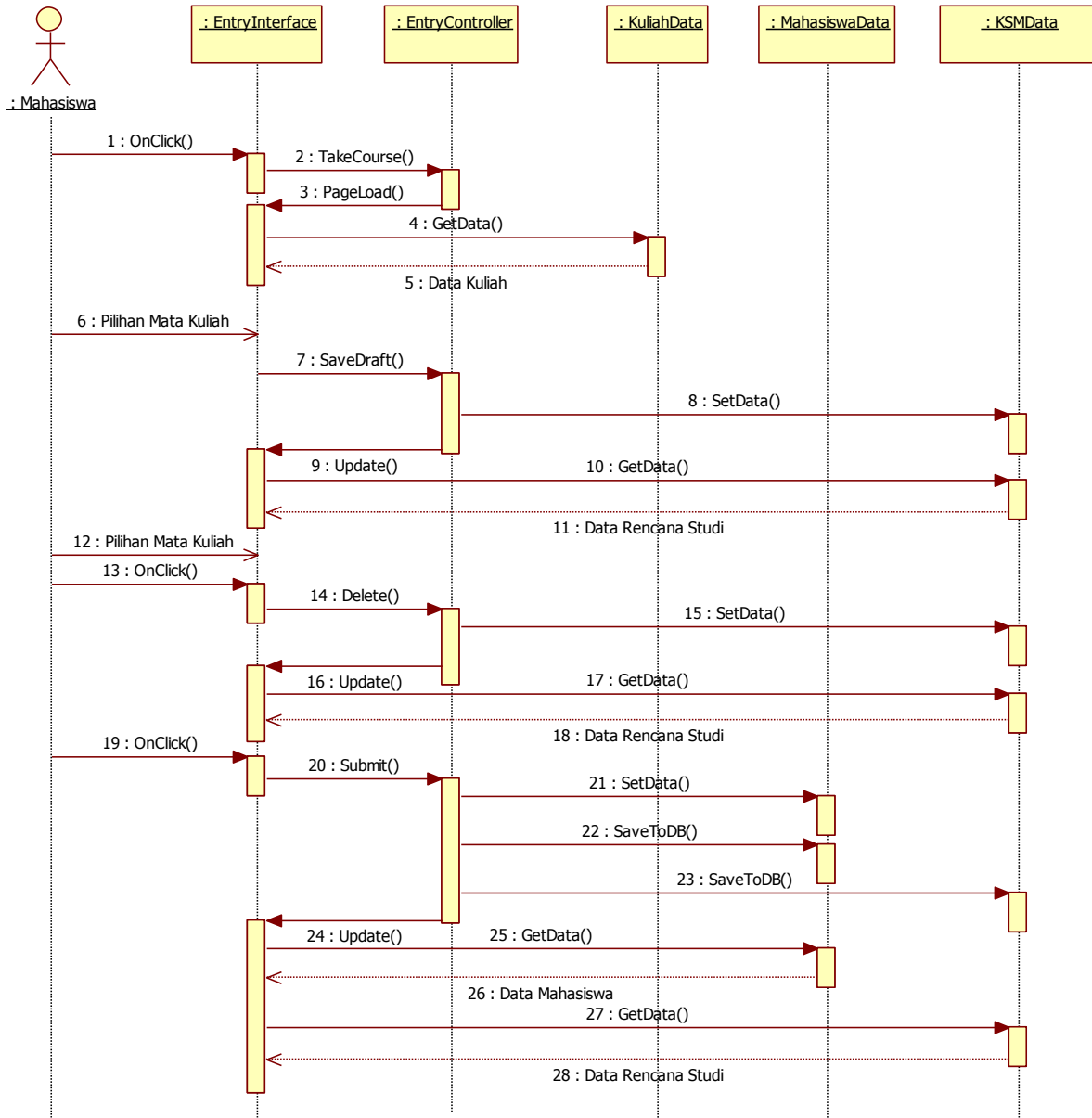
Gambar 3-6 SQ-02-03 Sequence Diagram untuk Skenario SC-02-03 pada UC-02

### 3.1.2.2.4 Kembali Mengambil Mata Kuliah Setelah Simpan



Gambar 3-7 SQ-02-04 Sequence Diagram untuk Skenario SC-02-04 pada UC-02

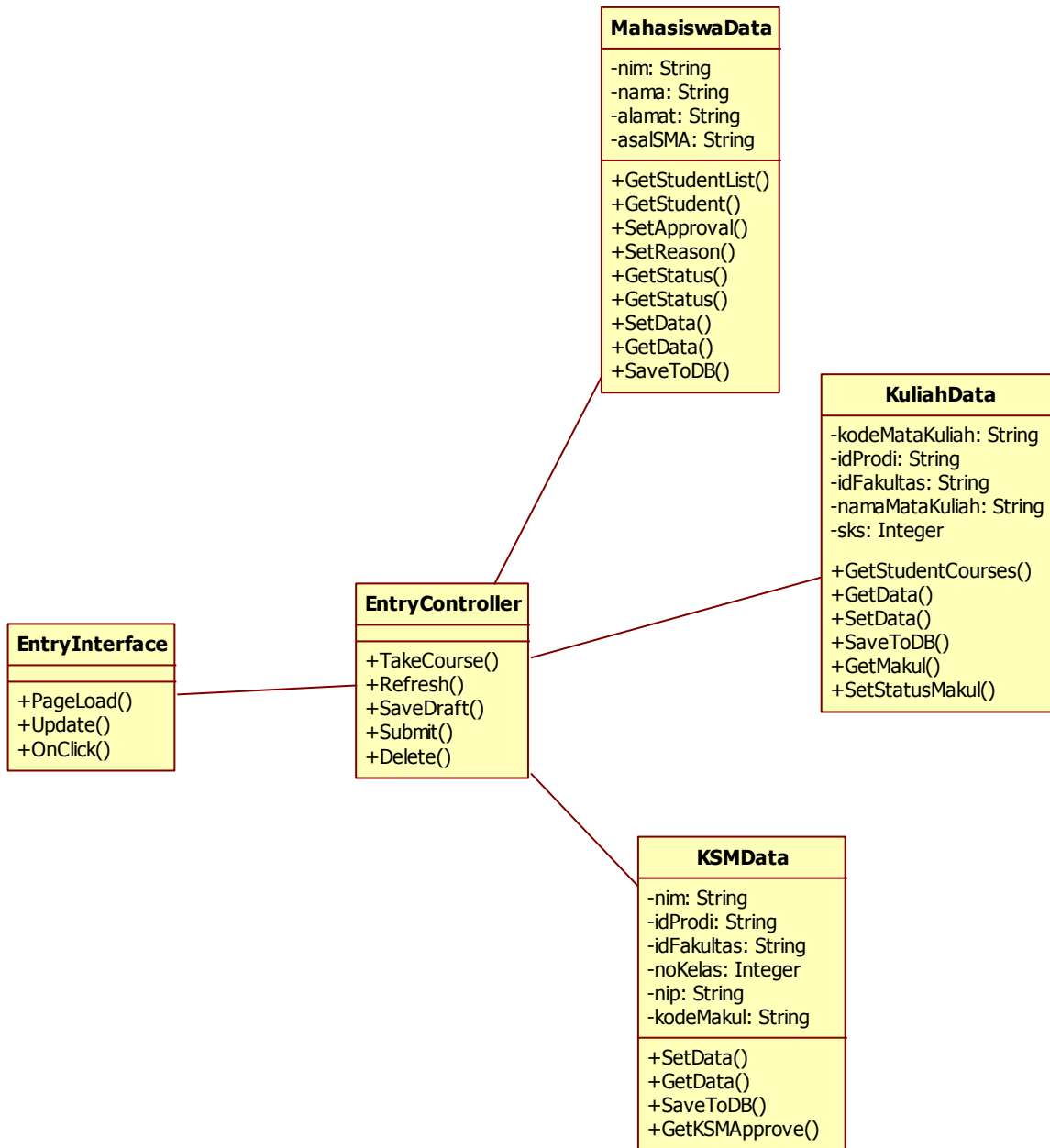
### 3.1.2.2.5 Hapus Mata Kuliah



Gambar 3-8 SQ-02-05 Sequence Diagram untuk Skenario SC-02-05 pada UC-02



### 3.1.2.3 Diagram Kelas Perancangan



Gambar 3-9 CL-D-02 Diagram kelas perancangan untuk UC-02

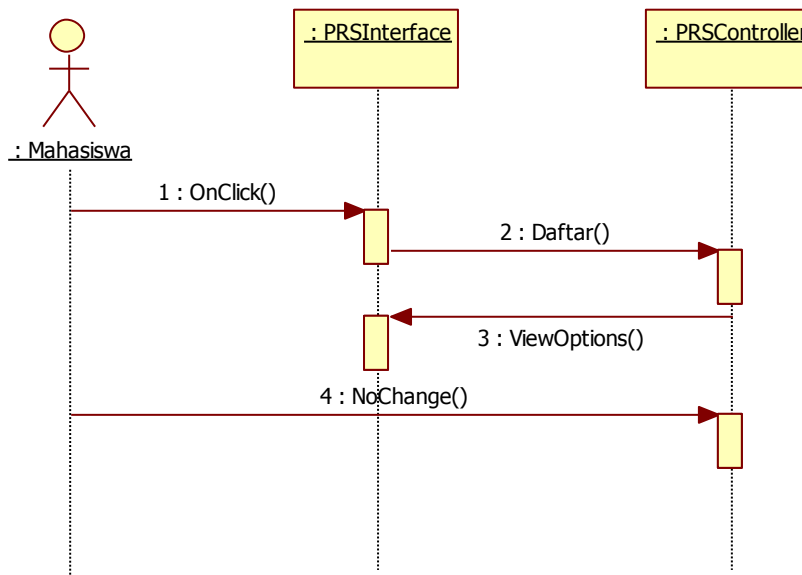
### 3.1.3 Use Case Daftar PRS

#### 3.1.3.1 Identifikasi Kelas Perancangan

No	Nama Kelas	Jenis Kelas (Interface, Control, Entity)
1.	PRSController	Control
2.	PRSInterface	Interface

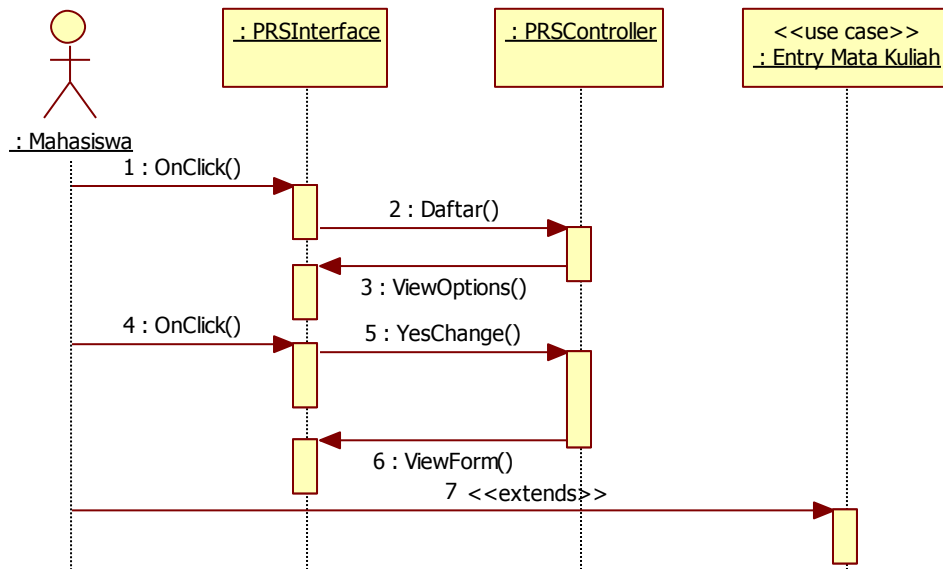
#### 3.1.3.2 Sequence Diagram

##### 3.1.3.2.1 Daftar PRS tanpa mengubah Rencana Studi



Gambar 3-10 SQ-03-01 Sequence Diagram untuk skenario SC-03-01 pada UC-03

### 3.1.3.2.2 Daftar PRS dengan mengubah Rencana Studi



Gambar 3-11 SQ-03-02 Sequence Diagram untuk Skenario SC-03-02 pada UC-03

### 3.1.3.3 Diagram Kelas Perancangan



Gambar 3-12 CL-D-03 Diagram Kelas Perancangan untuk UC-03

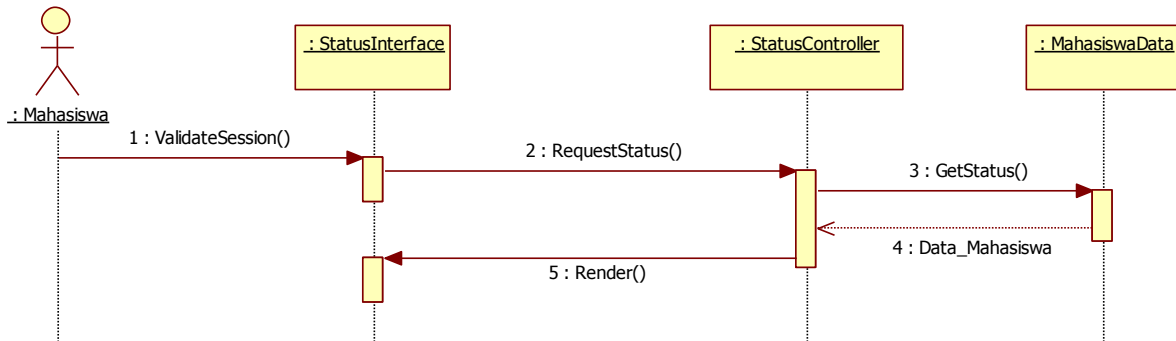
## 3.1.4 Use Case Melihat Status Mahasiswa

### 3.1.4.1 Identifikasi Kelas Perancangan

No	Nama Kelas	Jenis Kelas (Interface, Control, Entity)
1.	StatusInterface	Interface
2.	LoginInterface	Interface
3.	StatusController	Control
4.	MahasiswaData	Entity

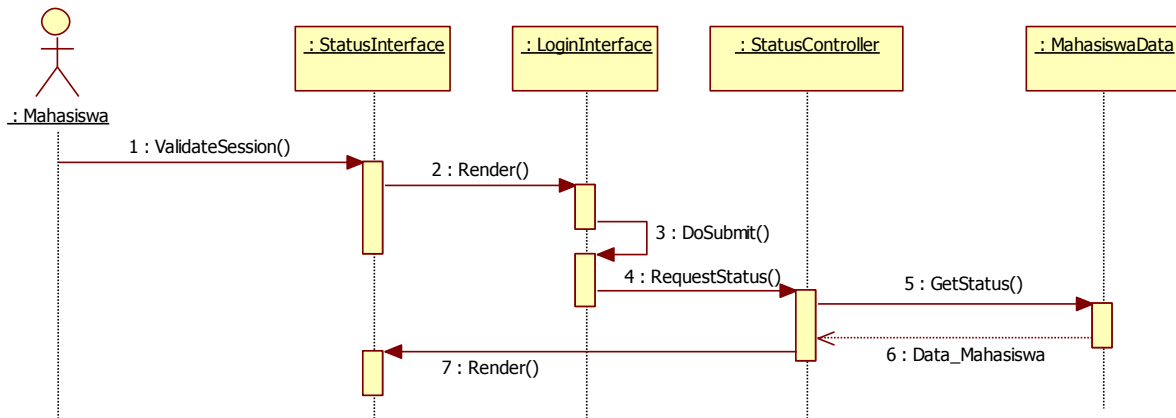
### 3.1.4.2 Sequence Diagram

#### 3.1.4.2.1 Melihat Status Mahasiswa



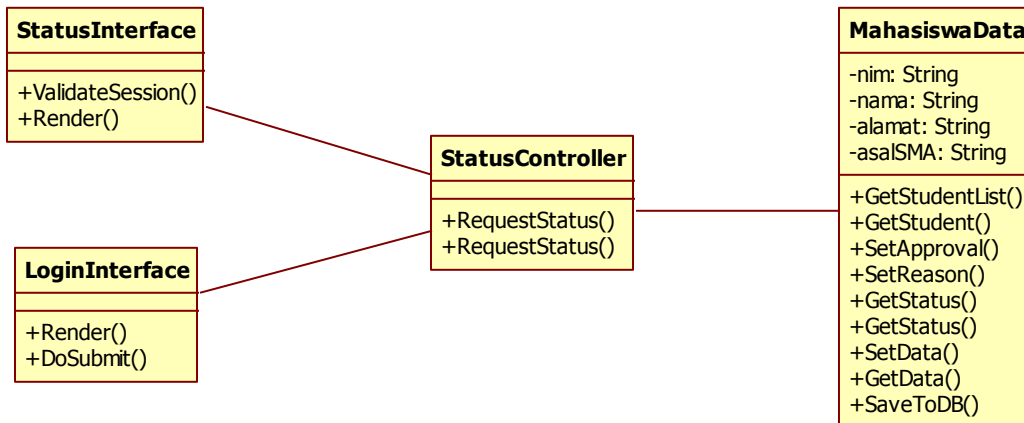
Gambar 3-13 SQ-04-01 Sequence Diagram untuk Skenario SC-04-01 pada UC-04

#### 3.1.4.2.2 Melihat Status Mahasiswa sebelum Log In



Gambar 3-14 SQ-04-02 Sequence Diagram untuk Skenario SC-04-02 pada UC-04

### 3.1.4.3 Diagram Kelas Perancangan



Gambar 3-15 CL-D-04 Diagram Kelas Perancangan untuk UC-04

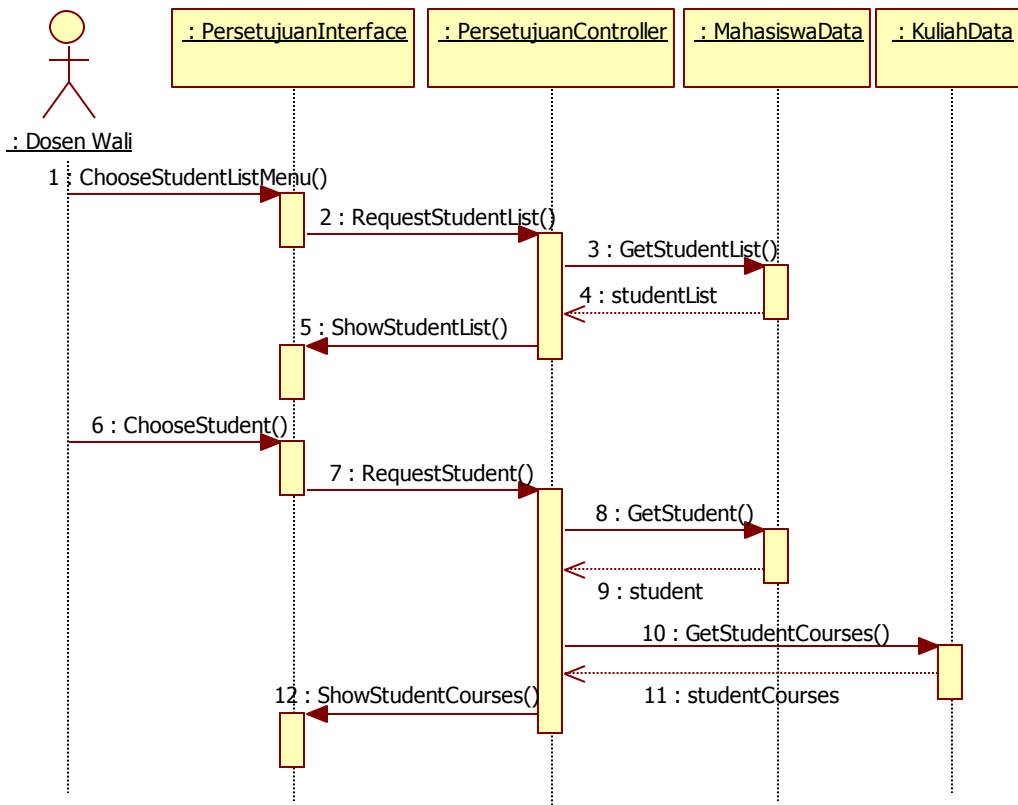
### 3.1.5 Use Case Melihat Entri Mata Kuliah

#### 3.1.5.1 Identifikasi Kelas Perancangan

No	Nama Kelas	Jenis Kelas (Interface, Control, Entity)
1.	PersetujuanInterface	Interface
2.	PersetujuanController	Control
3.	MahasiswaData	Entity
4.	KuliahData	Entity

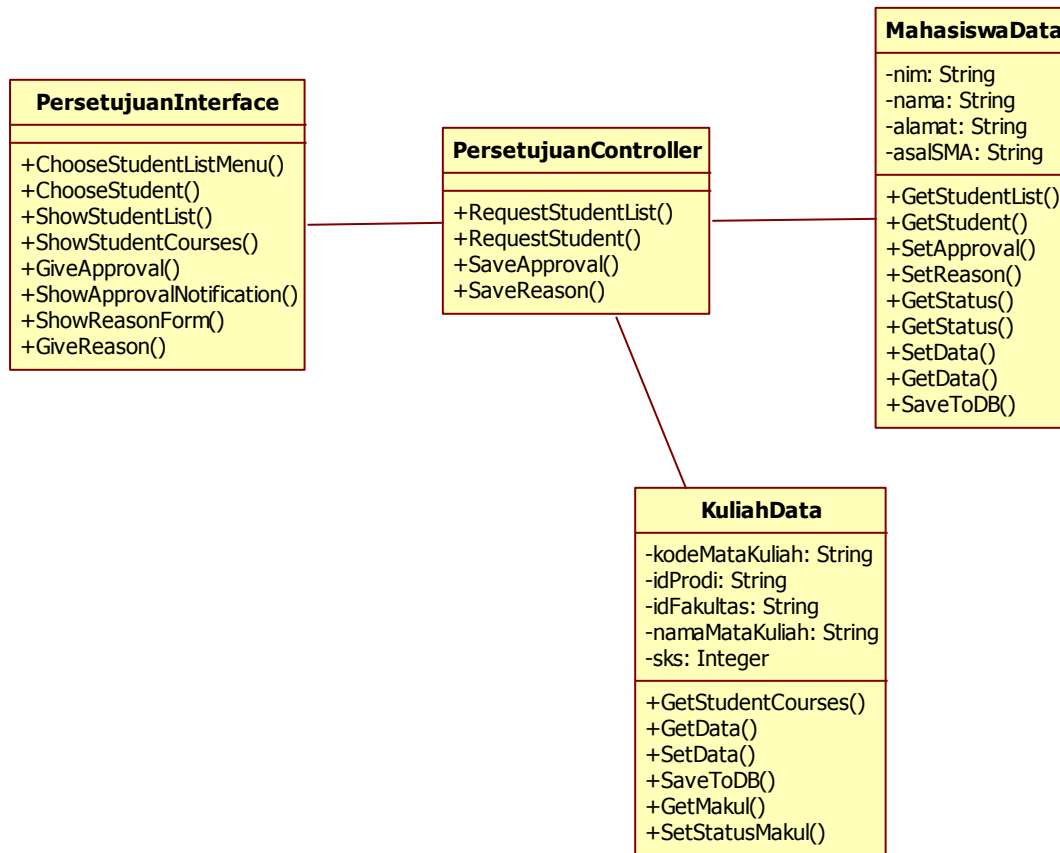
#### 3.1.5.2 Sequence Diagram

##### 3.1.5.2.1 Melihat Entri Mata Kuliah



Gambar 3-16 SQ-05-01 Sequence Diagram untuk Skenario SC-05-01 pada UC-05

### 3.1.5.3 Diagram Kelas Perancangan



Gambar 3-17 CL-D-05 Diagram Kelas Perancangan untuk UC-05

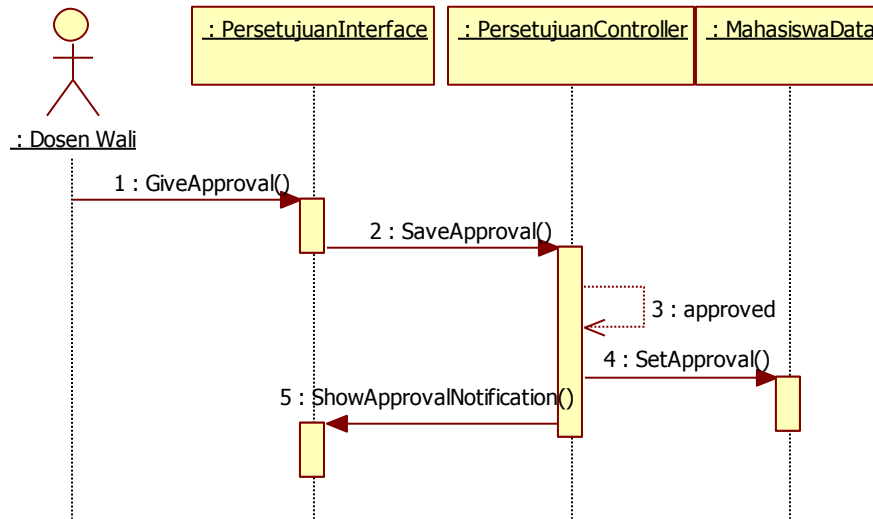
### 3.1.6 Use Case Memberikan Persetujuan

#### 3.1.6.1 Identifikasi Kelas Perancangan

No	Nama Kelas	Jenis Kelas (Interface, Control, Entity)
1.	PersetujuanInterface	Interface
2.	PersetujuanController	Control
3.	MahasiswaData	Entity

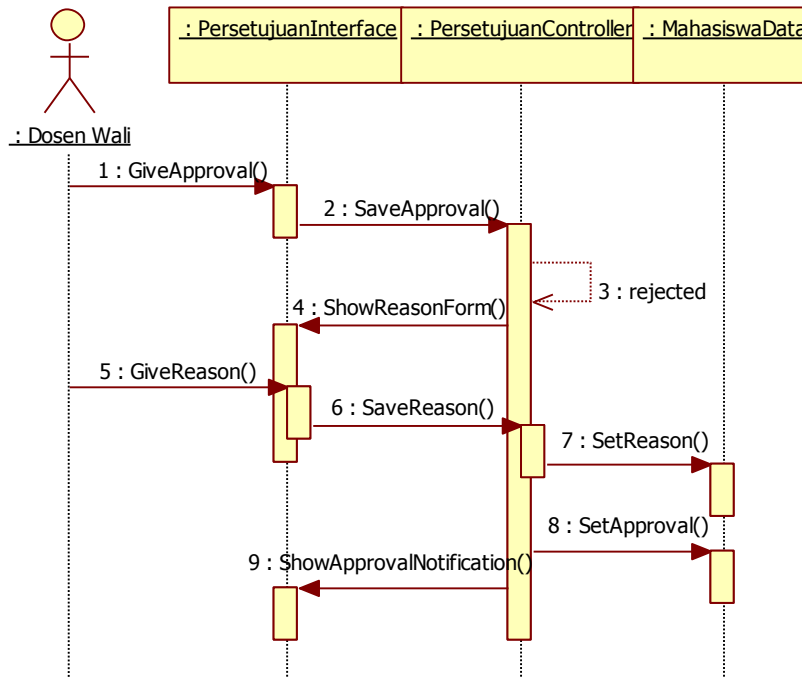
#### 3.1.6.2 Sequence Diagram

### 3.1.6.2.1 Memberikan Keputusan Menyetujui



Gambar 3-18 SQ-06-01 Sequence Diagram untuk Skenario SC-06-01 pada UC-06

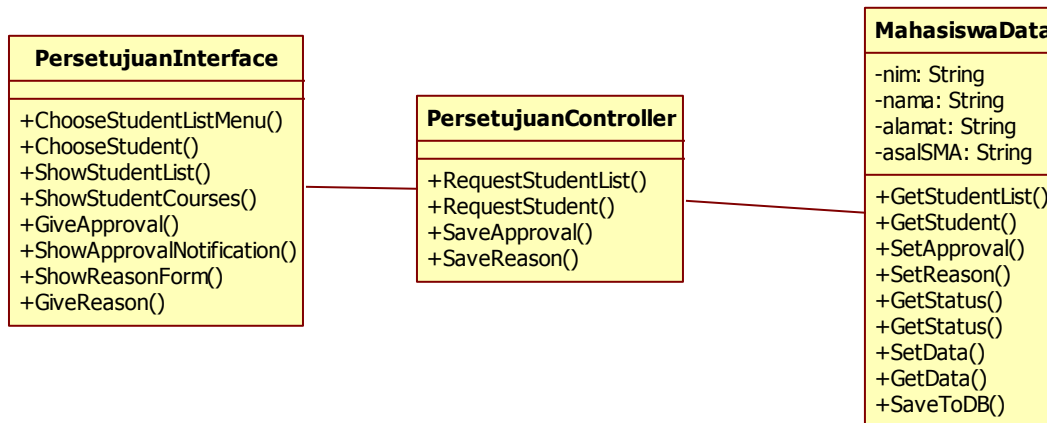
### 3.1.6.2.2 Memberikan Keputusan Menolak



Gambar 3-19 SQ-06-02 Sequence Diagram untuk Skenario SC-06-02 pada UC-06



### 3.1.6.3 Diagram Kelas Perancangan



Gambar 3-20 CL-D-06 Diagram Kelas Perancangan untuk UC-06

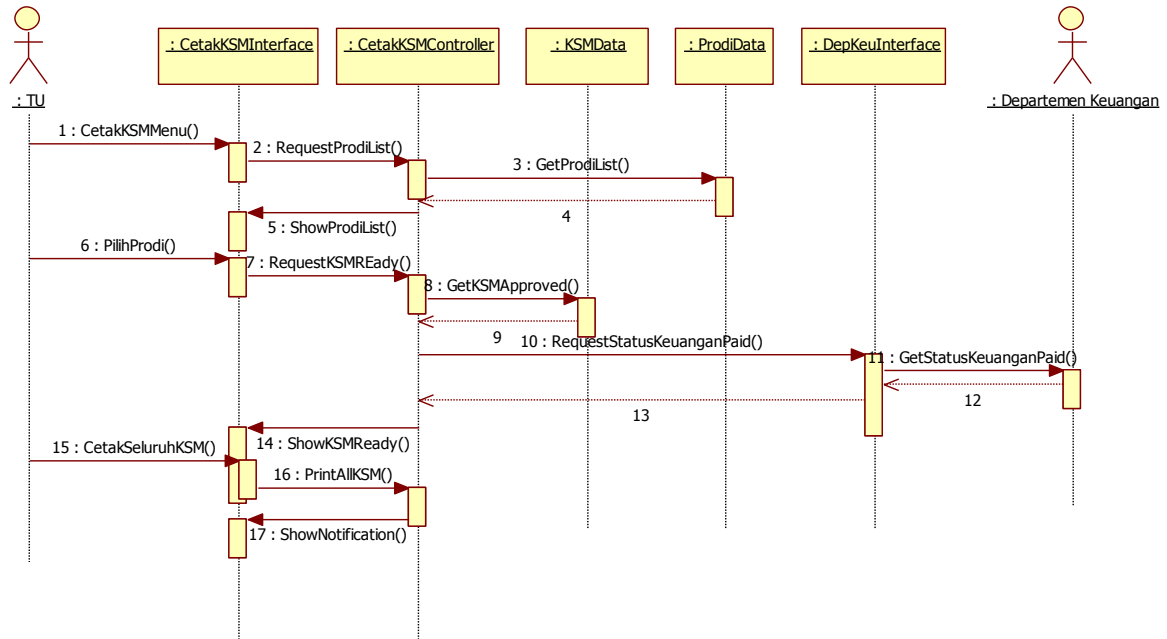
### 3.1.7 Use Case Mencetak KSM

#### 3.1.7.1 Identifikasi Kelas Perancangan

No	Nama Kelas	Jenis Kelas (Interface, Control, Entity)
1.	CetakKSMInterface	Interface
2.	CetakKSMController	Control
3.	DepkeuInterface	Interface
4.	ProdiData	Entity
5.	KSMDData	Entity

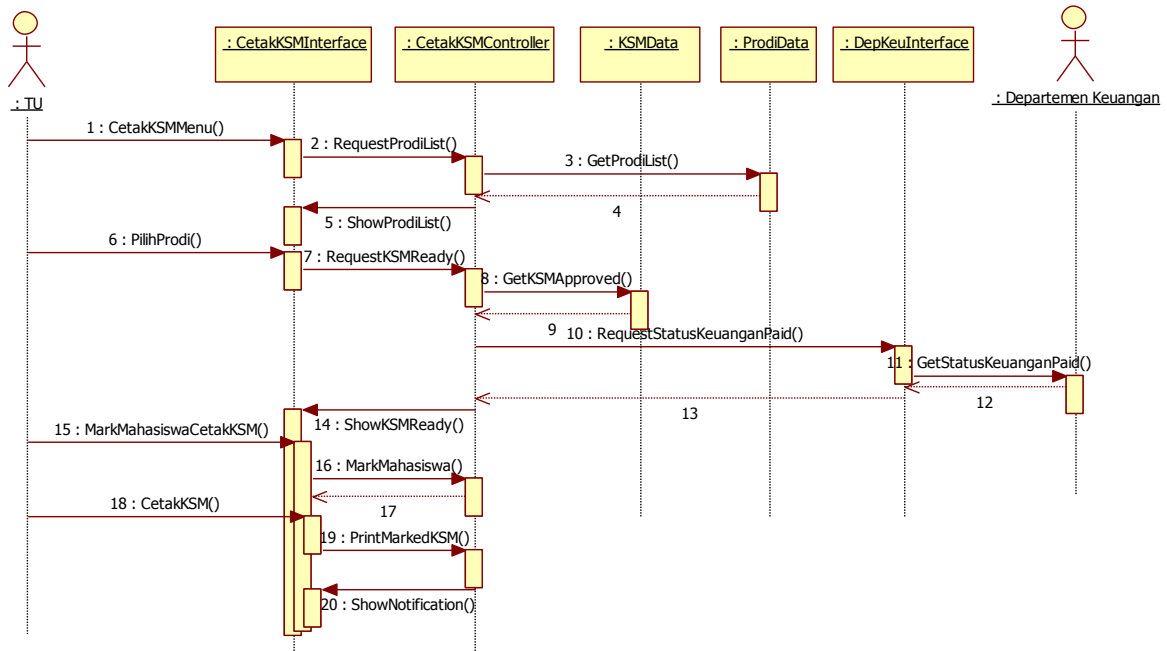
#### 3.1.7.2 Sequence Diagram

### 3.1.7.2.1 Mencetak Seluruh KSM



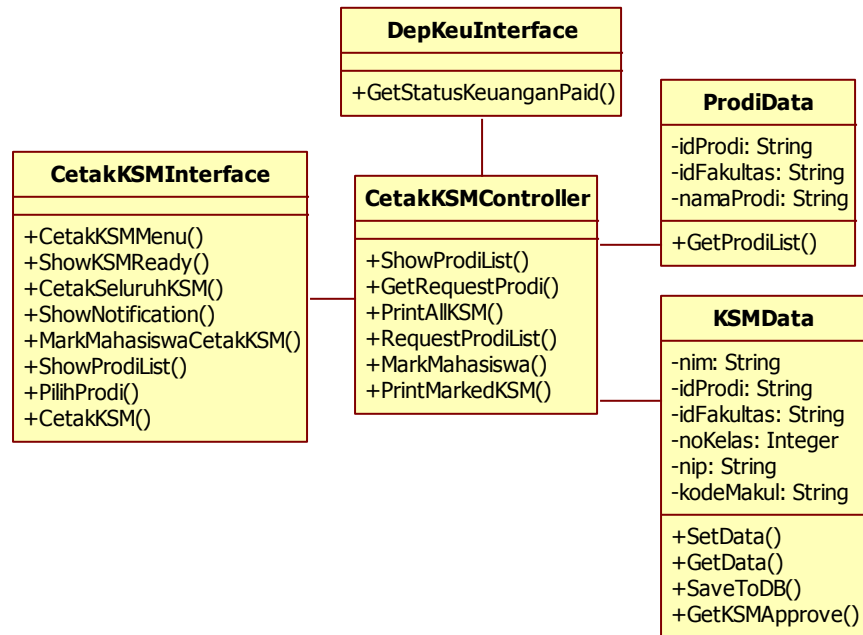
Gambar 3-21 SQ-07-01 Sequence Diagram untuk Skenario SC-07-01 pada UC-07

### 3.1.7.2.2 Mencetak Sebagian KSM



Gambar 3-22 SQ-07-02 Sequence Diagram untuk Skenario SC-07-02 pada UC-07

### 3.1.7.3 Diagram Kelas Perancangan



Gambar 3-23 CL-D-07 Diagram Kelas Perancangan untuk UC-07

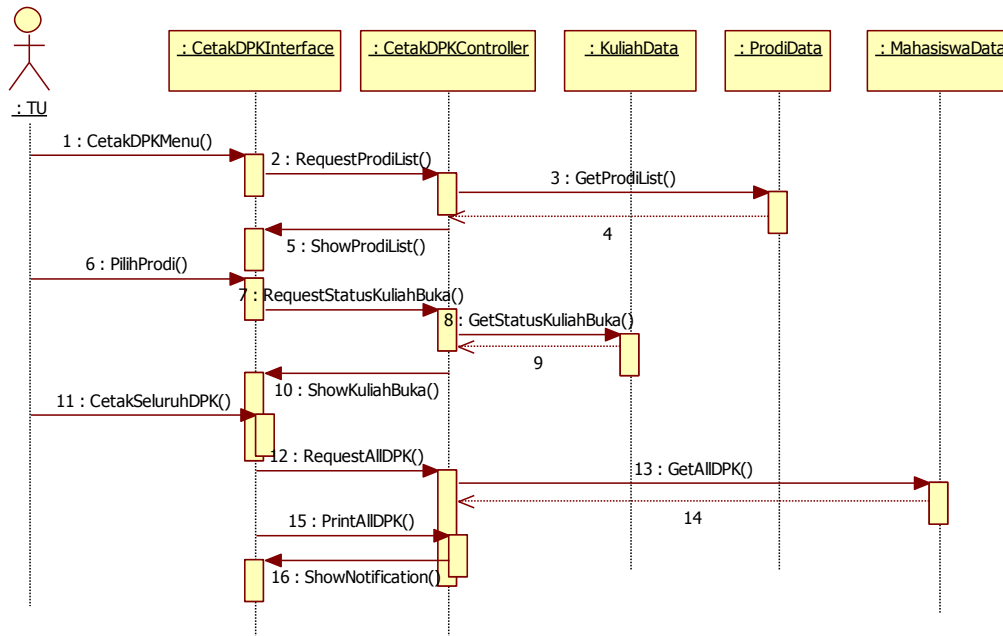
### 3.1.8 Use Case Mencetak DPK

#### 3.1.8.1 Identifikasi Kelas Perancangan

No	Nama Kelas	Jenis Kelas (Interface, Control, Entity)
1.	CetakDPKInterface	Interface
2.	CetakDPKController	Control
3.	KuliahData	Entity
4.	ProdiData	Entity
5.	MahasiswaData	Entity

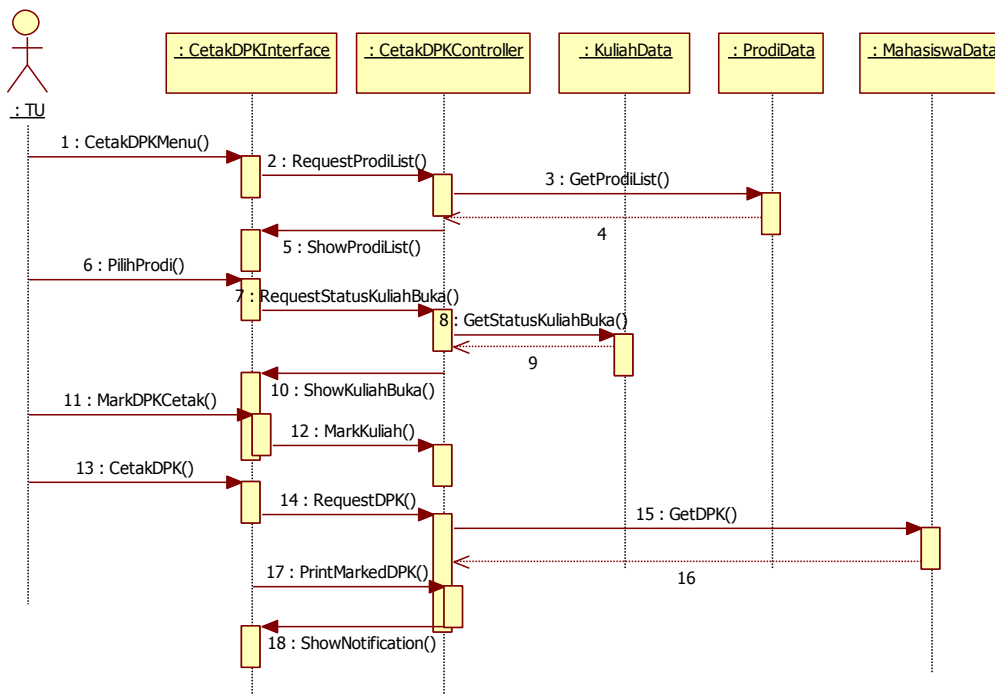
#### 3.1.8.2 Sequence Diagram

##### 3.1.8.2.1 Mencetak Seluruh DPK



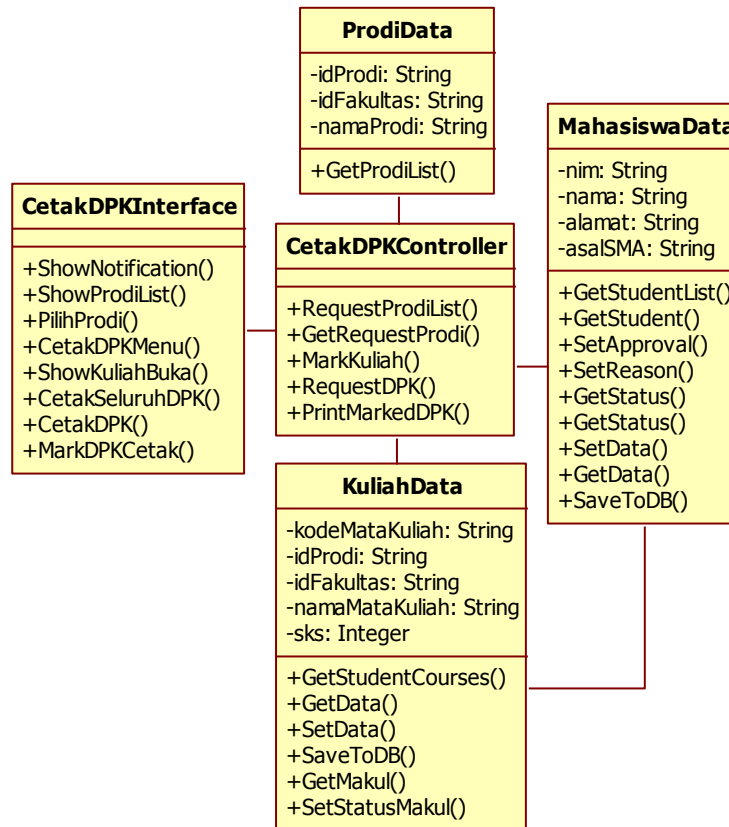
Gambar 3-24 SQ-08-01 Sequence Diagram untuk Skenario SC-08-01 pada UC-08

### 3.1.8.2.2 Mencetak Sebagian DPK



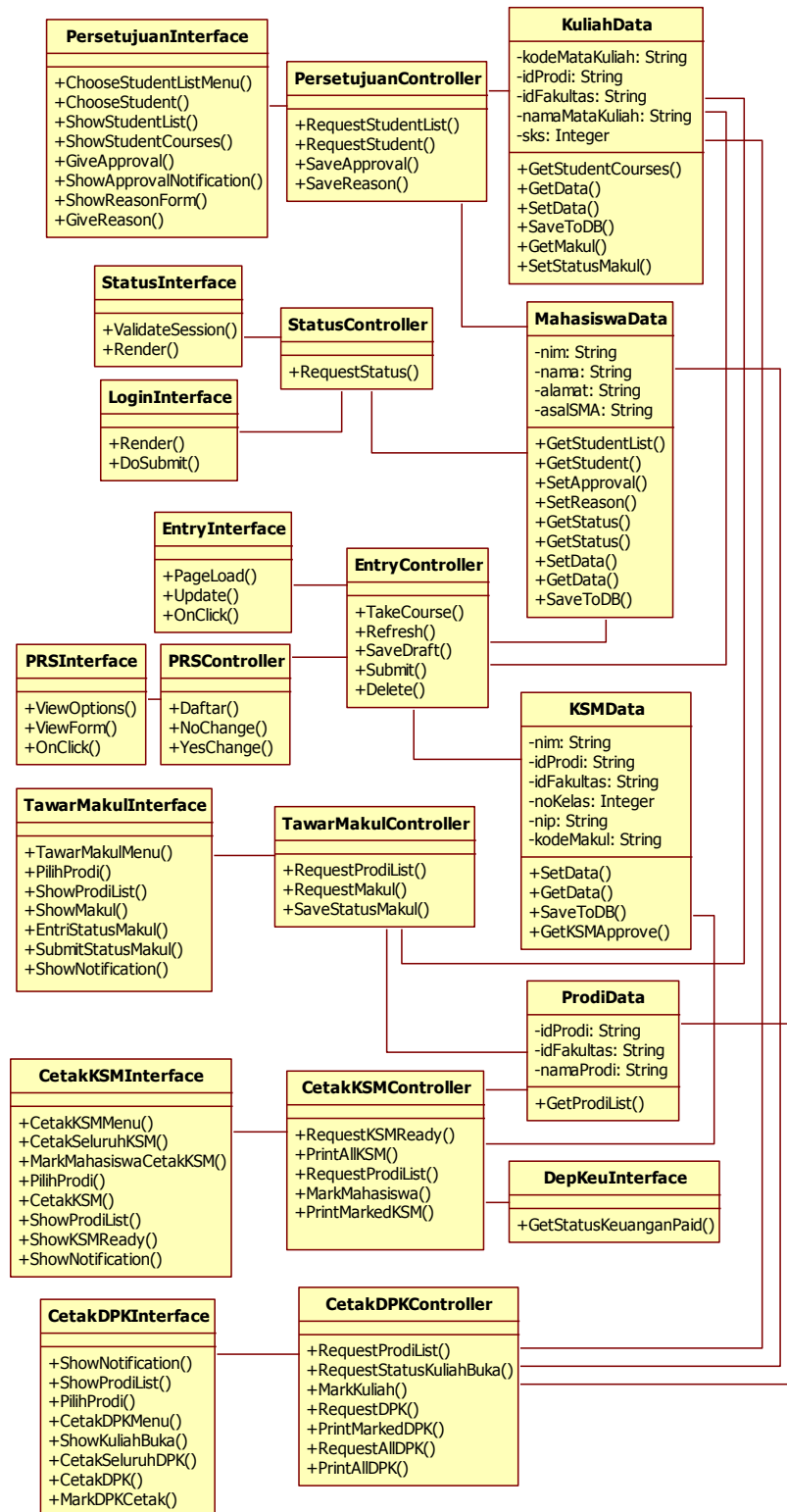
Gambar 3-25 SQ-08-02 Sequence Diagram untuk Skenario SC-08-02 pada UC-08

### 3.1.8.3 Diagram Kelas Perancangan



Gambar 3-26 CL-D-08 Diagram Kelas Perancangan untuk UC-08

## 3.2 Diagram Kelas Keseluruhan



Gambar 3-27 Diagram Kelas Keseluruhan

### 3.3 Rincian Kelas

#### 3.3.1 Kelas ProdiData

Deskripsi Kelas: Kelas entitas yang mengandung sekumpulan program studi yang terdapat pada sistem perwalian online.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
GetProdiList()	public	Melakukan retrieve data program studi dari basis data
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
idProdi	private	String
idFakultas	private	String
namaProdi	private	String

#### 3.3.2 Kelas KSMData

Deskripsi Kelas: Kelas entitas yang mengandung data dari KSM mahasiswa.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
GetData()	public	Melakukan retrieve data KSM mahasiswa dari basis data
SetData()	Public	Melakukan pengesetan data KSM mahasiswa
SaveToDB()	Public	Melakukan penyimpanan data KSM mahasiswa dari program aplikasi ke basis data
GetKSMApprove	Public	Mengambil data apakah KSM ybs sudah disetujui oleh dosen wali atau tidak
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
Nim	Public	String
idProdi	private	String
idFakultas	private	String
noKelas	private	String[]
nip	Public	String
Makul	Public	String[]

### 3.3.3 Kelas KuliahData

Deskripsi Kelas : Merupakan representasi Tabel Kuliah pada basis data

Nama Operasi	Visibility (private, public)	Keterangan
GetStudentCourse()	public	Operasi yang digunakan untuk memperoleh data prodi yang diambil mahasiswa yang terdapat di basis data
GetData()	public	Operasi yang digunakan untuk memperoleh data dari basis data
SetData()	public	Operasi yang digunakan untuk mengisi data ke basis data
SaveToDB()	public	Operasi yang digunakan untuk menyimpan data ke basis data
GetMakul()	public	Operasi yang digunakan untuk memperoleh data mata kuliah yang berada di basis data
SetStatusMakul()	public	Operasi yang digunakan untuk mengisi status mata kuliah yang dipilih pada data mata kuliah di basis data
Nama Atribut	Visibility (private, public)	Tipe
kodeMataKuliah	private	String
idProdi	private	String
idFakultas	private	String
namaMataKuliah	Private	String
Sks	Private	Integer

### 3.3.4 Kelas MahasiswaData

Deskripsi Kelas : Merupakan representasi Tabel Mahasiswa pada basis data

Nama Operasi	Visibility (private, public)	Keterangan
GetStudentList()	public	Operasi yang digunakan untuk memperoleh daftar mahasiswa yang terdapat di dalam basis data
GetStudent()	public	Operasi yang digunakan untuk memperoleh data mahasiswa di basis data
SetApproval()	public	Operasi yang digunakan untuk



		mengisi data mahasiswa yang mata kuliahnya disetujui oleh dosen wali
SetReason()	public	Operasi yang digunakan untuk mengisi data mahasiswa yang mata kuliahnya ditolak dosen wali
GetStatus()	public	Operasi yang digunakan untuk memperoleh status perwalian mahasiswa dari basis data
SetStatus()	public	Operasi yang digunakan untuk mengisi status perwalian mahasiswa di basis data
SetData()	public	Operasi yang digunakan untuk mengisi data ke basis data
GetData()	public	Operasi yang digunakan untuk memperoleh data dari basis data
SaveToDB()	public	Operasi yang digunakan untuk menyimpan data ke basis data
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
nim	private	String
nama	private	String
alamat	private	String
asalSMA	private	String

### 3.3.5 Kelas StatusController

Deskripsi Kelas: Kelas controller yang mengatur terjadinya transaksi data yang harus ditunjukkan dalam status mahasiswa

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
RequestStatus()	public	Meminta data mahasiswa dalam MahasiswaData memberikan data yang sesuai
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>

### 3.3.6 Kelas StatusInterface

Deskripsi Kelas: Kelas interface yang menampilkan isi status perwalian mahasiswa yang bersangkutan.

Nama Operasi	Visibility (private, public)	Keterangan
ValidateSession()	public	Melakukan validasi terhadap <i>user session</i> dan mengarahkan user pada fungsi <i>dispatcher</i> yang sesuai.
Render()	Public	Menampilkan isi status mahasiswa ke layar.
Nama Atribut	Visibility (private, public)	Tipe

### 3.3.7 Kelas LoginInterface

Deskripsi Kelas: Kelas interface yang menampilkan *login form* mahasiswa ke sistem perwalian online.

Nama Operasi	Visibility (private, public)	Keterangan
DoSubmit()	public	Memasukkan data dari <i>login form</i> untuk validasi <i>username</i> dan <i>password</i> .
Render()	Public	Menampilkan <i>login form</i> ke layar
Nama Atribut	Visibility (private, public)	Tipe

### 3.3.8 Kelas TawarMakulInterface

Deskripsi Kelas : Merupakan kelas penghubung antara Kurikulum Management dengan sistem untuk melakukan penawaran mata kuliah, sekaligus memberikan status untuk mata kuliah.

Nama Operasi	Visibility (private, public)	Keterangan
TawarMakulMenu()	public	Operasi ini digunakan untuk menangani pilihan Menu Penawaran Mata Kuliah
PilihProdi()	public	Operasi ini digunakan untuk menangani pilihan Menu Pemilihan Program Studi
ShowProdiList()	public	Operasi ini digunakan untuk

		menampilkan daftar program studi
ShowMakul()	public	Operasi ini digunakan untuk menampilkan daftar program studi
EntriStatusMakul()	public	Operasi ini digunakan untuk menangani pengisian status mata kuliah
SubmitStatusMakul()	public	Operasi ini digunakan untuk meneruskan status mata kuliah yang telah diisi ke kelas TawarMakulController
ShowNotification()	public	Operasi ini digunakan untuk menampilkan notifikasi proses yang telah dilakukan
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

### 3.3.9 Kelas TawarMakulController

Deskripsi Kelas : Merupakan kelas yang mengelola *business logic* dari aksi Kurikulum Management dalam melakukan penawaran mata kuliah.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
RequestProdiList()	public	Operasi ini digunakan untuk mendapatkan daftar program studi dari kelas ProdiData
RequestMakul()	public	Operasi ini digunakan untuk mendapatkan mata kuliah dari kelas KuliahData
SaveStatusMakul()	public	Operasi ini digunakan untuk menyimpan status mata kuliah melalui kelas KuliahData
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

### 3.3.10 Kelas EntryInterface

Deskripsi Kelas : Merupakan representasi kelas penghubung antara Mahasiswa dengan sistem untuk melakukan aksi entri mata kuliah

Nama Operasi	Visibility (private, public)	Keterangan
PageLoad()	public	Operasi ini digunakan untuk menginisialisasi antarmuka awal untuk entri matakuliah
Update()	public	Operasi ini digunakan untuk memperbarui antarmuka entri matakuliah setelah aktor melakukan aksi
OnClick()	public	Operasi ini digunakan untuk menangkap input dari aktor yang menekan tombol
Nama Atribut	Visibility (private, public)	Tipe
-	-	-

### 3.3.11 Kelas EntryController

Deskripsi Kelas : Merupakan representasi kelas yang mengelola *business logic* dari aksi mahasiswa dalam melakukan entri mata kuliah

Nama Operasi	Visibility (private, public)	Keterangan
TakeCourse()	public	Operasi ini digunakan ketika mahasiswa akan melakukan pengambilan matakuliah.
Refresh()	public	Operasi ini digunakan untuk memilih fakultas dan prodi lain dalam pengambilan matakuliah.
SaveDraft()	public	Operasi ini digunakan untuk menyimpan sementara matakuliah yang telah diambil sebagai draf.
Submit()	public	Operasi ini digunakan untuk mengajukan rencana studi sementara berdasarkan draf matakuliah yang diambil.
Delete()	public	Operasi ini digunakan untuk menghapus matakuliah pada draf matakuliah yang diambil.
Nama Atribut	Visibility	Tipe

	(private, public)	
-	-	-

### 3.3.12 Kelas PRSInterface

Deskripsi Kelas : Merupakan representasi kelas yang menghubungkan mahasiswa dengan sistem ketika melakukan aksi daftar PRS

Nama Operasi	Visibility (private, public)	Keterangan
ViewOptions()	public	Operasi ini digunakan untuk menampilkan pilihan menu mengubah rencana studi setelah daftar PRS.
ViewForm()	public	Operasi ini digunakan untuk menampilkan rencana studi saat ini ketika ingin diubah.
OnClick()	public	Operasi ini digunakan untuk menangkap input dari aktor yang menekan tombol
Nama Atribut	Visibility (private, public)	Tipe
-	-	-

### 3.3.13 Kelas PRSController

Deskripsi Kelas : Merupakan representasi kelas yang mengelola *business logic* dari aksi mahasiswa dalam melakukan aksi daftar PRS

Nama Operasi	Visibility (private, public)	Keterangan
Daftar()	public	Operasi ini digunakan untuk melakukan aksi daftar PRS.
YesChange()	public	Operasi ini digunakan untuk melakukan perubahan rencana studi setelah daftar PRS.
NoChange()	public	Operasi ini digunakan untuk tidak melakukan perubahan rencana studi setelah daftar PRS.
Nama Atribut	Visibility (private, public)	Tipe
-	-	-

### 3.3.14 Kelas PersetujuanInterface

Deskripsi Kelas : Merupakan representasi kelas penghubung antara Dosen dengan sistem untuk melihat entri mata kuliah mahasiswa dan memberikan persetujuan.

Nama Operasi	Visibility (private, public)	Keterangan
ChooseStudentListMenu()	public	Operasi ini digunakan untuk memilih menu melihat daftar mahasiswa.
ChooseStudent()	public	Operasi ini digunakan untuk melihat data rinci dari mahasiswa.
ShowStudentList()	public	Operasi ini digunakan untuk menampilkan daftar mahasiswa.
ShowStudentCourses()	public	Operasi ini digunakan untuk menampilkan daftar mata kuliah yang diambil mahasiswa.
GiveApproval()	public	Operasi ini digunakan untuk memberikan persetujuan terhadap mata kuliah yang diambil mahasiswa agar dapat disimpan ke dalam basis data.
ShowApprovalNotification()	public	Operasi ini digunakan untuk menampilkan notifikasi pemberian persetujuan.
ShowReasonForm()	public	Operasi ini digunakan untuk menampilkan form alasan jika mata kuliah yang diambil mahasiswa ditolak oleh dosen.
GiveReason()	public	Operasi ini digunakan untuk memberikan alasan agar dapat disimpan ke dalam basis data.
Nama Atribut	Visibility (private, public)	Tipe
-	-	-

### 3.3.15 Kelas PersetujuanController

Deskripsi Kelas : Merupakan representasi kelas yang mengelola *business logic* dari aksi dosen dalam melihat entri mata kuliah mahasiswa dan memberikan persetujuan.

Nama Operasi	Visibility (private, public)	Keterangan
RequestStudentList()	public	Operasi ini digunakan untuk

		menghasilkan daftar mahasiswa dari entitas MahasiswaData.
RequestStudent()	public	Operasi ini digunakan untuk menghasilkan data rinci mahasiswa dari entitas MahasiswaData.
SaveApproval()	public	Operasi ini digunakan untuk melakukan penyimpanan persetujuan.
SaveReason()	public	Operasi ini digunakan untuk melakukan penyimpanan alasan penolakan persetujuan.
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

### 3.3.16 Kelas CetakKSMInterface

Deskripsi Kelas : Merupakan kelas penghubung antara Tata Usaha dengan sistem untuk mencetak KSM.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
PilihProdi()	public	Operasi ini digunakan untuk menangani pilihan Menu Pemilihan Program Studi
CetakKSMMenu()	public	Operasi ini digunakan untuk menangani pilihan Menu Pencetakan KSM
CetakSeluruhKSM()	public	Operasi ini digunakan untuk menangani pilihan Menu Pencetakan Seluruh KSM
MarkMahasiswaCetakKSM()	public	Operasi ini digunakan untuk menangani <i>marking</i> terhadap data mahasiswa yang akan dicetak KSMnya
CetakKSM()	public	Operasi ini digunakan untuk menangani pilihan Menu Pencetakan KSM untuk yang di- <i>mark</i> saja
ShowProdiList()	public	Operasi ini digunakan untuk menampilkan daftar program studi
ShowKSMReady()	public	Operasi ini digunakan untuk menampilkan KSM yang siap

		untuk dicetak
ShowNotification()	public	Operasi ini digunakan untuk menampilkan notifikasi proses yang telah dilakukan
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

### 3.3.17 Kelas CetakKSMController

Deskripsi Kelas : Merupakan kelas yang mengelola *business logic* dari aksi Tata Usaha dalam mencetak KSM.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
RequestProdiList()	public	Operasi ini digunakan untuk mendapatkan daftar program studi dari kelas ProdiData
RequestKSMReady	public	Operasi ini digunakan untuk mendapatkan data KSM yang telah siap dicetak
PrintAllKSM()	public	Operasi ini digunakan untuk mencetak seluruh KSM yang siap cetak
MarkMahasiswa()	public	Operasi ini digunakan untuk menandai mahasiswa yang KSMnya akan dicetak
PrintMarkedKSM()	public	Operasi ini digunakan untuk mencetak KSM mahasiswa yang ditandai
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

### 3.3.18 Kelas CetakDPKInterface

Deskripsi Kelas : Merupakan kelas penghubung antara Tata Usaha dengan sistem untuk mencetak DPK.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
ShowNotification()	public	Operasi ini digunakan untuk menampilkan notifikasi proses yang telah dilakukan



ShowProdiList()	public	Operasi ini digunakan untuk menampilkan daftar program studi
PilihProdi()	public	Operasi ini digunakan untuk menangani Menu Pemilihan Program Studi
CetakDPKMenu()	public	Operasi ini digunakan untuk menangani Menu Pencetakan DPK
ShowKuliahBuka()	public	Operasi ini digunakan untuk menampilkan daftar kuliah yang dibuka
CetakSeluruhDPK()	public	Operasi ini digunakan untuk menangani Menu Pencetakan Seluruh DPK
CetakDPK()	public	Operasi ini digunakan untuk menangani Menu Pencetakan DPK yang dipilih
MarkDPKCetak()	public	Operasi ini digunakan untuk menangani <i>marking</i> terhadap DPK yang akan dicetak
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

### 3.3.19 Kelas CetakDPKController

Deskripsi Kelas : Merupakan kelas yang mengelola *business logic* dari aksi Tata Usaha dalam mencetak DPK.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
RequestProdiList()	public	Operasi ini digunakan untuk mendapatkan daftar program studi dari kelas ProdiData
RequestStatusKuliahBuka()	public	Operasi ini digunakan untuk mendapatkan daftar kuliah yang statusnya dibuka
RequestAllDPK()	public	Operasi ini digunakan untuk mendapatkan seluruh DPK
PrintAllDPK()	public	Operasi ini digunakan untuk mencetak seluruh DPK
MarkKuliah()	public	Operasi ini digunakan untuk menandai kuliah yang akan

		dicetak DPKnya
RequestDPK()	public	Operasi ini digunakan untuk mendapatkan DPK untuk mata kuliah tertentu
PrintMarkedDPK()	public	Operasi ini digunakan untuk mencetak DPK yang dipilih
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

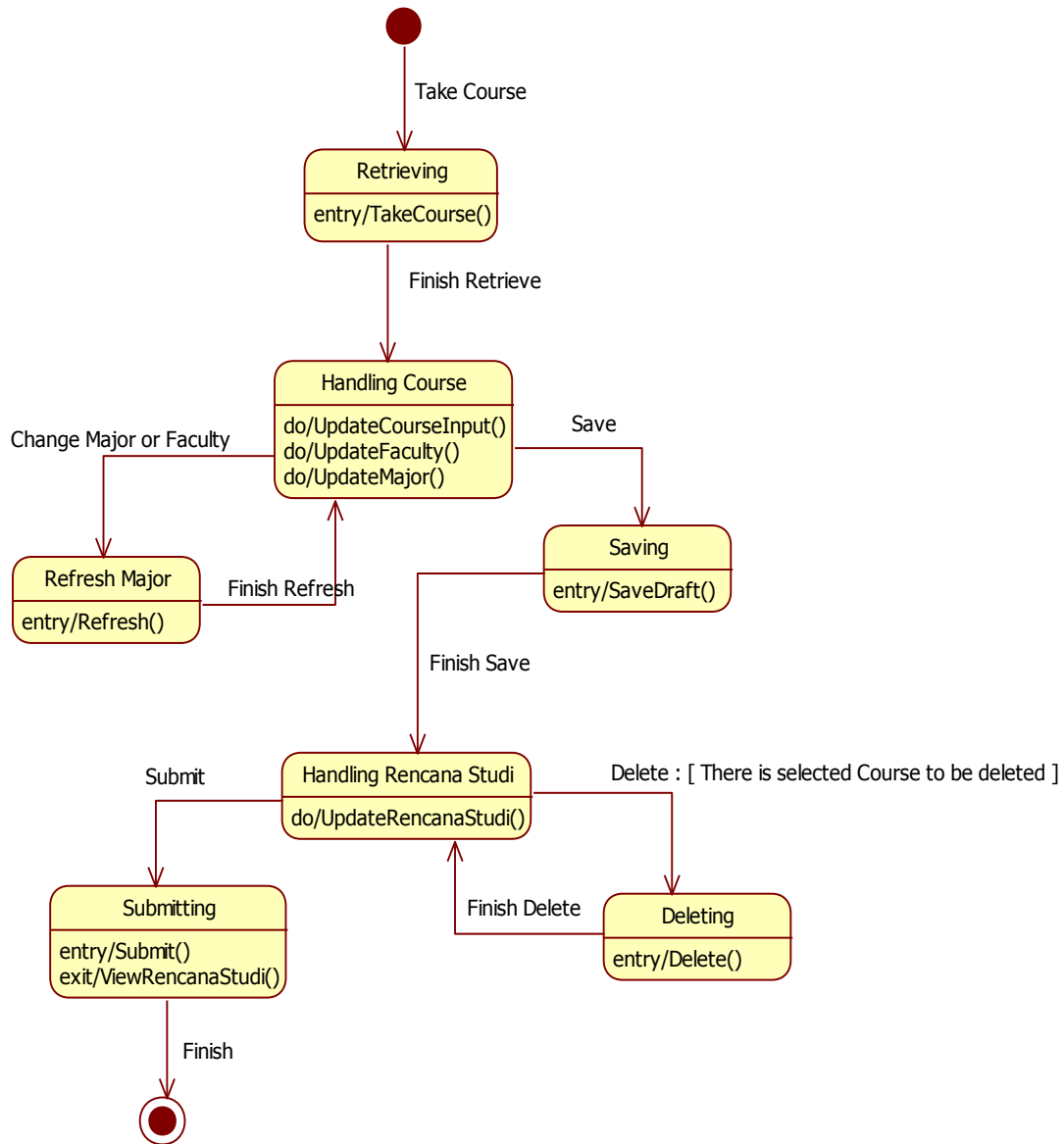
### 3.3.20 Kelas DepKeulInterface

Deskripsi Kelas : Merupakan kelas yang menghubungkan Sistem Akademik dengan Departemen Keuangan dalam mendapatkan status keuangan mahasiswa.

<b>Nama Operasi</b>	<b>Visibility (private, public)</b>	<b>Keterangan</b>
GetStatusKeuanganPaid()	public	Operasi ini digunakan untuk mendapatkan data mahasiswa yang status keuangannya sudah membaya
<b>Nama Atribut</b>	<b>Visibility (private, public)</b>	<b>Tipe</b>
-	-	-

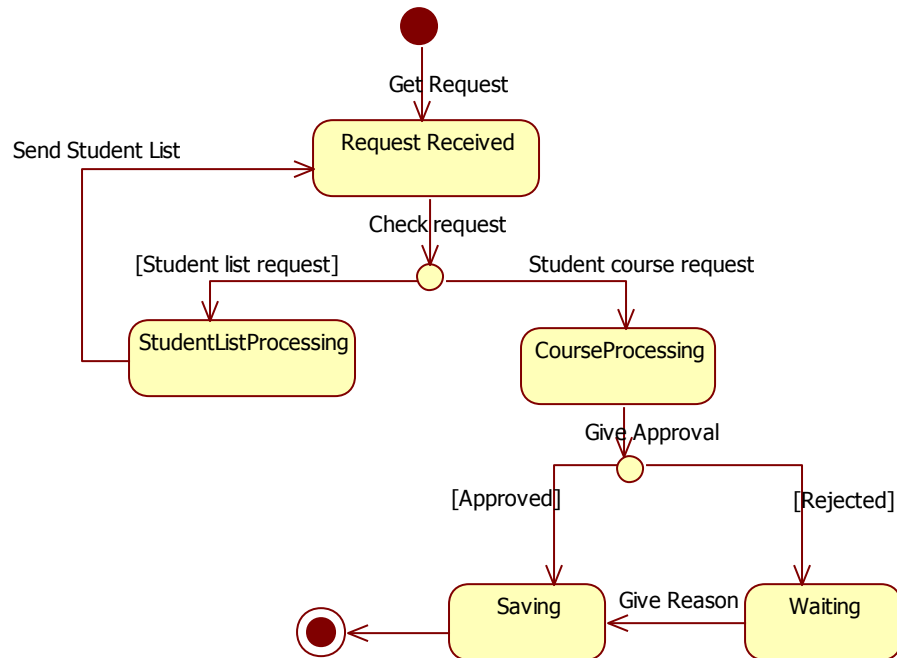
### 3.4 State Chart Diagram

#### 3.4.1 Kelas EntryController



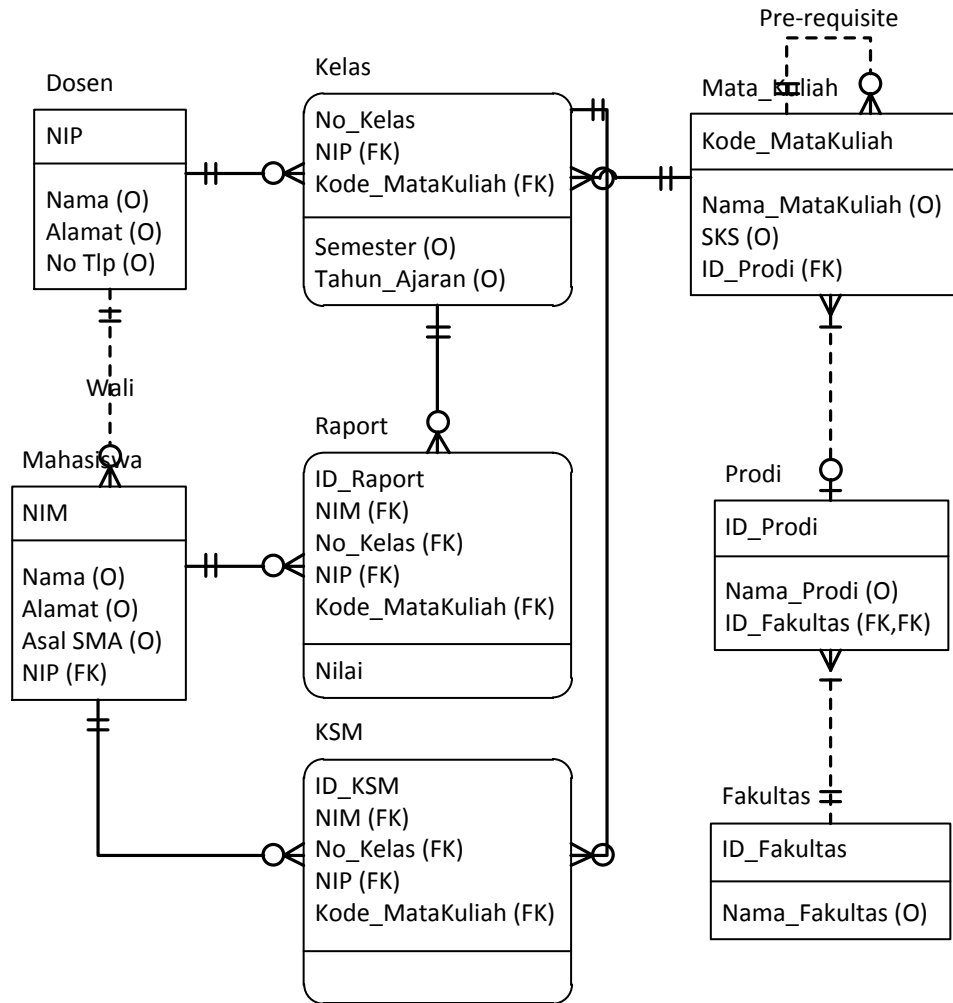
Gambar 3-28 Statechart Diagram untuk kelas EntryController

### 3.4.2 Kelas PersetujuanController



Gambar 3-29 Statechart Diagram untuk kelas PersetujuanController

## 4 Model Perancangan Basis Data



Gambar 4-1 Model Perancangan Basis Data

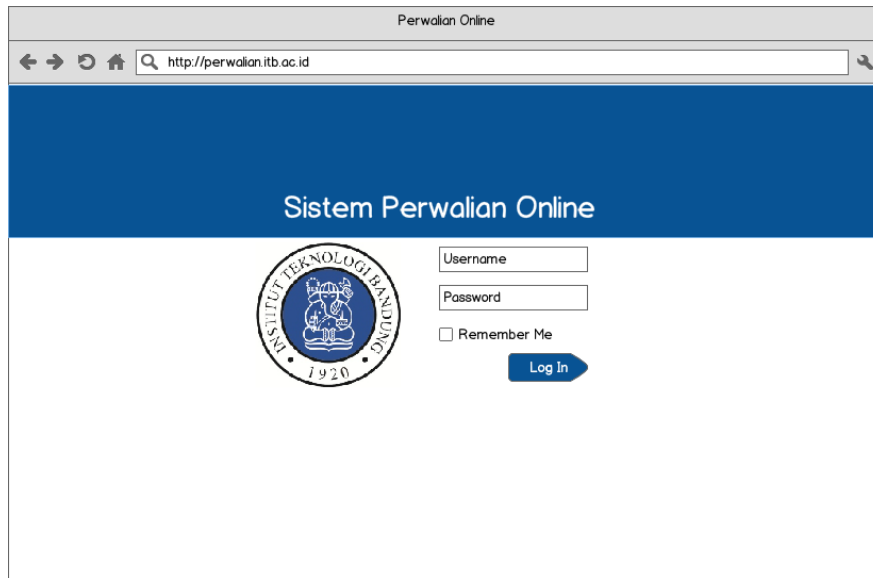
Gambar 4-1 merupakan model perancangan basis data untuk Sistem Informasi Akademik yang dirancang. Adapun asumsi-asumsi yang digunakan dalam merancang basis data ini adalah sebagai berikut:

1. Setiap fakultas dapat memiliki banyak program studi, namun minimal harus terdapat satu program studi. Sebaliknya, setiap prodi hanya dimiliki hanya oleh satu fakultas.
2. Setiap program studi dapat membuka banyak mata kuliah, namun minimal harus terdapat satu mata kuliah yang dibuka. Sebaliknya, setiap mata kuliah hanya dimiliki oleh satu program studi dan mungkin saja tidak dibuka untuk suatu semester.
3. Setiap mata kuliah dapat memiliki lebih dari satu mata kuliah pre-requisite, dan dapat pula tidak memiliki pre-requisite sama sekali.

4. Setiap mata kuliah dapat diajar oleh lebih dari satu dosen, dimana untuk setiap kelasnya diidentifikasi dengan nomor kelas, NIP dosen yang mengajar, dan kode mata kuliah. Untuk setiap kelas yang dibuka, akan dicatat semester dan tahun ajarannya.
5. Setiap kelas mata kuliah yang diambil mahasiswa untuk suatu semester tertentu akan dicatat dalam Kartu Rencana Studi (KSM).
6. Ketika perkuliahan selesai, nilai mahasiswa untuk setiap kelas mata kuliah dicatat dalam Raport.
7. Seorang dosen dapat menjadi dosen wali untuk beberapa mahasiswa sekaligus dan mungkin saja tidak menjadi dosen wali sama sekali. Namun, setiap mahasiswa pasti memiliki hanya satu dosen wali.

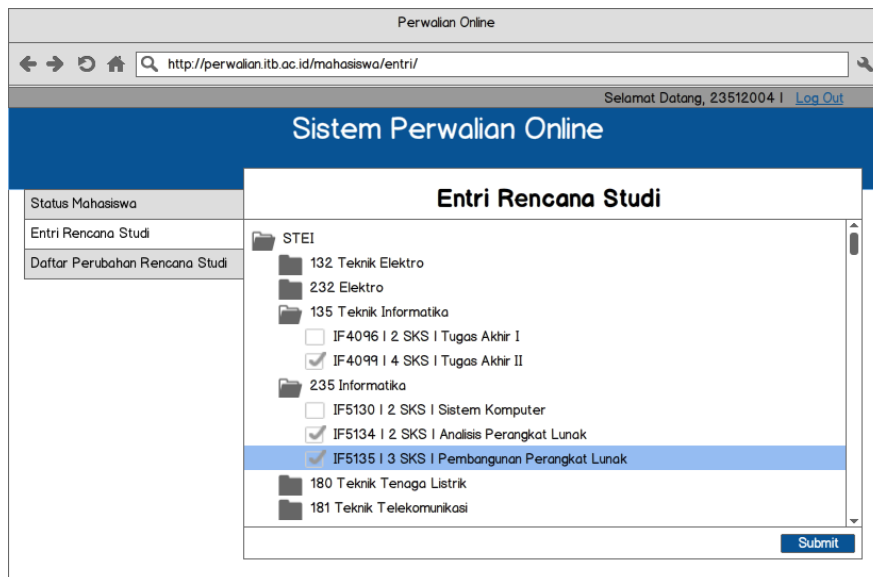
## 5 Model Perancangan Antarmuka/Navigasi

### 5.1 Antarmuka Homepage



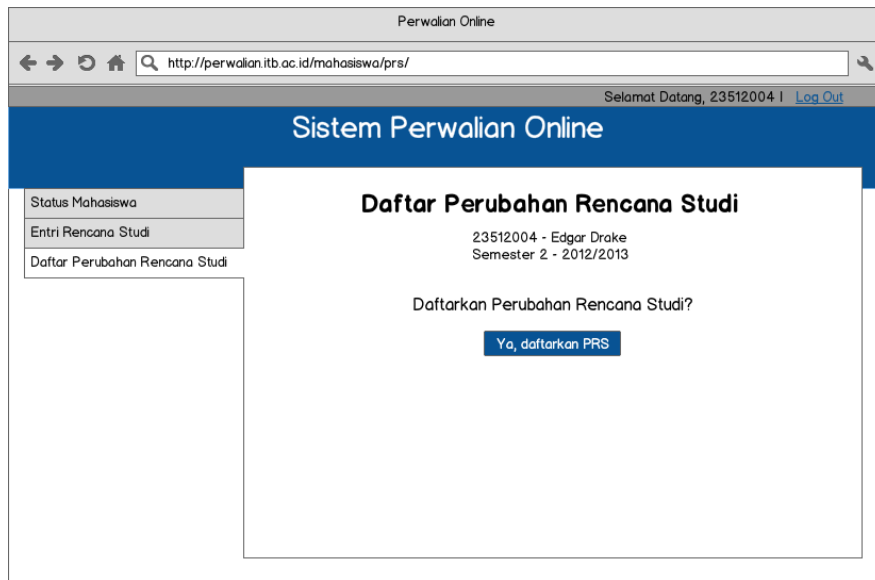
Gambar 5-1 Antarmuka Homepage

### 5.2 Antarmuka Entri FRS



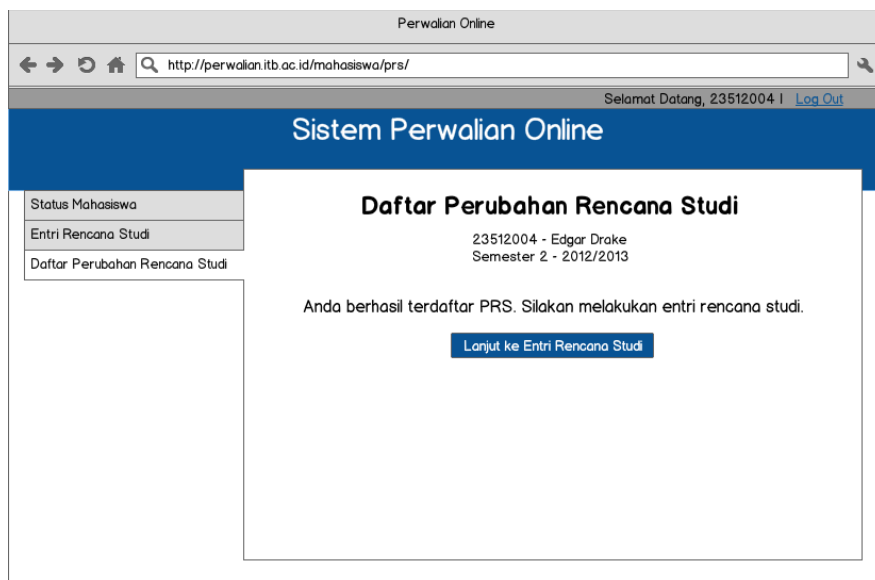
Gambar 5-2 Antarmuka Entri FRS

### 5.3 Antarmuka Pendaftaran PRS



Gambar 5-3 Antarmuka Pendaftaran PRS

### 5.4 Antarmuka PRS Sukses



Gambar 5-4 Antarmuka PRS Sukses



## 5.5 Antarmuka Status Mahasiswa

Perwalian Online

http://perwalian.itb.ac.id/mahasiswa/status/

Selamat Datang, 23512004 | [Log Out](#)

### Sistem Perwalian Online

Status Mahasiswa

Entri Rencana Studi

Daftar Perubahan Rencana Studi

#### Status Mahasiswa

23512004 - Edgar Drake  
Semester 2 - 2012/2013

Kuliah	SKS	Nama Mata Kuliah	Kelas	Dosen
IF5130	2	Sistem Komputer	K1	MA
IF5131	3	Algoritma Pemrograman	K1	IL
IF5132	2	Manajemen Informasi	K1	MLK
IF5133	2	Rekayasa Kebutuhan Perangkat Lunak	K1	HLZ
IF5134	2	Kualitas Perangkat Lunak	K1	HLZ
IF5135	3	Pembangunan Perangkat Lunak	K1	SA

Gambar 5-5 Antarmuka Status Mahasiswa

## 5.6 Antarmuka List Mahasiswa oleh Dosen Wali

Perwalian Online

http://perwalian.itb.ac.id/wali/mahasiswa/status/

Selamat Datang, 19800203 | [Log Out](#)

### Sistem Perwalian Online

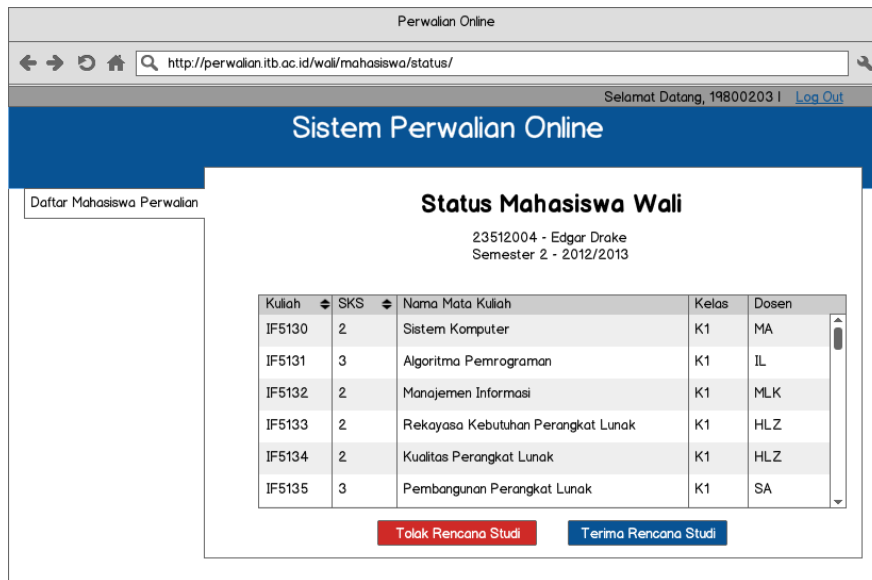
Daftar Mahasiswa Perwalian

19800203 - Dr. Achmad H., ST, MT

NIM	Nama	Rencana Studi	Dosen Wali	IPK-	IPK-	Total
13510001	Rahman Arief	Belum	Dr. Achmad H., ST,	3.5	3.7	20
13511003	Ridhwan Kamil	Sudah	Dr. Achmad H., ST,	3.1	3.0	19
13511007	Mia	Belum	Dr. Achmad H., ST,	2.5	3.1	20
2351200	Edgar Drake	Belum	Dr. Achmad H., ST,	3.5	3.5	18
2351200	Fira Lestari	Belum	Dr. Achmad H., ST,	3.5	3.2	20

Gambar 5-6 Antarmuka List Mahasiswa oleh Dosen Wali

## 5.7 Antarmuka Status Mahasiswa oleh Dosen Wali



Gambar 5-7 Antarmuka Status Mahasiswa oleh Dosen Wali

## 6 Traceability Use Case

Use Case ID	Skenario	Perancangan		
		Sequence Diagram	Class Diagram	Activity Diagram
UC-01	SC-01-01	SQ-01-01	CL-D-01	AC-01
	SC-01-02	SQ-01-02		
UC-02	SC-02-01	SQ-02-01	CL-D-02	AC-02
	SC-02-02	SQ-02-02		
	SC-02-03	SQ-02-03		
	SC-02-04	SQ-02-04		
	SC-02-05	SQ-02-05		
UC-03	SC-03-01	SQ-03-01	CL-D-03	AC-03
	SC-03-02	SQ-03-02		
UC-04	SC-04-01	SQ-04-01	CL-D-04	AC-04
	SC-04-02	SQ-04-01		
UC-05	SC-05-01	SQ-05-01	CL-D-05	AC-05
UC-06	SC-06-01	SQ-06-01	CL-D-06	AC-06
	SC-06-02	SQ-06-02		
UC-07	SC-07-01	SQ-07-01	CL-D-07	AC-07
	SC-07-02	SQ-07-02		
UC-08	SC-08-01	SQ-08-01	CL-D-08	AC-08
	SC-08-02	SQ-08-02		

## 7 Tanggapan Hasil Review

Reviewer:

13509042 – Hanny Fauzia

23512006 – Dian Sa'adillah Maylawati

23512032 – Yeti Nugraheni

23512090 – Rina Praptini

No	Software Design	Checklist	Keterangan Alasan
1.	Apakah batasan sistem sudah digambarkan?	sudah direvisi	Batasan sistem belum secara eksplisit dijelaskan
2.	Apakah kelengkapan spesifikasi (diagram yang dibuat lengkap atau tidak?)	sudah direvisi	Belum ada usecase cetak DPK
3.	Apakah database sudah menyimpan data yang diperlukan atau tidak? (sesuai dengan class)	√	Database yang dirancang sudah sesuai dengan class diagram yang ada
4.	Apakah Usecase sudah mengcover semua fungsi yang seharusnya ada?	sudah direvisi	Ada fungsi yang belum ada yaitu fungsi cetak DPK
5.	Apakah navigasi UI lengkap atau tidak?	√	Secara keseluruhan UI yang dirancang sudah lengkap
6.	Apakah navigasi UI mudah dimengerti atau tidak?	√	Secara keseluruhan UI yang dirancang mudah dimengerti
7.	Apakah semua fungsi yang ada di squence ada di class diagram?	sudah direvisi	Disquence Entri Mata Kuliah ada method pageLoad tetapi di class EntryInterface tidak ada
8.	Apakah didalam class diagram ada hubungan antar komponen yang tidak diperbolehkan? (Model MVC)	√	Secara keseluruhan notasi yang digambarkan benar
9.	Apakah squence diagram ada hubungan antar komponen yang tidak diperbolehkan atau tidak? (Model MVC)	sudah direvisi	disquence Entri Mata Kuliah ada hubungan yang tidak diperbolehkan yaitu aktor langsung ke controller
10.	Apakah alur di squence diagram kontinyu atau tidak?	√	Secara keseluruhan penggambaran squence diagram digambarkan secara kontinyu
11.	Apakah komponennya dirancang reuseable?	masing-masing komponen menjalankan fungsi spesifik (tidak revisi)	Belum reuseable, terutama untuk view dan controllernya
12.	Apakah desainnya modular?	√	
13.	Apakah arsitektur softwarenya sudah dipartisi untuk mempermudah implementasi?	√	
14.	Apakah semua kelas sudah dipetakan ke dalam diagram komponen?	sudah direvisi	Belum, ada kelas yang tidak tercantum di diagram komponen, seperti kelas DepKeuInterface
15.	Apakah ada arsitektur softwarenya	√	Ya, software architecture di jelaskan dalam <i>architectural pattern</i> MVC
16.	Apakah semua penjelasan interface dijelaskan?	√	Ya,
17.	Apakah subsistem software atau komponen utama dijelaskan?	√	Ya, subsistem dan major komponen didefinisikan
18.	Apakah desain dielaborasi dalam diagram-	√	Ya, desain dielaborasi di usecase

	diagram yang biasa ada untuk memenuhi level detail?		dan didetailkan dengan sequence diagram dan activity diagram
19.	Apakah desain atau bagian sistem mengikuti arsitektur <i>pattern</i> yang sudah banyak digunakan?	√	Ya, mengikuti MVC pattern
20.	Apakah desain atau bagian dari desain mengikuti pola desain terkenal?	√	
21.	Apakah desain keseluruhan mengimplementasi semua requirement (fungsional)? Apakah tabel traceability telah dikembangkan?	sudah direvisi	Ada requirement yang belum diimplementasi yaitu cetak DPK
22.	Apakah desain direpresentasikan dalam bentuk yang mudah dimengerti oleh pihak lain?	√	cukup mudah dimengerti dengan menggunakan diagram yang telah dijelaskan
23.	Adalah notasi desain standar? Konsisten?	√	Secara keseluruhan penggunaan desain standar meskipun ada beberapa notasi yang perlu dijelaskan lebih lanjut
24.	Apakah konsep <i>information hiding</i> dan <i>functional independence</i> sudah mengikuti desain keseluruhan?	√	Ya. Terlihat di dalam kelas, ada private dan public, ada getter dan setter.
25.	Apakah desain sudah <i>loose coupling</i> ?	√	Ya. Terlihat dari model perancangan arsitektural yang berbasis komponen
26.	Apakah desain sudah <i>high cohesion</i> ?	√	Ya. Terlihat dari model perancangan arsitektural dan diagram kelas, termasuk level kohesi object dan level kohesi fungsional.
27.	Apakah <i>design specification</i> tersedia untuk membangun <i>software</i> ?	sudah direvisi	Belum lengkap. Tidak ada definisi method untuk diagram kelas. Sebagai programmer maupun orang awam, perlu mengetahui definisi, fungsi, atau kegunaan dari masing-masing method. Definisi kelas masing-masing pun tidak ada, sehingga tidak jelas apa kegunaannya.

Jawaban pertanyaan tambahan:

1. Di sequence diagram Kembali Memilih Mata Kuliah dari Jurusan Lain ada panah dari aktor ke controller yang tidak diproses lebih lanjut.

**Jawab:**

Karena aksi aktor tersebut ditangani pada level interface, sampai aktor melakukan *klik* yang ditangani method OnClick() baru diproses lebih lanjut.

2. KSM mengapa dibuat menjadi tabel?

**Jawab:**

KSM merupakan entitas yang tercipta karena adanya kebutuhan untuk mencatat mata kuliah apa saja yang diambil oleh mahasiswa. Data ini juga diperlukan untuk menentukan Daftar Peserta Kelas (DPK).

