

BAB 6

KESIMPULAN DAN SARAN

Pada bab ini dijelaskan kesimpulan dari pembahasan tugas akhir secara keseluruhan dan saran untuk pengembangan lebih lanjut.

7.1 Kesimpulan

1. Beberapa Rumah sakit belum memanfaatkan sistem komputer secara efektif dalam kinerjanya. Untuk membantu penanganan beban kerja dibutuhkan sistem yang terkomputerisasi, dalam hal ini dengan teknologi komputer berbasis web.
2. Dengan dikembangkannya sistem informasi Rumah Sakit maka permasalahan-permasalahan yang timbul dapat ditekan seminimal mungkin, terutama pada sistem basis data pada setiap transaksi yang dilakukan.
3. Dengan pengembangan aplikasi berbasis web (*web based application development*) memudahkan dalam proses instalasi dan dalam pengembangan sistem selanjutnya.

7.2 Saran

Beberapa saran dan masukan yang dapat disampaikan penulis terhadap pembuatan Pengembangan Sistem Informasi Rumah Sakit adalah :

1. Aplikasi ini diharapkan bisa dikembangkan lebih lanjut dengan memberi tambahan pada layanan pemeriksaan rawat jalan seperti penanganan operasi, layanan kehamilan.

2. Pembangunan sistem pendaftaran secara *online* sehingga lebih memudahkan pengujung untuk dapat melakukan pendaftaran ke rumah sakit.
3. Pencetakan hasil lab untuk pemeriksaan laboratorium dan radiologi.
4. Pembangunan untuk piranti mobile untuk bagian pendaftaran agar pengujung lebih mudah melakukan pengaksesan



Daftar Pustaka

- Abdul Kadir. 2002. *Dasar Pemrograman Web Dinamis Menggunakan PHP*. Yogyakarta: Penerbit Andi.
- Adriansya, Satya, Mawan, A., Dewi, Mondong Jefri A., Ramlan, L., Ramlan, Pamela. 2006. *Php Dengan Framework Codeigniter*.
- Andri, Kristanto. 2003. *Perancangan Sistem Informasi Dan Aplikasinya*, Penerbit Gaya Media Yogyakarta.
- Arbi. 2004. *Manajemen Database Dengan MySQL*. Yogyakarta :Penerbit Andi.
- Chen, P., Kataria, G., Krishnan, R. 2011. *Correlated Failures, Diversification, and Information Security Risk Management*.
- Chen, K.Y., Chen, M.C., dan Liu, W.Y.. 2008. *Designing Data Warehouses For Equipment Anagement System with Genetic Algorithms*.
- Damayanti, Yemima. 2004. *Perancangan Sistem Informasi Penjualan Barang dan Piutang pada CV ALLIANCE TECHNIK*. Bandung : Manajemen Informatika LIKMI.
- Fhatansyah, Ir. 1999. *Basis Data*, Penerbit Bandung: Informatika.
- Girsang, R.. *Sistem Informasi Biodata Guru, Pegawai, dan Siswa Pada SMA Negeri 1 Seribudolok*. Medan : Program Studi D-3 Ilmu Komputer Universitas Sumatra Utara.
- Hartanto, Steven, 2010. *Pembangunan Sistem Informasi Berbasis Dekstop dan Web Multimedia dengan Framework Condeigniter*. Yogyakarta : Program Studi Teknik Informatika Universitas Atma Jaya.
- Haryono, Eko. 2010. *Analisis dan Perancangan Sistem Informasi Pengolahan Data Siswa di SMA N 1 Bayat Klaten Jawa Tengah*.
- Imbar, V. dan Tirta. Eric. 2006. *Analisa, Perancangan dan Implementasi Sistem Informasi Penjualan*

Pelumas. Bandung : Jurusan Sistem Informasi Universitas Kristen Maranatha.

Jamaiti, Esy. 2006. *Sistem Informasi Geografis Geomorfologi Samarinda*. Bandung : Sekolah Sains dan Teknologi Indonesia ST-INTEN.

Jogiyanto Hartanto, MBA, Ph.D.. 1999. *Analisa Dan Desain Sistem Informasi Pendekatan Terstruktur Teori Dan Praktek Aplikasi Bisnis*. Yogyakarta : Penerbit Andi.

Leitao, Paulo, 2011, *Holonic Disturbance Management Architecture for Flexible Manufacturing Systems*.

Midjan, La. 1989. *Sistem Informasi Akuntansi 1*. Bandung:Lembaga Informatika Akuntansi (LIA).

Mulyadi. 1993. *Sistem Informasi Manajemen*. Jakarta : Prehalindo.

Myer, Thomas. 2008. *Professional CodeIgniter*. IndianaPolis: Wiley Publishing.

Rahayu, Sapti. 2006. *Bahan Kuliah Sistem Informasi*. Universitas Atma Jaya Yogyakarta.

Setyanto, Arief. 2007. *Sistem Informasi Siswa Berbasis Mobile Celluler*.

Sutarman. 2003. *Membangun aplikasi Web Dengan PHP dan MySQL*. Yogyakarta:Graha Ilmu.

Tao, Gen. 2012. *International Journal of Manajemen Sumber Daya Manusia*.

Tejoyuwono, Notohadiprawiro. 2006. *Sistem Informasi Pengertian dan Kepentingannya*. Yogyakarta : Ilmu Tanah Universitas Gajahmada.

Utdirartatmo, Firar. 2003. *Mengelola Database Server MySQL di Linux dan Windows*, Penerbit Andi:Yogyakarta.

Widhyaestoeti, Dahlia. 2011. *Rancang Bangun Database Nilai Siswa Tingkat Sekolah Menengah*.

LAMPIRAN



SKPL

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

SIRS

(Sistem Informasi Rumah Sakit)

Untuk :

Universitas Atmajaya Yogyakarta


Dipersiapkan oleh:

Yulius Aditnya Primandaru / 09 07 05835

Program Studi Teknik Informatika - Fakultas Teknologi

Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halaman
		SKPL-SIRS		1/71

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

DAFTAR ISI

1	Pendahuluan	10
1.1	Tujuan	10
1.2	Lingkup Masalah	10
1.3	Definisi, Akronim dan Singkatan	11
1.4	Referensi	11
1.5	Deskripsi umum (Overview)	11
2	Deskripsi Kebutuhan	12
2.1	Perspektif produk	12
2.2	Fungsi Produk	13
2.3	Karakteristik Pengguna	26
2.4	Batasan - batasan	26
2.5	Asumsi dan Ketergantungan	26
3	Kebutuhan khusus	26
3.1	Kebutuhan antarmuka eksternal	26
3.1.1	Antarmuka pemakai	27
3.1.2	Antarmuka perangkat keras	27
3.1.3	Antarmuka perangkat lunak	27
3.1.4	Antarmuka Komunikasi	28
3.2	Kebutuhan fungsionalitas Perangkat Lunak	29
3.2.1	Use Case Diagram	29
	Gambar 2. Use Case Diagram	30
4	Spesifikasi Rinci Kebutuhan	30
4.1	Spesifikasi Kebutuhan Fungsionalitas	30
4.1.1	Use case Spesification : Login	30
1.	Brief Description	30
2.	Primary Actor	30
3.	Supporting Actor	30
4.	Basic Flow	30
5.	Alternative Flow	30
6.	Error Flow	30
7.	PreConditions	30
8.	PostConditions	31
4.1.2	Use case Spesification : Edit Password	31
1.	Brief Description	31
2.	Primary Actor	31
3.	Supporting Actor	31
4.	Basic Flow	31
5.	Alternative Flow	31
6.	Error Flow	31

7. PreConditions	32
8. PostConditions	32
4.1.3 Use case Spesification : Pengelolaan Data Pengguna.....	32
1. Brief Description	32
2. Primary Actor	32
3. Supporting Actor	32
4. Basic Flow	32
5. Alternative Flow	33
6. Error Flow	34
7. PreConditions	34
8. PostConditions	34
4.1.4 Use case Spesification : Pengelolaan Data Poliklinik	34
1. Brief Description	34
2. Primary Actor	35
3. Supporting Actor	35
4. Basic Flow	35
5. Alternative Flow	35
6. Error Flow	36
7. PreConditions	37
8. PostConditions	37
4.1.5 Use case Spesification : Pengelolaan Data Pegawai.....	37
1. Brief Description	37
2. Primary Actor	37
3. Supporting Actor	37
4. Basic Flow	37
5. Alternative Flow	38
6. Error Flow	39
7. PreConditions	39
8. PostConditions	39
4.1.6 Use case Spesification : Pengelolaan Jadwal Dokter.....	40
1. Brief Description	40
2. Primary Actor	40
3. Supporting Actor	40
4. Basic Flow	40
5. Alternative Flow	41
6. Error Flow	41
7. PreConditions	42
8. PostConditions	42
4.1.7 Use case Spesification : Pengelolaan Data Obat ..	42

1. Brief Description	42
2. Primary Actor	42
3. Supporting Actor	42
4. Basic Flow	42
5. Alternative Flow	43
6. Error Flow	43
7. PreConditions	43
8. PostConditions	43
4.1.8 Use case Spesification : Pengelolaan Bidang Medis	43
1. Brief Description	43
2. Primary Actor	44
3. Supporting Actor	44
4. Basic Flow	44
5. Alternative Flow	44
6. Error Flow	45
7. PreConditions	45
8. PostConditions	45
4.1.9 Use case Spesification : Pengelolaan Data Pasien.....	46
1. Brief Description	46
2. Primary Actor	46
3. Supporting Actor	46
4. Basic Flow	46
5. Alternative Flow	47
6. Error Flow	48
7. PreConditions	49
8. PostConditions	50
4.1.10 Use case Spesification : Pengelolaan Data Resep	50
1. Brief Description	50
2. Primary Actor	50
3. Supporting Actor	50
4. Basic Flow	50
5. Alternative Flow	51
6. Error Flow	51
7. PreConditions	51
8. PostConditions	51
4.1.11 Use case Spesification : Pengelolaan data kamar	51
1. Brief Description	51
2. Primary Actor	52
3. Supporting Actor	52

4. Basic Flow	52
5. Alternative Flow	52
6. Error Flow	53
7. PreConditions	54
8. PostConditions	54
4.1.12 Use case Spesification : Pengelolaan Kelas Inap	54
1. Brief Description	54
2. Primary Actor	54
3. Supporting Actor	54
4. Basic Flow	54
5. Alternative Flow	55
6. Error Flow	56
7. PreConditions	56
8. PostConditions	56
4.1.13 Use case Spesification : Pengelolaan Rawat Inap	56
1. Brief Description	56
2. Primary Actor	57
3. Supporting Actor	57
4. Basic Flow	57
5. Alternative Flow	57
6. Error Flow	58
7. PreConditions	58
8. PostConditions	58
4.1.14 Use case Spesification : Pendaftaran Pemeriksaan	59
1. Brief Description	59
2. Primary Actor	59
3. Supporting Actor	59
4. Basic Flow	59
5. Alternative Flow	59
6. Error Flow	60
7. PreConditions	60
8. PostConditions	60
4.1.15 Use case Spesification : Pendaftaran Pemeriksaan	60
1. Brief Description	60
2. Primary Actor	60
3. Supporting Actor	60
4. Basic Flow	60
5. Alternative Flow	61
6. Error Flow	61
7. PreConditions	61
8. PostConditions	62
4.1.16 Use case Spesification : Pemeriksaan Awal	62
1. Brief Description	62
2. Primary Actor	62
3. Supporting Actor	62

4. Basic Flow	62
5. Alternative Flow	63
6. Error Flow	63
7. PreConditions	63
8. PostConditions	63
4.1.17 Use case Spesification : Pemeriksaan Dokter	63
1. Brief Description	63
2. Primary Actor	63
3. Supporting Actor	63
4. Basic Flow	63
5. Alternative Flow	64
6. Error Flow	65
7. PreConditions	65
8. PostConditions	65
4.1.18 Use case Spesification : Pembayaran Rumah sakit	65
1. Brief Description	65
2. Primary Actor	65
3. Supporting Actor	65
4. Basic Flow	65
5. Alternative Flow	66
6. Error Flow	66
7. PreConditions	66
4.1.19 Use case Spesification :Pengelolaan Layanan Lab	66
1. Brief Description	66
2. Primary Actor	66
3. Supporting Actor	66
4. Basic Flow	66
5. Alternative Flow	67
6. Error Flow	68
7. PreConditions	68
8. PostConditions	68
4.1.20 Use case Spesification : Pengelolaan Hasil Lab	68
1. Brief Description	68
2. Primary Actor	68
3. Supporting Actor	68
4. Basic Flow	68
5. Alternative Flow	69
6. Error Flow	70
7. PreConditions	70
8. PostConditions	70
4.1.21 Use case Spesification : Pengelolaan Radiologi	70
1. Brief Description	70
2. Primary Actor	71

3. Supporting Actor	71
4. Basic Flow	71
5. Alternative Flow	71
6. Error Flow	72
7. PreConditions	72
8. PostConditions	72
4.1.22 Use case Spesification : Pengelolaan Hasil radiologi.....	73
1. Brief Description	73
2. Primary Actor	73
3. Supporting Actor	73
4. Basic Flow	73
5. Alternative Flow	74
6. Error Flow	74
7. PreConditions	75
8. PostConditions	75
Gambar 3. ERD	Error! Bookmark not defined.

Daftar Gambar

1. Arsitektur Perangkat Lunak SC3	13
2. Use Case Diagram	30
3. ERD	77

1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SIRS (*Sistem Informasi Rumah Sakit*) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan pengguna) performansi (kemampuan perangkat lunak dari segi kecepatan, tempat penyimpanan yang dibutuhkan, serta keakuratan), dan atribut (*feature-feature* tambahan yang dimiliki sistem), serta mendefinisikan fungsi perangkat lunak. SKPL-SIRS ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak SIRS dikembangkan dengan tujuan untuk :

1. Pengelolaan data pasien, dokter dan karyawan di Rumah Sakit.
2. Pengelolaan data pemeriksaan.

Program Studi Teknik Informatika	SKPL – SIRS	10/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.3 Definisi, Akronim dan Singkatan

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SIRS-XXX	Kode yang merepresentasikan kebutuhan pada SIRS (Sistem Informasi Rumah Sakit) dimana XXX merupakan nomor fungsi produk.
SIRS	Perangkat lunak untuk mengelola data rumah sakit. (berbasis web)

1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Hantana, Juli Sapta Putra. 2006. *SKPL SC3 (Smart Client for Cyber Community)*. Yogyakarta: Universitas Atma Jaya Yogyakarta
2. Wibisono, Aryo . 2006. *SKPL SIAMA (Sistem Informasi Akademik Mahasiswa)*. Yogyakarta : Universitas Atma Jaya Yogyakarta
3. Putra, Tri Warsono. 2011. *SKPL SIRSPS (SISTEM INFORMASI RUMAH SAKIT PATMASARI)*. Yogyakarta : Universitas Atma Jaya Yogyakarta

1.5 Deskripsi umum (Overview)

Secara umum dokumen SKPL ini terbagi atas 3 bagian utama. Bagian utama berisi penjelasan mengenai dokumen SKPL tersebut yang mencakup tujuan pembuatan SKPL, ruang lingkup masalah dalam pengembangan perangkat lunak tersebut, definisi, referensi dan deskripsi umum tentang dokumen SKPL ini.

Bagian kedua berisi penjelasan umum tentang perangkat lunak SIRS yang akan dikembangkan, mencakup

Program Studi Teknik Informatika	SKPL – SIRS	11/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

perspektif produk yang akan dikembangkan, fungsi produk perangkat lunak, karakteristik pengguna, batasan dalam penggunaan perangkat lunak dan asumsi yang dipakai dalam pengembangan perangkat lunak SIRS tersebut.

Bagian ketiga berisi penjelasan secara lebih rinci tentang kebutuhan perangkat lunak SIRS yang akan dikembangkan.

2 Deskripsi Kebutuhan

2.1 Perspektif produk

SIRS merupakan perangkat lunak yang dikembangkan untuk membantu pengelolaan sistem informasi untuk rumah sakit. Sistem ini menangani pengelolaan data-data pasien, dokter, dan karyawan juga pengelolaan pemeriksaan dan pembayaran yang ada di rumah sakit supaya pengelolaannya lebih cepat dan akurat.

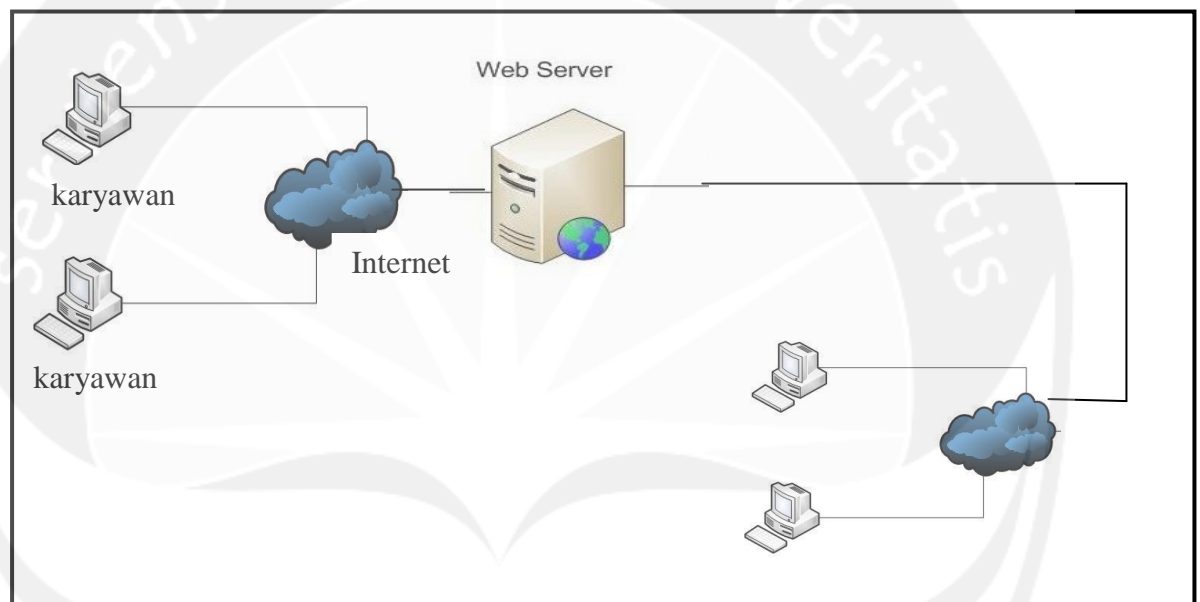
Secara garis besar, sistem ini dibangun untuk memudahkan karyawan rumah sakit untuk mengelola data-datanya. Proses bisnis dari sistem ini dimulai oleh administrator yang bertugas untuk mengelola data-data awal. Kemudian resepsionis yang bertugas untuk mengelola pendaftaran pasien, perawat untuk melakukan pemeriksaan awal, dokter untuk melakukan pemeriksaan, dan kasir untuk melakukan pembayaran. Masing-masing pengguna yang ada akan saling terkait.

Perangkat lunak SIRS ini berjalan pada platform Windows CE untuk aplikasi web, dan dibuat menggunakan framework CodeIgniter.

Pengguna akan berinteraksi dengan sistem melalui antarmuka GUI (Graphical Pengguna Interface). Pada sistem ini, seperti terlihat pada gambar 1, arsitektur perangkat lunak yang digunakan berupa aplikasi web, di

Program Studi Teknik Informatika	SKPL – SIRS	12/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

mana semua data disimpan di basisdata. Pengguna dapat mengakses data yang ada di server tersebut secara on-line. Inputan data yang dimasukkan akan disimpan dalam basisdata server, sehingga jika ada pencarian data, maka data yang diinginkan akan dicari ke basisdata server.



Gambar 1. Arsitektur Perangkat Lunak

2.2 Fungsi Produk

Fungsi produk perangkat lunak SIRS adalah sebagai berikut :

1. Fungsi Login (SKPL-SIRS-001)

Merupakan fungsi yang digunakan oleh petugas Rumah Skit Sahabat Sehat untuk dapat masuk dalam sistem dan mempunyai hak akses sesuai dengan peran yang dimiliki oleh pengguna. Pengguna harus

mempunyai username dan password yang terdaftar ke basisdata.

1. *Fungsi Edit Password (SKPL-SIRS-002)*

Merupakan fungsi yang di gunakan pengguna untuk mengganti password dari username yang dimilikinya.

2. *Fungsi Pengelolaan Data Pengguna (SKPL-SIRS-003)*

Merupakan fungsi yang digunakan oleh pengguna admin untuk mengelola data pengguna.

Fungsi Pengelolaan Data Pengguna mencakup :

a. *Fungsi Input Data Pengguna (SKPL-SIRS-003-01)*

Fungsi ini digunakan untuk menambah data pengguna atau data pengguna meliputi, nama pengguna, username dan rolenya.

b. *Fungsi Ubah Data Pengguna (SKPL-SIRS-003-02)*

Fungsi ini digunakan untuk mengubah data pengguna yang sudah ada sebelumnya.

c. *Fungsi Hapus Data Pengguna (SKPL-SIRS-003-03)*

Fungsi ini digunakan untuk menghapus data pengguna yang sudah ada.

3. *Fungsi Pengelolaan Data Poliklinik (SKPL-SIRS-004)*

Merupakan fungsi yang digunakan oleh pengguna admin untuk mengelola data poliklinik .

Fungsi Pengelolaan Data Karyawan mencakup :

a. *Fungsi Input Data Poliklinik (SKPL-SIRS-004-01)*

Fungsi ini digunakan untuk menambah data poliklinik meliputi nama poliklinik.

b. Fungsi Ubah Data Poliklinik (SKPL-SIRS-004-02)

Fungsi ini digunakan untuk mengubah data poliklinik yang sudah ada sebelumnya.

c. Fungsi Hapus Data Poliklinik (SKPL-SIRS-004-03)

Fungsi ini digunakan untuk menghapus data poliklinik yang sudah ada.

4. Fungsi Pengelolaan Data Pegawai (SKPL-SIRS-005)

Merupakan fungsi yang digunakan oleh pengguna admin untuk mengelola data pegawai.

Fungsi Pengelolaan Data Pegawai mencakup :

a. Fungsi Input Data Pegawai (SKPL-SIRS-005-01)

Fungsi ini digunakan untuk menambah data pegawai meliputi nama pegawai, jenis kelamin, bidang medis dll.

b. Fungsi Ubah Data Pegawai (SKPL-SIRS-005-02)

Fungsi ini digunakan untuk mengubah data pegawai yang sudah ada sebelumnya.

c. Fungsi Hapus Data Pegawai (SKPL-SIRS-005-03)

Fungsi ini digunakan untuk menghapus data pegawai yang sudah ada.

5. Fungsi Pengelolaan Jadwal Dokter (SKPL-SIRS-006)

Program Studi Teknik Informatika	SKPL – SIRS	15/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Merupakan fungsi yang digunakan oleh pengguna admin untuk mengelola jadwal kerja dokter meliputi hari dan waktu dokter bekerja.

Fungsi Pengelolaan Jadwal Dokter mencakup :

a. *Fungsi Input Jadwal Dokter (SKPL-SIRS-006-01)*

Fungsi ini digunakan untuk menambah data jadwal dokter.

b. *Fungsi Ubah Jadwal Dokter (SKPL-SIRS-006-02)*

Fungsi ini digunakan untuk mengubah jadwal dokter yang sudah ada sebelumnya.

c. *Fungsi Hapus Jadwal Dokter (SKPL-SIRS-006-03)*

Fungsi ini digunakan untuk menghapus jadwal dokter yang sudah tidak di gunakan lagi dalam basis data.

6. *Fungsi Pengelolaan Data Obat (SKPL-SIRS-007)*

Merupakan fungsi yang digunakan oleh pengguna admin untuk mengelola data obat.

Fungsi Pengelolaan Data Obat mencakup :

a. *Fungsi Input Data Obat (SKPL-SIRS-007-01)*

Fungsi ini digunakan untuk menambah data obat yang meliputi nama obat, jenis obat, stock obat, harga jual, tanggal kadaluarsa, harga dasar, keterangan dll.

b. *Fungsi Ubah Data Obat (SKPL-SIRS-007-02)*

Fungsi ini digunakan untuk mengubah data obat yang sudah ada sebelumnya.

c. *Fungsi Hapus Data Obat (SKPL-SIRS-007-03)*

Program Studi Teknik Informatika	SKPL – SIRS	16/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi ini digunakan untuk menghapus data obat yang sudah ada.

7. Fungsi Pengelolaan Bidang Medis (SKPL-SIRS-008)

Fungsi ini merupakan fungsi yang digunakan untuk mengelola Bidang medis.

Fungsi Pengelolaan Bidang Medis meliputi :

a. Fungsi Input Bidang Medis (SKPL-SIRS-008-01)

Fungsi ini digunakan untuk menambah bidang medis diantaranya nama bidang, no bidang medis dst.

b. Fungsi Ubah Bidang Medis (SKPL-SIRS-008-02)

Fungsi ini digunakan untuk mengubah data bidang medis yang sudah ada sebelumnya.

c. Fungsi Hapus Bidang Medis (SKPL-SIRS-008-03)

Fungsi ini digunakan untuk menghapus data bidang medis yang sudah ada.

8. Fungsi Pengelolaan Data Pasien (SKPL-SIRS-009)

Fungsi ini merupakan fungsi yang digunakan untuk mengelola data pasien.

Fungsi Pengelolaan Data Pasien meliputi :

a. Fungsi Input Data Pasien (SKPL-SIRS-009-01)

Fungsi ini digunakan untuk menambah data pasien antara lain id pasien, nama pasien, alamat, tempat dan tanggal lahir dll.

b. Fungsi Ubah Data Pasien (SKPL-SIRS-009-02)

Fungsi ini digunakan untuk mengubah data pasien yang sudah ada sebelumnya.

Program Studi Teknik Informatika	SKPL – SIRS	17/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

c. *Fungsi Hapus Data Pasien (SKPL-SIRS-009-03)*

Fungsi ini digunakan untuk menghapus data pasien yang sudah ada.

9. *Fungsi Pengelolaan Resep (SKPL-SIRS-010)*

Fungsi ini merupakan fungsi yang digunakan untuk mengelola resep yang dilakukan oleh apoteker.

Fungsi Pengelolaan Kecantikan meliputi :

a. *Fungsi Display Resep (SKPL-SIRS-010-01)*

Fungsi ini digunakan untuk menampilkan resep dengan status tertentu.

b. *Fungsi Display Detail Resep (SKPL-SIRS-010-02)*

Fungsi ini digunakan untuk menampilkan detail resep dari no resep tertentu.

c. *Fungsi Update Resep (SKPL-SIRS-010-03)*

Fungsi ini digunakan untuk melakukan perubahan status resep.

10. *Fungsi Pengelolaan Data Kamar (SKPL-SIRS-011)*

Fungsi ini merupakan fungsi yang digunakan untuk mengelola Data kamar yang digunakan untuk rawat inap.

a. *Fungsi Input Data Kamar (SKPL-SIRS-011-03)*

Fungsi ini digunakan untuk menambah data kamar seperti tanggal masuk dan keluar maupun status kamar.

b. *Fungsi Ubah Data Kamar (SKPL-SIRS-011-02)*

Program Studi Teknik Informatika	SKPL – SIRS	18/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi ini digunakan untuk mengubah data kamar yang sudah ada sebelumnya.

c. Fungsi Hapus Data Kamar (SKPL-SIRS-011-03)

Fungsi ini digunakan untuk menghapus data kamar.

11. Fungsi Pengelolaan Kelas Rawat Inap (SKPL-SIRS-012)

Fungsi ini merupakan fungsi yang digunakan untuk mengelola Kelas Rawat Inap.

a. Fungsi Input Kelas Rawat Inap (SKPL-SIRS-012-01)

Fungsi ini digunakan untuk menambah kelas rawat inap baru kedalam basis data.

b. Fungsi Ubah Data Kelas Rawat Inap (SKPL-SIRS-012-02)

Fungsi ini digunakan untuk mengubah data kelas rawat inap yang sudah ada sebelumnya.

c. Fungsi Hapus Data Kelas Rawat Inap (SKPL-SIRS-012-03)

Fungsi ini digunakan untuk menghapus data kelas rawat inap.

12. Fungsi Pengelolaan Rawat Inap (SKPL-SIRS-013)

Fungsi ini merupakan fungsi yang digunakan untuk mengelola resep yang di lakukan oleh apoteker.

a. Fungsi Input Rawat Inap (SKPL-SIRS-013-01)

Fungsi ini digunakan untuk menambah rawat inap baru kedalam basis data.

b. Fungsi Ubah Data Rawat Inap (SKPL-SIRS-013-02)

Fungsi ini digunakan untuk mengubah data inap yang sudah ada sebelumnya.

c. Fungsi Hapus Kelas Rawat Inap (SKPL-SIRS-013-03)

Fungsi ini digunakan untuk menghapus data kelas rawat inap.

13. Fungsi Pengelolaan Pendaftaran Pemeriksaan (SKPL-SIRS-014)

Fungsi ini merupakan fungsi yang digunakan untuk melakukan pendaftaran pemeriksaan.

Fungsi Pengelolaan Data Pasien meliputi :

a. Fungsi Tambah Pasien (SKPL-SIRS-014-01)

Merupakan fungsi yang digunakan untuk menambah pasien baru kedalam basis data.

b. Cetak Bukti Pendaftaran (SKPL-SIRS-014-02)

Merupakan fungsi untuk mencetak bukti pendaftaran sebagai pasien di rumah sakit.

c. Fungsi Mencari Pasien (SKPL-SIRS-014-03)

Fungsi ini digunakan untuk mencari data pasien yang nama pasiennya di inputkan.

d. Fungsi Daftar Pemeriksaan (SKPL-SIRS-014-04)

Fungsi ini digunakan untuk melakukan pendaftaran pemeriksaan, dimulai dengan memilih nama rumah sakit kemudian memilih dokter yang ada dalam rumah sakit tersebut, kemudian mendaftarkan nama pasien ke antrian.

Program Studi Teknik Informatika	SKPL – SIRS	20/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

14. Fungsi Pengelolaan Pendaftaran Rawat Inap (**SKPL-SIRS-015**)

Fungsi ini merupakan fungsi yang digunakan untuk mengelola pendaftaran rawat inap.

Fungsi Pendaftaran Perawatan meliputi :

a. Fungsi Tambah Pasien Rawat Inap (**SKPL-SIRS-015-01**)

Fungsi ini digunakan untuk menambah pasien baru yang melakukan rawat inap kedalam basis data.

b. Fungsi Ubah Pasien Rawat Inap (**SKPL-SIRS-015-02**)

Fungsi ini digunakan untuk mengubah data pasien rawat inap yang sudah ada sebelumnya.

c. Fungsi Hapus Pasien Rawat Inap (**SKPL-SIRS-015-03**)

Fungsi ini digunakan untuk menghapus data pasien rawat inap yang sudah ada.

d. Fungsi Mencari Pasien Rawat Inap (**SKPL-SIRS-015-04**)

Fungsi ini digunakan untuk mencari data pasien yang nama pasien rawat inap.

2. Fungsi Pemeriksaan Awal (**SKPL-SIRS-016**)

Fungsi ini merupakan fungsi yang digunakan untuk melakukan pemeriksaan awal yang dilakukan oleh perawat, dimana data pasien yang ada adalah data pasien yang status antriannya Masuk, kemudian perawat melakukan pemeriksaan awal.

Fungsi Pengelolaan Data Pasien meliputi :

Program Studi Teknik Informatika	SKPL – SIRS	21/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

a. Fungsi Entry Data Pemeriksaan Awal (SKPL-SIRS-016-01)

Fungsi ini digunakan untuk menginputkan data pemeriksaan awal pasien data yang diinputkan meliputi antara lain tensi, berat badan, tinggi badan suhu badan dll. Dan status berubah mendari masuk menjadi periksa.

b. Fungsi Ubah Data Pemeriksaan Awal (SKPL-SIRS-016-02)

Fungsi ini digunakan untuk mengubah data pasien yang sudah melakukan pemeriksaan awal. Dan status berubah mendari masuk menjadi periksa.

15. Fungsi Pemeriksaan Dokter (SKPL-SIRS-017)

Fungsi ini merupakan fungsi yang digunakan untuk melakukan penyimpanan data pemeriksaan yang dilakukan oleh dokter, dimana data pasien yang ada adalah data pasien yang status antriannya Periksa, yang artinya sudah di periksa oleh perawat.

Fungsi Pemeriksaan Dokter meliputi :

a. Fungsi Entry Hasil Pemeriksaan Pasien (SKPL-SIRS-017-01)

Fungsi ini digunakan untuk menginputkan hasil dari pemeriksaan oleh dokter misalnya hasil konsultasi, resep obat yang di berikan, pilihan tindakan yang dilakukan, dll.

b. Fungsi Ubah Hasil Pemeriksaan Pasien (SKPL-SIRS-017-02)

Program Studi Teknik Informatika	SKPL – SIRS	22/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi ini digunakan untuk melakukan perubahan pemeriksaan yang dilakukan oleh dokter yang telah dimasukkan dalam basis data.

c. Fungsi Kelola Resep Dokter (SKPL-SIRS-017-03)

Fungsi ini digunakan untuk membuat resep dokter untuk pasien yang melakukan pemeriksaan.

16. Fungsi Pembayaran Rumah sakit (SKPL-SIRS-018)

Fungsi ini merupakan fungsi yang digunakan untuk melakukan pembayaran rumah sakit atau pembayaran perawatan.

Fungsi pembayaran rumah sakit meliputi :

a. Fungsi Pembayaran Rawat Inap (SKPL-SIRS-018-01)

Fungsi ini digunakan untuk melakukan pembayaran pasien rawat inap, dimana kita bisa mengetahui total pembayaran dari pasien.

b. Fungsi Cetak Pembayaran Rawat Inap (SKPL-SIRS-018-02)

Fungsi ini digunakan untuk melakukan cetak nota pembayaran pasien rawat inap.

c. Fungsi Pembayaran Rawat Jalan (SKPL-SIRS-018-03)

Fungsi ini digunakan untuk melakukan pembayaran pasien rawat jalan.

d. Fungsi Cetak Pembayaran Rawat Jalan (SKPL-SIRS-018-03)

Program Studi Teknik Informatika	SKPL – SIRS	23/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi ini digunakan untuk melakukan cetak nota pembayaran pasien rawat jalan.

17. *Fungsi Pengelolaan Layanan Lab (SKPL-SIRS-019)*

Fungsi ini merupakan fungsi yang digunakan untuk mengelola layanan lab yang dilakukan oleh petugas lab.

a. *Fungsi Input Layanan Lab (SKPL-SIRS-019-01)*

Fungsi ini digunakan untuk menambah data Layanan Lab seperti test darah dst.

b. *Fungsi Ubah Layanan Lab (SKPL-SIRS-019-02)*

Fungsi ini digunakan untuk mengubah layanan lab jika terjadi perubahan.

c. *Fungsi Hapus Layanan Lab (SKPL-SIRS-019-03)*

Fungsi ini digunakan untuk menghapus data layanan lab.

18. *Fungsi Pengelolaan Hasil Lab (SKPL-SIRS-020)*

Fungsi ini merupakan fungsi yang digunakan untuk mengelola hasil lab yang dilakukan oleh petugas lab.

a. *Fungsi Input Hasil Lab (SKPL-SIRS-020-01)*

Fungsi ini digunakan untuk menambah data hasil Lab seperti hasil test darah dst.

b. *Fungsi Ubah Hasil Lab (SKPL-SIRS-020-02)*

Fungsi ini digunakan untuk mengubah hasil lab jika terjadi perubahan.

c. *Fungsi Hapus Hasil Lab (SKPL-SIRS-020-03)*

Fungsi ini digunakan untuk menghapus data hasil lab.

Program Studi Teknik Informatika	SKPL – SIRS	24/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

19. Fungsi Pengelolaan Radiologi (**SKPL-SIRS-021**)

Fungsi ini merupakan fungsi yang digunakan untuk mengelola data radiologi yang dilakukan oleh petugas radiologi.

a. Fungsi Input Data Radiologi (**SKPL-SIRS-021-01**)

Fungsi ini digunakan untuk menambah data Radiologi.

b. Fungsi Ubah Data Radiologi (**SKPL-SIRS-021-02**)

Fungsi ini digunakan untuk mengubah data Radiologi yang sudah ada.

c. Fungsi Hapus Data Radiologi (**SKPL-SIRS-021-03**)

Fungsi ini digunakan untuk menghapus data radiologi yang sudah tidak diperlukan.

20. Fungsi Pengelolaan Hasil Radiologi (**SKPL-SIRS-022**)

Fungsi ini merupakan fungsi yang digunakan untuk mengelola hasil radiologi yang dilakukan oleh petugas radiologi.

a. Fungsi Input Data Radiologi (**SKPL-SIRS-022-01**)

Fungsi ini digunakan untuk menambah data Hasil Radiologi.

b. Fungsi Ubah Data Hasil Radiologi (**SKPL-SIRS-022-02**)

Fungsi ini digunakan untuk mengubah data Hasil Radiologi yang sudah ada.

c. Fungsi Hapus Data Hasil Radiologi (**SKPL-SIRS-022-03**)

Program Studi Teknik Informatika	SKPL – SIRS	25/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Fungsi ini digunakan untuk menghapus data hasil radiologi yang sudah tidak diperlukan.

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak SIRS adalah :

1. Pengguna harus memahami pengoperasian komputer (PC).
2. Pengguna diharapkan memahami secara dalam tentang internet.

2.4 Batasan - batasan

Batasan-batasan dalam pengembangan perangkat lunak dalam aplikasi SIRS ini antara lain :

- a. Kebijakan Umum
Berpedoman pada tujuan dari pengembangan perangkat Lunak SIRS.
- b. Keterbatasan perangkat keras
Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

2.5 Asumsi dan Ketergantungan

Sistem ini dapat di gunakan untuk aplikasi yang diakses melalui web browser.

3 Kebutuhan khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antar muka eksternal pada perangkat lunak SC3 meliputi kebutuhan antarmuka pemakai, antarmuka

Program Studi Teknik Informatika	SKPL – SIRS	26/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

3.1.1 Antarmuka pemakai

Pengguna berinteraksi dengan antarmuka yang ditampilkan dalam bentuk form-form serta halaman website.

3.1.2 Antarmuka perangkat keras

Antarmuka perangkat keras yang digunakan dalam perangkat lunak SIRS adalah:

1. Perangkat mobile (Smart Phone, Pocket PC, PDA, dll).
2. Personal Computer (PC)
3. Mouse
4. Keyboard
5. Server : Web Server dan Server Basisdata

3.1.3 Antarmuka perangkat lunak

Perangkat lunak yang dibutuhkan untuk mengoperasikan perangkat lunak SIRS adalah sebagai berikut :

1. Nama : MySQL
Sumber : Microsoft
Sebagai basisdata management system (DBMS) yang digunakan untuk menyimpan data di sisi server.
2. Nama : Windows XP / Windows 7
Sumber : Microsoft.
Sebagai sistem operasi untuk desktop-pengguna dimana aplikasi akan di jalankan.
3. Nama : IIS
Sumber : Microsoft.
Sebagai web server.
4. Nama : Framework CI

Sumber : CodeIgniter.

Sebagai framework untuk membangun aplikasi.

5. Nama : Windows XP, Vista, Seven.

Sumber : Microsoft

Sebagai sistem operasi untuk desktop-pengguna.

6. Nama : Microsoft Internet Explorer, Netscape Navigator, Opera, Safari, Mozilla Firefox, Google Chrome, dll.

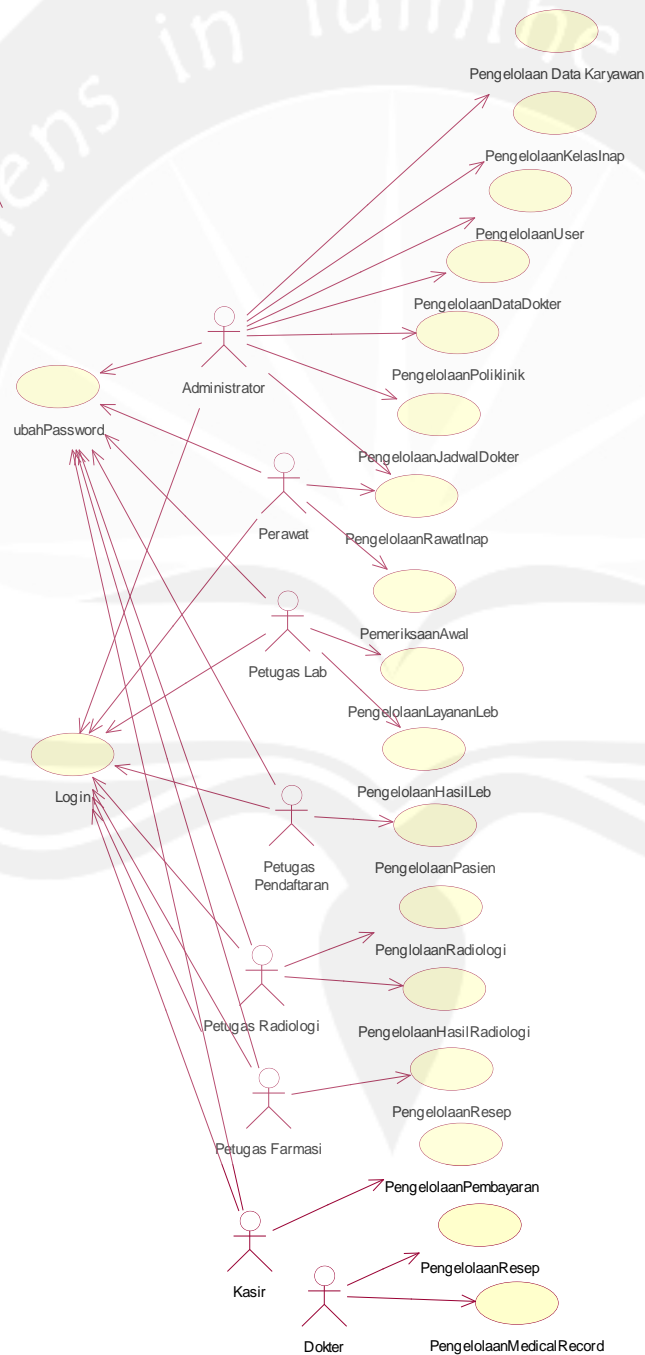
Sebagai penjelajah situs (web browser) berbasis grafis atau teks untuk web-server.

3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak SIRS menggunakan protocol HTTP.

3.2 Kebutuhan fungsionalitas Perangkat Lunak

3.2.1 Use Case Diagram



Gambar 2. Use Case Diagram

4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

4.1.1 Use case Spesification : Login

1. Brief Description

Use Case ini digunakan oleh aktor untuk memperoleh akses ke sistem. Login didasarkan pada sebuah id unik yaitu username dan password yang berupa rangkaian karakter.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor akan masuk kedalam sistem dan harus login terlebih dahulu
2. Aktor memasukkan username dan password
3. Sistem memeriksa username dan password yang diinputkan aktor
 - E-1 Username atau password yang diinputkan oleh aktor salah.
4. Sistem memberikan akses ke aktor
5. Use Case ini selesai dilakukan.

5. Alternative Flow

none

6. Error Flow

- E-1 Password atau nama pengguna tidak sesuai
- 6. Sistem menampilkan peringatan bahwa username atau password tidak sesuai.
- 7. Kembali ke Basic Flow langkah ke 2.

7. PreConditions

None

Program Studi Teknik Informatika	SKPL – SIRS	30/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. PostConditions

1. Aktor memasuki sistem dan dapat menggunakan fungsi- fungsi pada sistem.

4.1.2 Use case Spesification : Edit Password

1. Brief Description

Use Case ini digunakan oleh administrator, dokter, role, kasir, apoteker untuk mengedit password.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan edit password.
2. Sistem menampilkan antarmuka untuk edit password.
3. Aktor memasukkan password lama dan password baru.
4. Sistem mengecek password lama yang di masukkan dalam basisdata.
E-1 Password lama yang diinputkan salah
5. Sistem mengganti password lama dengan password baru
6. Use Case ini selesai dilakukan.

5. Alternative Flow

None

6. Error Flow

E-1 Password lama yang dimasukkan salah

Program Studi Teknik Informatika	SKPL – SIRS	31/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

7. Sistem memberikan pesan peringatan bahwa password lama yang diinputkan salah.

8. Kembali ke Basic Flow langkah ke 3.

7. PreConditions

1. Use Case Login telah dilakukan.

2. Aktor telah memasuki sistem.

8. PostConditions

9. Password dari actor sudah berubah.

4.1.3 Use case Spesification : Pengelolaan Data Pengguna

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data pengguna. Aktor dapat melakukan entry data pengguna, edit data pengguna, delete data pengguna.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pengguna.

2. Sistem memberikan pilihan untuk melakukan entry data pengguna, edit data pengguna, delete data pengguna.

3. Aktor memilih melakukan entry data pengguna.

A-1 Aktor memilih untuk melakukan edit data pengguna

A-2 Aktor memilih untuk melakukan delete data pengguna

Program Studi Teknik Informatika	SKPL – SIRS	32/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4. Sistem menampilkan form untuk menginputan data pengguna baru.
5. Aktor menginputkan data pengguna baru.
 - E-1 Form nama Pengguna kosong.
 - E-2 Form username kosong.
6. Aktor meminta sistem untuk menyimpan data pengguna yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan edit data pengguna.
 1. Aktor memilih data yang akan di edit.
 2. Sistem menampilkan form untuk mengedit data pengguna.
 3. Aktor mengedit data pengguna yang sudah di tampilkan.
 - E-3 Form Nama Pengguna kosong.
 - E-4 Form Username kosong.
 4. Aktor meminta sistem untuk menyimpan data pengguna yang telah di edit.
 5. Sistem menyimpan data pengguna yang baru yang telah di edit.
 6. Berlanjut ke Basic Flow langkah 7
- A-2 Aktor memilih untuk melakukan delete data pengguna
 1. Sistem menampilkan data-data data pengguna.
 2. Aktor memilih data pengguna yang akan di hapus.
 3. Sistem menghapus data pengguna yang sudah dipilih.

4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form Nama Pengguna kosong.

1. Sistem memberikan peringatan bahwa form Nama Pengguna harus diisi.
2. Berlanjut ke basic flow ke 5.

E-2 Form Username kosong.

1. Sistem memberikan peringatan bahwa form Username harus diisi.
2. Berlanjut ke basic flow ke 5.

E-3 Form Nama Pengguna kosong.

1. Sistem memberikan peringatan bahwa form Nama Pengguna harus diisi.
2. Berlanjut ke A-1 no 3.

E-4 Form Username kosong.

1. Sistem memberikan peringatan bahwa form Username harus diisi.
2. Berlanjut ke A-1 no 3.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data Pengguna di basisdata telah terupdate.

4.1.4 Use case Spesification : Pengelolaan Data Poliklinik

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data poliklinik. Aktor dapat melakukan entry data poliklinik, edit data poliklinik, delete data poliklinik.

Program Studi Teknik Informatika	SKPL – SIRS	34/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data poliklinik.
2. Sistem memberikan pilihan untuk melakukan entry data poliklinik, edit data poliklinik, delete data poliklinik.
3. Aktor memilih melakukan entry data poliklinik.
 - A-1 Aktor memilih untuk melakukan edit data poliklinik
 - A-2 Aktor memilih untuk melakukan delete data poliklinik
4. Sistem menampilkan form untuk menginputan data poliklinik baru.
5. Aktor menginputkan data poliklinik baru.
 - E-1 Form id poliklinik kosong.
 - E-2 Form nama poliklinik kosong.
6. Aktor meminta sistem untuk menyimpan data poliklinik yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan edit data poliklinik.
1. Aktor memilih data yang akan di edit.
 2. Sistem menampilkan form untuk mengedit data poliklinik.
 3. Aktor mengedit data poliklinik yang sudah di tampilkan.

Program Studi Teknik Informatika	SKPL – SIRS	35/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-3 Form Id Poliklinik kosong.

E-4 Form Nama Poliklinik kosong.

4. Aktor meminta sistem untuk menyimpan data poliklinik yang telah di edit.

5. Sistem menyimpan data poliklinik yang baru yang telah di edit.

6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data poliklinik

5. Sistem menampilkan data-data data poliklinik.

6. Aktor memilih data poliklinik yang akan di hapus.

7. Sistem menghapus data poliklinik yang sudah dipilih.

8. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form Id Poliklinik kosong.

1. Sistem memberikan peringatan bahwa Id Poliklinik harus diisi.

2. Berlanjut ke basic flow ke 5.

E-2 Form Nama Poliklinik kosong.

1. Sistem memberikan peringatan bahwa form Nama poliklinik harus diisi.

2. Berlanjut ke basic flow ke 5.

E-3 Form Id Poliklinik kosong.

1. Sistem memberikan peringatan bahwa form Id Poliklinik harus diisi.

2. Berlanjut ke A-1 no 3.

E-4 Form Nama Poliklinik kosong.

1. Sistem memberikan peringatan bahwa form Nama poliklinik harus diisi.

2. Berlanjut ke A-1 no 3.

7. PreConditions

1. Use Case Login telah dilakukan.

2. Aktor telah memasuki sistem.

8. PostConditions

Data Poliklinik di basisdata telah terupdate.

4.1.5 Use case Spesification : Pengelolaan Data Pegawai

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data pegawai. Aktor dapat melakukan entry data pegawai baru, edit data pegawai, delete data pegawai.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pegawai.

2. Sistem memberikan pilihan untuk melakukan entry data pegawai baru, edit data pegawai, delete data pegawai.

3. Aktor memilih melakukan entry data pegawai.

A-1 Aktor memilih untuk melakukan edit data pegawai.

A-2 Aktor memilih untuk melakukan delete data pegawai.

4. Sistem menampilkan form untuk menginputan jadwal pegawai baru.

Program Studi Teknik Informatika	SKPL – SIRS	37/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

5. Aktor menginputkan data pegawai baru.
- E-1 Form nama pegawai harus diisi
 - E-2 Form Alamat pegawai harus diisi
 - E-3 Form no telp pegawai harus diisi.
6. Aktor meminta sistem untuk menyimpan data pegawai yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit data pegawai.

1. Aktor memilih data yang akan di edit.
2. Sistem menampilkan form untuk mengedit data pegawai baru.
3. Aktor mengedit data pegawai yang sudah ditampilkan.

E-4 Form nama pegawai harus diisi

E-5 Form Alamat pegawai harus diisi

E-6 Form no telp pegawai harus diisi.

4. Aktor meminta sistem untuk menyimpan data pegawai yang telah di edit.
5. Sistem menyimpan data pegawai yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data pegawai.

1. Sistem menampilkan data-data data pegawai.
2. Aktor memilih data pegawai yang akan di hapus.
3. Sistem menghapus data pegawai yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form nama pegawai harus diisi

1. Sistem memberikan peringatan bahwa form nama pegawai harus diisi.
2. Berlanjut ke basic flow ke 5.

E-2 Form Alamat pegawai harus diisi

1. Sistem memberikan peringatan bahwa form Alamat harus diisi.
2. Berlanjut ke basic flow ke 5.

E-3 Form no telp pegawai harus diisi.

1. Sistem memberikan peringatan bahwa form no telp harus diisi.
2. Berlanjut ke basic flow ke 5.

E-4 Form nama pegawai harus diisi

1. Sistem memberikan peringatan bahwa form nama pegawai harus diisi.
2. Berlanjut ke A-1 no 3.

E-5 Form Alamat pegawai harus diisi

1. Sistem memberikan peringatan bahwa form Alamat harus diisi.
2. Berlanjut ke A-1 no 3.

E-6 Form no telp doker harus diisi.

1. Sistem memberikan peringatan bahwa form no telp harus diisi.
2. Berlanjut ke A-1 no 3.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki siste.

8. PostConditions

1. Data Pegawai di basisdata telah terupdate.

4.1.6 Use case Spesification : Pengelolaan Jadwal Dokter

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola jadwal dokter. Aktor dapat melakukan entry jadwal dokter, edit jadwal, delete jadwal dokter.

2. Primary Actor

Administrator

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan jadwal dokter.
2. Sistem memberikan pilihan untuk melakukan entry jadwal dokter baru, edit jadwal dokter, delete jadwal dokter.
3. Aktor memilih melakukan entry jadwal dokter.
 - A-1 Aktor memilih untuk melakukan edit jadwal dokter.
 - A-2 Aktor memilih untuk melakukan delete jadwal dokter.
4. Sistem menampilkan form untuk menginputan jadwal dokter baru.
5. Aktor menginputkan jadwal dokter baru.
 - E-1 Form waktu kosong.
6. Aktor meminta sistem untuk menyimpan jadwal dokter yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit jadwal dokter.

1. Aktor memilih data yang akan di edit.
2. Sistem menampilkan form untuk mengedit jadwal dokter baru.
3. Aktor mengedit jadwal dokter yang sudah di tampilkan.

E-2 Form waktu kosong.

1. Aktor meminta sistem untuk menyimpan jadwal dokter yang telah di edit.
2. Sistem menyimpan jadwal dokter yang baru yang telah di edit.
3. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete jadwal dokter.

1. Sistem menampilkan data-data jadwal dokter.
2. Aktor memilih jadwal dokter yang akan di hapus.
3. Sistem menghapus jadwal dokter yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form waktu kosong.

1. Sistem memberikan peringatan bahwa form waktu harus diisi.
2. Berlanjut ke basic flow ke 5.

E-2 Form waktu kosong.

1. Sistem memberikan peringatan bahwa form waktu harus diisi.
2. Berlanjut ke A-1 no 3.

7. PreConditions

4. Use Case Login telah dilakukan.
5. Aktor telah memasuki sistem.

8. PostConditions

1. Jadwal Dokter di basisdata telah terupdate.

4.1.7 Use case Spesification : Pengelolaan Data Obat

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data obat. Aktor dapat melakukan entry data obat, edit data obat, delete data obat.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data obat.
2. Sistem memberikan pilihan untuk melakukan entry data obat, edit data obat, delete data obat.
3. Aktor memilih melakukan entry data obat.
 - A-1 Aktor memilih untuk melakukan edit data obat
 - A-2 Aktor memilih untuk melakukan delete data obat
4. Sistem menampilkan form untuk menginputan data obat baru.
5. Aktor menginputkan data obat baru.
6. Aktor meminta sistem untuk menyimpan data obat yang baru di inputkan

Program Studi Teknik Informatika	SKPL – SIRS	42/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit data obat.

1. Aktor memilih data yang akan di edit.
2. Sistem menampilkan form untuk mengedit data obat.
3. Aktor mengedit data obat yang sudah di tampilkan.
4. Aktor meminta sistem untuk menyimpan data obat yang telah di edit.
5. Sistem menyimpan data obat yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data obat

1. Sistem menampilkan data-data data obat.
2. Aktor memilih data obat yang akan di hapus.
3. Sistem menghapus data obat yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

None

7. PreConditions

7. Use Case Login telah dilakukan.
8. Aktor telah memasuki sistem.

8. PostConditions

1. Data Obat di basisdata telah terupdate.

4.1.8 Use case Spesification : Pengelolaan Bidang Medis

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola bidang medis .Meliputi Id_bidang_medis, nama bidang medis, dan biaya.

Program Studi Teknik Informatika	SKPL – SIRS	43/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

Administrator

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor Kelola Bidang Medis.
2. Sistem menampilkan data bidang medis yang sudah ada.
3. Aktor memilih tambah data bidang medis.
 - A-1 Aktor memilih ubah data bidang medis.
 - A-2 Aktor memilih delete data bidang medis.
4. Sistem menampilkan form berisi nama tindakan, harga dan keterangan.
5. Aktor menginputkan data bidang medis.
 - E-1 Form nama tindakan harus diisi.
 - E-2 Form harga harus diisi
6. Sistem menyimpan dan menampilkan semua data tindakan medis termasuk data yang baru ditambah.
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih ubah data bidang medis.
 1. Sistem menampilkan form untuk mengubah data.
 2. Aktor mengedit data yang dipilih.
 - E-3 Form nama tindakan harus diisi.
 - E-4 Form harga harus diisi
 3. Sistem menyimpan data yang sudah diedit dan menampilkan semua data bidang medis.
 4. Berlanjut ke Basic Flow ke 7

Program Studi Teknik Informatika	SKPL – SIRS	44/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

A-2 Aktor memilih delete data bidang medis.

1. Aktor memilih data yang akan dihapus.
2. Sistem menghapus data yang dipilih.
3. Berlanjut ke Basic Flow ke 7

6. Error Flow

E-1 Form nama tindakan harus diisi.

1. Sistem memberikan peringatan bahwa form Nama tindakan harus diisi.
2. Berlanjut ke basic flow no 3.

E-2 Form harga harus diisi

1. Sistem memberikan peringatan bahwa form Nama harga harus diisi.
2. Berlanjut ke basic flow no 3.

E-3 Form nama tindakan harus diisi.

1. Sistem memberikan peringatan bahwa form Nama tindakan harus diisi.
2. Berlanjut ke A-1 no 2.

E-4 Form harga harus diisi

1. Sistem memberikan peringatan bahwa form harga harus diisi.
2. Berlanjut ke A-1 no 2.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data bidang medis telah terupdate.

4.1.9 Use case Spesification : Pengelolaan Data Pasien

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data pasien yang meliputi entry data pasien, ubah data pasien, delete data pasien.

2. Primary Actor

Resepsionis

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data pasien.
2. Sistem memberikan pilihan untuk melakukan entry data pasien, edit data pasien, delete data pasien.
3. Aktor memilih melakukan entry data pasien.
 - A-1 Aktor memilih untuk melakukan edit data pasien
 - A-2 Aktor memilih untuk melakukan delete data pasien
4. Sistem menampilkan form untuk menginputan data pasien baru.
5. Aktor menginputkan data pasien baru.
 - E-1 Form Id Pasien harus diisi
 - E-2 Form jenis kelamin harus diisi
 - E-3 Form Tanggal lahir harus diisi
 - E-4 Form status nikah harus diisi
 - E-5 Form pekerjaan harus diisi
 - E-6 Form warga negara harus diisi
 - E-7 Form asuransi harus diisi

Program Studi Teknik Informatika	SKPL – SIRS	46/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Aktor meminta sistem untuk menyimpan data pasien yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit data pasien

1. Aktor memilih data yang akan di edit.
2. Sistem menampilkan form untuk mengedit data pasien
3. Aktor mengedit data pasien yang sudah di tampilkan.

E-8 Form Id Pasien harus diisi

E-9 Form jenis kelamin harus diisi

E-10 Form Tanggal lahir harus diisi

E-11 Form status nikah harus diisi

E-12 Form pekerjaan harus diisi

E-13 Form warga negara harus diisi

E-14 Form asuransi harus diisi

4. Aktor meminta sistem untuk menyimpan data pasien yang telah di edit.
5. Sistem menyimpan data pasien yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data pasien

1. Sistem menampilkan data-data data pasien
2. Aktor memilih data pasien yang akan di hapus.
3. Sistem menghapus data pasien yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

Program Studi Teknik Informatika	SKPL – SIRS	47/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Error Flow

E-1 Form Id Pasien harus diisi

1. Sistem akan error karena id pasien primary key.
2. Berlanjut ke basic flow no 5.

E-2 Form jenis kelamin harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke basic flow no 5.

E-3 Form Tanggal lahir harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke basic flow no 5.

E-4 Form status nikah harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke basic flow no 5.

E-5 Form pekerjaan harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke basic flow no 5.

E-6 Form warga negara harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke basic flow no 5.

E-7 Form asuransi harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke basic flow no 5.

E-8 Form Id Pasien harus diisi

1. Sistem akan error karena id pasien primary key.
2. Berlanjut ke A-1 no 3.

E-9 Form jenis kelamin harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke A-1 no 3.

E-10 Form Tanggal lahir harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke A-1 no 3.

E-11 Form status nikah harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke A-1 no 3.

E-12 Form pekerjaan harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke A-1 no 3.

E-13 Form warga negara harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke A-1 no 3.

E-14 Form asuransi harus diisi

1. Sistem akan error karena inputan pada combobox tidak boleh kosong.
2. Berlanjut ke A-1 no 3.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

Program Studi Teknik Informatika	SKPL – SIRS	49/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

8. PostConditions

1. Data pasien di basisdata telah terupdate.

4.1.10 Use case Spesification : Pengelolaan Data

Resep

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data resep. Aktor dapat melakukan entry data resep, edit data resep, delete data resep.

2. Primary Actor

Dokter

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data resep.
2. Sistem memberikan pilihan untuk melakukan entry data resep, edit data resep, delete data resep.
3. Aktor memilih melakukan entry data resep.
 - A-1 Aktor memilih untuk melakukan edit data resep
 - A-2 Aktor memilih untuk melakukan delete data resep
4. Sistem menampilkan form untuk menginputan data resep baru.
5. Aktor menginputkan data resep baru.
6. Aktor meminta sistem untuk menyimpan data resep yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit data resep.

1. Aktor memilih data yang akan di edit.
2. Sistem menampilkan form untuk mengedit data resep.
3. Aktor mengedit data resep yang sudah di tampilkan.
4. Aktor meminta sistem untuk menyimpan data resep yang telah di edit.
5. Sistem menyimpan data resep yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data resep

5. Sistem menampilkan data-data data resep.
6. Aktor memilih data resep yang akan di hapus.
7. Sistem menghapus data resep yang sudah dipilih.
8. Berlanjut ke Basic Flow langkah 7

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Data Resep di basisdata telah terupdate.

4.1.11 Use case Spesification : Pengelolaan data kamar

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data kamar. Aktor dapat melakukan entry data kamar, edit data kamar, delete data kamar.

Program Studi Teknik Informatika	SKPL – SIRS	51/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan kamar.
2. Sistem memberikan pilihan untuk melakukan entry data kamar, edit data kamar, delete data kamar.
3. Aktor memilih melakukan entry data kamar.
 - A-1 Aktor memilih untuk melakukan edit data kamar
 - A-2 Aktor memilih untuk melakukan delete data kamar
4. Sistem menampilkan form untuk menginputan data kamar baru.
5. Aktor menginputkan data kamar baru.
 - E-1 Form Nama Kamar harus diisi
 - E-2 Form Biaya Kamar harus diisi
6. Aktor meminta sistem untuk menyimpan data kamar yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan edit data kamar.
1. Aktor memilih data yang akan di edit.
 2. Sistem menampilkan form untuk mengedit data kamar.
 3. Aktor mengedit data Kelas Inap yang sudah di tampilkan.

Program Studi Teknik Informatika	SKPL – SIRS	52/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-3 Form Nama Kamar harus diisi

E-4 Form Biaya Kamar harus diisi

4. Aktor meminta sistem untuk menyimpan data kamar yang telah di edit.
5. Sistem menyimpan data kamar yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data kamar

1. Sistem menampilkan data-data data kamar.
2. Aktor memilih data kamar yang akan di hapus.
3. Sistem menghapus data kamar yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form Nama kamar harus diisi

1. Sistem memberikan peringatan bahwa form Nama paket harus diisi.
2. Berlanjut ke basic flow no 5.

E-2 Form Biaya paket harus diisi

1. Sistem memberikan peringatan bahwa form Biaya paket harus diisi.
2. Berlanjut ke basic flow no 5.

E-3 Form Nama kamar harus diisi

1. Sistem memberikan peringatan bahwa form Nama kamar harus diisi.
2. Berlanjut ke A-1 no 3.

E-4 Form Biaya kamar harus diisi

1. Sistem memberikan peringatan bahwa form Biaya kamar harus diisi.

2. Berlanjut ke A-1 no 3.

7. PreConditions

1. Use Case Login telah dilakukan.

2. Aktor telah memasuki sistem.

8. PostConditions

1. Data kamar di basisdata telah terupdate.

4.1.12 Use case Spesification : Pengelolaan Kelas Inap

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data Kelas Inap. Aktor dapat melakukan entry data Kelas Inap, edit data Kelas Inap, delete data Kelas Inap.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan Kelas Inap.

2. Sistem memberikan pilihan untuk melakukan entry data Kelas Inap, edit data Kelas Inap, delete data Kelas Inap.

3. Aktor memilih melakukan entry data Kelas Inap.

A-1 Aktor memilih untuk melakukan edit data Kelas Inap

A-2 Aktor memilih untuk melakukan delete data Kelas Inap

4. Sistem menampilkan form untuk menginputan data Kelas Inap baru.
5. Aktor menginputkan data Kelas Inap baru.
 - E-1 Form Nama Kelas harus diisi
 - E-2 Form Biaya Kelas harus diisi
6. Aktor meminta sistem untuk menyimpan data Kelas Inap yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit data Kelas Inap.

1. Aktor memilih data yang akan di edit.
2. Sistem menampilkan form untuk mengedit data Kelas Inap.
3. Aktor mengedit data Kelas Inap yang sudah di tampilkan.

E-3 Form Nama Perawatan harus diisi

E-4 Form Biaya Perawatan harus diisi

4. Aktor meminta sistem untuk menyimpan data Kelas Inap yang telah di edit.
5. Sistem menyimpan data Kelas Inap yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data Kelas Inap

1. Sistem menampilkan data-data data Kelas Inap.
2. Aktor memilih data Kelas Inap yang akan di hapus.

3. Sistem menghapus data Kelas Inap yang sudah dipilih.

4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form Nama Perawatan harus diisi

1. Sistem memberikan peringatan bahwa form Nama Perawatan harus diisi.
2. Berlanjut ke basic flow no 5.

E-2 Form Biaya Perawatan harus diisi

1. Sistem memberikan peringatan bahwa form Biaya Perawatan harus diisi.
2. Berlanjut ke basic flow no 5.

E-3 Form Nama Perawatan harus diisi

1. Sistem memberikan peringatan bahwa form Nama Perawatan harus diisi.
2. Berlanjut ke A-1 no 3.

E-4 Form Biaya Perawatan harus diisi

1. Sistem memberikan peringatan bahwa form Biaya Perawatan harus diisi.
2. Berlanjut ke A-1 no 3.

7. PreConditions

7. Use Case Login telah dilakukan.
8. Aktor telah memasuki sistem.

8. PostConditions

1. Data Kelas Inap di basisdata telah terupdate.

4.1.13 Use case Spesification : Pengelolaan Rawat Inap

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola rawat inap. Aktor dapat melakukan entry rawat Inap, edit Rawat Inap, delete Rawat Inap.

Program Studi Teknik Informatika	SKPL – SIRS	56/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

Administrator

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan rawat inap.
2. Sistem memberikan pilihan untuk melakukan entry rawat inap baru, edit rawat inap, delete rawat inap.
3. Aktor memilih melakukan entry rawat inap.
 - A-1 Aktor memilih untuk melakukan edit rawat inap.
 - A-2 Aktor memilih untuk melakukan delete rawat inap.
4. Sistem menampilkan form untuk menginputan jadwal dokter baru.
5. Aktor menginputkan rawat inap baru.
 - E-1 Form waktu harus diisi
6. Aktor meminta sistem untuk menyimpan rawat inap yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan edit rawat inap.
 2. Aktor memilih data yang akan di edit.
 3. Sistem menampilkan form untuk mengedit rawat inap baru.
 4. Aktor mengedit rawat inap yang sudah di tampilkan.
 - E-2 Form waktu harus diisi

Program Studi Teknik Informatika	SKPL – SIRS	57/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

5. Aktor meminta sistem untuk menyimpan rawat inap yang telah di edit.
 6. Sistem menyimpan rawat inap yang baru yang telah di edit.
 7. Berlanjut ke Basic Flow langkah 7
- A-2 Aktor memilih untuk melakukan delete rawat inap.
1. Sistem menampilkan data-data rawat inap.
 2. Aktor memilih rawat inap yang akan di hapus.
 3. Sistem menghapus rawat inap yang sudah dipilih.
 4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form waktu harus diisi

1. Sistem memberikan peringatan bahwa form waktu harus diisi.
2. Berlanjut ke basic flow ke 5.

E-2 Form waktu harus diisi

1. Sistem memberikan peringatan bahwa form waktu harus diisi.
2. Berlanjut ke A-1 no 3.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.

8. PostConditions

1. Rawat inap di basisdata telah terupdate.

4.1.14 Use case Spesification : Pendaftaran Pemeriksaan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola pendaftaran pemeriksaan. Aktor dapat melakukan pendaftaran pasien berdasarkan rumah sakit dan dokternya, aktor dapat mengubah status antrian pasien.

2. Primary Actor

Resepsionis

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan daftar pemeriksaan.
2. Sistem menampilkan form untuk memilih nama rumah sakit.
3. Aktor memilih rumah sakit.
4. Sistem menampilkan form untuk memilih dokter.
5. Aktor memilih dokter.
6. Sistem menampilkan daftar pasien yang ada sesuai rumah sakit dan dokter yang sudah dipilih.
7. Aktor memasukkan pasien ke daftar antrian.
 - A-1 Mencari data pasien sesuai dengan nama pasien.
8. Sistem menampilkan form antrian.
9. Aktor mengubah status antrian pasien.
10. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Mencari data pasien sesuai dengan nama pasien.

Program Studi Teknik Informatika	SKPL – SIRS	59/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Aktor memasukkan nama yang dicari
 - E-1 Data yang dicari tidak ada.
2. Sistem menampilkan data yang di cari.
3. Berlanjut ke Basic Flow ke 10

6. Error Flow

E-1 Data yang dicari tidak ada.

1. Sistem memberikan peringatan bahwa data yang dicari tidak ada
2. Berlanjut ke A-1 ke 3.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.
3. Aktor telah berhasil mendaftarkan pasien ke antrian.

8. PostConditions

1. Pasien telah masuk antrian.

4.1.15 Use case Spesification : Pendaftaran Pemeriksaan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola pendaftaran rawat inap. Aktor dapat melakukan pendaftaran pasien berdasarkan rumah sakit dan dokternya, aktor dapat mengubah status antrian pasien rawat inap.

2. Primary Actor

Petugas Pendaftaran

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan daftar rawat inap.

Program Studi Teknik Informatika	SKPL – SIRS	60/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Sistem menampilkan form untuk memilih nama rumah sakit.
3. Aktor memilih rumah sakit.
4. Sistem menampilkan form untuk memilih kamar.
5. Aktor memilih kamar.
6. Sistem menampilkan daftar pasien yang ada sesuai rumah sakit dan kamar yang telah dipilih.
7. Aktor memasukkan pasien ke daftar pasien rawat inap.
 - A-1 Mencari data pasien sesuai dengan nama pasien.
8. Sistem menampilkan form pasien yang melakukan rawat inap.
9. Aktor mencetak buti pendaftaran pasien.
10. Aktor mengubah status pasien.
11. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Mencari data pasien sesuai dengan nama pasien.

1. Aktor memasukkan nama yang dicari

E-1 Data yang dicari tidak ada.

2. Sistem menampilkan data yang di cari.
3. Berlanjut ke Basic Flow ke 10

6. Error Flow

E-1 Data yang dicari tidak ada.

1. Sistem memberikan peringatan bahwa data yang dicari tidak ada
2. Berlanjut ke A-1 ke 3.

7. PreConditions

1. Use Case Login telah dilakukan.

Program Studi Teknik Informatika	SKPL – SIRS	61/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Aktor telah memasuki sistem.
3. Aktor telah berhasil mendaftarkan pasien ke pasien rawat inap.

8. PostConditions

1. Pasien telah masuk kedalam pasien rawat inap.

4.1.16 Use case Spesification : Pemeriksaan Awal

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan pemeriksaan awal meliputi entry data pemeriksaan awal dan ubah data pemeriksaan.

2. Primary Actor

Perawat

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pemeriksaan awal
2. Sistem menampilkan data pasien yang melakukan pemeriksaan yang masuk dalam antrian pada hari itu.
3. Aktor memasukkan pasien ke pemeriksaan.
4. Sistem menampilkan form untuk mencatat hasil periksa pasien misalnya tekanan darah, suhu, dll.
5. Aktor menginputkan hasil pemeriksaan pasien antara lain tekanan darah, suhu, dll.
6. Sistem menyimpan dan menampilkan data pemeriksaan pasien.
7. Aktor bisa melakukan ubah data pemeriksaan.
8. Use case telah selesai dilakukan.

Program Studi Teknik Informatika	SKPL – SIRS	62/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

5. Alternative Flow

None

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.
3. Status Pasien Masuk belum diperiksa.

8. PostConditions

1. Status Pasien sudah Periksa yang artinya sudah melakukan pemeriksaan awal.

4.1.17 Use case Spesification : Pemeriksaan Dokter

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan pemeriksaan awal meliputi entry data pemeriksaan awal dan ubah data pemeriksaan.

2. Primary Actor

Dokter

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pemeriksaan dokter.
2. Sistem menampilkan data pasien yang sudah melakukan pemeriksaan awal.
3. Aktor melakukan pemeriksaan dan menginputkan data hasil dari pemeriksaannya yaitu konsultasi, dianogsa , dan di rujuk ke mana pasien tersebut.

4. Aktor melakukan Pilih Resep memilih obat yang di perlukan.

A-1 Aktor search data obat berdasarkan nama obat dan id obat.

5. Sistem menyimpan resep obat yang sudah di pilih.

6. Aktor melakukan Pilih Tindakan memasukkan tindakan yang dilakukan saat pemeriksaan.

A-2 Aktor search data tindakan berdasarkan nama tindakan dan id jenis tindakan.

7. Sistem menyimpan tindakan yang sudah di lakukan.

8. Sistem menyimpan semua data yang sudah di inputkan oleh actor.

9. Use case telah selesai dilakukan.

5. Alternative Flow

A-1 Aktor search data obat berdasarkan nama obat dan id obat.

1. Sistem menampilkan semua data obat.

2. Aktor menginputkan data obat yang dicari berdasarkan dengan nama obat atau id obat.

3. Sistem menampilkan data obat yang dicari.

4. Berlanjut ke Basic Flow 4.

A-2 Aktor search data tindakan berdasarkan nama tindakan dan id jenis tindakan.

1. Sistem menampilkan semua data tindakan.

2. Aktor menginputkan data tindakan yang dicari berdasarkan dengan nama tindakan atau id jenis tindakan.

3. Sistem menampilkan data tindakan yang dicari.

4. Berlanjut ke Basic Flow 6.

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.
3. Pasien belum diperiksa oleh dokter.

8. PostConditions

1. Pasien sudah mendapatkan pemeriksaan dari dokter.

4.1.18 Use case Spesification : Pembayaran Rumah sakit

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan pembayaran rumah sakit dan juga melakukan cetak pembayaran.

2. Primary Actor

Kasir

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pembayaran rumah sakit.
2. Sistem menampilkan data pasien yang sudah melakukan pembayaran rumah sakit.
3. Aktor melakukan pembayaran dan menginputkan uang cash yang di berikan pasien.
4. Sistem menyimpan resep obat yang sudah di pilih.
5. Aktor melakukan cetak pembayaran.

Program Studi Teknik Informatika	SKPL – SIRS	65/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

6. Use case telah selesai dilakukan.

5. Alternative Flow

None

6. Error Flow

None

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki sistem.
3. Pasien belum melakukan pembayaran.

8. PostConditions

1. Pasien sudah selesai melakukan pembayaran.

4.1.19 Use case Spesification : Pengelolaan Layanan Lab

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola layanan lab. Aktor dapat melakukan entry layanan lab, edit layanan lab, delete layanan lab.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan layanan lab.
2. Sistem memberikan pilihan untuk melakukan entry data layanan lab, edit layanan lab, delete layanan lab.
3. Aktor memilih melakukan entry data dokter.
 - A-1 Aktor memilih untuk menambah layanan lab.
 - A-2 Aktor memilih untuk melakukan layanan lab.

Program Studi Teknik Informatika	SKPL – SIRS	66/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4. Sistem menampilkan form untuk menginputkan layanan lab.
5. Aktor menginputkan data dokter baru.
 - E-1 Form nama layanan harus diisi
 - E-2 Form jenis layanan harus diisi
6. Aktor meminta sistem untuk menyimpan data layanan lab di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit layanan lab.

7. Aktor memilih data yang akan di edit.

8. Sistem menampilkan form untuk mengedit layanan lab.

9. Aktor mengedit data dokter yang sudah ditampilkan.

E-4 Form nama layanan lab harus diisi

E-5 Form jenis layanan lab harus diisi

10. Aktor meminta sistem untuk menyimpan data dokter yang telah di edit.

11. Sistem menyimpan layanan lab yang baru yang telah di edit.

12. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data layanan lab.

5. Sistem menampilkan data-data data layanan lab.

6. Aktor memilih layanan lab yang akan di hapus.

7. Sistem menghapus layanan lab yang sudah dipilih.

8. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form nama layanan lab harus diisi

1. Sistem memberikan peringatan bahwa form nama layanan lab harus diisi.
2. Berlanjut ke basic flow ke 5.

E-2 Form jenis layanan lab harus diisi

1. Sistem memberikan peringatan bahwa form jenis layanan lab harus diisi.
2. Berlanjut ke basic flow ke 5.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki siste.

8. PostConditions

1. Data Dokter di basisdata telah terupdate.

4.1.20 Use case Spesification : Pengelolaan Hasil Lab

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola layanan lab. Aktor dapat melakukan entry layanan lab, edit hasil lab, delete hasil lab.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan hasil lab.

Program Studi Teknik Informatika	SKPL – SIRS	68/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Sistem memberikan pilihan untuk melakukan entry data hasil lab, edit hasil lab, delete hasil lab.
3. Aktor memilih melakukan entry data dokter.
 - A-1 Aktor memilih untuk menambah hasil lab.
 - A-2 Aktor memilih untuk melakukan hasil lab.
4. Sistem menampilkan form untuk menginputan hasil lab.
5. Aktor menginputkan data dokter baru.
 - E-1 Form nama layanan harus diisi
 - E-2 Form jenis layanan harus diisi
6. Aktor meminta sistem untuk menyimpan data hasil lab di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan edit hasil lab.
1. Aktor memilih data yang akan di edit.
 2. Sistem menampilkan form untuk mengedit hasil lab.
 3. Aktor mengedit data dokter yang sudah ditampilkan.
 - E-4 Form nama layanan lab harus diisi
 - E-5 Form jenis layanan lab harus diisi
 4. Aktor meminta sistem untuk menyimpan data dokter yang telah di edit.
 5. Sistem menyimpan hasil lab yang baru yang telah di edit.
 6. Berlanjut ke Basic Flow langkah 7
- A-2 Aktor memilih untuk melakukan delete data hasil lab.

9. Sistem menampilkan data-data data hasil lab.
10. Aktor memilih hasil lab yang akan di hapus.
11. Sistem menghapus hasil lab yang sudah dipilih.
12. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form nama layanan lab harus diisi

3. Sistem memberikan peringatan bahwa form nama layanan lab harus diisi.
4. Berlanjut ke basic flow ke 5.

E-2 Form jenis layanan lab harus diisi

3. Sistem memberikan peringatan bahwa form jenis layanan lab harus diisi.
4. Berlanjut ke basic flow ke 5.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki siste.

8. PostConditions

1. Data Dokter di basisdata telah terupdate.

4.1.21 Use case Spesification : Pengelolaan Radiologi.

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola layanan radiologi. Aktor dapat melakukan entry layanan radiologi, edit layanan radiologi, delete layanan radiologi.

Program Studi Teknik Informatika	SKPL – SIRS	70/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan layanan radiologi.
2. Sistem memberikan pilihan untuk melakukan entry data layanan radiologi, edit layanan radiologi, delete layanan radiologi.
3. Aktor memilih melakukan entry data dokter.
 - A-1 Aktor memilih untuk menambah layanan radiologi.
 - A-2 Aktor memilih untuk melakukan layanan radiologi.
4. Sistem menampilkan form untuk menginputkan layanan radiologi.
5. Aktor menginputkan data dokter baru.
 - E-1 Form nama kategori harus diisi
 - E-2 Form tarif harus diisi
6. Aktor meminta sistem untuk menyimpan data layanan radiologi di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan edit layanan radiologi.
 1. Aktor memilih data yang akan di edit.
 2. Sistem menampilkan form untuk mengedit layanan radiologi.
 3. Aktor mengedit data dokter yang sudah ditampilkan.

Program Studi Teknik Informatika	SKPL – SIRS	71/76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

E-4 Form kaegori radiologi harus diisi

E-5 Form tarif harus diisi

4. Aktor meminta sistem untuk menyimpan data dokter yang telah di edit.
5. Sistem menyimpan layanan radiologi yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data layanan radiologi.

1. Sistem menampilkan data-data data layanan radiologi.
2. Aktor memilih layanan radiologi yang akan di hapus.
3. Sistem menghapus layanan radiologi yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form kategori radiologi harus diisi

1. Sistem memberikan peringatan bahwa form nama layanan radiologi harus diisi.
2. Berlanjut ke basic flow ke 5.

E-2 Form jenis layanan radiologi harus diisi

1. Sistem memberikan peringatan bahwa form jenis layanan radiologi harus diisi.
2. Berlanjut ke basic flow ke 5.

7. PreConditions

1. Use Case Login telah dilakukan.
2. Aktor telah memasuki siste.

8. PostConditions

2. Data Dokter di basisdata telah terupdate.

Program Studi Teknik Informatika	SKPL – SIRS	72/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

4.1.22 Use case Spesification : Pengelolaan Hasil radiologi

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola hasil radiologi. Aktor dapat melakukan entry hasil radiologi, edit hasil radiologi, delete hasil radiologi.

2. Primary Actor

Administrator

3. Supporting Actor

None

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan hasil radiologi.
2. Sistem memberikan pilihan untuk melakukan entry data hasil radiologi, edit hasil radiologi, delete hasil radiologi.
3. Aktor memilih melakukan entry data dokter.
 - A-1 Aktor memilih untuk menambah hasil radiologi.
 - A-2 Aktor memilih untuk melakukan hasil radiologi.
4. Sistem menampilkan form untuk menginputan hasil radiologi.
5. Aktor menginputkan data dokter baru.
 - E-1 Form nama layanan harus diisi
 - E-2 Form jenis layanan harus diisi
6. Aktor meminta sistem untuk menyimpan data hasil radiologi di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan edit hasil radiologi.

1. Aktor memilih data yang akan di edit.
2. Sistem menampilkan form untuk mengedit hasil radiologi.
3. Aktor mengedit data dokter yang sudah ditampilkan.

E-4 Form nama hasil radiologi harus diisi

E-5 Form jenis hasil radiologi harus diisi

4. Aktor meminta sistem untuk menyimpan data dokter yang telah di edit.
5. Sistem menyimpan hasil radiologi yang baru yang telah di edit.
6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan delete data hasil radiologi.

1. Sistem menampilkan data-data data hasil radiologi.
2. Aktor memilih hasil radiologi yang akan di hapus.
3. Sistem menghapus hasil radiologi yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form nama hasil radiologi harus diisi

Program Studi Teknik Informatika	SKPL – SIRS	74/ 76
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Sistem memberikan peringatan bahwa form nama hasil radiologi harus diisi.

2. Berlanjut ke basic flow ke 5.

E-2 Form jenis hasil radiologi harus diisi

1. Sistem memberikan peringatan bahwa form jenis layanan radiologi harus diisi.

2. Berlanjut ke basic flow ke 5.

7. PreConditions

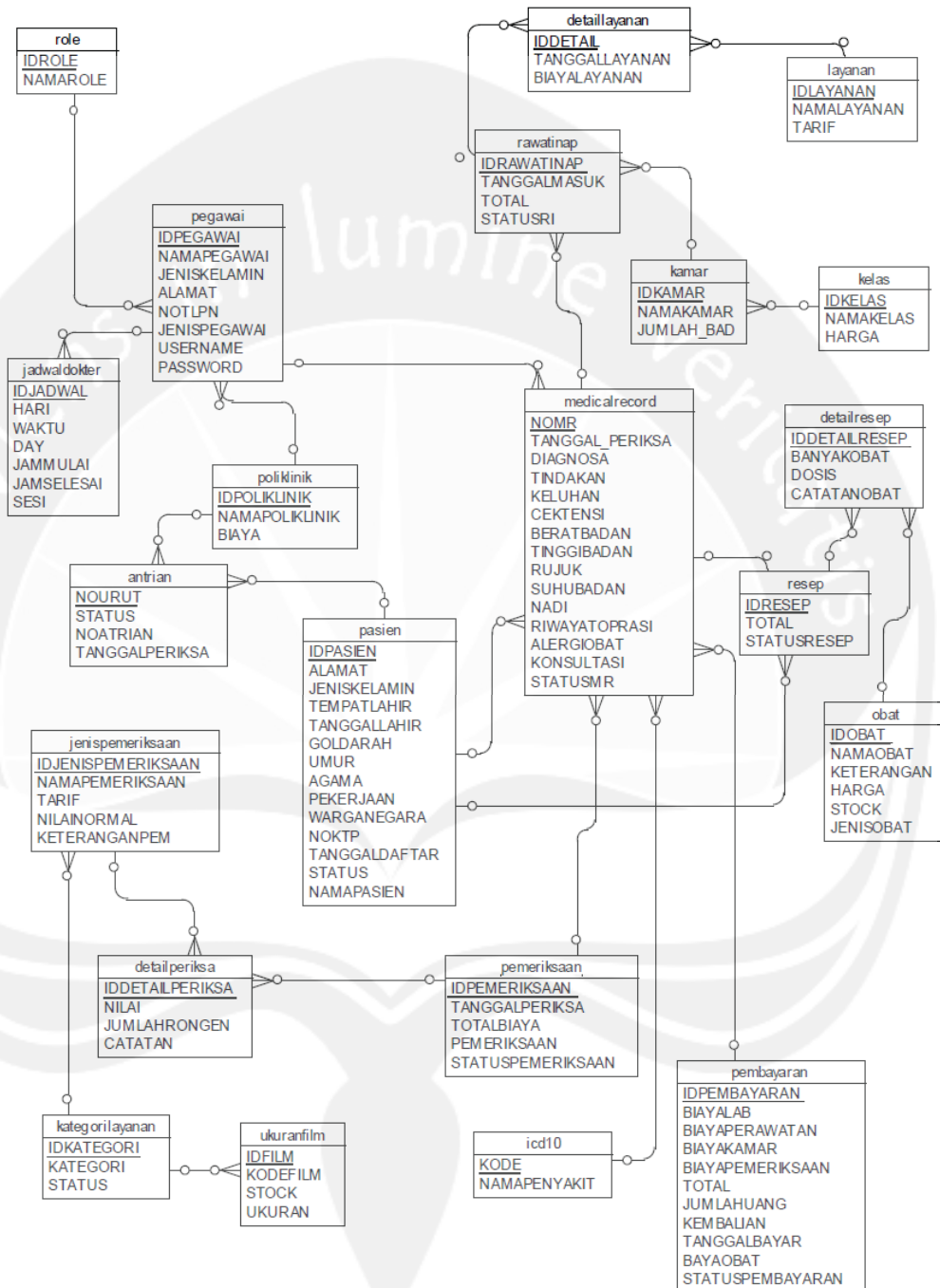
1. Use Case Login telah dilakukan.

2. Aktor telah memasuki siste.

8. PostConditions

1. Data Dokter di basisdata telah terupdate.

1. ERD



Gambar 3. ERD

DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Sistem Informasi Rumah Sakit

(SIRS)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Yulius Aditya Primandaru / 5835

Program Studi Teknik Informatika - Fakultas Teknologi
Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika	Nomor Dokumen		Halama
		DPPL-SIRS		1/161
		Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperik sa oleh								
Disetuj ui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

DAFTAR ISI

DAFTAR ISI	4
Pendahuluan	10
1.1 Tujuan	17
1.2 Ruang Lingkup	17
1.3 Definisi dan Akronim	18
1.4 Referensi	19
Perancangan Sistem	20
1.5 Perancangan Arsitektur	20
Gambar 1.1 Rancangan Arsitektur SIRS	20
1.6 Perancangan Detail	21
1.6.1 Sequence Diagram	21
1.6.1.1 Login	21
1.6.1.2 Change Password	22
1.6.1.3 Kelola Data Pegawai	22
1.6.1.4 Kelola Data Poliklinik	26
1.6.1.5 Kelola Jadwal Dokter	30
1.6.1.6 Kelola Data Obat	33
1.6.1.7 Kelola Bidang Medis	38
1.6.1.8 Kelola Data Pasien	39
1.6.1.9 Kelola Data Resep	43
1.6.1.10 Kelola Data Kamar	46
1.6.1.11 Kelola Kelas Rawat Inap	49
1.6.1.12 Pendaftaran Pasien	53
1.6.1.13 Keperawatan Pasien	55
1.6.1.14 Display Medical Record	56
1.6.1.15 Edit Pemeriksaan Awal	57
1.6.1.16 Pemeriksaan dan Diagnosis Pasien Dokter	58
1.6.1.17 Kelola Pembayaran Rumah Sakit	60
1.6.1.18 Pembayaran Rawat Inap	61
1.6.1.19 Pembelian Obat	62
1.6.1.20 Kelola Layanan Lab	63
1.6.1.21 Display Layanan Lab	65

Program Studi Teknik Informatika	DPPL – SIRS	4/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.6.1.22	Kelola Hasil Lab	66
1.6.1.23	DeleteHasil Lab	69
1.6.1.24	Kelola Radiologi	70
1.6.1.25	Kelola Hasil Radiologi	73
1.6.2	Class Diagram	77
1.6.3	Spesifikasi Deskripsi Kelas Diagram	80
1.6.3.1	Spesifikasi Deskripsi Kelas Login_V	80
1.6.3.2	Spesifikasi Deskripsi Kelas ChangePassword_V	80
1.6.3.3	Spesifikasi Deskripsi Kelas Pegawai_V	80
1.6.3.4	Spesifikasi Deskripsi Kelas poliklinik_V	81
1.6.3.5	Spesifikasi Deskripsi Kelas pengelolaanKaryawan_V	82
1.6.3.6	Spesifikasi Deskripsi Kelas pengelolaanJadwalDokter_V	82
1.6.3.7	Spesifikasi Deskripsi Kelas obat_V	83
1.6.3.8	Spesifikasi Deskripsi Kelas bidangMedis_V	84
1.6.3.9	Spesifikasi Deskripsi Kelas obat_V	84
1.6.3.10	Spesifikasi Deskripsi Kelas pasien_V	85
1.6.3.11	Spesifikasi Deskripsi Kelas kamar_V	86
1.6.3.12	86
1.6.3.13	Spesifikasi Deskripsi Kelas kelasRawatInap_V	86
1.6.3.14	Spesifikasi Deskripsi Kelas pendaftaran_V	87
1.6.3.15	Spesifikasi Deskripsi Kelas perawat_V	88
1.6.3.16	Spesifikasi Deskripsi Kelas Dokter_V	88
1.6.3.17	Spesifikasi Deskripsi Kelas apoteker_V	89
1.6.3.18	Spesifikasi Deskripsi Kelas kasir_V	90
1.6.3.19	Spesifikasi Deskripsi Kelas Lab_V	90
1.6.3.20	Spesifikasi Deskripsi Kelas Lab_V	91
1.6.3.21	Spesifikasi Deskripsi Kelas login_C	91
1.6.3.22	Spesifikasi Deskripsi Kelas UbahPassword_C	92
1.6.3.23	Spesifikasi Deskripsi Kelas pengelolaanPegawai_C	92

1.6.3.24	Spesifikasi Deskripsi Kelas pengelolaanPoliKlinik_C	93
1.6.3.25	Spesifikasi Deskripsi Kelas jadwalDokter_C ..	94
1.6.3.26	Spesifikasi Deskripsi Kelas pengelolaanObat_C	95
1.6.3.27	Spesifikasi Deskripsi Kelas bidangMedis_C ...	96
1.6.3.28	Spesifikasi Deskripsi Kelas kamar_C	96
1.6.3.29	Spesifikasi Deskripsi Kelas KelasRawatInap_C	97
1.6.3.30	Spesifikasi Deskripsi Kelas pasien_C	98
1.6.3.31	Spesifikasi Deskripsi Kelas pendaftaran_C ...	99
1.6.3.32	Spesifikasi Deskripsi Kelas perawat_C	99
1.6.3.33	Spesifikasi Deskripsi Kelas Dokter_C	100
1.6.3.34	Spesifikasi Deskripsi Kelas apoteker_C	101
1.6.3.35	Spesifikasi Deskripsi Kelas kasir_C	102
1.6.3.36	Spesifikasi Deskripsi Kelas LayananLab_C ...	103
1.6.3.37	Spesifikasi Deskripsi Kelas HasilLab_C	104
1.6.3.38	Spesifikasi Deskripsi Kelas Radiologi_C	104
1.6.3.39	Spesifikasi Deskripsi Kelas HasilRadiologi_C	105
1.6.3.40	Spesifikasi Deskripsi Kelas Login_M	106
1.6.3.41	Spesifikasi Deskripsi Kelas UbahPassword_M.	106
1.6.3.42	Spesifikasi Deskripsi Kelas Pegawai_M	107
1.6.3.43	Spesifikasi Deskripsi Kelas Poliklinik_M ...	108
1.6.3.44	Spesifikasi Deskripsi Kelas Karyawan_M	109
1.6.3.45	Spesifikasi Deskripsi Kelas JadwalDokter_M.	109
1.6.3.46	Spesifikasi Deskripsi Kelas Obat_M	110
1.6.3.47	Spesifikasi Deskripsi Kelas BidangMedis_M ..	111
1.6.3.48	Spesifikasi Deskripsi Kelas Pasien_M	112
1.6.3.49	Spesifikasi Deskripsi Kelas Kamar_M	113
1.6.3.50	Spesifikasi Deskripsi Kelas KelasRawatInap_M	
	114	
1.6.3.51	Spesifikasi Deskripsi Kelas MedicalRecord_M	114
1.6.3.52	Spesifikasi Deskripsi Kelas Resep_M	116

1.6.3.53	Spesifikasi Deskripsi Kelas Kasir_M.....	117
1.6.3.54	Spesifikasi Deskripsi Kelas LayananLab_M...	118
1.6.3.55	Spesifikasi Deskripsi Kelas HasilLab_M.....	119
1.6.3.56	Spesifikasi Deskripsi Kelas Radiologi_M....	120
1.6.3.57	Spesifikasi Deskripsi Kelas HasilRadiologi_M	121
Perancangan Data		122
1.7	Dekomposisi Data	122
1.7.1	Deskripsi Entitas Data ICD10	122
1.7.2	Deskripsi Entitas Data Detil Layanan	122
1.7.3	Deskripsi Entitas Data rawatinap	122
1.7.4	Deskripsi Entitas Data Kamar	123
1.7.5	Deskripsi Entitas Data kelas	123
1.7.6	Deskripsi Entitas Data Pegawai	124
1.7.7	Deskripsi Entitas Data Jadwal Dokter	124
1.7.8	Deskripsi Entitas Data Detail layanan	125
1.7.9	Deskripsi Entitas Dataukuran film	126
1.7.10	Deskripsi Entitas Data Medical Record	126
1.7.11	Deskripsi Entitas Data Antrian	128
1.7.12	Deskripsi Entitas Data Obat	128
	Deskripsi Entitas Data Pasien	129
1.7.13	Deskripsi Entitas Data Pembayaran	130
1.7.14	Deskripsi Entitas Data Pemeriksaan	131
1.7.15	Deskripsi Entitas Data Pegawai	132
1.7.16	Deskripsi Entitas Data Poliklinik	133
1.7.17	Deskripsi Entitas Data Resep	133
1.7.18	Deskripsi Entitas DataJenis Pemeriksaan	133
1.7.19	Deskripsi Entitas DataKategori layanan	134
1.8	Deskripsi Perancangan UI	134
1.8.1	Antarmuka Login	134
1.8.2	Antarmuka Manu Utama	136
1.8.3	Antaramuka Edit Password	137
1.8.4	Antarmuka Pengelolaan Data Pegawai	138

1.8.5	Antarmuka Tambah Data Pengelolaan Data Pegawai	. 139
1.8.6	Antarmuka Edit Data Pengelolaan Data Pegawai	... 140
1.8.7	Antarmuka Delete Data Pengelolaan Data Pegawai	. 141
1.8.8	Antarmuka Pengelolaan Poliklinik 142
1.8.9	Antarmuka Tambah Data Pengelolaan Poliklinik	... 143
1.8.10	Antarmuka Edit Data Pengelolaan Poliklinik 144
1.8.11	Antarmuka Delete Data Pengelolaan Poliklinik	... 146
1.8.12	Antarmuka Pengelolaan Jadwal Dokter 147
1.8.13	Antarmuka Tambah Data Pengelolaan Jadwal Dokter	148
1.8.14	Antarmuka Edit Data Pengelolaan Jadwal Dokter	.. 149
1.8.15	Antarmuka Delete Pengelolaan Jadwal Dokter 150
1.8.16	Antarmuka Pengelolaan Data Obat 151
1.8.17	Antarmuka Tambah Data Pengelolaan Data Obat	... 152
1.8.18	Antarmuka Edit Data Pengelolaan Data Obat 153
1.8.19	Antarmuka Delete Data Pengelolaan Data Obat	... 154
1.8.20	Antarmuka Search Data Obat 155
1.8.21	Antarmuka Pengelolaan Kamar 156
1.8.22	Antarmuka Tambah Data Kamar 157
1.8.23	Antarmuka Edit Data Kamar 158
1.8.24	Antarmuka Delete Data Kamar 159
1.8.25	Antarmuka Tambah Data Pasien 160
1.8.26	Antarmuka Pendaftaran Pemeriksaan 161
1.8.27	Antarmuka Pendaftaran Rawat Inap 162
1.8.28	Antarmuka Tambah Layanan 163
1.8.29	Antarmuka Cetak Bukti Pendaftaran 164
1.8.30	Antarmuka Pemeriksaan Awal 164
1.8.31	Antarmuka Ubah Pemeriksaan Awal 166
1.8.32	Antarmuka Pemeriksaan Dokter 167
1.8.33	Antarmuka Pengelolaan Resep 168
1.8.33.1	Antarmuka Pembayaran Rumah Sakit 169



DAFTAR GAMBAR

Gambar 1.1 Rancangan Arsitektur SIRS.....	20
Gambar 1.2 Sequence Diagram untuk Login.....	21
Gambar 1.3 Sequence Diagram untuk Change Password... Entry Data Pegawai.....	22
Gambar 1.4 Sequence Diagram untuk Entry Data Pegawai Edit Data Pegawai.....	22
Display Data Pegawai.....	24
Gambar 1.6 Sequence Diagram untuk Display Data Pegawai	24
Delete Data Pegawai.....	25
Gambar 1.7 Sequence Diagram untuk Delete Data Pegawai Entry Data Poliklinik.....	25
Gambar 1.8 Sequence Diagram untuk Entry Data Poliklinik	26
Edit Data Poliklinik.....	27
Gambar 1.9 Sequence Diagram untuk Edit Data Poliklinik Display Data Poliklinik.....	27
Gambar 1.10 Sequence Diagram untuk Display Data Poliklinik.....	28
Delete Data Poliklinik.....	29
Gambar 1.11 Sequence Diagram untuk Delete Data Poliklinik.....	29
Entry Jadwal Dokter.....	30
Gambar 1.12 Sequence Diagram untuk Entry Jadwal Dokter	30
Edit Jadwal Dokter.....	31
Gambar 1.13 Sequence Diagram untuk Edit Jadwal Dokter Display Jadwal Dokter.....	31
	32

Gambar 1.14 Sequence Diagram untuk Display Jadwal Dokter.....	32
Delete Jadwal Dokter.....	33
Gambar 1.15 Sequence Diagram untuk Delete Jadwal Dokter	33
Entry Data Obat.....	33
Gambar 1.16 Sequence Diagram untuk Entry Data Obat..	34
Edit Data Obat.....	35
Gambar 1.17 Sequence Diagram untuk Edit DataObat....	35
Display Data Obat.....	36
Gambar 1.18 Sequence Diagram untuk Display Data Obat	36
Delete Data Obat.....	37
Gambar 1.20 Sequence Diagram untuk Delete Data Obat.	37
Entry Bidang Medis.....	38
Gambar 1.21 Sequence Diagram untuk Entry Bidang Medis	38
Entry Data Pasien.....	39
Gambar 1.22 Sequence Diagram untuk Entry Data Pasien	39
Edit Data Pasien.....	40
Gambar 1.23 Sequence Diagram untuk Edit Data Pasien.	40
Display Data Pasien.....	41
Gambar 1.24 Sequence Diagram untuk Display Data Pasien	41
Delete Data Pasien.....	42
Gambar 1.25 Sequence Diagram untuk Delete Data Pasien	42
Entry Data Resep.....	43
Gambar 1.26 Sequence Diagram untuk Entry Data Resep.	43
Edit Data Resep.....	44
Gambar 1.27 Sequence Diagram untuk Edit Data Resep..	44
Display Data Resep.....	45
Gambar 1.28 Sequence Diagram untuk Display Data Resep	45

Entry Data Kamar	46
Gambar 1.29 Sequence Diagram untuk Entry Data Kamar ..	46
Edit Data Kamar	47
Gambar 1.30 Sequence Diagram untuk Edit Data Kamar ..	47
Display Data Kamar	48
Gambar 2.33 Sequence Diagram untuk Display Data Kamar	48
Delete Data Kamar	48
Gambar 1.31 Sequence Diagram untuk Delete Data Kamar	48
Entry Kelas Rawat Inap	49
Gambar 1.32 Sequence Diagram untuk Entry Data Kelas Rawat Inap	49
Edit Kelas Rawat Inap	50
Gambar 1.33 Sequence Diagram untuk Edit Data Kelas Rawat Inap	50
Display Kelas Rawat Inap	51
Gambar 1.34 Sequence Diagram untuk Dispay Data Kelas Rawat Inap	51
Delete Kelas Rawat Inap	52
Gambar 1.35 Sequence Diagram untuk Delete Data Kelas Rawat Inap	52
Pendaftaran PasienPeriksa	53
Gambar 1.37 Sequence Diagram untuk Pendaftaran Pasien Periksa	53
Pendaftaran Pasien Rawat Inap	54
Gambar 1.38 Sequence Diagram untuk Pendaftaran Pasien Rawat Inap	54
Entry Pemeriksaan Awal	55
Gambar 1.39 Sequence Diagram untuk Entry Pemeriksaan Awal	55

Gambar 1.40 Sequence Diagram untuk Display Medical Record.....	56
Gambar 1.41 Sequence Diagram untuk Edit Pemeriksaan Awal.....	57
Dokter: Pemeriksaan Awal Ubah Nilai Medical Record..	58
Gambar 1.42 Sequence Diagram untuk Pemeriksaan Awal Ubah Nilai Medical Record.....	58
Dokter: Diagnosis Penyakit.....	59
Gambar 1.43 Sequence Diagram untuk Diagnosis Penyakit Pasien.....	59
Pembayaran Pemeriksaan.....	60
Gambar 1.44 Sequence Diagram untuk Pembayaran Pemeriksaan.....	60
Gambar 1.45 Sequence Diagram untuk Pembayaran Rawat Inap	61
Gambar 1.46 Sequence Diagram untuk Pembelian Obat... Entry Layanan Lab.....	62 63
Gambar 1.47 Sequence Diagram untuk Entry Layanan Lab Edit Layanan Lab.....	63 64
Gambar 1.48 Sequence Diagram untuk Edit Layanan Lab.	64
Gambar 1.49 Sequence Diagram untuk Display Layanan Lab	65
Entry Hasil Lab.....	66
Gambar 1.50 Sequence Diagram untuk Entry Hasil Lab.. Edit Hasil Lab.....	66 67
Gambar 1.51 Sequence Diagram untuk Edit Hasil Lab... Display Hasil Lab.....	67 68
Gambar 1.52 Sequence Diagram untuk Display Hasil Lab	68
Gambar 1.53 Sequence Diagram untuk Delete Hasil Lab. Entry Radiologi.....	69 70

Gambar 1.54 Sequence Diagram untuk Entry Radiologi ..	70
Edit Radiologi	71
Gambar 1.55 Sequence Diagram untuk Edit Radiologi ...	71
DisplayRadiologi	72
Gambar 1.56 Sequence Diagram untuk Display Radiologi	72
Entry Hasil Radiologi	73
Gambar 1.57 Sequence Diagram untuk Entry Hasil Radiologi	73
Edit Hasil Radiologi	74
Gambar 1.58 Sequence Diagram untuk Edit Hasil Radiologi	74
DisplayHasil Radiologi	75
Gambar 1.59 Sequence Diagram untuk Display Hasil Radiologi	75
DeleteHasil Radiologi	76
Gambar 1.60 Sequence Diagram untuk Delete Hasil Radiologi	76
Gambar 1.61 Class Diagram	79
Gambar 1.62 Rancangan Antarmuka Login	135
Gambar 1.63 Rancangan Antarmuka Menu Utama	136
Gambar 1.64 Rancangan Antarmuka Edit Password	137
Gambar 1.66 Rancangan Antarmuka Tambah Data Pengelolaan Data Pegawai	139
Gambar 1.67 Rancangan Antarmuka Edit Data Pengelolaan Data Pegawai	140
Gambar 1.68 Rancangan Antarmuka Delete Data Pengelolaan Data Pegawai	141
Gambar 1.69 Rancangan Antarmuka Pengelolaan Poliklinik	142

Gambar 1.70 Rancangan Antarmuka Tambah Data Pengelolaan Poliklinik.....	143
Gambar 1.71 Rancangan Antarmuka Edit Data Pengelolaan Poliklinik.....	144
Gambar 1.73 Rancangan Antarmuka Pengelolaan Jadwal Dokter.....	147
Gambar 1.75 Rancangan Antarmuka Edit Data Pengelolaan Jadwal Dokter.....	149
Gambar 1.77 Rancangan Antarmuka Pengelolaan Data Obat	151
Gambar 1.78 Rancangan Antarmuka Tambah Data Obat...	152
Gambar 1.79 Rancangan Antarmuka Edit Data Obat.....	153
Gambar 1.80 Rancangan Antarmuka Delete Data Obat...	154
Gambar 1.81 Rancangan Antarmuka Search Data Obat...	155
Gambar 1.82 Rancangan Antarmuka Pengelolaan Kamar..	156
Gambar 1.83 Rancangan Antarmuka Tambah Data Kamar..	157
Gambar 1.84 Rancangan Antarmuka Edit Data Kamar....	158
Gambar 1.85 Rancangan Antarmuka Delete Data Kamar..	159
1.86 Rancangan Antarmuka Input Data Data Pasien....	160
Gambar 1.87 Rancangan Antarmuka daftar pemeriksaan pilih Poliklinik.....	161
Gambar 1.89 Rancangan Antarmuka Pendaftaran Rawat Inap	162
Gambar 1.90 Rancangan Antarmuka Daftar Pasien Periksa	163
Gambar 1.91 Rancangan Antarmuka Cetak Bukti Pendaftaran	164
Gambar 1.92 Rancangan Antarmuka Input Pemeriksaan Awal	165

Gambar 1.93 Rancangan Antarmuka Ubah Hasil Pemeriksaan	166
Gambar 1.94 Rancangan Antarmuka Pemeriksaan Dokter.	167
Gambar 1.95 Rancangan Antarmuka Inputan Pemeriksaan Dokter.....	168
Gambar 1.96 Rancangan Antarmuka Detail Pembayaran..	169



Pendahuluan

1.1 Tujuan

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.

1.2 Ruang Lingkup

Perangkat Lunak SIRS dikembangkan dengan tujuan untuk :

1. Melakukan data pegawai.
2. Melakukan pengelolaan obat.
3. Melakukan pengelolaan bidang medis.
4. Melakukan pengelolaan data pasien.
5. Melakukan pengelolaan data Poliklinik.
6. Melakukan pengelolaan pendaftaran Pemeriksaan.
7. Melakukan pengelolaan pendaftaran perawatan.
8. Melakukan pengelolaan pemeriksaan awal.
9. Melakukan pengelolaan pemeriksaan dokter.
10. Melakukan pengelolaan pembayaran rumah sakit.
11. Melakukan pengelolaan resep.

Program Studi Teknik Informatika	DPPL – SIRS	17/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

12. Melakukan pengelolaan data Kamar.
13. Melakukan pengelolaan Kelas Rawat Inap.
14. Melakukan pengelolaan layanan lab.
15. Melakukan pengelolaan hasil lab.
16. Melakukan pengelolaan radiologi.
17. Melakukan pengelolaan hasil radiologi.
18. Menampilkan laporan pendapatan rawat inap.
19. Menampilkan laporan pendapatan rawat jalan.

1.3 Definisi dan Akronim

Daftar definisi akronim dan singkatan :

Keyword/Phrase	Definisi
DPPL	Deskripsi Perancangan Perangkat Lunak disebut juga Software Design Description (SDD) merupakan deskripsi dari perancangan produk/perangkat lunak yang akan dikembangkan.
SIRS	Sistem Informasi Rumah Sakit , merupakan sistem informasi yang digunakan untuk proses bisnis Rumah Sakit
GUI	Graphic User Interface, tampilan antarmuka sistem untuk pegawai
Database	Kumpulan data terkait yang diorganisasikan dalam struktur

Program Studi Teknik Informatika	DPPL – SIRS	18/161
----------------------------------	-------------	--------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

	tertentu dan dapat diakses dengan cepat
--	---

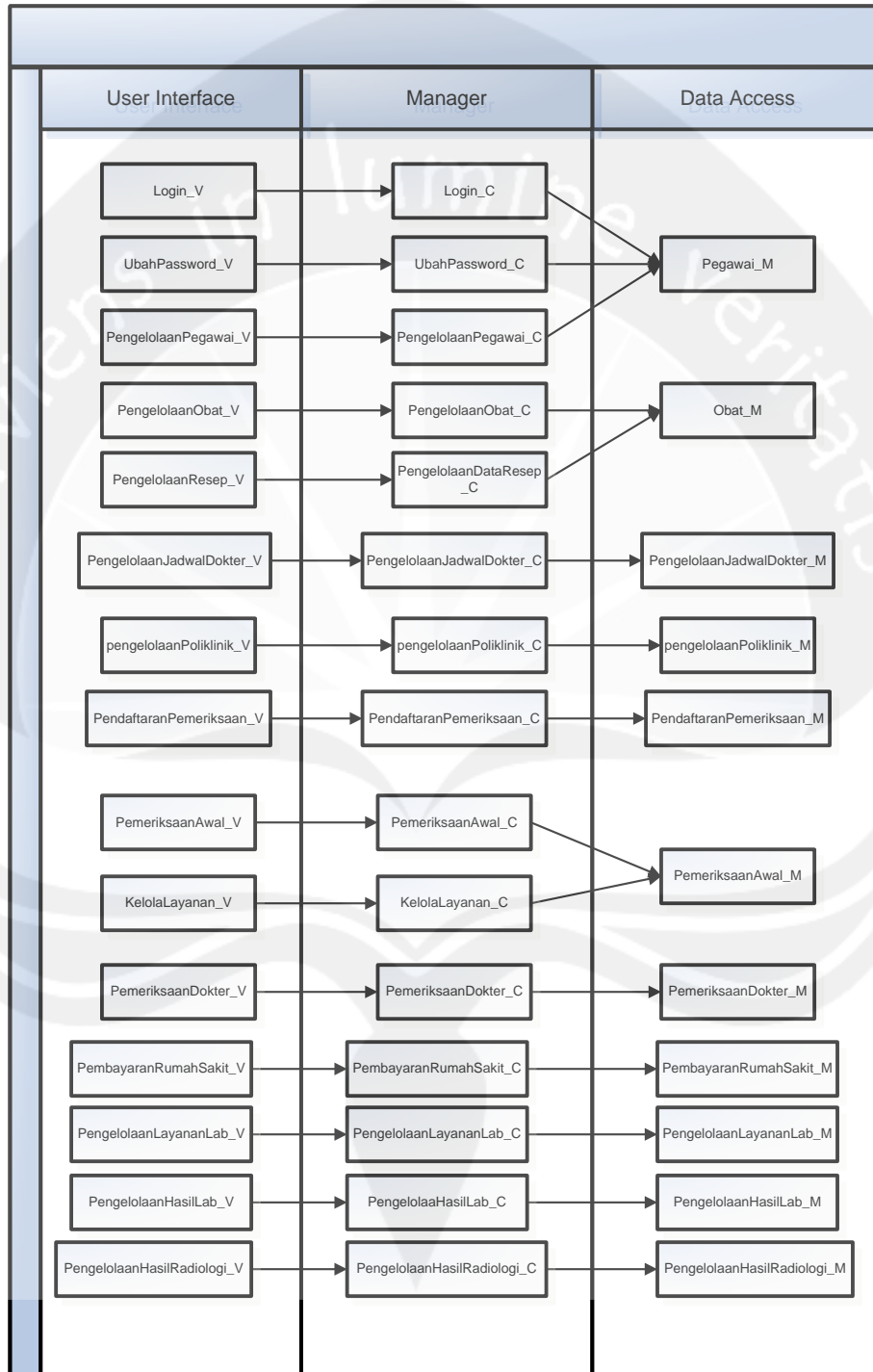
1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

1. Bennet Simon, McRobb Steve, Farmer Ray, *Object-Oriented System Analysis and Design Using UML*, McGraw-Hill Companies, 2002.
2. Boggs Wendy, Boggs Michael, *Mastering UML with Rational Rose 2002*, SYBEX Inc, 2002.
3. Deitel, *C# How to Program*, Prentice-Hall Inc, 2002.
4. *MSDN Library*-October 2005, Microsoft, 2005.
5. Aditya Primandaru, Yulius. *DPPL SPP (Sistem Penjualan Perabotan)*. Universitas Atma Jaya Yogyakarta. 2012
6. Wibisono, Aryo. *DPPL SIAMA (Sistem Informasi Akademik Mahasiswa)*. Universitas Atma Jaya Yogyakarta. 2010.

Perancangan Sistem

1.5 Perancangan Arsitektur

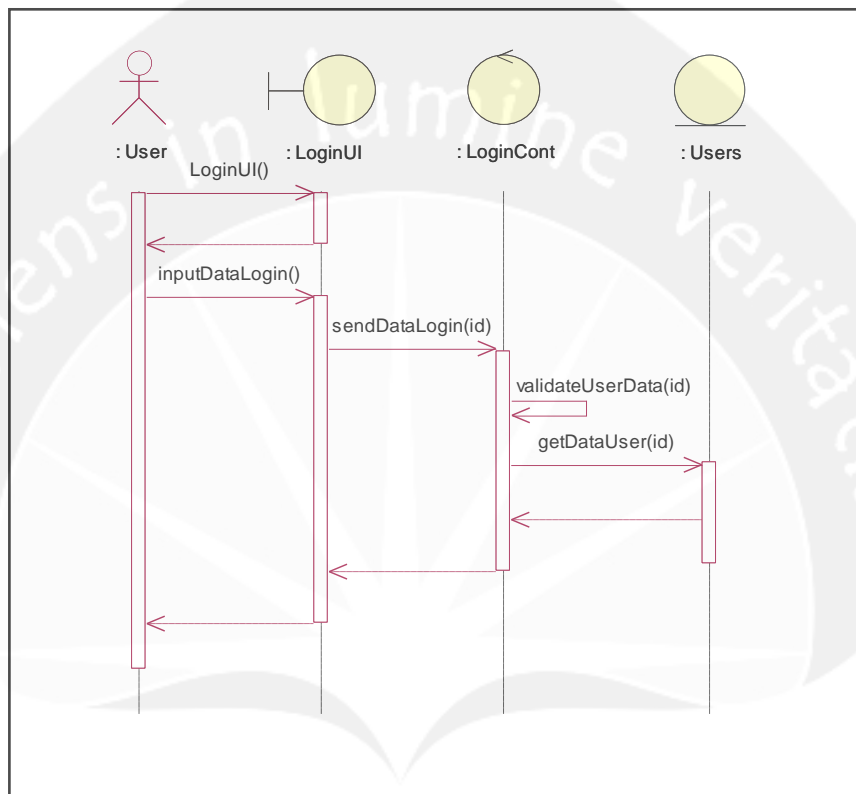


Gambar 1.1 Rancangan Arsitektur SIRS

1.6 Perancangan Detail

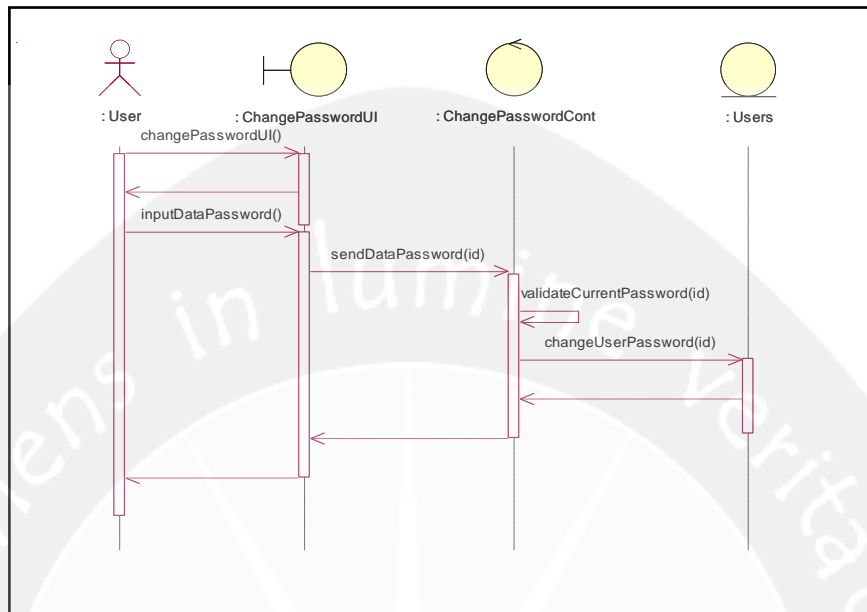
1.6.1 Sequence Diagram

1.6.1.1 Login



Gambar 1.2 Sequence Diagram untuk Login

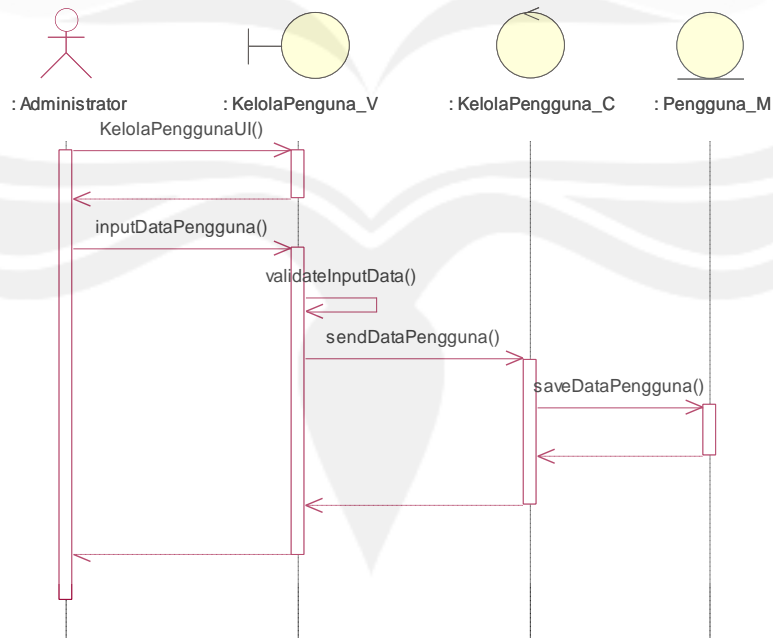
1.6.1.2 Change Password



Gambar 1.3 Sequence Diagram untuk Change Password

1.6.1.3 Kelola Data Pegawai

Entry Data Pegawai



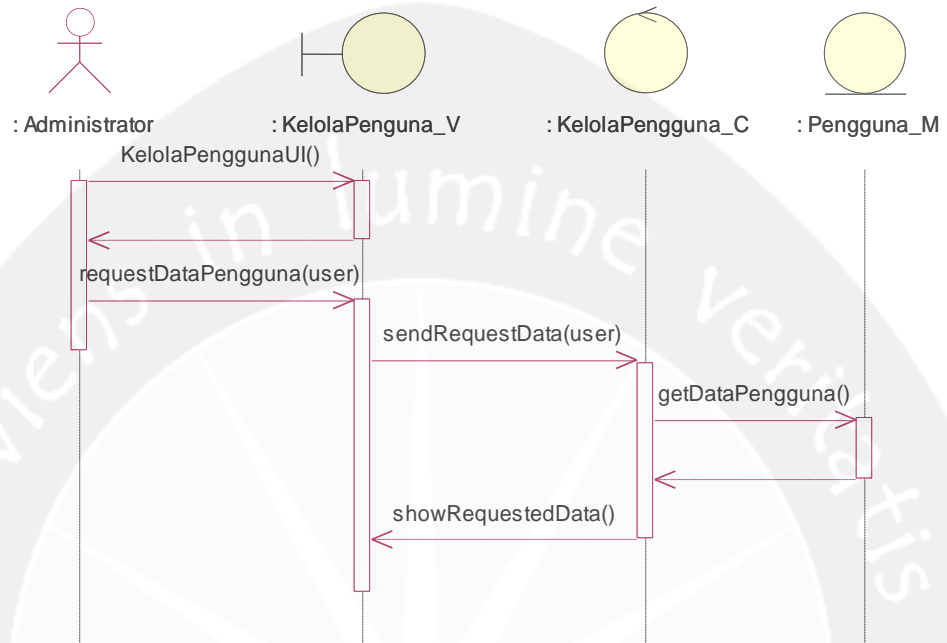
Gambar 1.4 Sequence Diagram untuk Entry Data Pegawai

Edit Data Pegawai



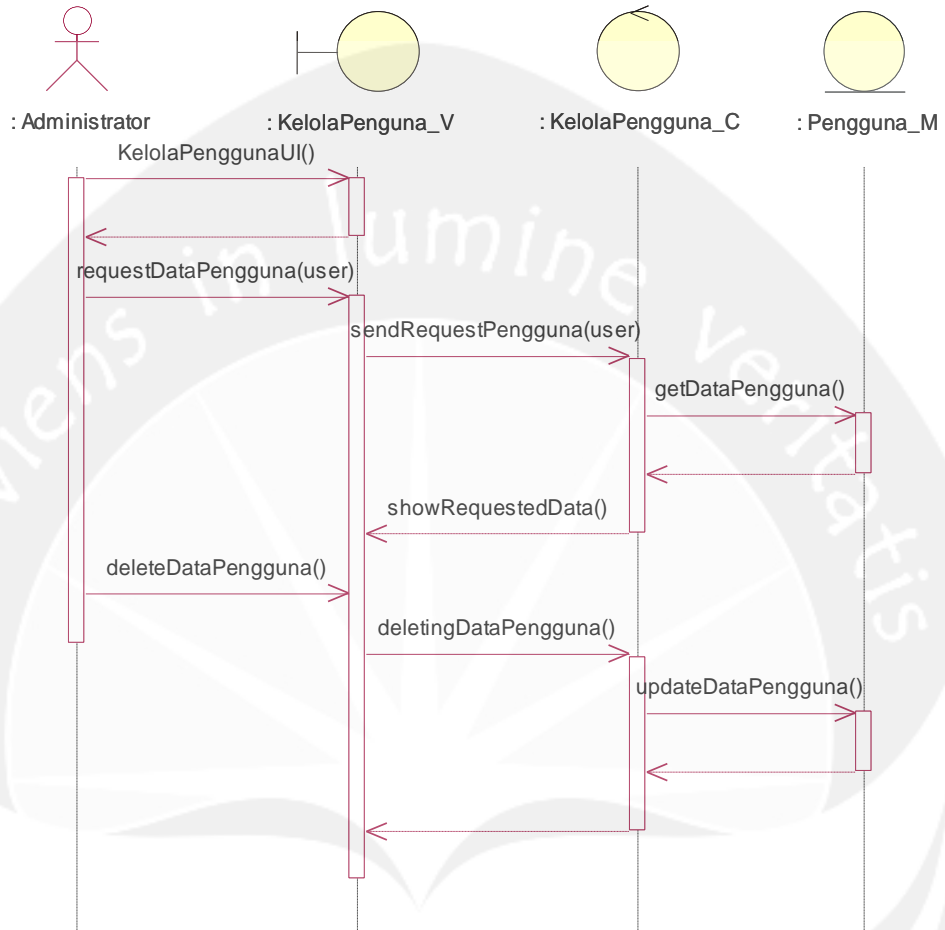
Gambar 1.5 Sequence Diagram untuk Edit Data Pegawai

Display Data Pegawai



Gambar 1.6 Sequence Diagram untuk Display Data Pegawai

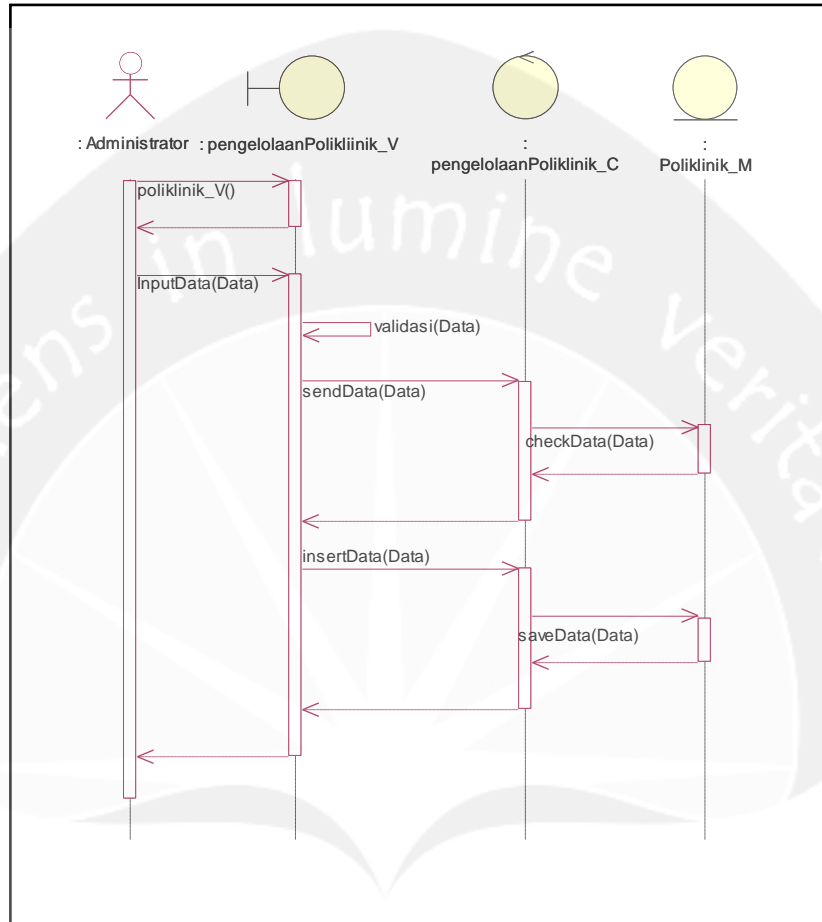
Delete Data Pegawai



Gambar 1.7 Sequence Diagram untuk Delete Data Pegawai

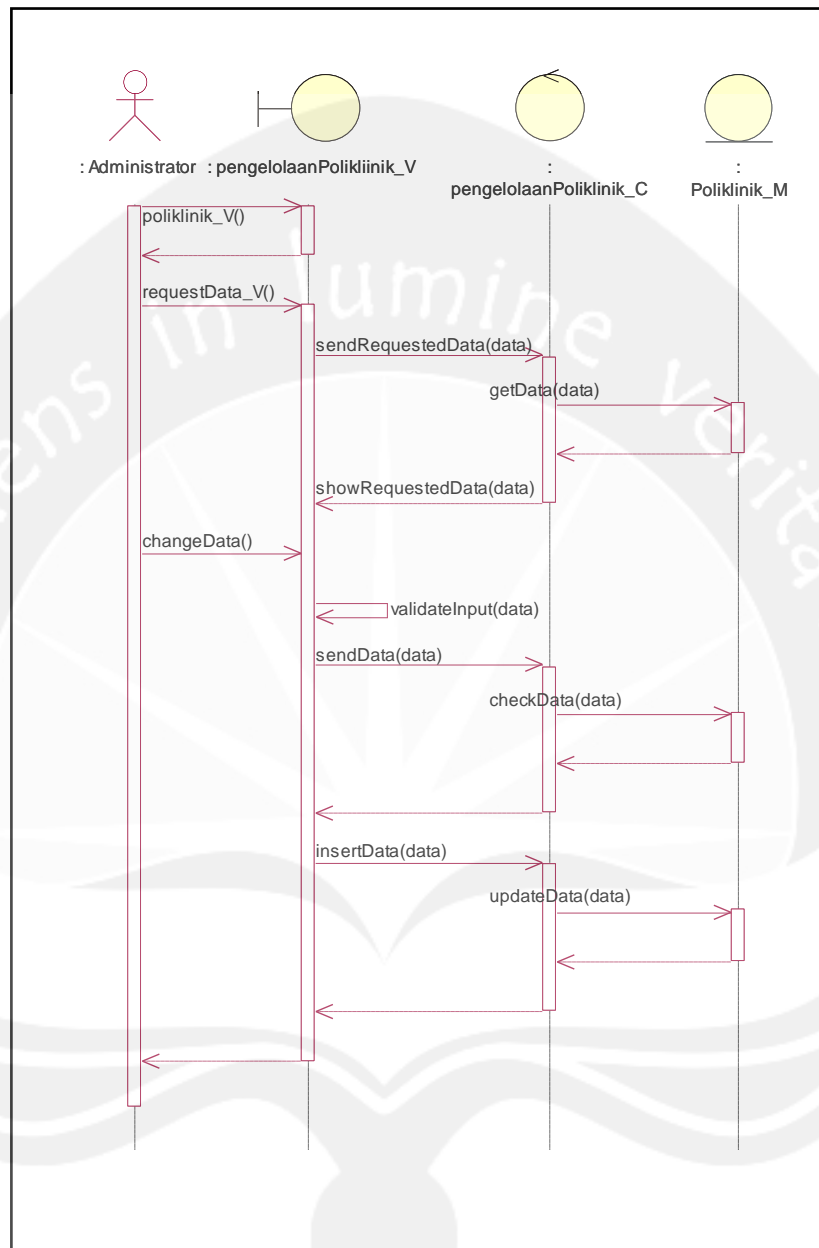
1.6.1.4 Kelola Data Poliklinik

Entry Data Poliklinik



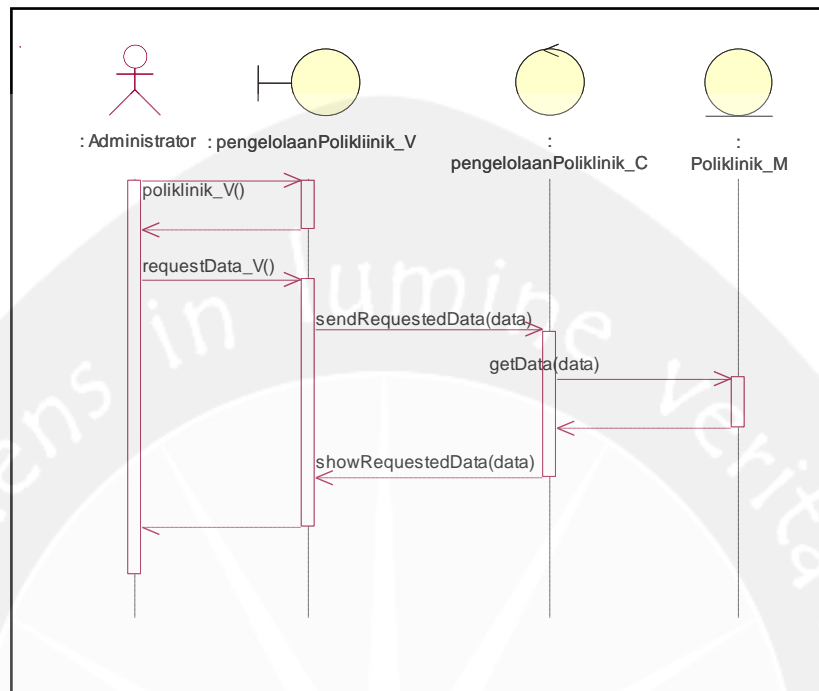
Gambar 1.8 Sequence Diagram untuk Entry Data Poliklinik

Edit Data Poliklinik



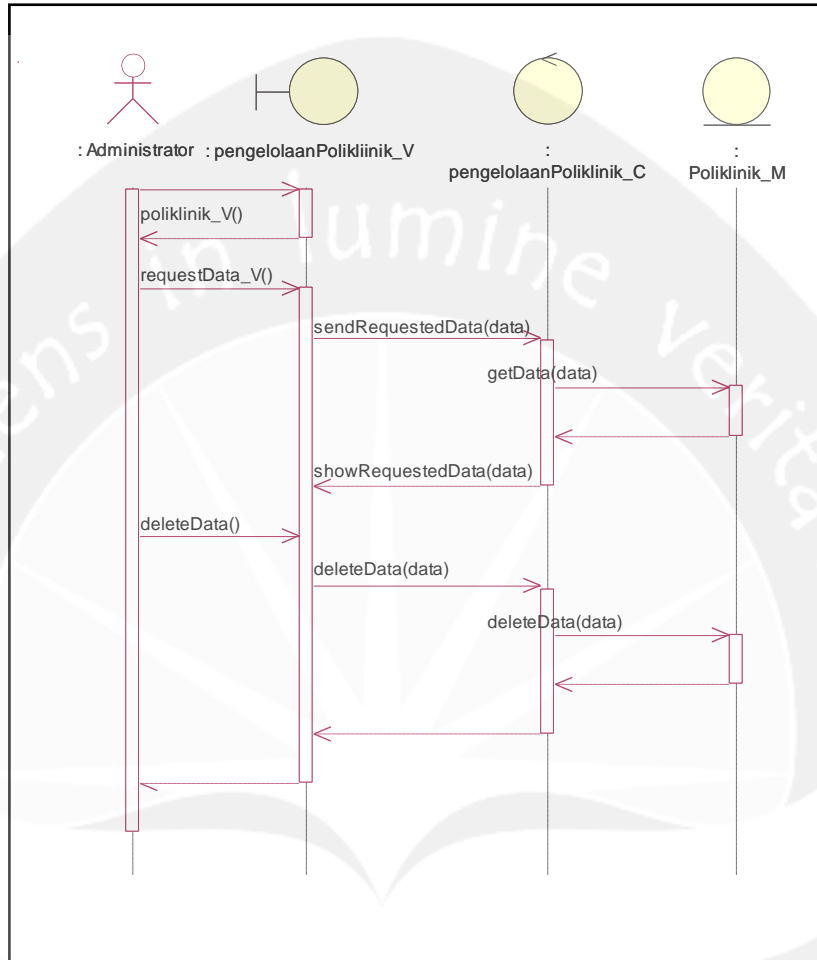
Gambar 1.9 Sequence Diagram untuk Edit Data Poliklinik

Display Data Poliklinik



Gambar1.10 Sequence Diagram untuk Display Data Poliklinik

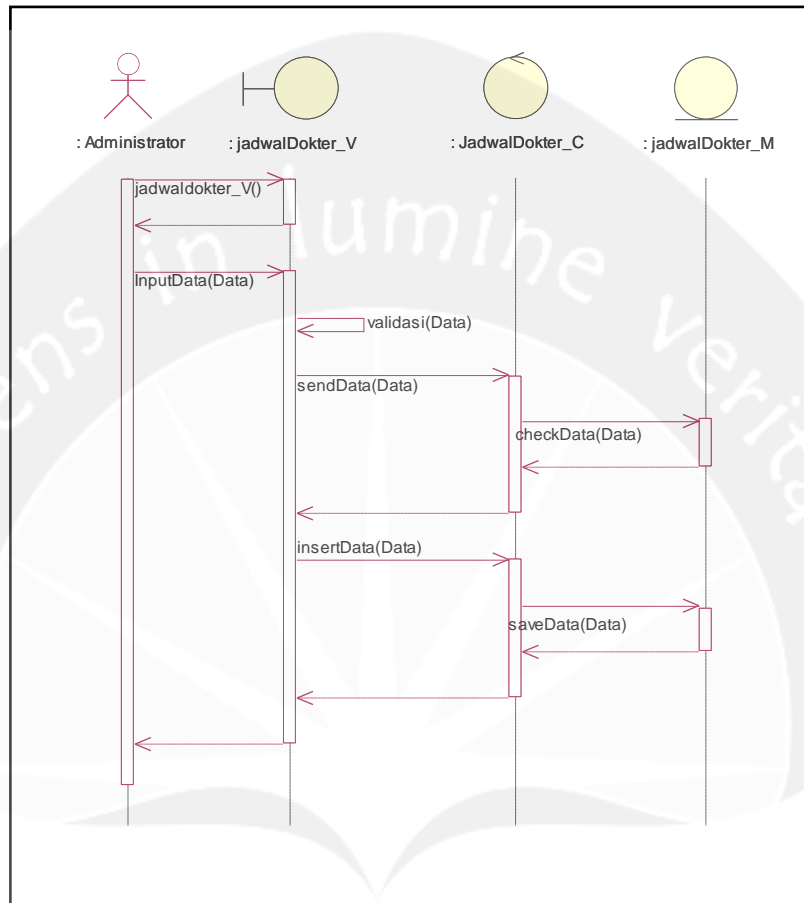
Delete Data Poliklinik



Gambar 1.11 Sequence Diagram untuk Delete Data Poliklinik

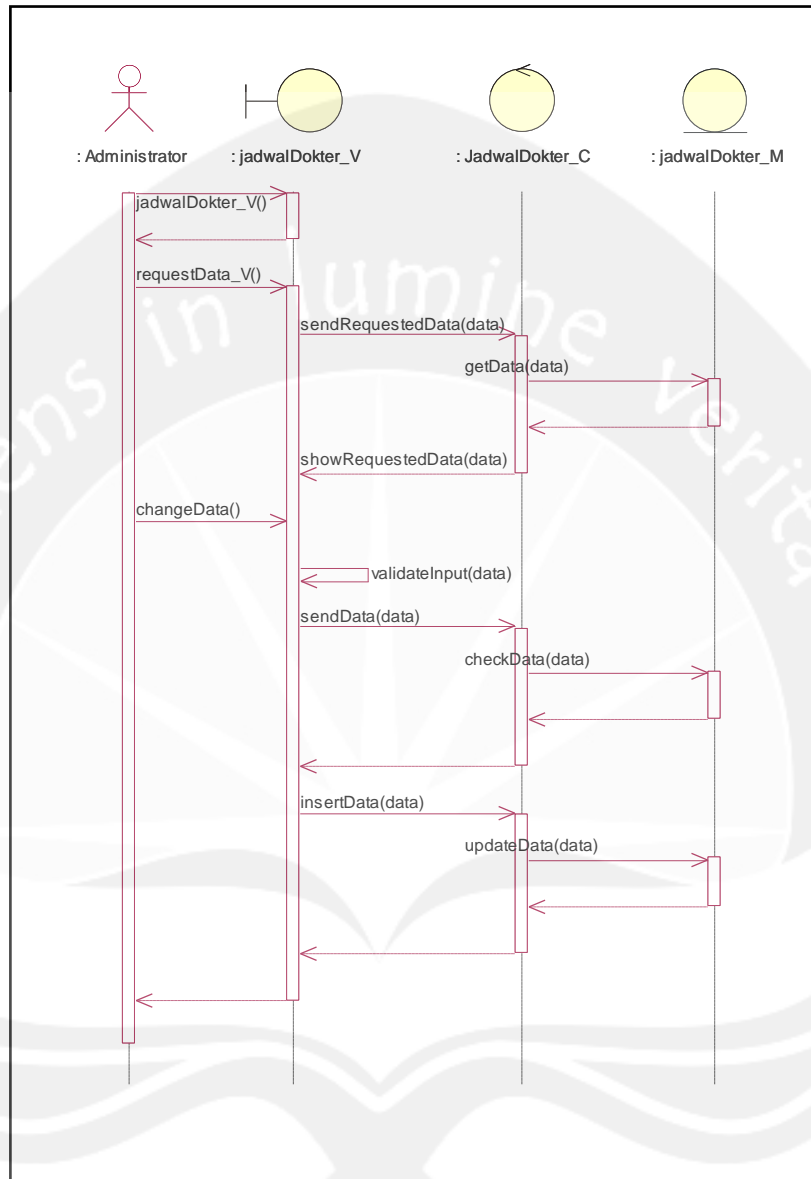
1.6.1.5 Kelola Jadwal Dokter

Entry Jadwal Dokter



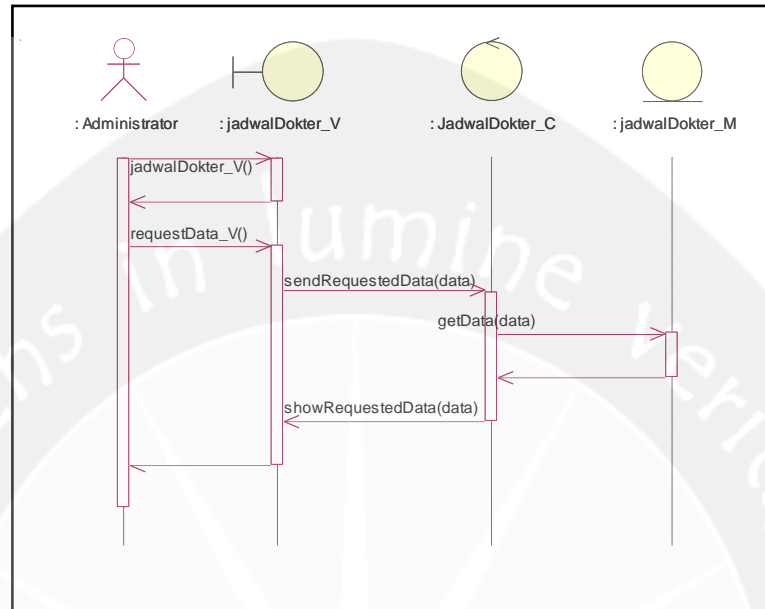
Gambar 1.12 Sequence Diagram untuk Entry Jadwal Dokter

Edit Jadwal Dokter



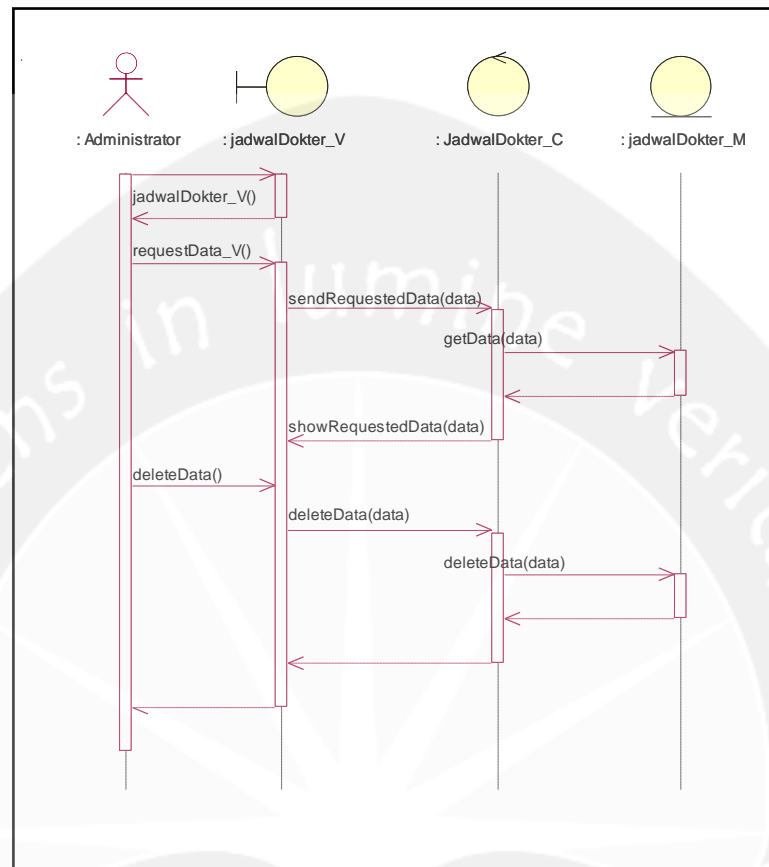
Gambar 1.13 Sequence Diagram untuk Edit Jadwal Dokter

Display Jadwal Dokter



Gambar 1.14 Sequence Diagram untuk Display Jadwal Dokter

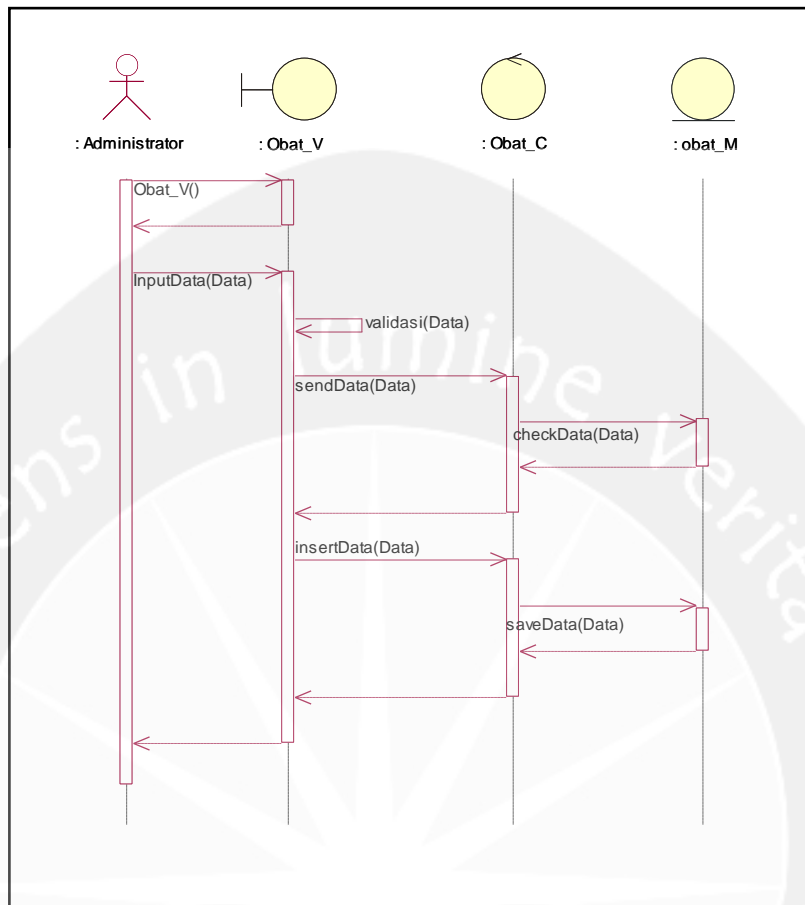
Delete Jadwal Dokter



Gambar 1.15 Sequence Diagram untuk Delete Jadwal Dokter

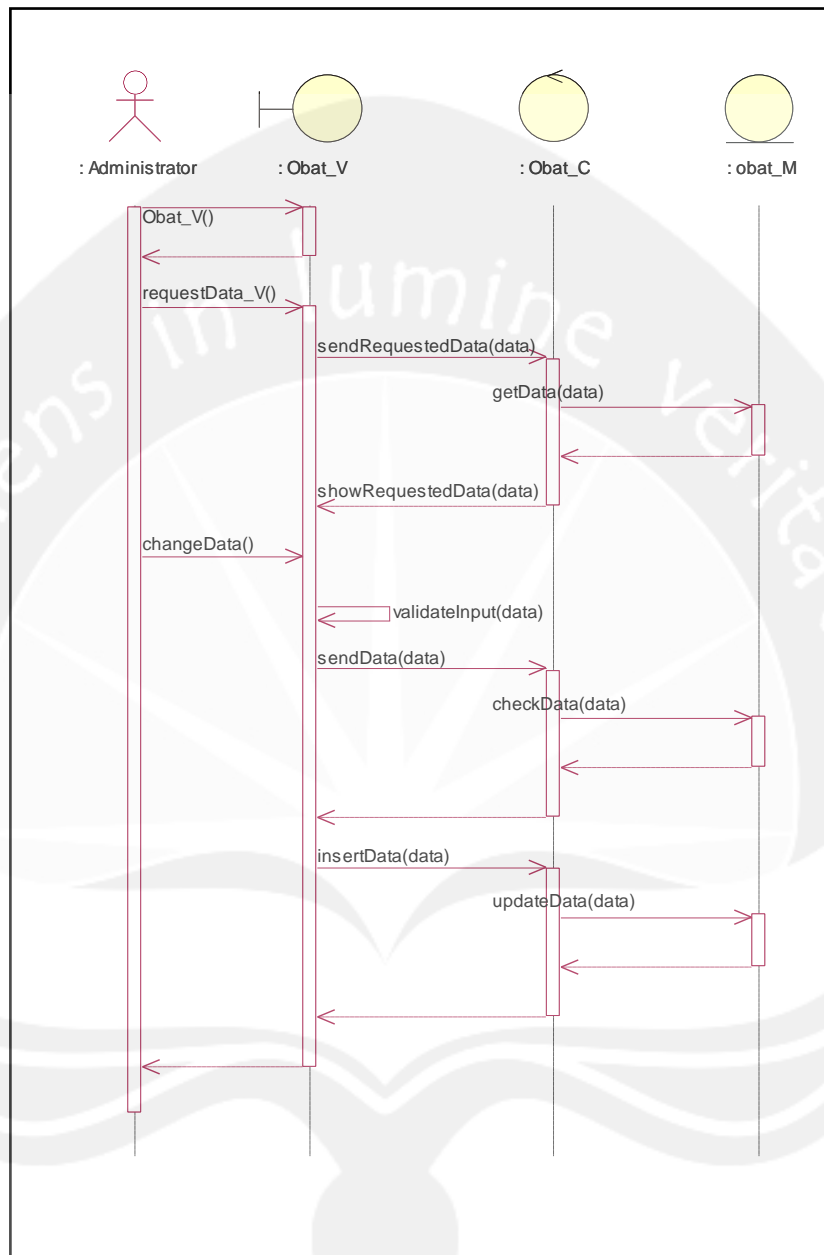
1.6.1.6 Kelola Data Obat

Entry Data Obat



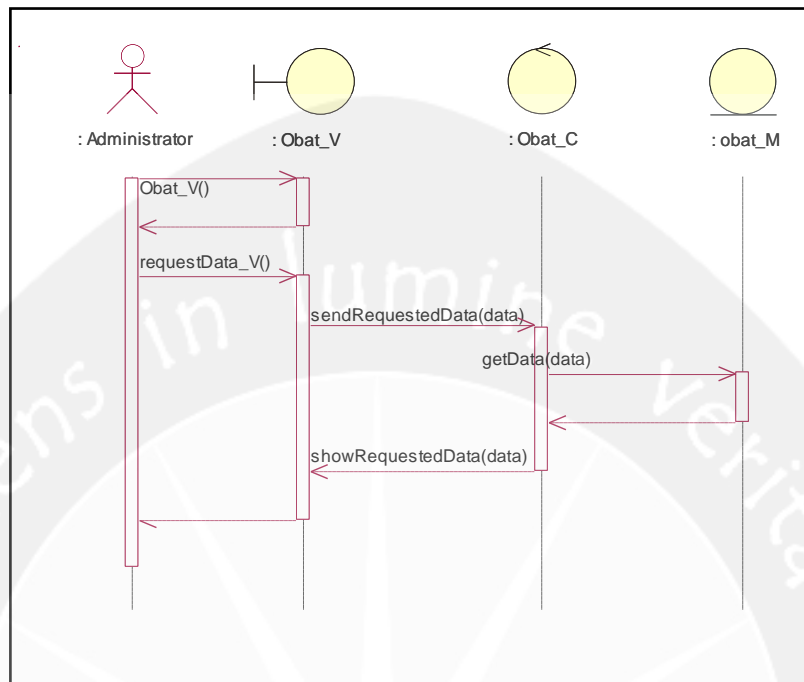
Gambar 1.16 Sequence Diagram untuk Entry Data Obat

Edit Data Obat



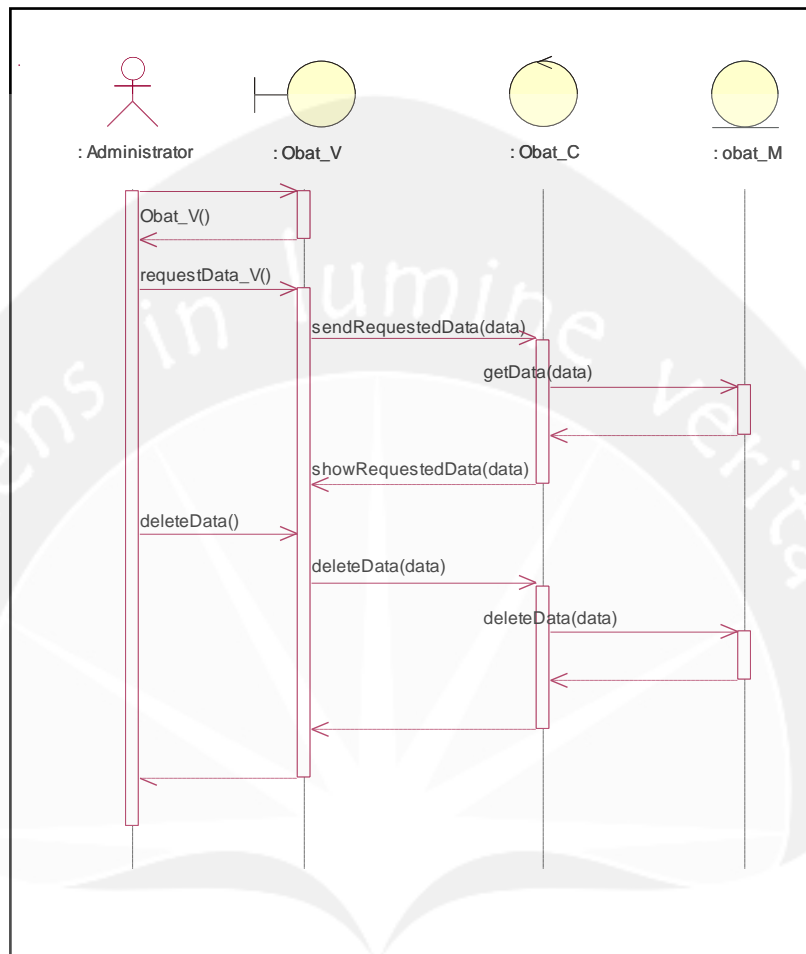
Gambar 1.17 Sequence Diagram untuk Edit Data Obat

Display Data Obat



Gambar 1.18 Sequence Diagram untuk Display Data Obat

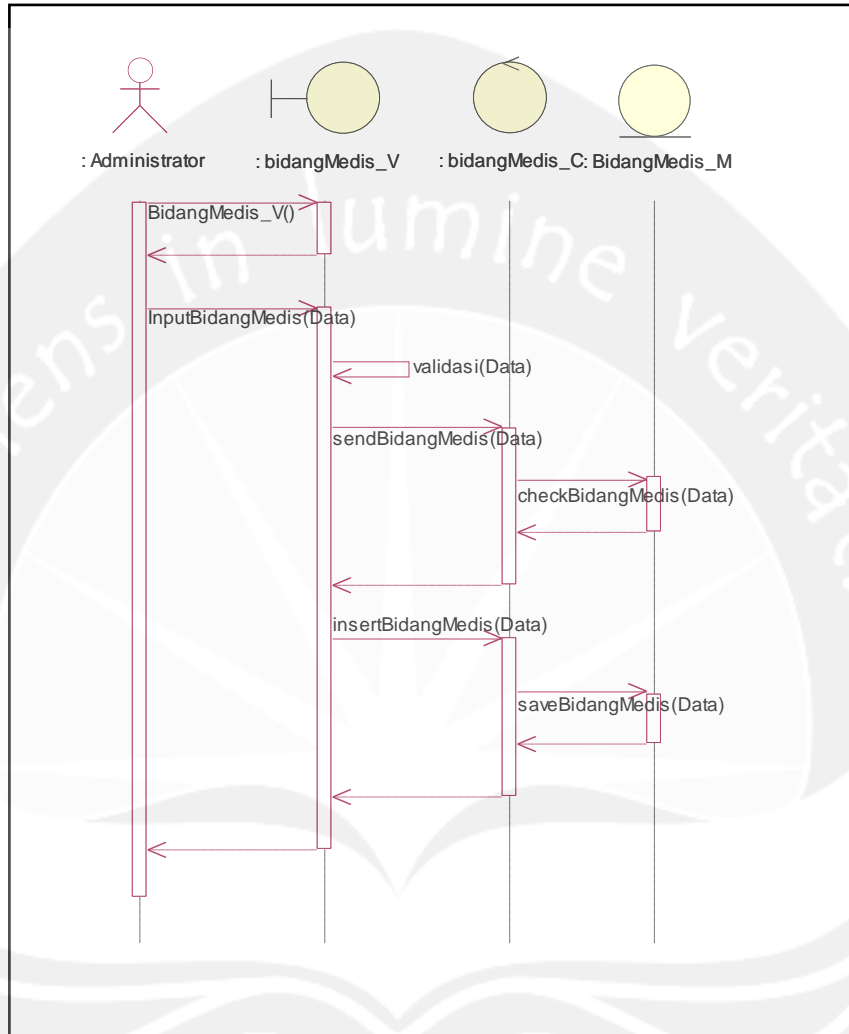
Delete Data Obat



Gambar 1.20 Sequence Diagram untuk Delete Data Obat

1.6.1.7 Kelola Bidang Medis

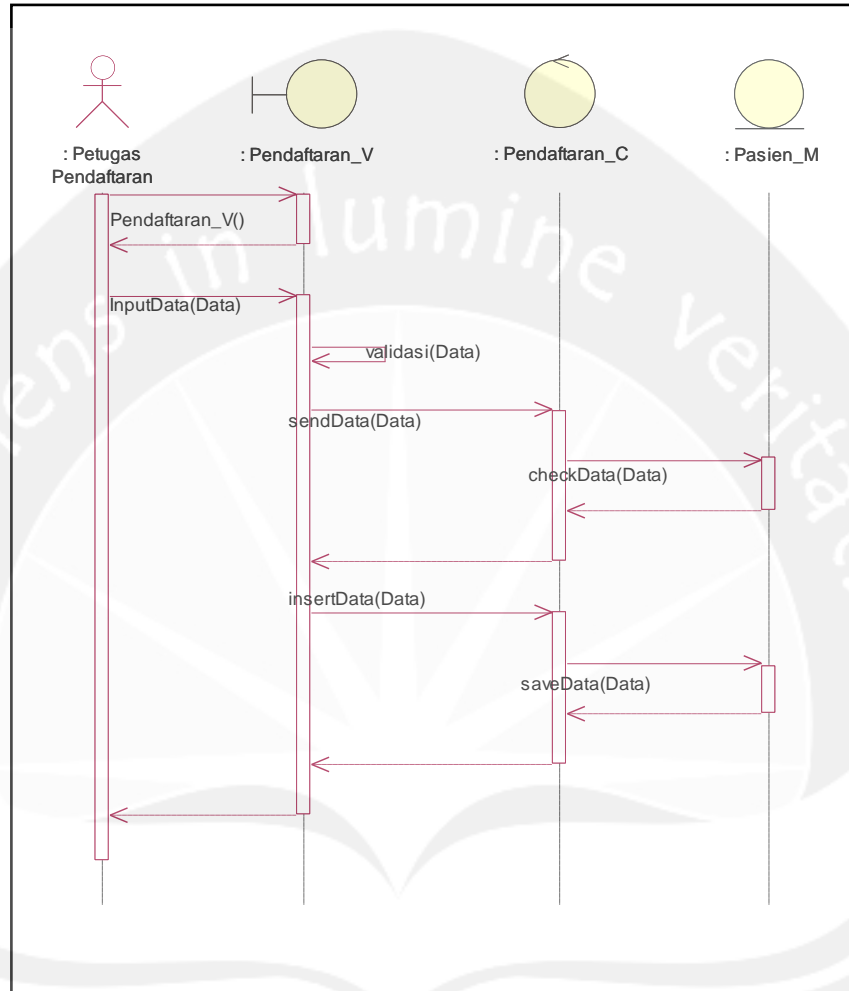
Entry Bidang Medis



Gambar 1.21 Sequence Diagram untuk Entry Bidang Medis

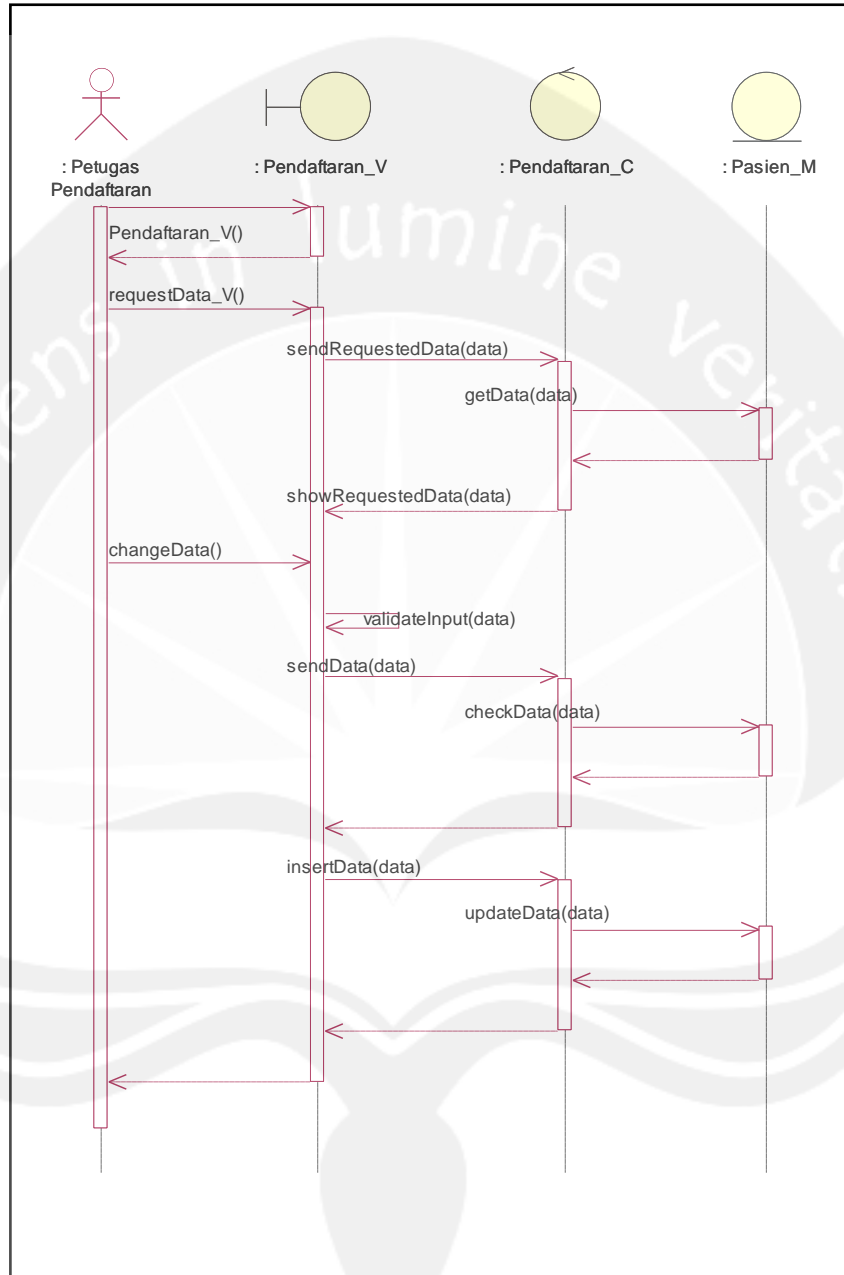
1.6.1.8 Kelola Data Pasien

Entry Data Pasien



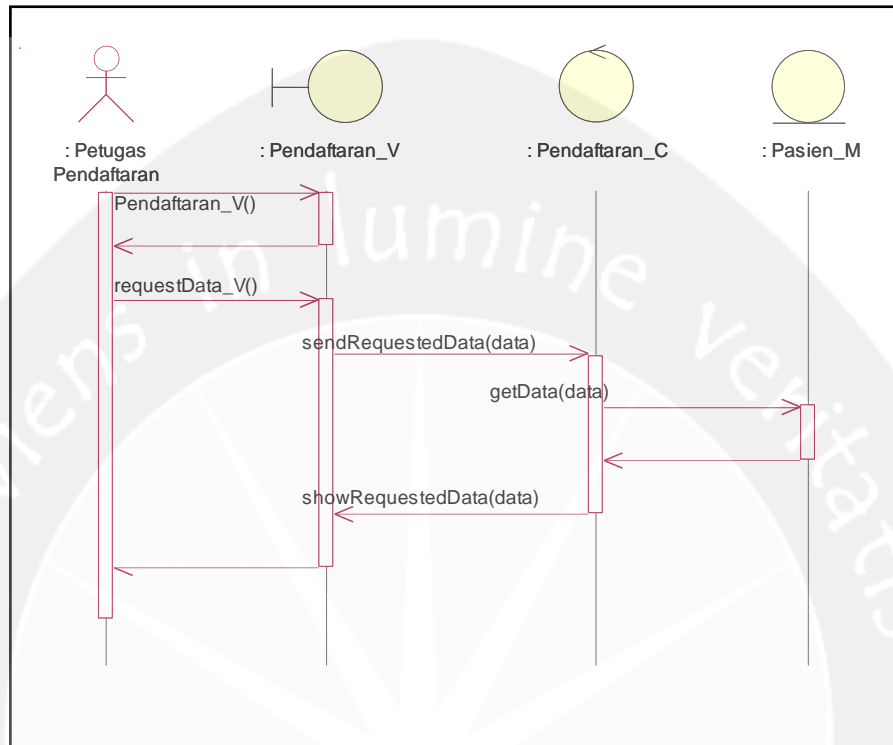
Gambar 1.22 Sequence Diagram untuk Entry Data Pasien

Edit Data Pasien



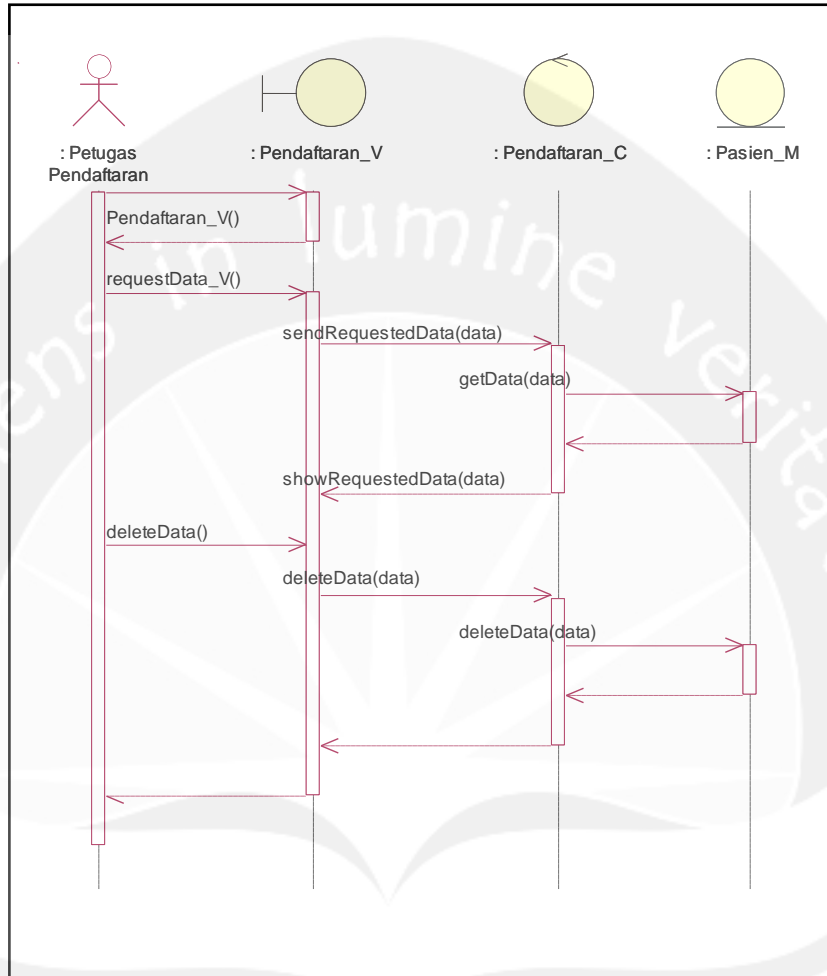
Gambar 1.23 Sequence Diagram untuk Edit Data Pasien

Display Data Pasien



Gambar 1.24 Sequence Diagram untuk Display Data Pasien

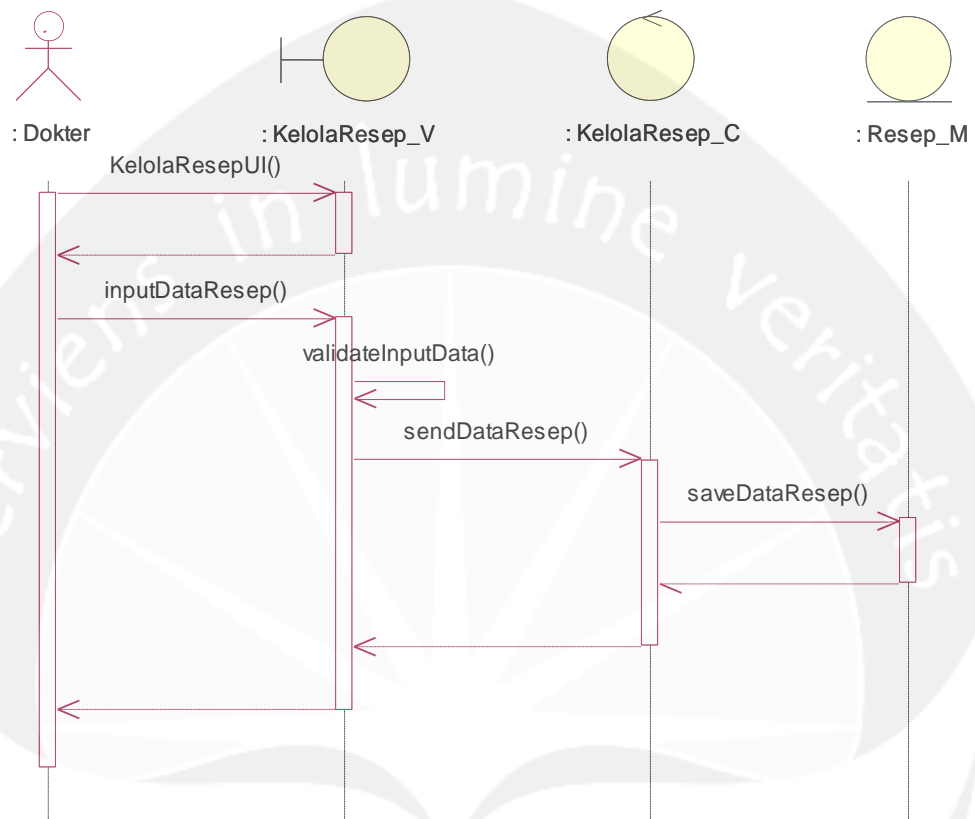
Delete Data Pasien



Gambar 1.25 Sequence Diagram untuk Delete Data Pasien

1.6.1.9 Kelola Data Resep

Entry Data Resep



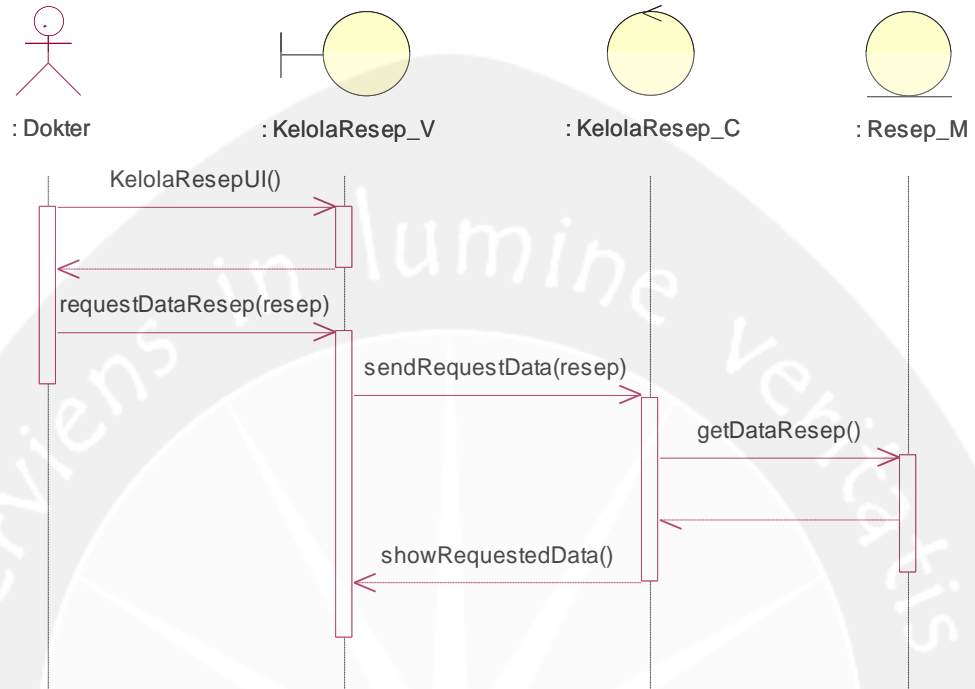
Gambar 1.26 Sequence Diagram untuk Entry Data Resep

Edit Data Resep



Gambar 1.27 Sequence Diagram untuk Edit Data Resep

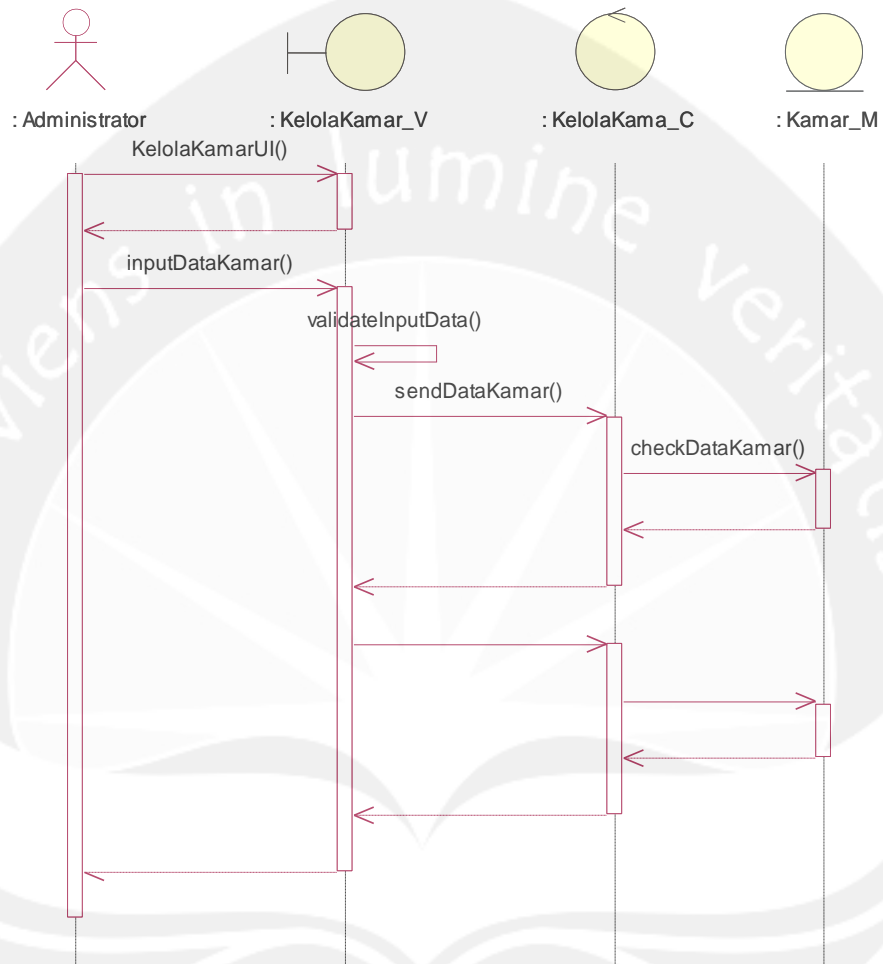
Display Data Resep



Gambar 1.28 Sequence Diagram untuk Display Data Resep

1.6.1.10 Kelola Data Kamar

Entry Data Kamar



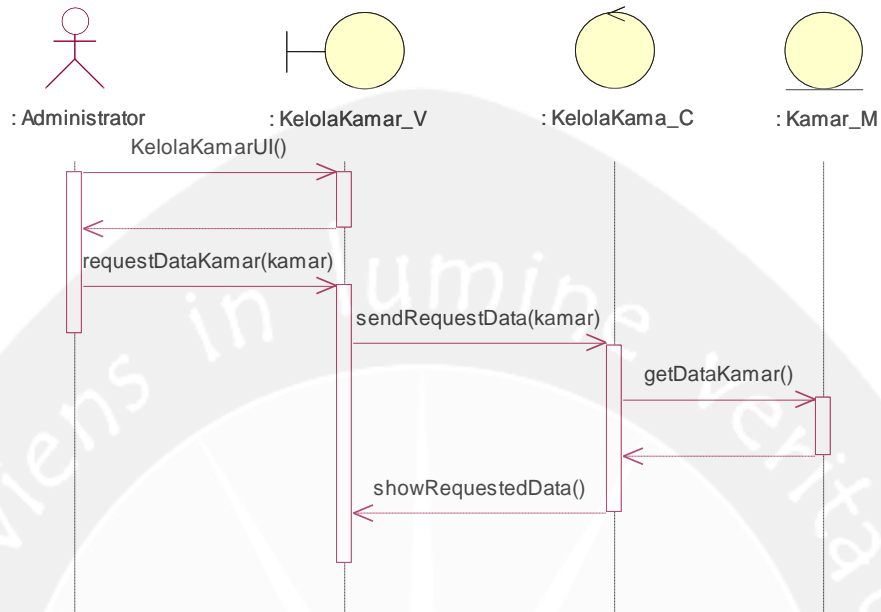
Gambar 1.29 Sequence Diagram untuk Entry Data Kamar

Edit Data Kamar



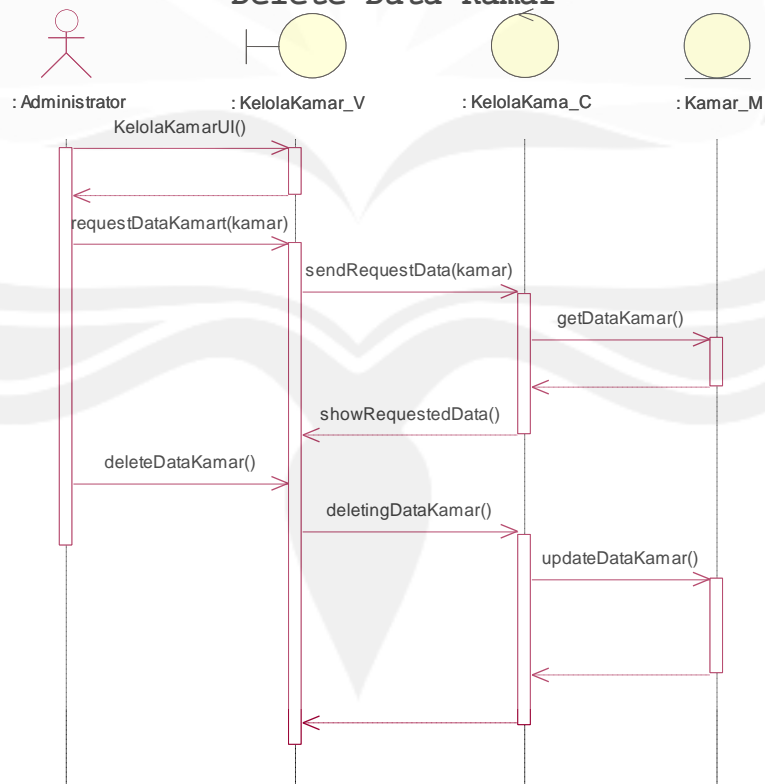
Gambar 1.30 Sequence Diagram untuk Edit Data Kamar

Display Data Kamar



Gambar 2.33 Sequence Diagram untuk Display Data Kamar

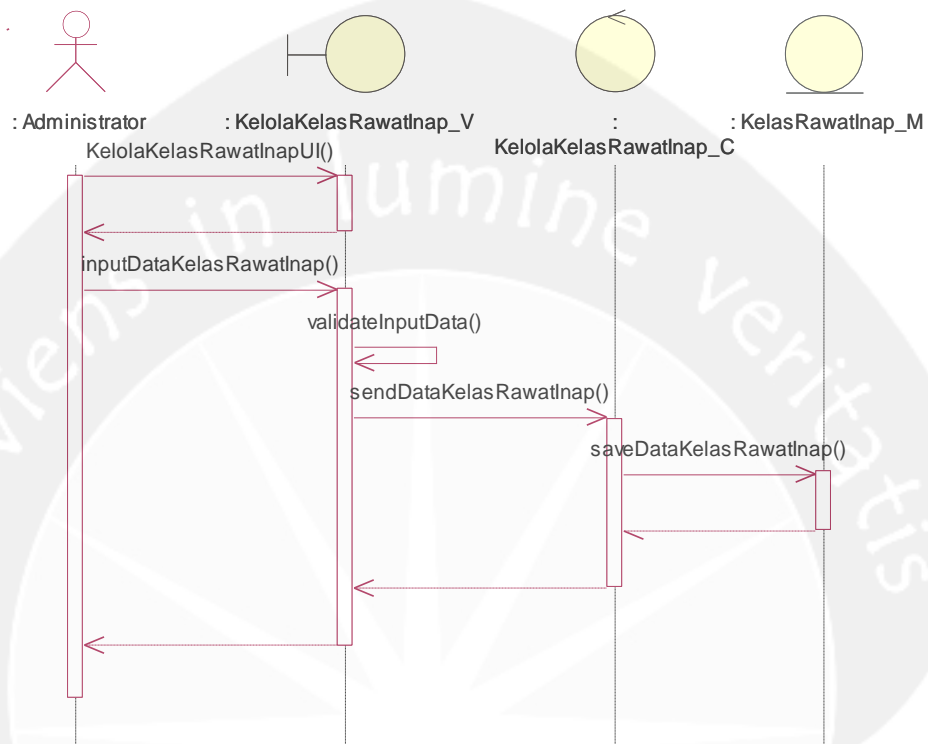
Delete Data Kamar



Gambar 1.31 Sequence Diagram untuk Delete Data Kamar

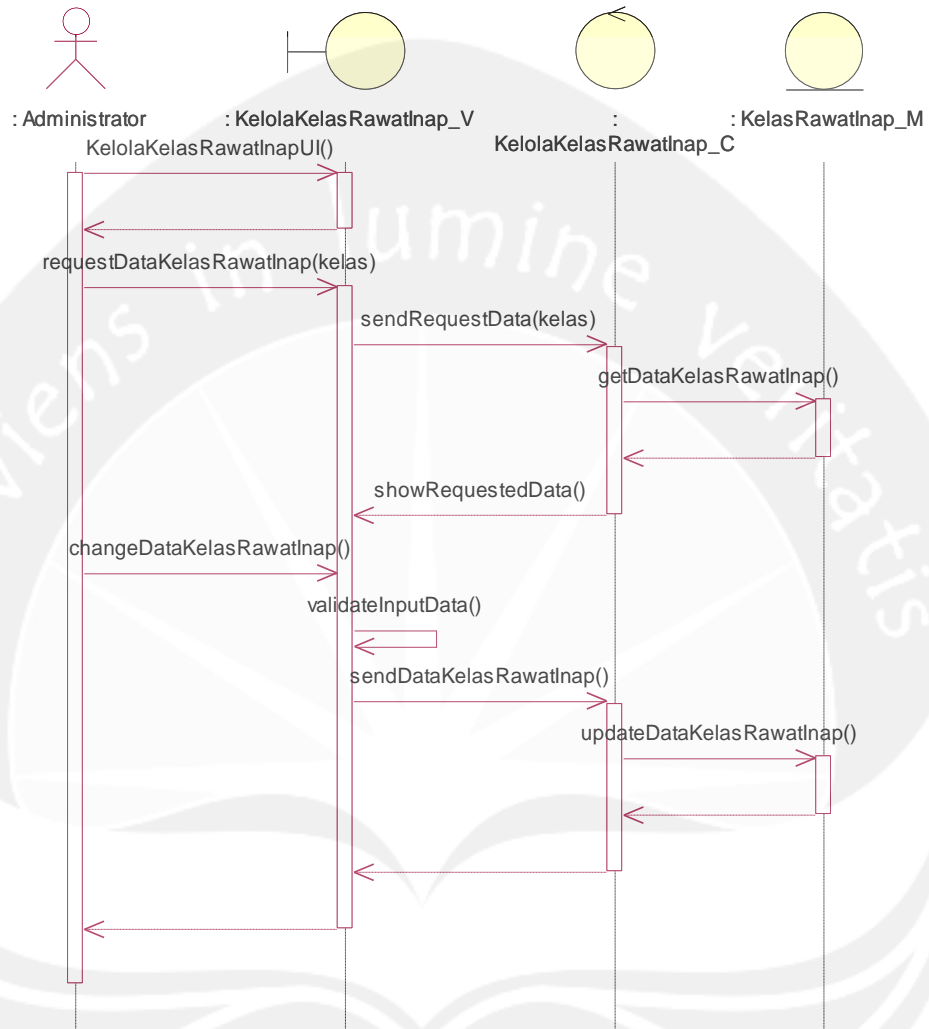
1.6.1.11 Kelola Kelas Rawat Inap

Entry Kelas Rawat Inap



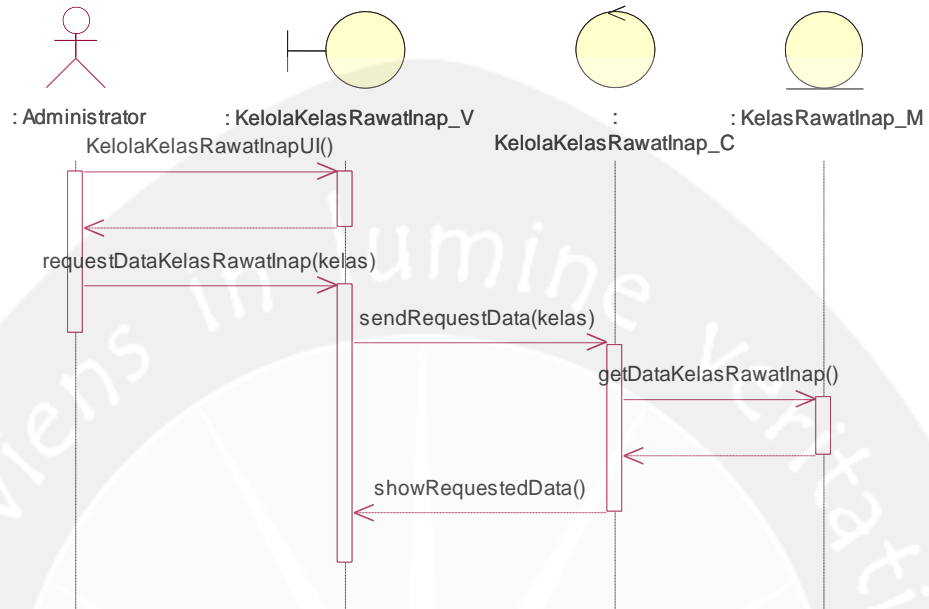
Gambar 1.32 Sequence Diagram untuk Entry Data Kelas Rawat Inap

Edit Kelas Rawat Inap



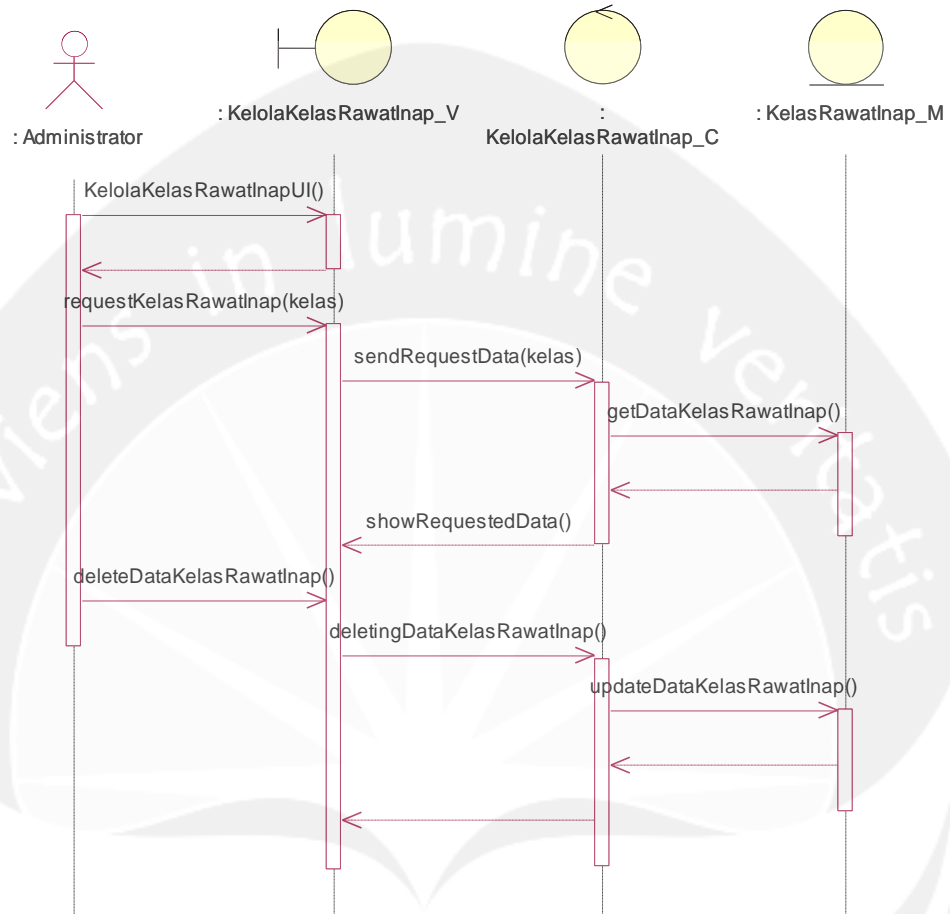
Gambar 1.33 Sequence Diagram untuk Edit Data Kelas Rawat Inap

Display Kelas Rawat Inap



Gambar 1.34 Sequence Diagram untuk Display Data Kelas Rawat Inap

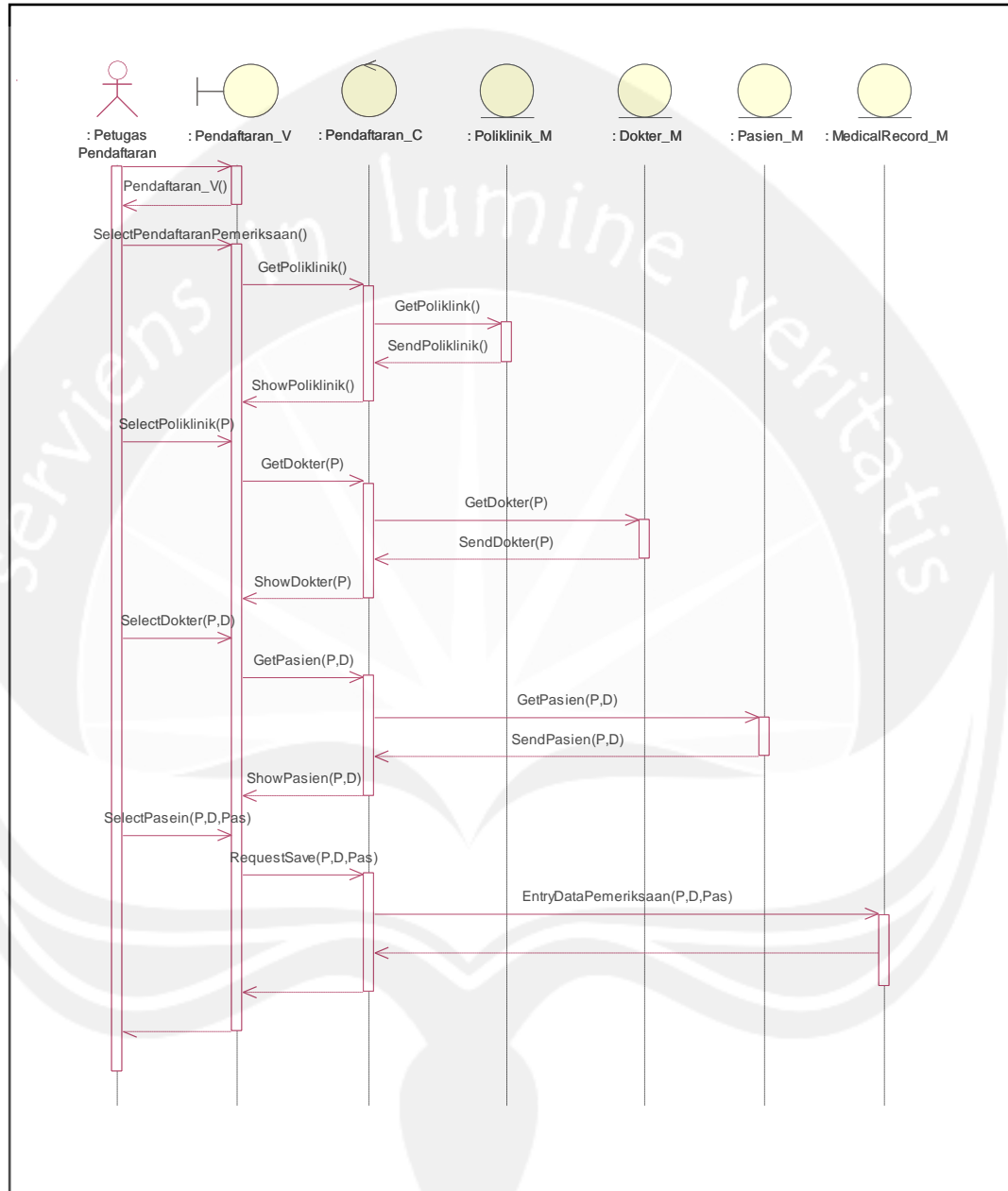
Delete Kelas Rawat Inap



Gambar 1.35 Sequence Diagram untuk Delete Data Kelas Rawat Inap

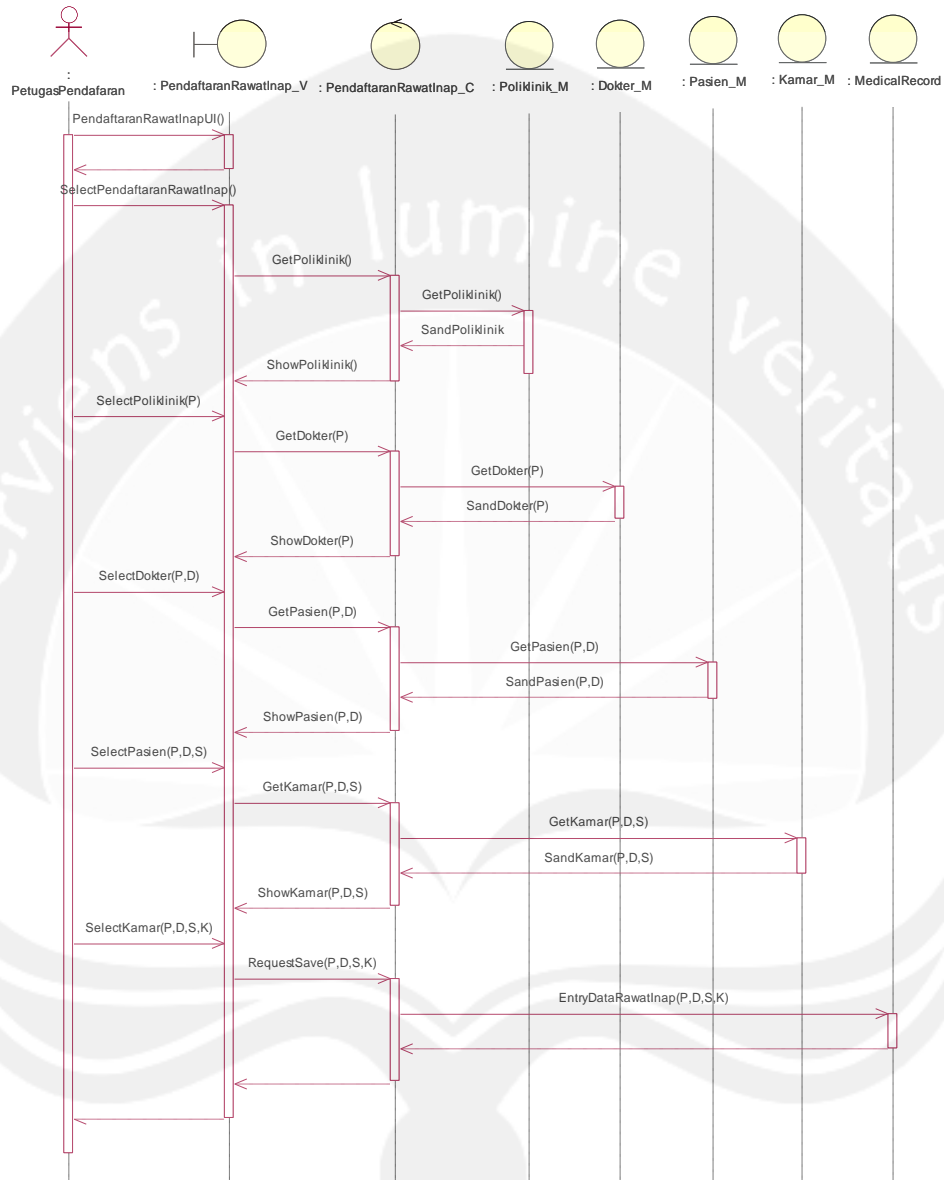
1.6.1.12 Pendaftaran Pasien

Pendaftaran PasienPeriksa



Gambar 1.37 Sequence Diagram untuk Pendaftaran Pasien Periksa

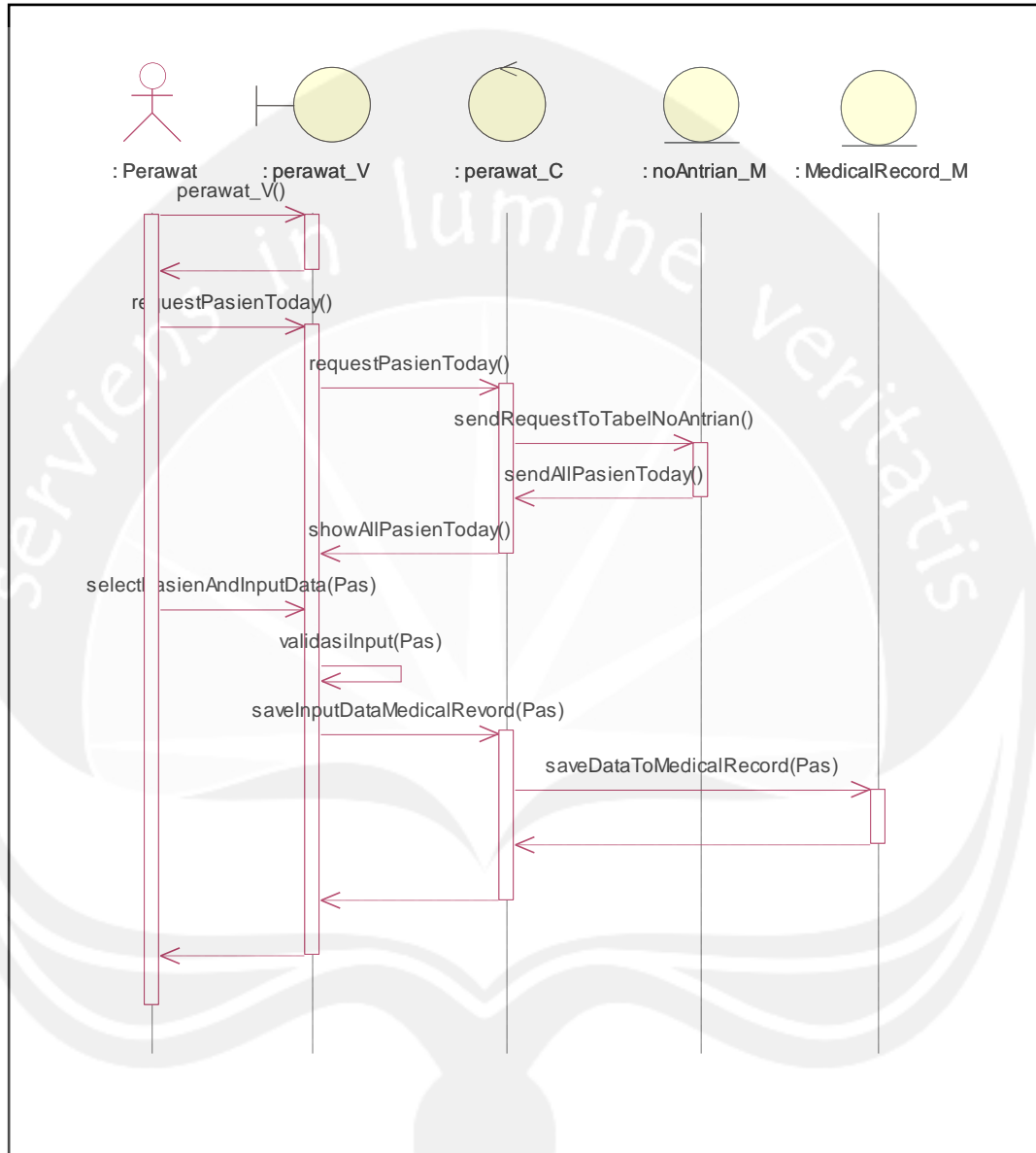
Pendaftaran Pasien Rawat Inap



Gambar 1.38 Sequence Diagram untuk Pendaftaran Pasien Rawat Inap

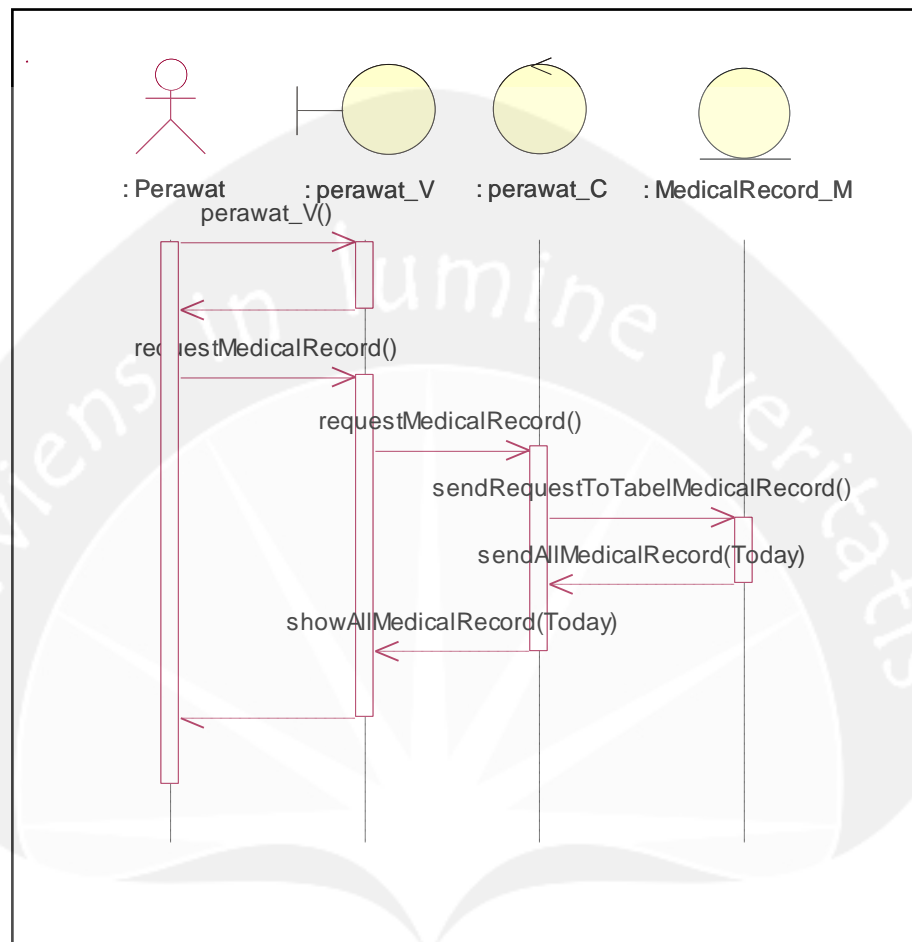
1.6.1.13 Keperawatan Pasien

Entry Pemeriksaan Awal



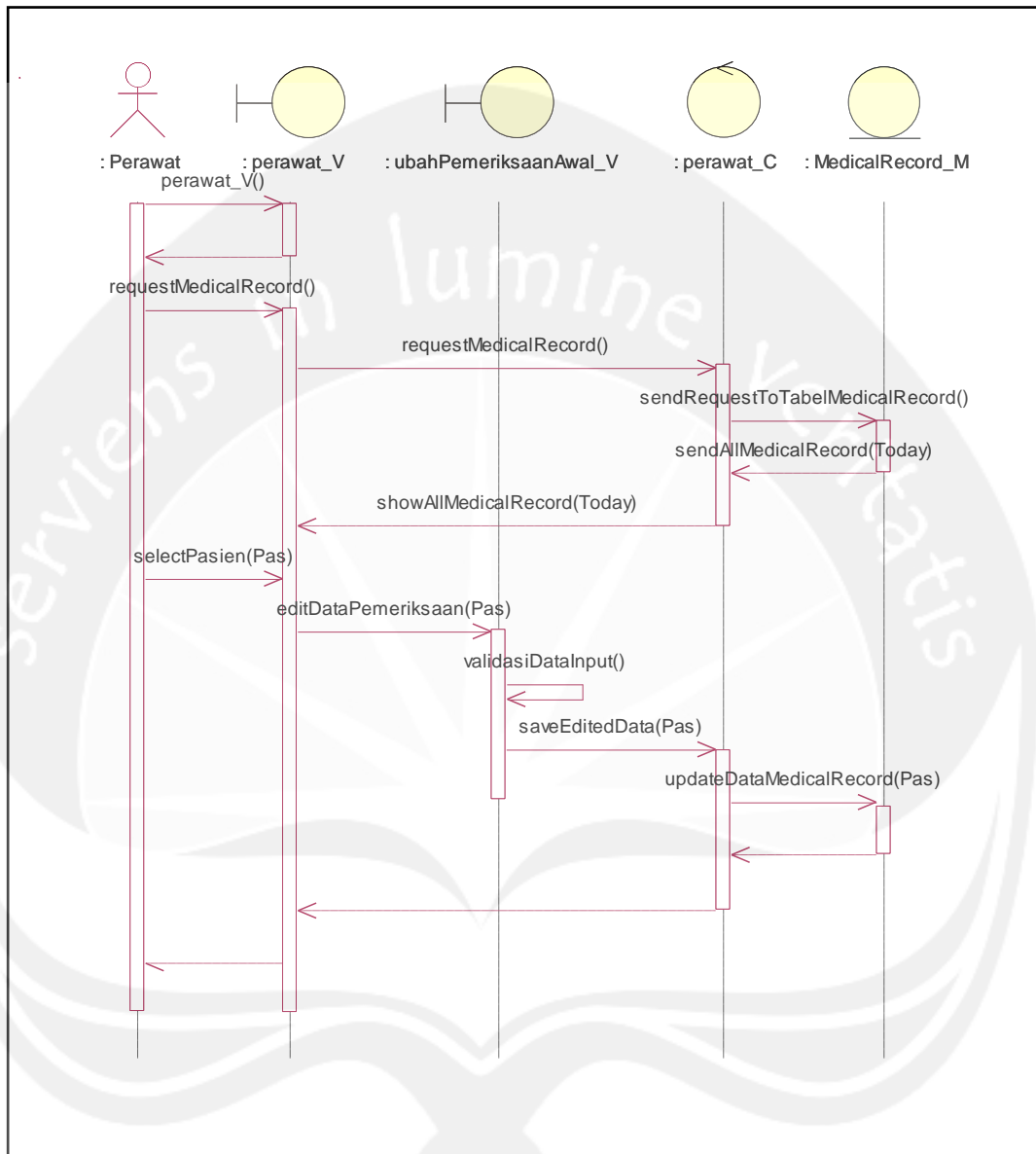
Gambar 1.39 Sequence Diagram untuk Entry Pemeriksaan Awal

1.6.1.14 Display Medical Record



Gambar 1.40 Sequence Diagram untuk Display Medical Record

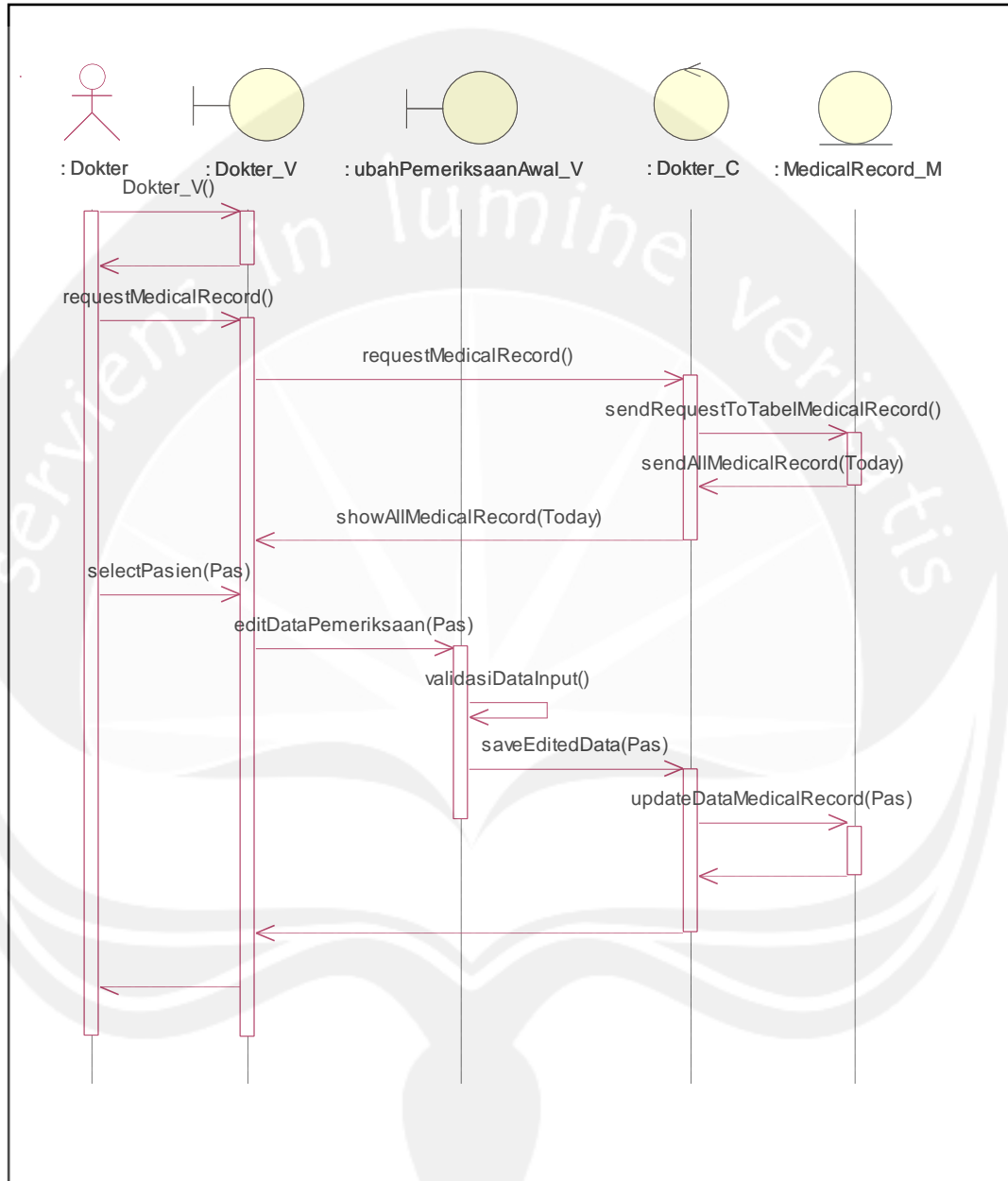
1.6.1.15 Edit Pemeriksaan Awal



Gambar 1.41 Sequence Diagram untuk Edit Pemeriksaan Awal

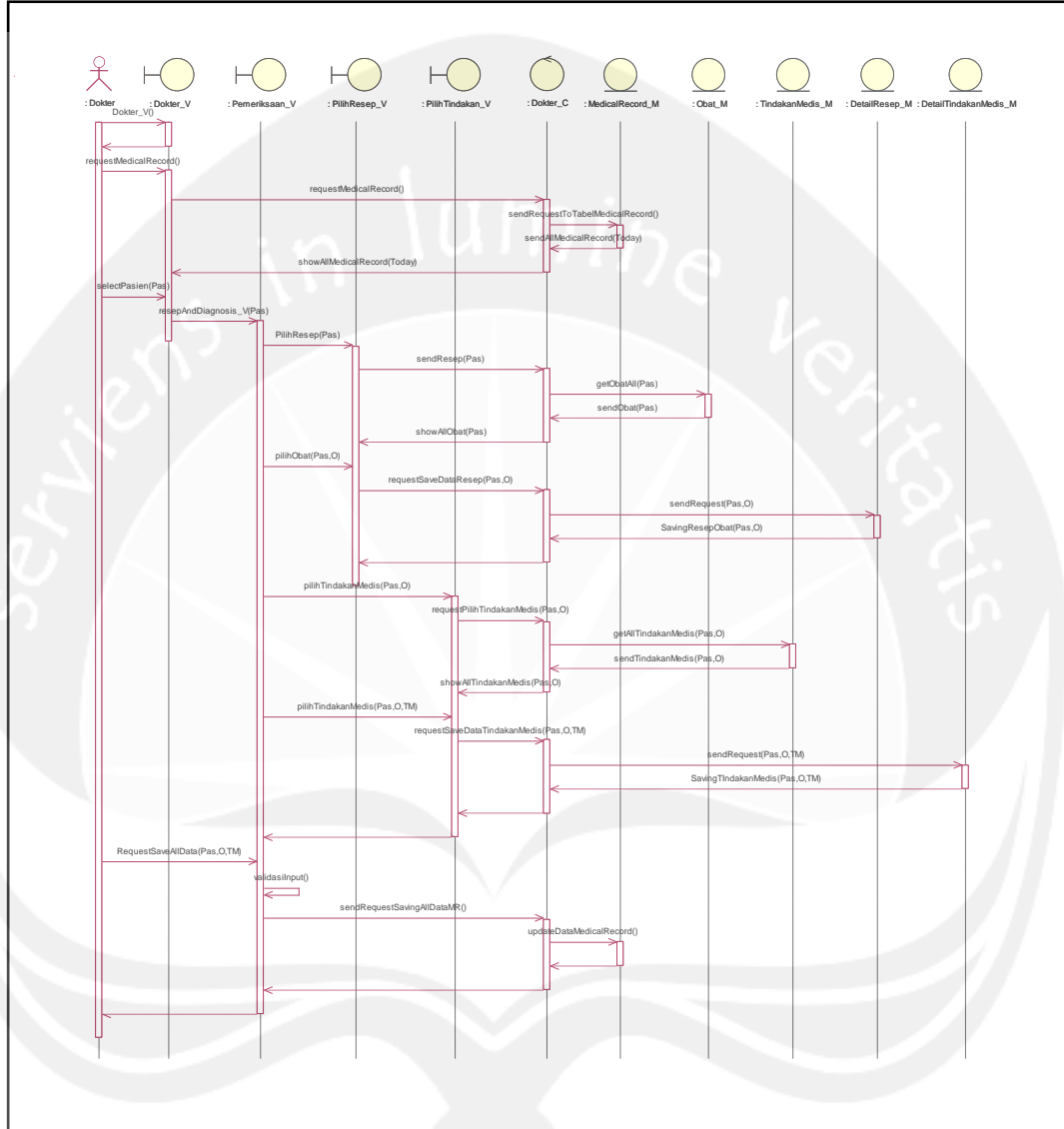
1.6.1.16 Pemeriksaan dan Diagnosis Pasien Dokter

Dokter: Pemeriksaan Awal Ubah Nilai Medical Record



Gambar 1.42 Sequence Diagram untuk Pemeriksaan Awal Ubah Nilai Medical Record

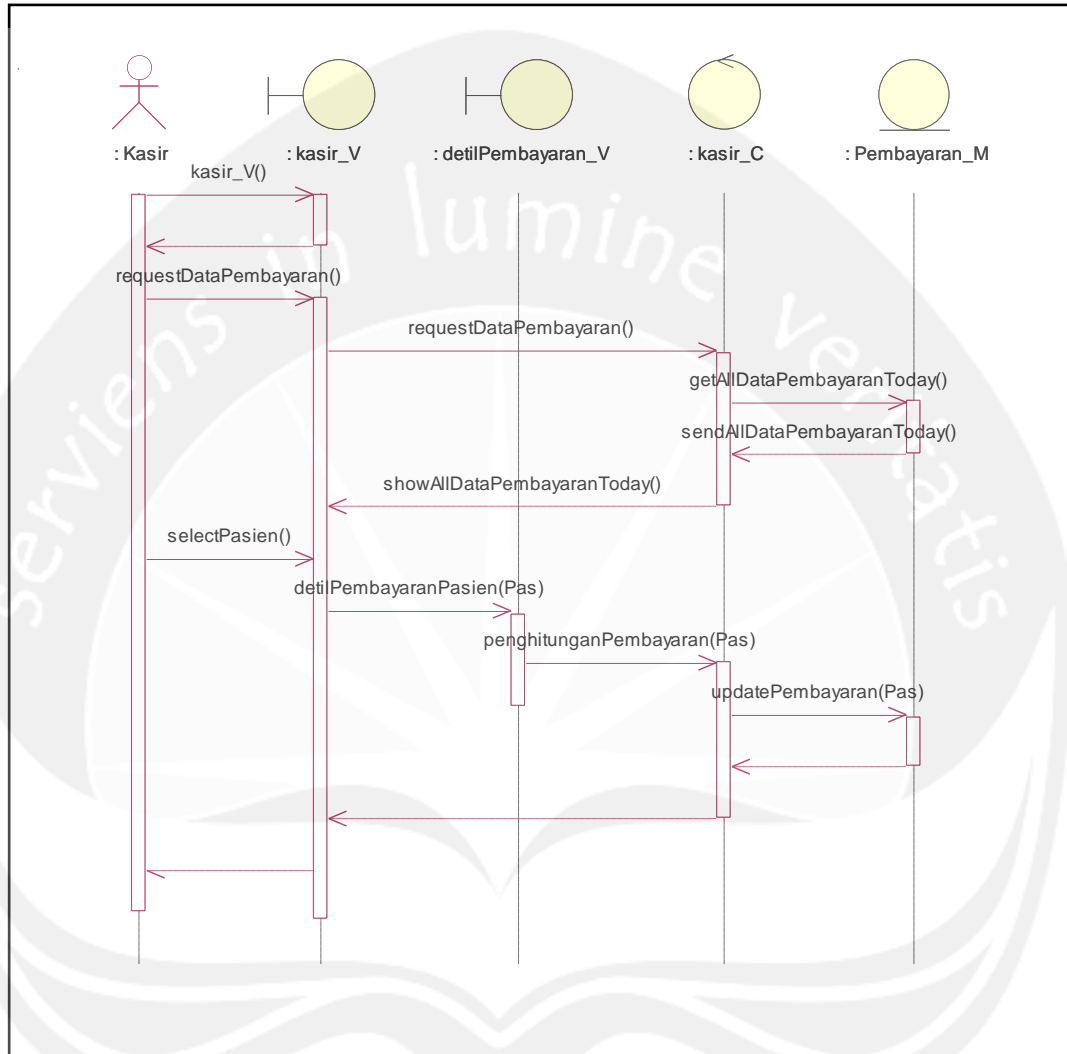
Dokter: Diagnosis Penyakit



Gambar 1.43 Sequence Diagram untuk Diagnosis Penyakit Pasien

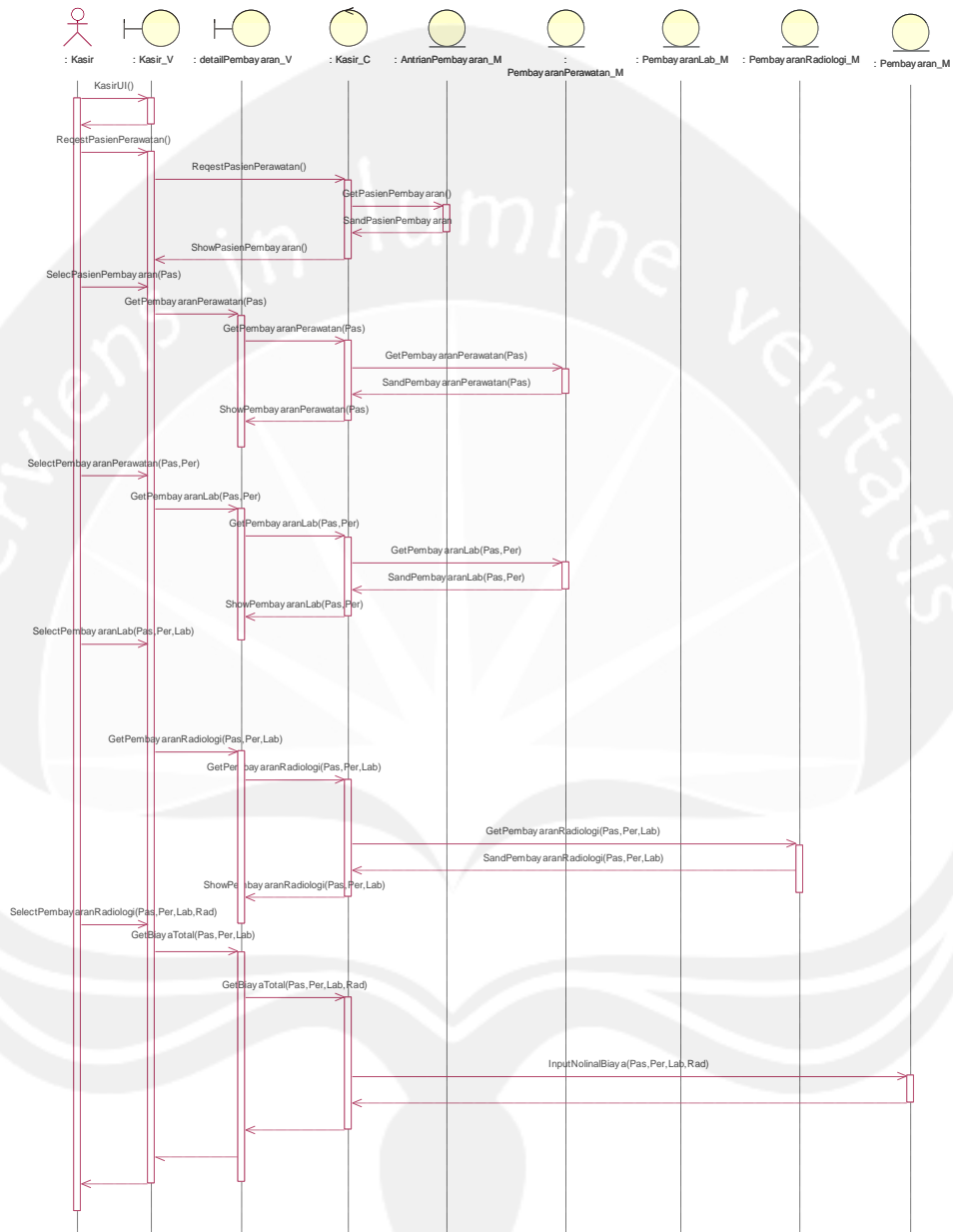
1.6.1.17 Kelola PembayaranRumah Sakit

Pembayaran Pemeriksaan



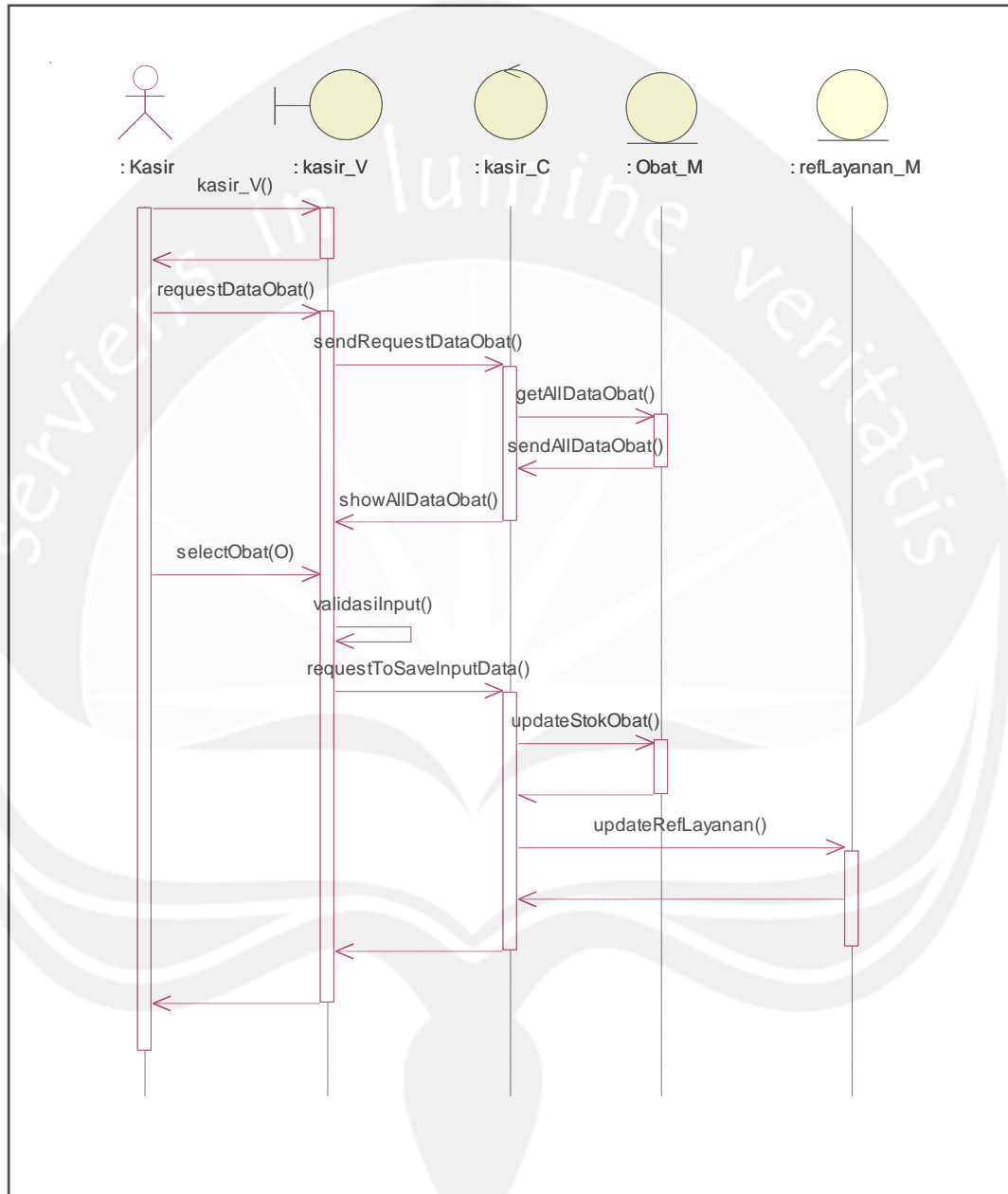
Gambar 1.44 Sequence Diagram untuk Pembayaran Pemeriksaan

1.6.1.18 Pembayaran Rawat Inap



Gambar 1.45 Sequence Diagram untuk Pembayaran Rawat Inap

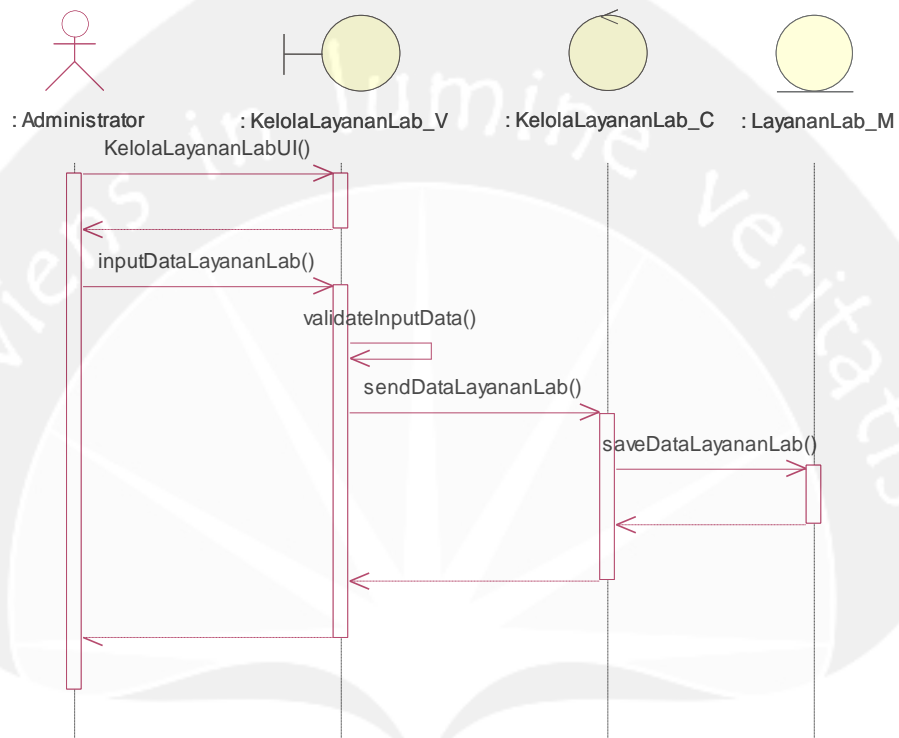
1.6.1.19 Pembelian Obat



Gambar 1.46 Sequence Diagram untuk Pembelian Obat

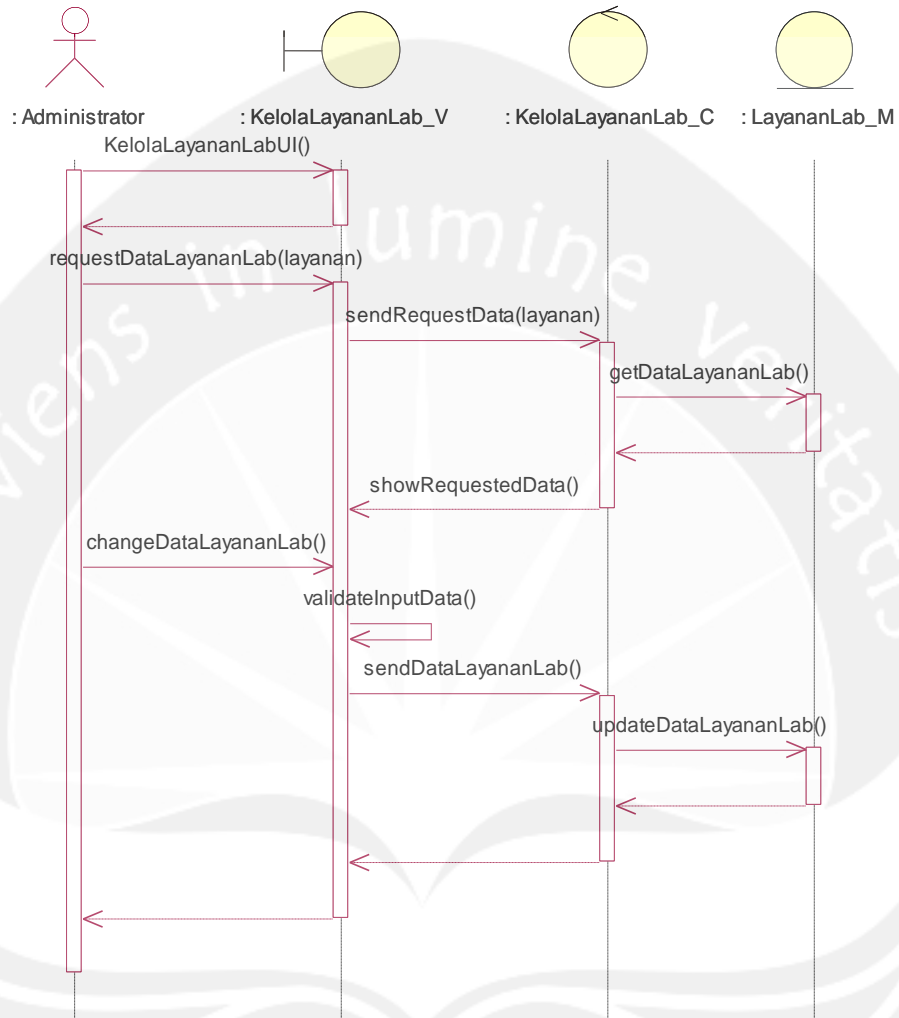
1.6.1.20 Kelola Layanan Lab

Entry Layanan Lab



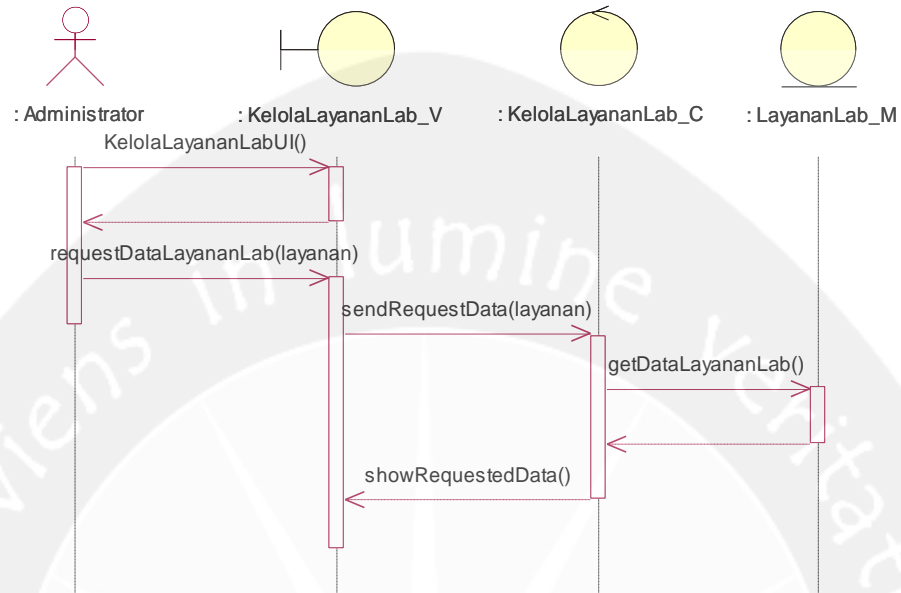
Gambar 1.47 Sequence Diagram untuk Entry Layanan Lab

Edit Layanan Lab



Gambar 1.48 Sequence Diagram untuk Edit Layanan Lab

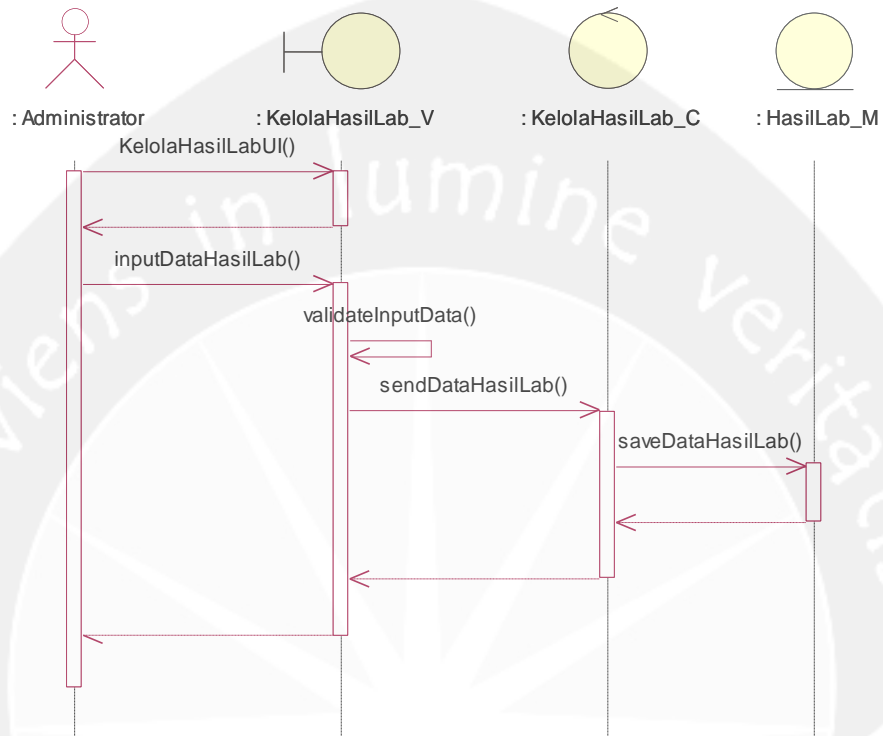
1.6.1.21 Display Layanan Lab



Gambar 1.49 Sequence Diagram untuk Display Layanan Lab

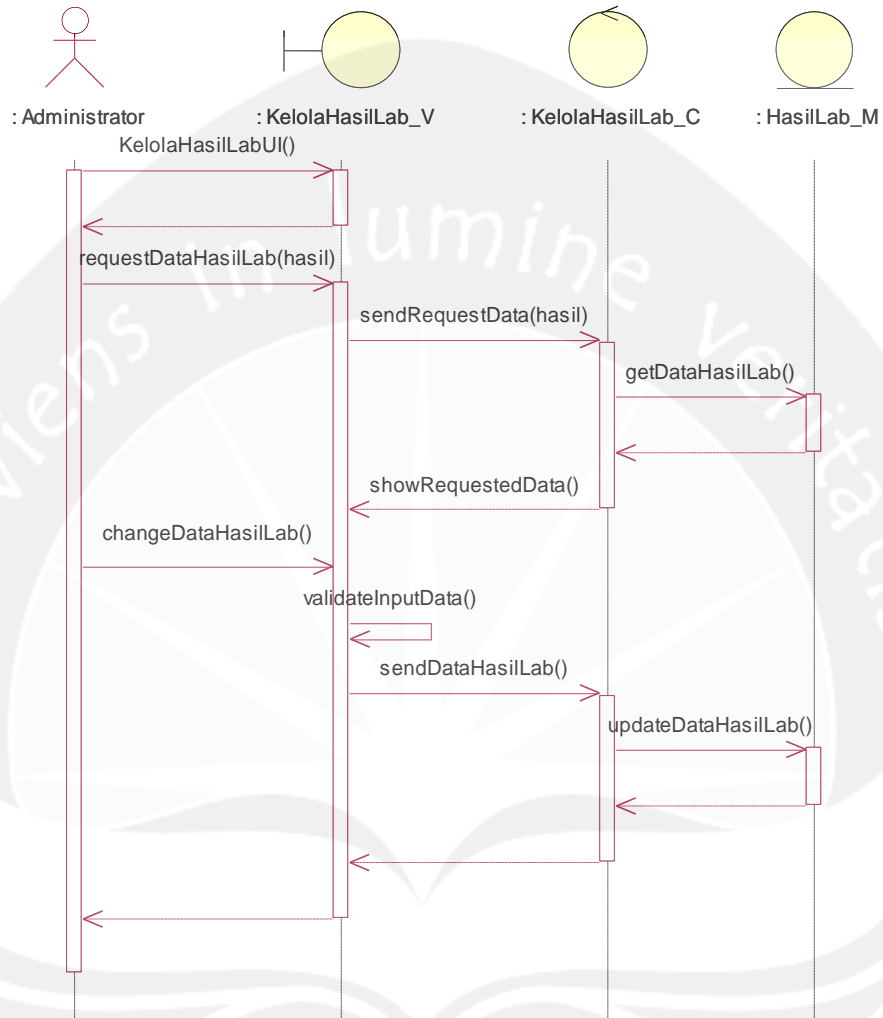
1.6.1.22 Kelola Hasil Lab

Entry Hasil Lab



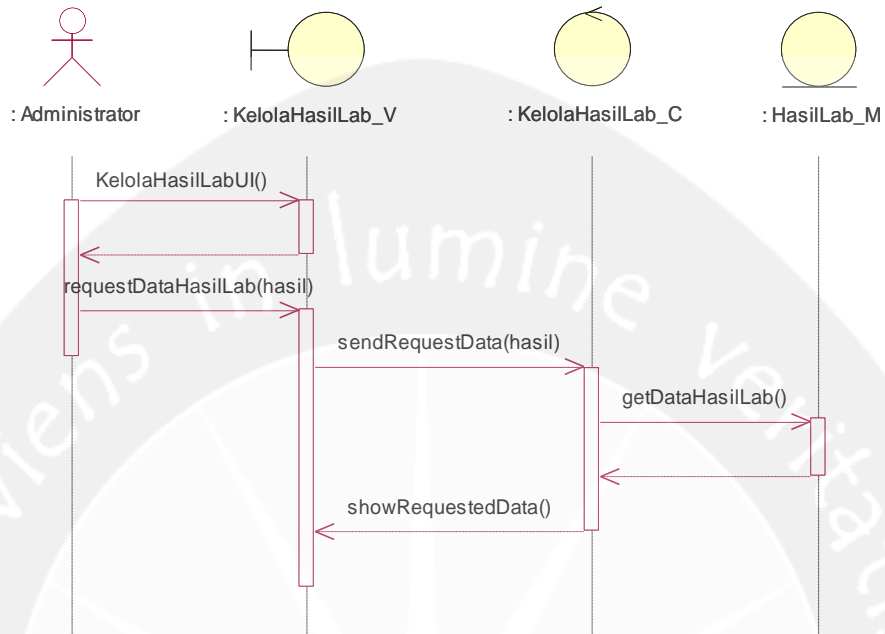
Gambar 1.50 Sequence Diagram untuk Entry Hasil Lab

Edit Hasil Lab



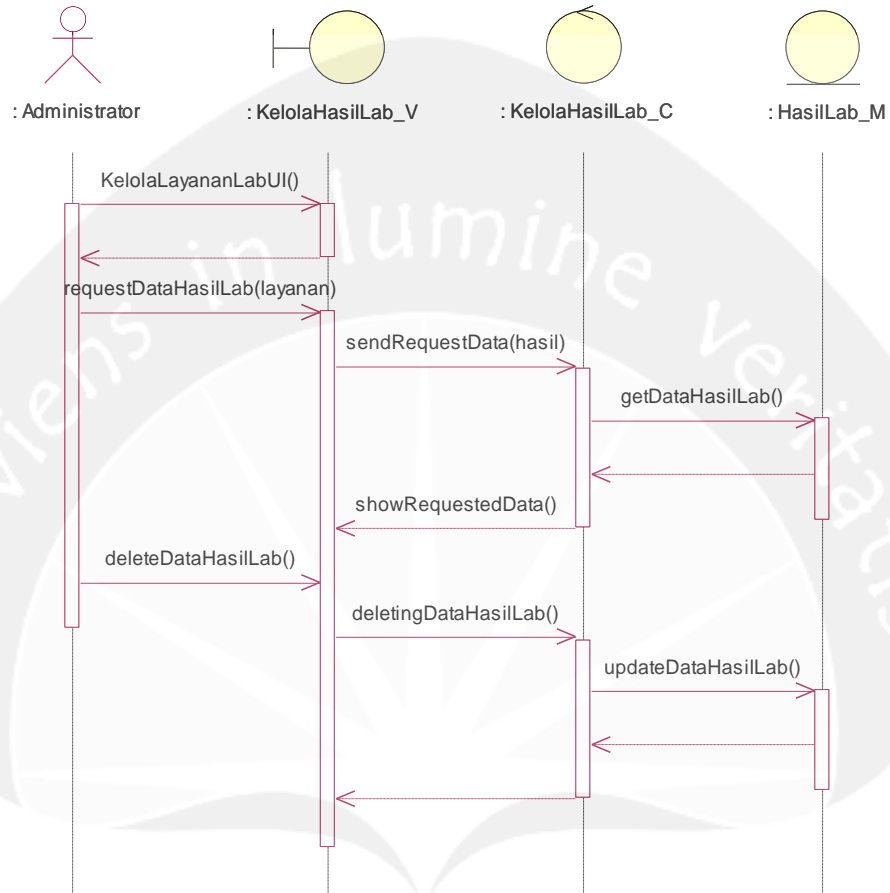
Gambar 1.51 Sequence Diagram untuk Edit Hasil Lab

DisplayHasil Lab



Gambar 1.52 Sequence Diagram untuk Display Hasil Lab

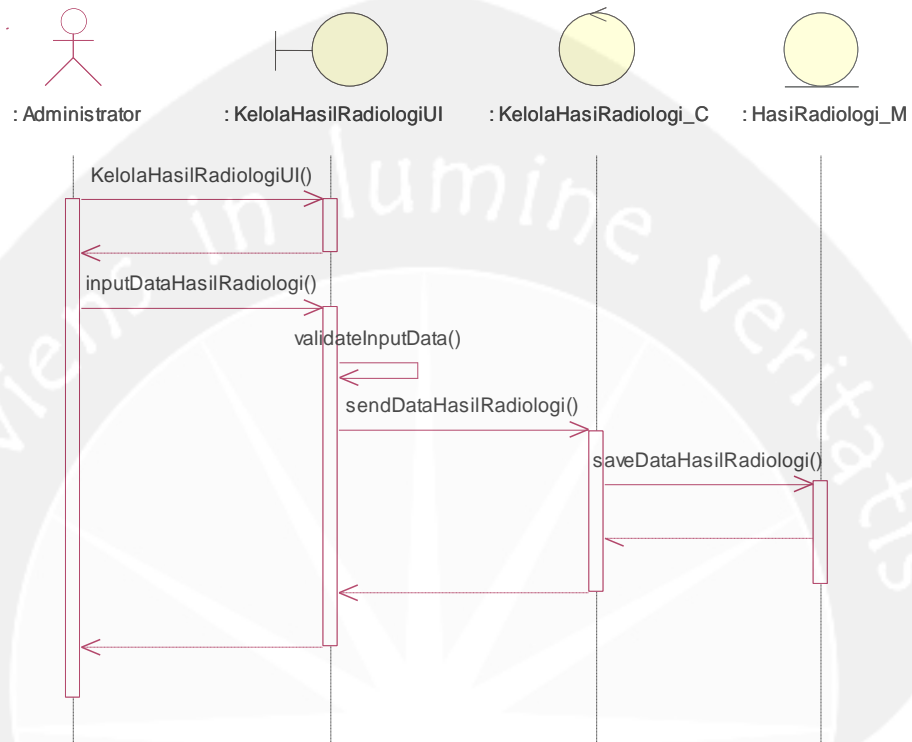
1.6.1.23 DeleteHasil Lab



Gambar 1.53 Sequence Diagram untuk Delete Hasil Lab

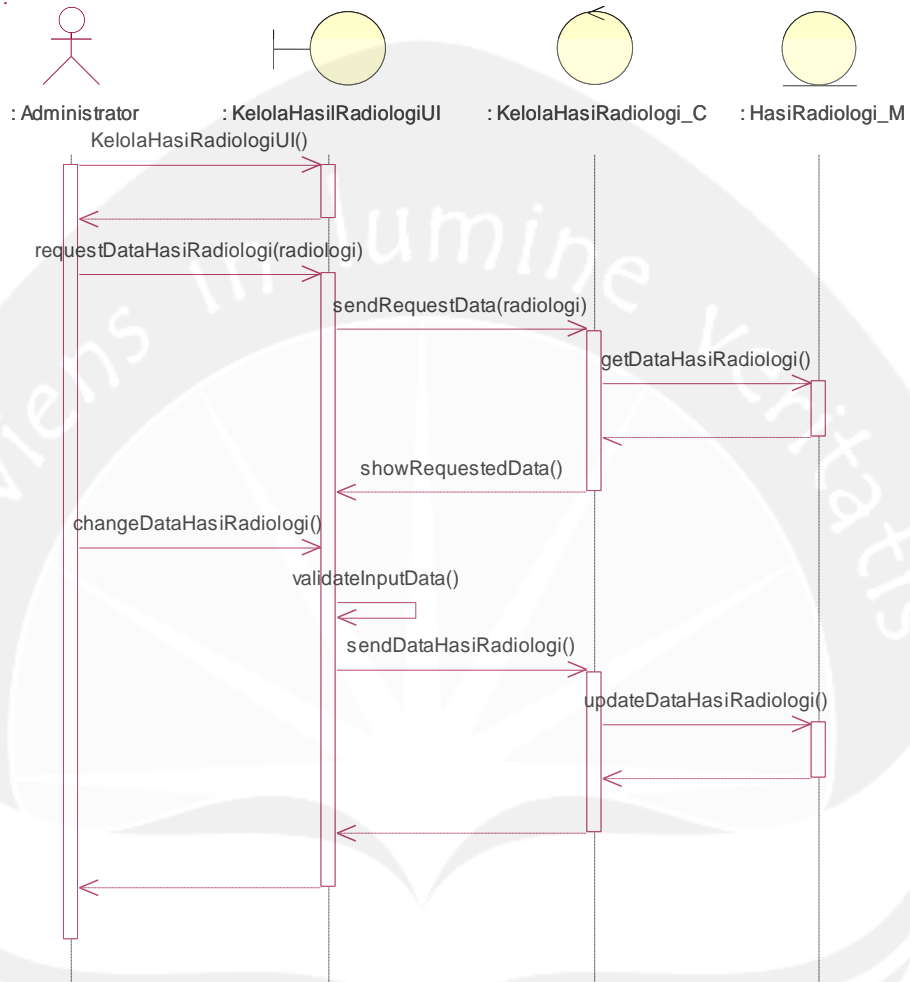
1.6.1.24 Kelola Radiologi

Entry Radiologi



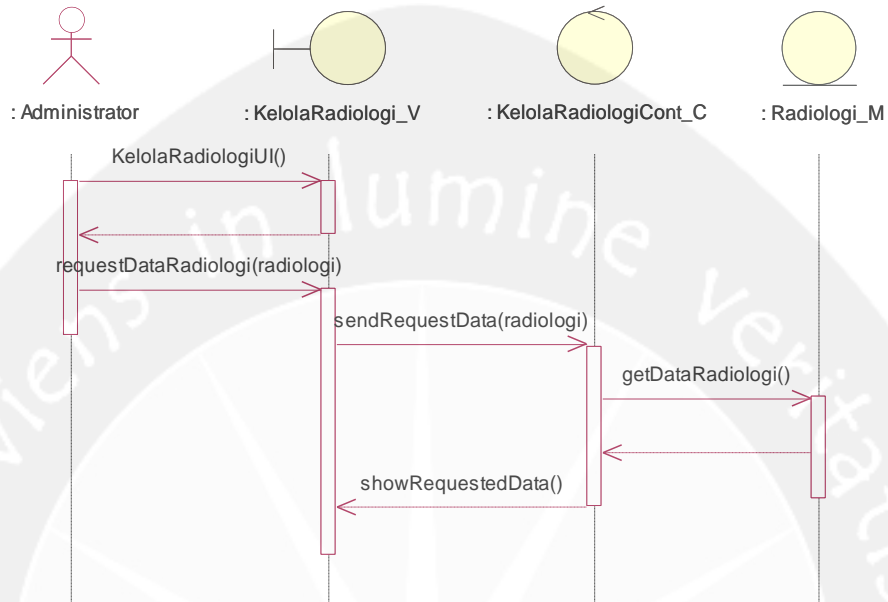
Gambar 1.54 Sequence Diagram untuk Entry Radiologi

Edit Radiologi



Gambar 1.55 Sequence Diagram untuk Edit Radiologi

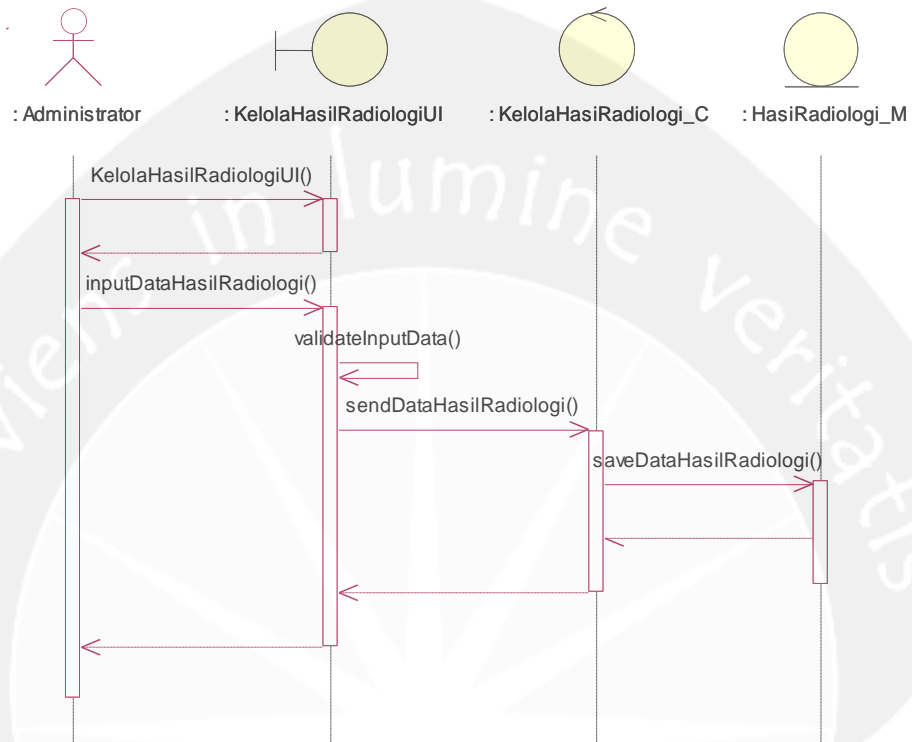
DisplayRadiologi



Gambar 1.56 Sequence Diagram untuk Display Radiologi

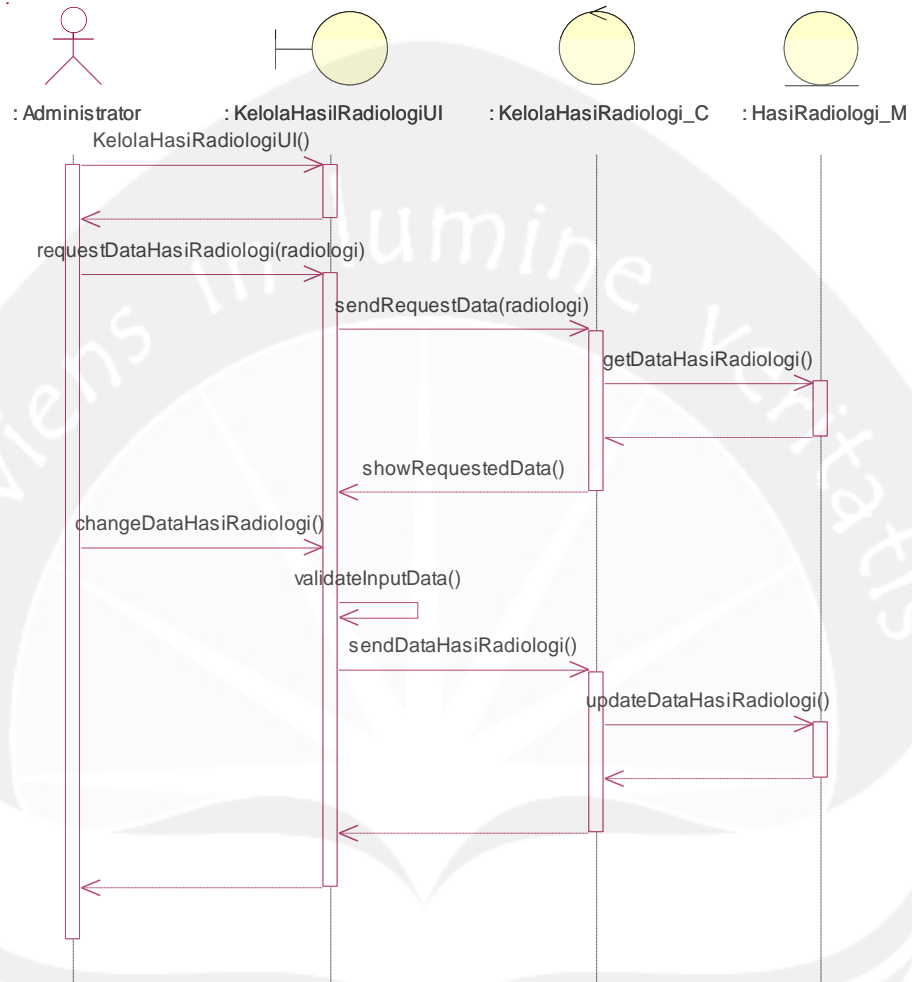
1.6.1.25 Kelola Hasil Radiologi

Entry Hasil Radiologi



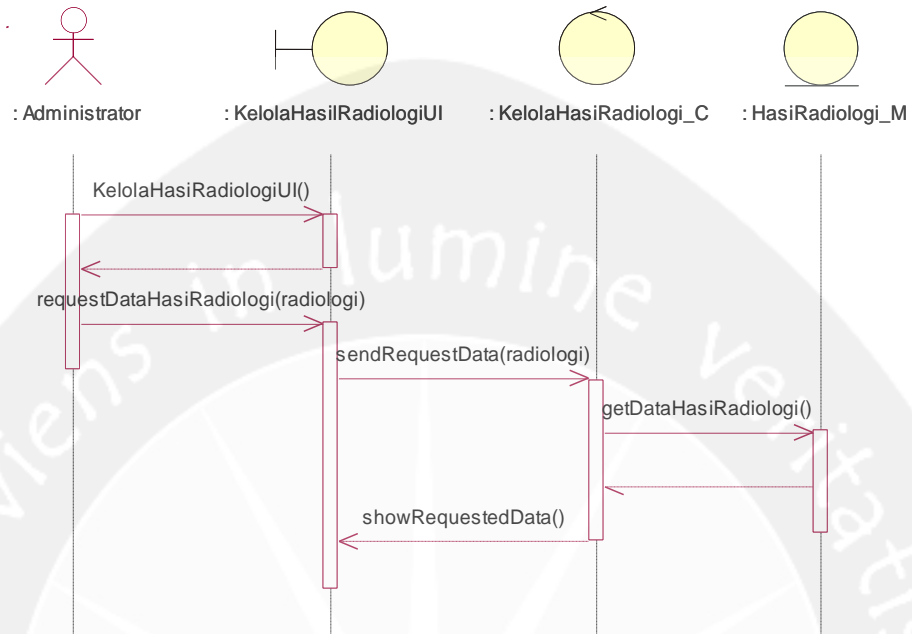
Gambar 1.57 Sequence Diagram untuk Entry Hasil Radiologi

Edit Hasil Radiologi



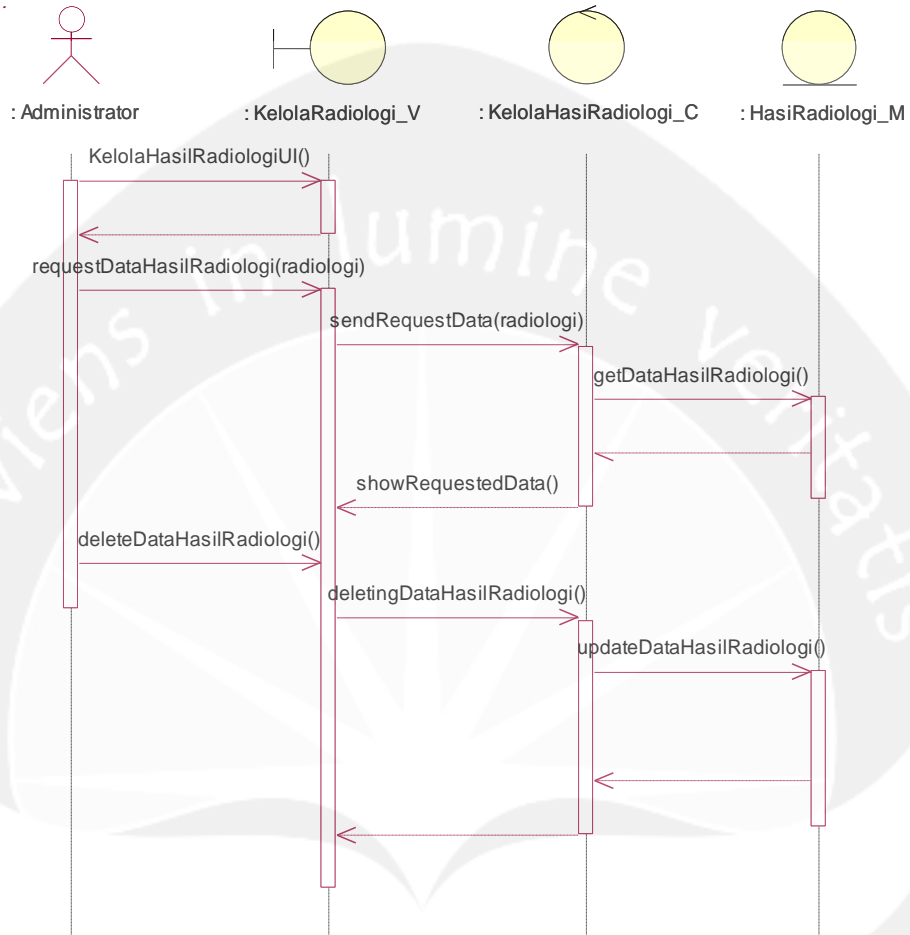
Gambar 1.58 Sequence Diagram untuk Edit Hasil Radiologi

DisplayHasil Radiologi



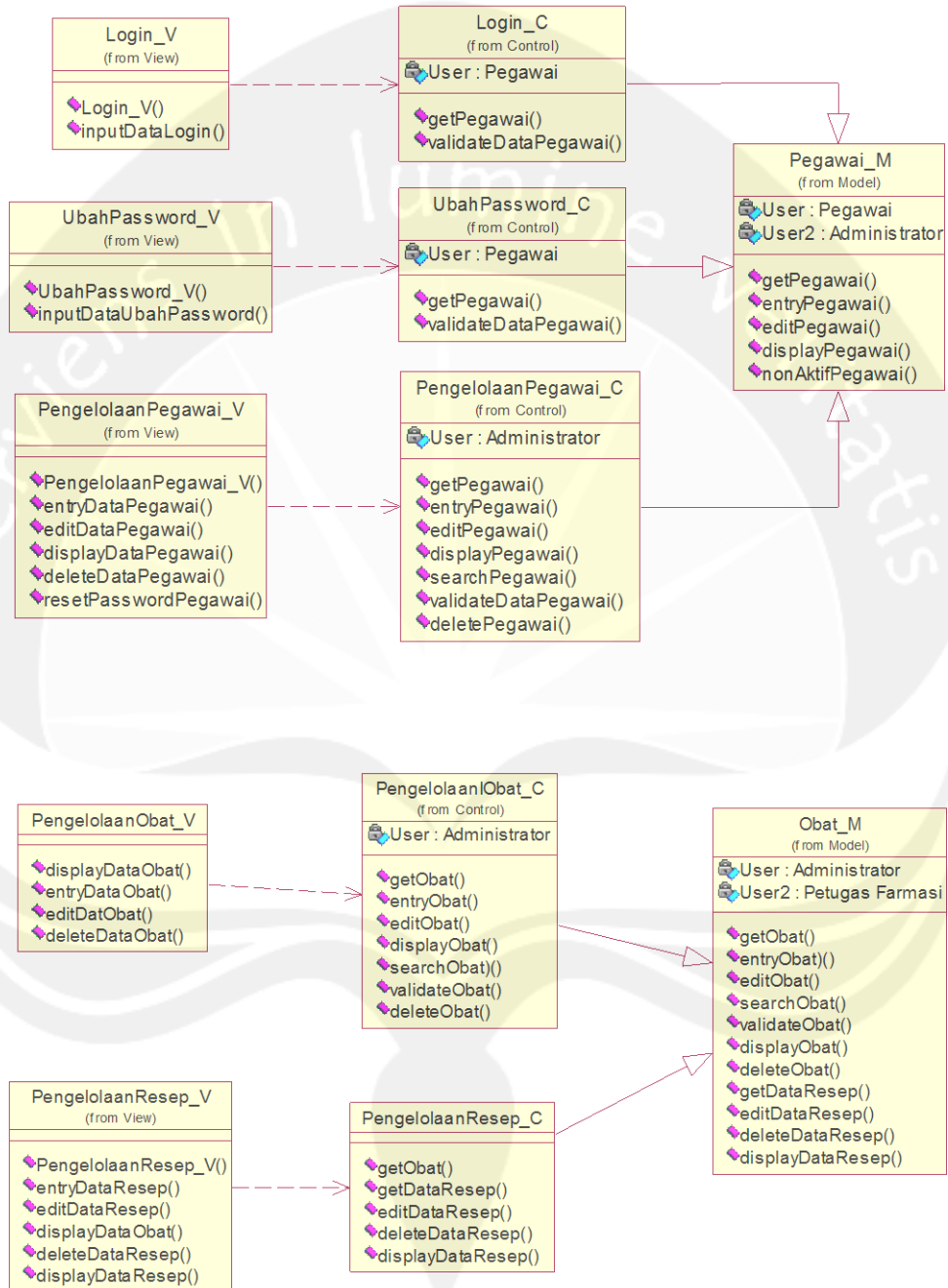
Gambar 1.59 Sequence Diagram untuk Display Hasil Radiologi

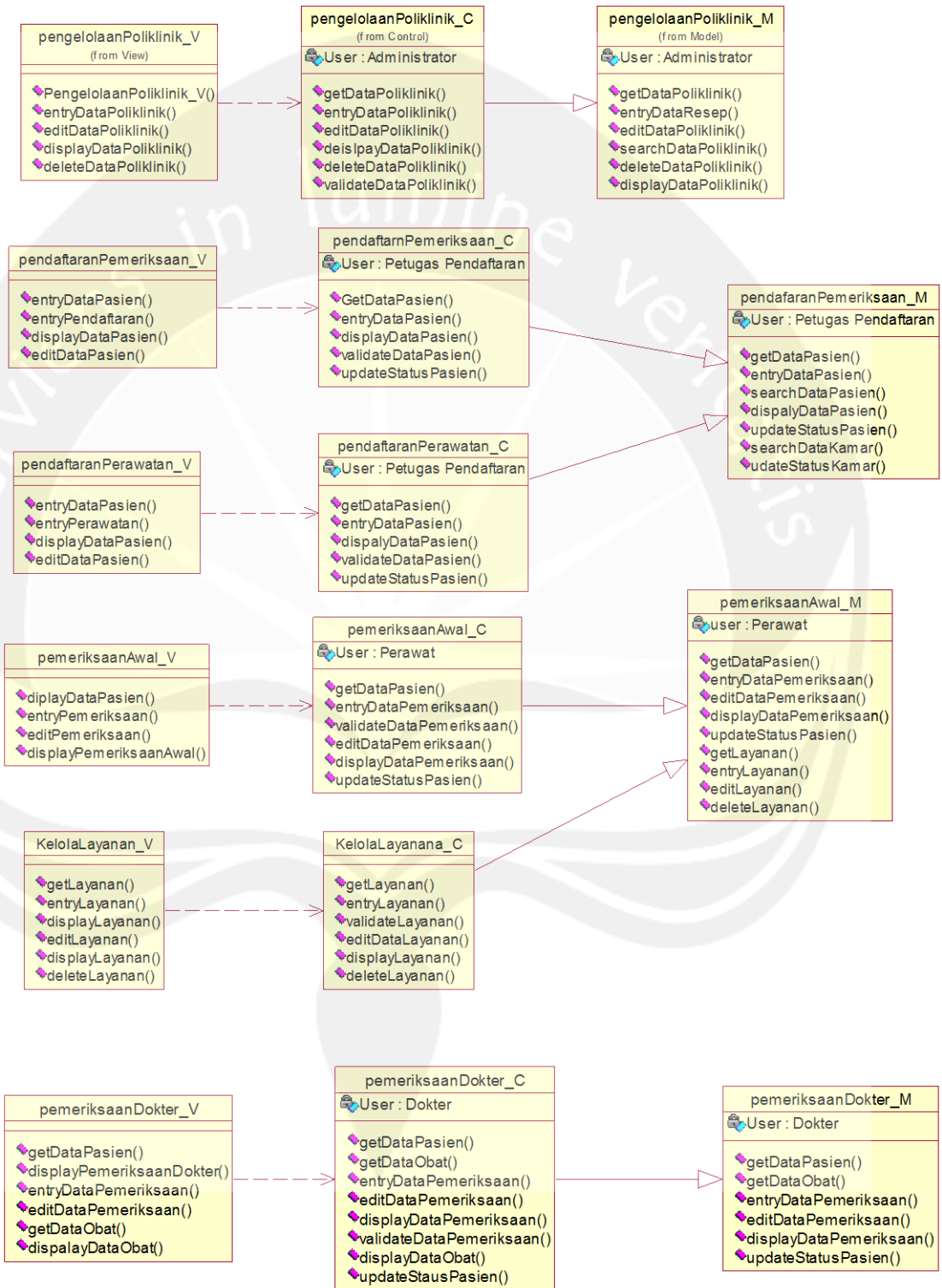
Delete Hasil Radiologi

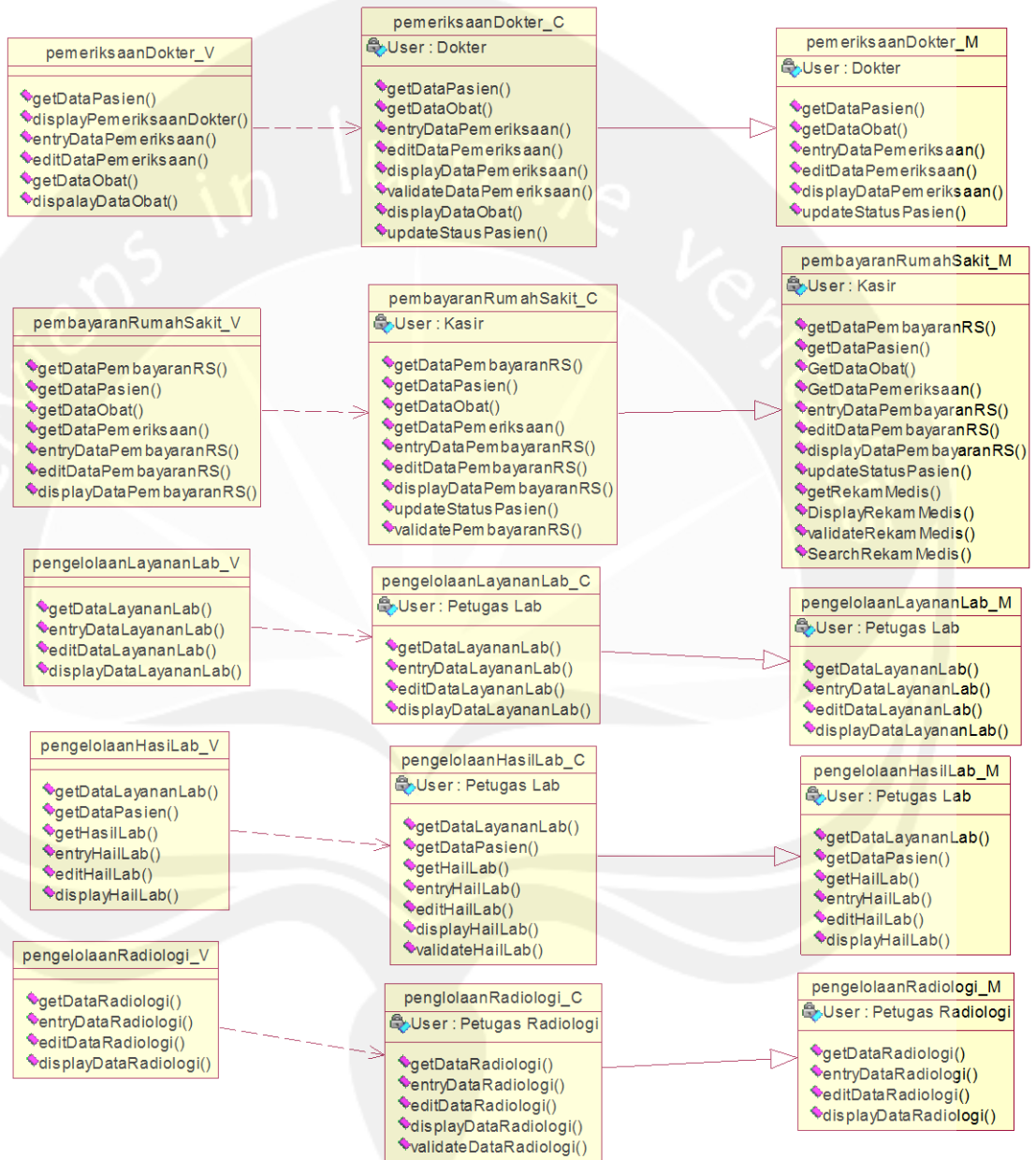


Gambar 1.60 Sequence Diagram untuk Delete Hasil Radiologi

1.6.2 Class Diagram







Gambar 1.61 Class Diagram

1.6.3 Spesifikasi Deskripsi Kelas Diagram

1.6.3.1 Spesifikasi Deskripsi Kelas Login_V

Login_V	<<View>>
+ Login_V() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+ InputDataLogin() Operasi ini untuk mendapatkan objek data user yang akan ditampilkan.	

1.6.3.2 Spesifikasi Deskripsi Kelas ChangePassword_V

ChangePassword_V	<<View>>
+ ChangePassword_V() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+InputDataPassword(User) Operasi ini digunakan untuk mengecek inputan user ke database dan digunakan untuk mengupdate password baru ke database.	

1.6.3.3 Spesifikasi Deskripsi Kelas Pegawai_V

Pegawai_V	<<View>>
+KelolaPegawai_V() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	

```
+entryPegawai (Pegawai)
```

Operasi ini untuk membuat data Pegawai yang akan dimasukkan ke database MasterPegawai.

```
+editPegawai (Pegawai)
```

Operasi ini untuk mengedit data Pegawai dari database Pegawai.

```
+displayPegawai (Pegawai) : Pegawai
```

Operasi ini untuk menampilkan data Pegawai dari database MasterPegawai.

```
+deleteUser (Pegawai) : Pegawai
```

Operasi ini untuk menghapus data Pegawai dari database MasterPegawai.

```
+ clearPassword (Pegawai) : Pegawai
```

Operasi ini untuk mereset password Pegawai dari database MasterPegawai.

1.6.3.4 Spesifikasi Deskripsi Kelas poliklinik_v

poliklinik_v	<<View>>
+Kelolapoliklinik_v () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+entryPoliklinik(poliklinik) Operasi ini untuk membuat data Poliklinik yang akan dimasukkan ke database Poliklinik.	
+editPoliklinik (poliklinik) Operasi ini untuk mengedit data Poliklinik dari database Poliklinik.	
+displayPoliklinik (poliklinik) : Poliklinik	

Operasi ini untuk menampilkan data Poliklinik dari database Poliklinik.

+deletePoliklinik (poliklinik): Poliklinik

Operasi ini untuk menghapus data Poliklinik dari database Poliklinik.

1.6.3.5 Spesifikasi Deskripsi Kelas

pengelolaanKaryawan_V

pengelolaanKaryawan _V	<< View>>
<p>+ pengelolaanKaryawan _V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+entryKaryawan (Karyawan) Operasi ini untuk membuat data Karyawan yang akan dimasukkan ke databaseKaryawan.</p> <p>+editKaryawan (Karyawan) Operasi ini untuk mengedit data Karyawan dari database Karyawan.</p> <p>+displayKaryawan (Karyawan) : Karyawan Operasi ini untuk menampilkan data Karyawan dari database Karyawan.</p> <p>+deleteKaryawan (Karyawan): Karyawan Operasi ini untuk menghapus data Karyawan dari database Karyawan.</p>	

1.6.3.6 Spesifikasi Deskripsi Kelas

pengelolaanJadwalDokter_V

pengelolaanJadwalDokter _V	<< View>>
-------------------------------	-----------

```

+ pengelolaanJadwalDokter _V ()
Default konstruktor, digunakan untuk inisialisasi
semua attribute dari kelas ini.
+entryJadwalDokter (JadwalDokter)
Operasi ini untuk membuat data JadwalDokter yang akan
dimasukkan ke databaseJadwalDokter.
+editJadwalDokter (JadwalDokter)
Operasi ini untuk mengeditdata JadwalDokter dari
database JadwalDokter.
+displayJadwalDokter (JadwalDokter) : JadwalDokter
Operasi ini untuk menampilkandata JadwalDokter dari
database JadwalDokter.
+deleteJadwalDokter (JadwalDokter): JadwalDokter
Operasi ini untuk menghapusdata JadwalDokter dari
database JadwalDokter.

```

1.6.3.7 Spesifikasi Deskripsi Kelas obat_V

obat _V	<< View>>
<pre> + obat _V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryObat (Obat) Operasi ini untuk membuat data Obat yang akan dimasukkan ke databaseObat. +editObat (Obat) Operasi ini untuk mengeditdata Obatdari database Obat. +displayObat (Obat) : Obat </pre>	

Operasi ini untuk menampilkan data Obat dari database Obat.

+deleteObat (Obat): Obat

Operasi ini untuk menghapus data Obat dari database Obat.

1.6.3.8 Spesifikasi Deskripsi Kelas bidangMedis_V

bidangMedis_V	<<View>>
<p>+Kelola bidangMedis_V() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+entryBidangMedis(bidangMedis) Operasi ini untuk membuat data Bidang Medis yang akan dimasukkan ke databaseBidangMedis.</p> <p>+editBidangMedis(bidangMedis) Operasi ini untuk mengedit data BidangMedis dari database BidangMedis.</p> <p>+displayBidangMedis(bidangMedis) : BidangMedis Operasi ini untuk menampilkan data BidangMedis dari database BidangMedis.</p> <p>+deleteBidangMedis (BidangMedis): BidangMedis Operasi ini untuk menghapus data BidangMedis dari database BidangMedis.</p>	

1.6.3.9 Spesifikasi Deskripsi Kelas obat_V

obat _V	<< View>>
<p>+ obat _V ()</p>	

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+entryObat (Obat)

Operasi ini untuk membuat data Obat yang akan dimasukkan ke databaseObat.

+editObat (Obat)

Operasi ini untuk mengeditdata Obatdari database Obat.

+displayObat (Obat) : Obat

Operasi ini untuk menampilkandata Obat dari database Obat.

+deleteObat (Obat): Obat

Operasi ini untuk menghapusdata Obat dari database Obat.

1.6.3.10 Spesifikasi Deskripsi Kelas pasien_V

pasien _V	<< View>>
-----------	-----------

+ pasien _V ()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+entrypasien (pasien)

Operasi ini untuk membuat data pasien yang akan dimasukkan ke databasepasien.

+editpasien (pasien)

Operasi ini untuk mengeditdata pasien dari database pasien.

+displaypasien (pasien) : pasien

Operasi ini untuk menampilkandata pasien dari

Program Studi Teknik Informatika	DPPL – SIRS	85/161
----------------------------------	-------------	--------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

database pasien.
+deletepasien (pasien): pasien
Operasi ini untuk menghapusdata pasien dari database pasien.

1.6.3.11 Spesifikasi Deskripsi Kelas kamar_V

kamar_V	<< View>>
<pre>+kamar_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entrykamar (kamar) Operasi ini untuk membuat data pasien yang akan dimasukkan ke databasekamar. +editkamar (kamar) Operasi ini untuk mengeditdata pasien dari database kamar. +displaykamar (kamar) : kamar Operasi ini untuk menampilkandata pasien dari database pasien. +deletekamar (kamar): kamar Operasi ini untuk menghapusdata kamardari database kamar.</pre>	

1.6.3.12

1.6.3.13 Spesifikasi Deskripsi Kelas kelasRawatInap_V

pengelolaan kelasRawatInap_V	<< View>>
<pre>+ pengelolaan kelasRawatInap_V ()</pre>	

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+entrykelasRawatInap (kelasRawatInap)

Operasi ini untuk membuat data PaketKecantikan yang akan dimasukkan ke databasekelasRawatInap.

+editkelasRawatInap (kelasRawatInap)

Operasi ini untuk mengeditdata kelasRawatInapdari database kelasRawatInap.

+displaykelasRawatInap (kelasRawatInap) :
PaketKecantikan

Operasi ini untuk menampilkandata kelasRawatInapdari database kelasRawatInap.

+deletekelasRawatInap (kelasRawatInap):
kelasRawatInap

Operasi ini untuk menghapusdata kelasRawatInapdari database kelasRawatInap.

1.6.3.14 Spesifikasi Deskripsi Kelas pendaftaran_V

pendaftaran _V	<< View>>
<p>+pendaftaran _V ()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+entryPasienPeriksa(pasien)</p> <p>Operasi ini untuk membuat data pasien periksayang akan dimasukkan ke databaseMedical Record.</p> <p>+entryPasienPerawatan(pasien)</p> <p>Operasi ini untuk membuat data pasien RawatInapyang akan dimasukkan ke databaseMedical Record.</p>	

1.6.3.15 Spesifikasi Deskripsi Kelas perawat_V

perawat_V	<< View>>
<pre>+ perawat_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryPemeriksaanAwal(pasien) Operasi ini untuk membuat data pasien periksa yang akan dimasukkan ke databaseMedical Record. +displayMedicalRecord (pasien): pasien Operasi ini untuk menampilkan data pasien periksa dari database Medical Record + editPemeriksaanAwal(pasien) Operasi ini untuk untuk mengedit data pasien periksa dari databaseMedical Record.</pre>	

1.6.3.16 Spesifikasi Deskripsi Kelas Dokter_V

Dokter_V	<< View>>
<pre>+ Dokter_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +updatePemeriksaanAwal(pasien) Operasi ini untuk mengubah/memperbaharuidata pemeriksaan awal pasien periksa dari perawat yang ada dalam databaseMedical Record. +diagnosisPenyakit(pasien) Operasi ini untuk membuat data pemeriksaan pasien yang antara lain berisi diagnose, resep obat, tindakan</pre>	

medis yang dilakukan, konsultasi yang kemudian akan dimasukkan ke database Medical Record.

1.6.3.17 Spesifikasi Deskripsi Kelas `apoteker_V`

<code>apoteker_V</code>	<< View >>
<pre>+ apoteker_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +displayResepDenganStatus(pasien) Operasi ini untuk menampilkan data resep dengan status seperti Lunas, Sedang Diproses, Belum Diproses, atau Sudah Diproses dari pasien yang ada dalam database Resep. +displayDetilResep(pasien) Operasi ini untuk menampilkan detil dari resep yang dipilih untuk melihat obat apa saja yang dibutuhkan. Untuk mendapatkan detil resep maka database yang digunakan juga bukan hanya Resep tapi juga menggunakan database detil resep yang mengacu pada nomor resep sesuai yang dibawa pasien. +updateCekObat(pasien) Operasi ini untuk memberikan status pada obat dan resep. Saat setelah pengambilan obat dari bagian apoteker. Bagian dari operasi cek obat ini antara lain mengecek obat apa saja dari no resep yang dibawa oleh system selain itu juga melakukan perubahan</pre>	

status dari belum diproses, menjadi sedang diproses, dan ketika selesai menjadi selesai diproses.

1.6.3.18 Spesifikasi Deskripsi Kelas kasir_V

kasir_V	<< View>>
<pre>+kasir_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +pembayaranKecantikan(pasien) Operasi ini untuk melakukan penghitungan biaya perawatan kecantikan dari database Antrian Kecantikan. +pembayaranRumah Sakit(pasien) Operasi ini untuk melakukan penghitungan biaya resep, tindakan medis, konsultasi, dan lainnya dari pasien pemeriksaan yang berasal dari database Medical Record.</pre>	

1.6.3.19 Spesifikasi Deskripsi Kelas Lab_V

Lab_V	<< View>>
<pre>+Lab _V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + PengelolaanLayananLab (Lab) Operasi ini untuk melakukan pengelolaan layanan</pre>	

Labdari database Layanan.
 + PengelolaanHasilLab(Lab)
 Operasi ini untuk melakukan pengelolaan Hasil Lab dari database MedicalRecord.

1.6.3.20 Spesifikasi Deskripsi Kelas Lab_V

Radiologi_V	<< View>>
<pre>+Radiologi_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. + PengelolaanRadiologi (radiologi) Operasi ini untuk melakukan pengelolaan Radiologi dari database Pemeriksaan. + PengelolaanHasil Radiologi radiologi) Operasi ini untuk melakukan pengelolaan Radiologi dari database MedicalRecord.</pre>	

1.6.3.21 Spesifikasi Deskripsi Kelas login_C

login_C	<<Control>>
<pre>+login_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getPegawai() : Pegawai Operasi ini untuk mengambil data Pegawaidari database MasterPegawai.</pre>	

+validateDataPegawai() : Pegawai

Operasi ini untuk mengecek kebenaran username dan password yang diinputkan Pegawai berdasarkan database MasterPegawai.

1.6.3.22 Spesifikasi Deskripsi Kelas UbahPassword_C

UbahPassword_C	<<Control>>
+ UbahPassword_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+GetPassword() Operasi ini untuk mengupdate password baru ke dalam database.	
+ValidateDataPegawai() : Pegawai Operasi ini untuk mengecek kebenaran username dan password yang diinputkan Pegawai berdasarkan database Pegawai.	

1.6.3.23 Spesifikasi Deskripsi Kelas pengelolaanPegawai_C

pengelolaanPegawai_C	<<Control>>
+ pengelolaanPegawai_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+AddPegawai(Pegawai)	

Program Studi Teknik Informatika	DPPL – SIRS	92/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Operasi ini untuk membuat data Pegawai yang akan dimasukkan ke database Pegawai.

+EditPegawai(Pegawai)

Operasi ini untuk mengedit data Pegawai dari database Pegawai.

+DisplayPegawai(Pegawai)

Operasi ini untuk menampilkan data Pegawai dari database Pegawai.

+DeletePegawai(Pegawai): Pegawai

Operasi ini untuk menghapus data Pegawai dari database Pegawai.

+getPegawai() : Pegawai

Operasi ini untuk mengambil data Pegawai dari database Pegawai.

+ValidatePegawai() : Pegawai

Operasi ini untuk mengecek kebenaran username dan password yang diinputkan Pegawai berdasarkan database Pegawai.

+ClearPassword()

Operasi ini untuk mereset password Pegawai ke dalam database.

1.6.3.24 Spesifikasi Deskripsi Kelas

pengelolaanPoliklinik_C

pengelolaanPoliklinik_C	<<Control>>
+pengelolaanPoliklinik_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	

Program Studi Teknik Informatika	DPPL – SIRS	93/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

+getPoliklinik() : Poliklinik
Operasi ini untuk mengambil data Poliklinik dari
database Poliklinik.
+entryPoliklinik (Poliklinik)
Operasi ini untuk membuat data Poliklinik yang akan
dimasukkan ke databasePoliklinik.
+editPoliklinik (Poliklinik)
Operasi ini untuk mengeditdata Poliklinik dari
database Poliklinik.
+displayPoliklinik (Poliklinik)
Operasi ini untuk menampilkan data Poliklinik dari
database Poliklinik.
+deletePoliklinik (Poliklinik): Poliklinik
Operasi ini untuk menghapusdata Poliklinik dari
database Poliklinik.

```

1.6.3.25 Spesifikasi Deskripsi Kelas `jadwalDokter_C`

<code>jadwalDokter_C</code>	<code><<Control>></code>
<pre> +jadwalDokter_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getJadwalDokter() : JadwalDokter Operasi ini untuk mengambil data JadwalDokter dari database JadwalDokter. +entryJadwalDokter (JadwalDokter) Operasi ini untuk membuat data JadwalDokter yang akan dimasukkan ke databaseJadwalDokter. +editJadwalDokter (JadwalDokter) </pre>	

Operasi ini untuk mengedit data JadwalDokter dari database JadwalDokter.

+displayJadwalDokter (JadwalDokter)

Operasi ini untuk menampilkan data JadwalDokter dari database JadwalDokter.

+deleteJadwalDokter (JadwalDokter): JadwalDokter

Operasi ini untuk menghapus data JadwalDokter dari database JadwalDokter.

1.6.3.26 Spesifikasi Deskripsi Kelas pengelolaanObat_C

pengelolaanObat_C	<<Control>>
-------------------	-------------

+pengelolaanObat_C()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+getObat() : Obat

Operasi ini untuk mengambil data Obat dari database Obat.

+entryObat (Obat)

Operasi ini untuk membuat data Obat yang akan dimasukkan ke database Obat.

+editObat (Obat)

Operasi ini untuk mengedit data Obat dari database Obat.

+displayObat (Obat)

Operasi ini untuk menampilkan data Obat dari database Obat.

+deleteObat (Obat): Obat

Operasi ini untuk menghapus data Obat dari database Obat.

1.6.3.27 Spesifikasi Deskripsi Kelas bidangMedis_C

bidangMedis_C	<<Control>>
<pre> +bidangMedis_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryBidangMedis(BidangMedis) Operasi ini untuk membuat data BidangMedis yang akan dimasukkan ke databaseBidangMedis. +editBidangMedis(BidangMedis) Operasi ini untuk mengeditdata BidangMedis dari database BidangMedis. +displayBidangMedis(BidangMedis) Operasi ini untuk menampilkan data BidangMedisdari database BidangMedis. +getBidangMedis() : BidangMedis Operasi ini untuk mengambil data BidangMedisdari database BidangMedis. +deleteBidangMedis(BidangMedis): BidangMedis Operasi ini untuk menghapusdata BidangMedisdari database BidangMedis. </pre>	

1.6.3.28 Spesifikasi Deskripsi Kelas kamar_C

kamar_C	<<Control>>
<pre> + kamar _C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. </pre>	

```

+getkamar() : kamar
Operasi ini untuk mengambil data kamar dari database kamar.
+entrykamar (kamar)
Operasi ini untuk membuat data kamar yang akan dimasukkan ke databasekamar.
+editkamar (kamar)
Operasi ini untuk mengeditdata kamar dari database kamar.
+displaykamar (kamar)
Operasi ini untuk menampilkan data kamar dari database kamar.
+deletekamar (kamar): kamar
Operasi ini untuk menghapusdata kamar dari database kamar.

```

1.6.3.29 Spesifikasi Deskripsi Kelas KelasRawatInap_C

kelasRawatInap_C	<<Control>>
<pre> + kelasRawatInap _C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +get kelasRawatInap () : kamar Operasi ini untuk mengambil data kelasRawatInap dari database kelasRawatInap. +entrykelasRawatInap (kelasRawatInap) Operasi ini untuk membuat data kelasRawatInap yang akan dimasukkan ke databasekelasRawatInap. +editkelasRawatInap (kelasRawatInap) </pre>	

Operasi ini untuk mengedit data kelasRawatInap dari database kelasRawatInap.

+display kelasRawatInap (kelasRawatInap)

Operasi ini untuk menampilkan data kelasRawatInap dari database kelasRawatInap.

+delete kelasRawatInap kelasRawatInap): kelasRawatInap

Operasi ini untuk menghapus data kamar dari database kelasRawatInap.

1.6.3.30 Spesifikasi Deskripsi Kelas pasien_C

pasien_C	<<Control>>
+pasien_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+getPasien() : Pasien Operasi ini untuk mengambil data Pasien dari database Pasien.	
+entryPasien (Pasien) Operasi ini untuk membuat data Pasien yang akan dimasukkan ke database Pasien.	
+editPasien (Pasien) Operasi ini untuk mengedit data Pasien dari database Pasien.	
+displayPasien (Pasien) Operasi ini untuk menampilkan data Pasien dari database Pasien.	
+deletePasien (Pasien): Pasien Operasi ini untuk menghapus data Pasien dari database Pasien.	

1.6.3.31 Spesifikasi Deskripsi Kelaspendaftaran_C

pendaftaran_C	<<Control>>
<pre>+pendaftaran_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getPasien() : Pasien Operasi ini untuk mengambil data Pasien dari database Pasien. +getDokter() : Dokter Operasi ini untuk mengambil data Dokter dari database Dokter. +getPoliklinik() : Poliklinik Operasi ini untuk mengambil data Poliklinik dari database Poliklinik. +getKamar() : Kamar Operasi ini untuk mengambil data Kamar dari database Kamar. +entryPasienPeriksa (Pasien) Operasi ini untuk memperbaharui data yang ada di Database Medical Record dengan Pasien yang periksa beserta Dokter dan Polikliniknya.</pre>	

1.6.3.32 Spesifikasi Deskripsi Kelasperawat_C

perawat_C	<<Control>>
<pre>+perawat_C() Default konstruktor, digunakan untuk inisialisasi</pre>	

Program Studi Teknik Informatika	DPPL – SIRS	99/161
<p>Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika</p>		

semua attribute dari kelas ini.

+getNoAntrian() : Pasien
Operasi ini untuk mengambil data No AntrianPasien dari database No Antrian.

+entryPemeriksaanAwal (Pasien)
Operasi ini untuk memperbaharuidata Medical Record yang akan dimasukkan ke databaseMedical Record.

+getMedicalRecord() : MedicalRecord
Operasi ini untuk mengambil data MedicalRecord Pasien dari database MedicalRecord.

+displayMedicalRecord (Pasien)
Operasi ini untuk menampilkan data MedicalRecord database Pasien.

+editPemeriksaanAwal (Pasien)
Operasi ini untuk memperbaharuidata Medical Record yang sudah ada didatabaseMedical Record.

1.6.3.33 Spesifikasi Deskripsi KelasDokter_C

Dokter_C	<<Control>>
<p>+Dokter_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+getObat() : Obat Operasi ini untuk mengambil dataObat dari database Obat.</p> <p>+getTindakanMedis() : TindakanMedis Operasi ini untuk mengambil dataTindakan Medis dari database Tindakan Medis.</p>	

Program Studi Teknik Informatika	DPPL – SIRS	100/161
<p>Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika</p>		

```

+getMedicalRecord() : MedicalRecord
Operasi ini untuk mengambil data MedicalRecord Pasien
dari database MedicalRecord.
+displayMedicalRecord (Pasien)
Operasi ini untuk menampilkan data MedicalRecord
database Pasien.
+updateMedicalRecord() : MedicalRecord
Operasi ini untuk mengubah data MedicalRecord Pasien
dari database MedicalRecord.

```

1.6.3.34 Spesifikasi Deskripsi Kelasapoteker_C

apoteker_C	<<Control>>
<pre> +apoteker_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getResepAllLunas() : Resep Operasi ini untuk mengambil dataResep dari database Resep yang memiliki status resep lunas. +getResepBelumDiproses() : Resep Operasi ini untuk mengambil data Resep dari database Resep yang memiliki status resep Belum diproses. +getResepAllSedangDiproses() : Resep Operasi ini untuk mengambil data Resep dari database Resep yang memiliki status resep sedang diproses. +getResepAllSelesaiDiproses() : Resep Operasi ini untuk mengambil data Resep dari database Resep yang memiliki status resep selesai diproses. + displayDetilResep (Pasien) </pre>	

Operasi ini untuk menampilkan data Detil Resep database Detil Resep.

+ updateCekObat(): Resep

Operasi ini untuk mengubah data status dari Detil Resep yang ada dalam database Detil Resep.

1.6.3.35 Spesifikasi Deskripsi Kelaskasir_C

kasir_C	<<Control>>
<p>+ kasir_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+displayDetilPerawatan(Pasien) Operasi ini untuk menampilkan data Detil Perawatan yang akan dimasukkan ke database Detil Perawatan.</p> <p>+totalBiayaKecantikan(Pasien) Operasi ini untuk menghitung biaya kecantikan dari Customer.</p> <p>+PembayaranKecantikan(Pasien): Pasien Operasi ini untuk mengelola pembayaran perawatan kecantikan Pasien dari database Pembayaran.</p> <p>+PembayaranRumah Sakit(Pasien): Pasien Operasi ini untuk mengelola pembayaran pemeriksaan Pasien di Rumah Sakit dari database Pembayaran.</p> <p>+pembelianObat(Pasien): Pasien Operasi ini untuk mengelola pembelian obat dari pelanggan non pasien Rumah Sakit dari database Layanan.</p> <p>+getPembayaranRumah Sakit(Pasien): Pasien Operasi ini untuk mengambil data Pembayaran dari</p>	

Program Studi Teknik Informatika	DPPL – SIRS	102/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

database Pembayaran Rumah Sakit.

+penghitunganUangKembali(Pasien): Pasien
 Operasi ini untuk melakukan penghitungan biaya dari pembayaran perawatan maupun pemeriksaan dari Pasien jika uang yang diberikan melebihi biaya tercantum.

+getAllObat() : Obat
 Operasi ini untuk megngambil data Obatdari database

+ObatBiayaLab() : Lab
 Operasi ini untuk megngambil data BiayaLabdari database.

+ObatBiayaRadiologi() : radiologi
 Operasi ini untuk megngambil data BiayaRadiologidari database.

1.6.3.36 Spesifikasi Deskripsi Kelas LayananLab_C

LayananLab_C	<<Control>>
<pre>+ layananLab_C () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entry layananLab (layananLab) Operasi ini untuk membuat data layananLabyang akan dimasukkan ke databaselayananLab. +edit layananLab (layananLab) Operasi ini untuk mengeditdata layananLabdari database layananLab. +display layananLab (layananLab) Operasi ini untuk menampilkan data layananLab dari database layananLab. +get layananLab () : layananLab</pre>	

Operasi ini untuk mengambil data BidangMedis dari database BidangMedis.

+delete layananLab (layananLab): layananLab

Operasi ini untuk menghapusdata layananLab dari database layananLab.

1.6.3.37 Spesifikasi Deskripsi Kelas HasilLab_C

HasilLab_C	<<Control>>
<p>+ HasilLab_C () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+entry HasilLab (HasilLab) Operasi ini untuk membuat data HasilLabyang akan dimasukkan ke databaseHasilLab.</p> <p>+edit HasilLab (HasilLab) Operasi ini untuk mengeditdata HasilLabdari database HasilLab.</p> <p>+display HasilLab (HasilLab) Operasi ini untuk menampilkan data HasilLab dari database HasilLab.</p> <p>+get HasilLab () : HasilLab Operasi ini untuk mengambil data BidangMedis dari database BidangMedis.</p> <p>+delete HasilLab (HasilLab): HasilLab Operasi ini untuk menghapusdata HasilLab dari database HasilLab.</p>	

1.6.3.38 Spesifikasi Deskripsi Kelas Radiologi_C

Radiologi_C	<<Control>>
-------------	-------------

Program Studi Teknik Informatika	DPPL – SIRS	104/161
<p>Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika</p>		

```

+ Radiologi_C ()
Default konstruktor, digunakan untuk inisialisasi
semua attribute dari kelas ini.
+entry Radiologi (Radiologi)
Operasi ini untuk membuat data Radiologiyang akan
dimasukkan ke databaseRadiologi.
+edit Radiologi (Radiologi)
Operasi ini untuk mengeditdata Radiologidari database
Radiologi.
+display Radiologi (Radiologi)
Operasi ini untuk menampilkan data Radiologi dari
database Radiologi.
+get Radiologi () : Radiologi
Operasi ini untuk mengambil data BidangMedis dari
database BidangMedis.
+delete Radiologi (Radiologi): Radiologi
Operasi ini untuk menghapusdata Radiologi dari
database Radiologi.

```

1.6.3.39 Spesifikasi Deskripsi Kelas HasilRadiologi_C

HasilRadiologi_C	<<Control>>
<pre> + HasilRadiologi_C () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entry HasilRadiologi (HasilRadiologi) Operasi ini untuk membuat data HasilRadiologiyang akan dimasukkan ke databaseHasilRadiologi. +edit HasilRadiologi (HasilRadiologi) </pre>	

Program Studi Teknik Informatika	DPPL – SIRS	105/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Operasi ini untuk mengedit data HasilRadiologi dari database HasilRadiologi.

```
+display HasilRadiologi (HasilRadiologi)
```

Operasi ini untuk menampilkan data HasilRadiologi dari database HasilRadiologi.

```
+get HasilRadiologi () : HasilRadiologi
```

Operasi ini untuk mengambil data BidangMedis dari database BidangMedis.

```
+delete HasilRadiologi (HasilRadiologi): HasilRadiologi
```

Operasi ini untuk menghapus data HasilRadiologi dari database HasilRadiologi.

1.6.3.40 Spesifikasi Deskripsi Kelas Login_M

Login_M	<<Model>>
-user : Pegawai Atribut ini digunakan untuk menyimpan data Pegawai pada database Pegawai.	
+ Login_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getPegawai () : Pegawai Operasi ini untuk mengambil data Pegawai dari database Pegawai.	

1.6.3.41 Spesifikasi Deskripsi Kelas UbahPassword_M

UbahPassword_M	<<Model>>
-user : Pegawai Atribut ini digunakan untuk menyimpan data	

Program Studi Teknik Informatika	DPPL – SIRS	106/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Pegawaipada database Pegawai.

+ UbahPassword_M()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+EditPassword (Pegawai)

Operasi ini untuk memperbaharuiPasswordPegawaidari database Pegawai.

1.6.3.42 Spesifikasi Deskripsi Kelas Pegawai_M

Pegawai_M

<<Model>>

-user : Pegawai

Atribut ini digunakan untuk menyimpan data Pegawai pada database Pegawai.

+Pegawai_M()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+AddPegawai (Pegawai)

Operasi ini untuk membuat data Pegawaiyang akan dimasukkan ke databasePegawai.

+EditPegawai (Pegawai)

Operasi ini untuk memperbaharuidataPegawaidari database Pegawai.

+DeletePegawai (Pegawai) : Pegawai

Operasi ini untuk menghapusdata Pegawaidari database Pegawai.

+DisplayPegawai (Pegawai) : Pegawai

Operasi ini untuk menampilkan data Pegawaidari database Pegawai.

+getPegawai () : Pegawai

Operasi ini untuk mengambil data Pegawai dari database Pegawai.

+ResetPassword()

Operasi ini untuk mereset password Pegawai di database Pegawai.

1.6.3.43 Spesifikasi Deskripsi Kelas Poliklinik_M

Poliklinik_M	<<Model>>
- Poliklinik: Poliklinik Atribut ini digunakan untuk menyimpan data Poliklinik pada database Poliklinik.	
+ Poliklinik_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+EntryPoliklinik (Poliklinik) Operasi ini untuk membuat data Poliklinik yang akan dimasukkan ke database Poliklinik.	
+EditPoliklinik (Poliklinik) Operasi ini untuk memperbaharui data Poliklinik dari database Poliklinik.	
+DeletePoliklinik (Poliklinik): Poliklinik Operasi ini untuk menghapus data Poliklinik dari database Poliklinik.	
+DisplayPoliklinik (Poliklinik): Poliklinik Operasi ini untuk menampilkan data Poliklinik dari database Poliklinik.	
+getPoliklinik () : Poliklinik Operasi ini untuk mengambil data Poliklinik dari database Poliklinik.	

1.6.3.44 Spesifikasi Deskripsi Kelas Karyawan_M

Karyawan_M	<<Model>>
<p>- Karyawan: karyawan Atribut ini digunakan untuk menyimpan data Karyawan pada database Karyawan.</p>	
<p>+ Karyawan_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+EntryKaryawan (Karyawan) Operasi ini untuk membuat data Karyawan yang akan dimasukkan ke databaseKaryawan.</p> <p>+EditKaryawan (Karyawan) Operasi ini untuk memperbaharuidataKaryawan dari database Karyawan.</p> <p>+DeleteKaryawan (Karyawan): Karyawan Operasi ini untuk menghapusdata Karyawan dari database Karyawan.</p> <p>+DisplayKaryawan (Karyawan): Karyawan Operasi ini untuk menampilkan data Karyawan dari database Karyawan.</p> <p>+getKaryawan () : Karyawan Operasi ini untuk mengambil data Karyawan dari database Karyawan.</p>	

1.6.3.45 Spesifikasi Deskripsi Kelas JadwalDokter_M

JadwalDokter_M	<<Model>>
<p>- JadwalDokter: jadwalDokter Atribut ini digunakan untuk menyimpan data JadwalDokter pada database JadwalDokter.</p>	

Program Studi Teknik Informatika	DPPL – SIRS	109/161
<p>Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika</p>		

```

+ JadwalDokter_M()
Default konstruktor, digunakan untuk inisialisasi
semua attribute dari kelas ini.
+EntryJadwalDokter (JadwalDokter)
Operasi ini untuk membuat data JadwalDokter yang akan
dimasukkan ke databaseJadwalDokter.
+EditJadwalDokter (JadwalDokter)
Operasi ini untuk memperbaharuidataJadwalDokter dari
database JadwalDokter.
+DeleteJadwalDokter (JadwalDokter): JadwalDokter
Operasi ini untuk menghapusdata JadwalDokter dari
database JadwalDokter.
+DisplayJadwalDokter (JadwalDokter): JadwalDokter
Operasi ini untuk menampilkan data JadwalDokter dari
database JadwalDokter.
+getJadwalDokter() : JadwalDokter
Operasi ini untuk mengambil data JadwalDokter dari
database JadwalDokter.

```

1.6.3.46 Spesifikasi Deskripsi Kelas Obat_M

Obat_M	<<Model>>
- Obat: obat Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.	
+ Obat_M() Default konstruktor, digunakan untuk inisialisasi	

Program Studi Teknik Informatika	DPPL – SIRS	110/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

semua attribute dari kelas ini.

+EntryObat (Obat)
 Operasi ini untuk membuat data Obat yang akan dimasukkan ke databaseObat.

+EditObat (Obat)
 Operasi ini untuk memperbaharuidataObat dari database Obat.

+DeleteObat (Obat): Obat
 Operasi ini untuk menghapusdata Obat dari database Obat.

+DisplayObat (Obat): Obat
 Operasi ini untuk menampilkan data Obat dari database Obat.

+getObat() : Obat
 Operasi ini untuk mengambil data Obat dari database Obat.

1.6.3.47 Spesifikasi Deskripsi Kelas BidangMedis_M

BidangMedis_M	<<Model>>
-Bidang Medis : bidangMedis Atribut ini digunakan untuk menyimpan data bidangMedispada database bidangMedis.	
+BidangMedis_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+EntryBidangMedis (BidangMedis) Operasi ini untuk membuat data BidangMedisyang akan dimasukkan ke databaseBidangMedis.	
+EditBidangMedis (BidangMedis) Operasi ini untuk memperbaharuidataBidangMedisdari	


```

database BidangMedis.
+DeleteBidangMedis (BidangMedis): BidangMedis
Operasi ini untuk menghapusdata BidangMedisdari
database BidangMedis.
+DisplayBidangMedis (BidangMedis): BidangMedis
Operasi ini untuk menampilkan data BidangMedisdari
database BidangMedis.
+getBidangMedis() : BidangMedis
Operasi ini untuk mengambil data BidangMedisdari
database BidangMedis.

```

1.6.3.48 Spesifikasi Deskripsi Kelas Pasien_M

Pasien_M	<<Model>>
<p>-Pasien: pasien Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.</p>	
<p>+Pasien_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p>	
<p>+EntryPasien(Pasien) Operasi ini untuk membuat data Pasienyang akan dimasukkan ke databasePasien.</p>	
<p>+EditPasien(Pasien) Operasi ini untuk memperbaharuidataPasiendari database Pasien.</p>	
<p>+DeletePasien(Pasien): Pasien Operasi ini untuk menghapusdata Pasiendari database Pasien.</p>	
<p>+DisplayPasien(Pasien):Pasien</p>	

Operasi ini untuk menampilkan data Pasien dari database Pasien.

+getPasien() : Pasien

Operasi ini untuk mengambil data Pasien dari database Pasien.

1.6.3.49 Spesifikasi Deskripsi Kelas Kamar_M

Kamar_M	<<Model>>
-Kamar: Kamar	
Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.	
+ Kamar_M()	
Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+EntryKamar(Kamar)	
Operasi ini untuk membuat data Kamar yang akan dimasukkan ke databaseKamar.	
+EditKamar (Kamar)	
Operasi ini untuk memperbaharuidataKamar dari database Kamar.	
+DeleteKamar(Kamar): Kamar	
Operasi ini untuk menghapusdata Kamar dari database Kamar.	
+DisplayKamar (Kamar): Kamar	
Operasi ini untuk menampilkan data Kamar dari database Kamar.	
+getKamar () : Kamar	
Operasi ini untuk mengambil data Kamar dari database Kamar.	

Program Studi Teknik Informatika	DPPL – SIRS	113/161
----------------------------------	-------------	---------

Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika

1.6.3.50 Spesifikasi Deskripsi Kelas KelasRawatInap_M

KelasRawatInap_M	<<Model>>
<p>- KelasRawatInap: KelasRawatInap</p> <p>Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.</p>	
<p>+ KelasRawatInap_M()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+EntryKelasRawatInap(KelasRawatInap)</p> <p>Operasi ini untuk membuat data KelasRawatInap yang akan dimasukkan ke databaseKelasRawatInap.</p> <p>+EditKelasRawatInap (KelasRawatInap)</p> <p>Operasi ini untuk memperbaharuidataKelasRawatInap dari database KelasRawatInap.</p> <p>+DeleteKelasRawatInap(KelasRawatInap): KelasRawatInap</p> <p>Operasi ini untuk menghapusdata KelasRawatInap dari database KelasRawatInap.</p> <p>+DisplayKelasRawatInap (KelasRawatInap): KelasRawatInap</p> <p>Operasi ini untuk menampilkan data KelasRawatInap dari database KelasRawatInap.</p> <p>+getKelasRawatInap () : KelasRawatInap</p> <p>Operasi ini untuk mengambil data KelasRawatInap dari database KelasRawatInap.</p>	

1.6.3.51 Spesifikasi Deskripsi Kelas MedicalRecord_M

MedicalRecord_M	<<Model>>
<p>- Pasien: pasien</p> <p>Atribut ini digunakan untuk menyimpan data Pasien pada database Pasien.</p>	

Program Studi Teknik Informatika	DPPL – SIRS	114/161
<p>Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika</p>		

-Dokter: dokter

Atribut ini digunakan untuk menyimpan data Dokter pada database Dokter.

-Poliklinik: Poliklinik

Atribut ini digunakan untuk menyimpan data Poliklinik pada database Poliklinik.

- Kecantikan: kecantikan

Atribut ini digunakan untuk menyimpan data Kecantikan pada database Kecantikan.

- Paket Kecantikan: paketKecantikan

Atribut ini digunakan untuk menyimpan data Paket Kecantikan pada database Paket Kecantikan.

- No Antrian: noAntrian

Atribut ini digunakan untuk menyimpan data No Antrian pada database No Antrian.

+MedicalRecord_M()

Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.

+EntryPasienPeriksa(Pasien)

Operasi ini untuk membuat data Pasien Periksa yang akan dimasukkan ke database Medical Record.

+EntryPasienPerawatan(Pasien)

Operasi ini untuk membuat data Pasien Perawatan yang akan dimasukkan ke database Antrian Kecantikan.

+EntryPemeriksaanAwal(Pasien)

Operasi ini untuk membuat data Pemeriksaan Awal Pasien yang akan dimasukkan ke database Medical Record.

+EditPemeriksaanAwal(Pasien)

Operasi ini untuk memperbaharui data Pemeriksaan Awal Pasien dari database Medical Record.

```

+DisplayMedicalRecord (MR):MR
Operasi ini untuk menampilkan data Medical
RecordPasien dari database Medical Record.
+getPasien () : Pasien
Operasi ini untuk mengambil data Pasien dari database
Pasien.
+getPoliklinik () : Poliklinik
Operasi ini untuk mengambil data Poliklinikdari
database Poliklinik.
+getDokter () : Dokter
Operasi ini untuk mengambil data Dokterdari database
Dokter.
+getNoAntrian () : NoAntrian
Operasi ini untuk mengambil data NoAntrian dari
database NoAntrian.
+getMedicalRecord () : MedicalRecord
Operasi ini untuk mengambil data MedicalRecord dari
database MedicalRecord.

```

1.6.3.52 Spesifikasi Deskripsi Kelas Resep_M

Resep_M	<<Model>>
<p>-Apoteker: apoteker</p> <p>Atribut ini digunakan untuk menyimpan data Apoteker pada database Dokter.</p>	
<p>+Resep_M()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+updateCekObat (Pasien)</p> <p>Operasi ini untuk memperbaharuidataResepPasien dari</p>	

database Medical Record.

+DisplayDetilResep (Pasien):Pasien
 Operasi ini untuk menampilkan dataDetil ResepPasien dari database Resep.

+getResepAllLunas() : Resep
 Operasi ini untuk mengambil data Resep dari database Resep yang memiliki status resep lunas.

+getResepBelumDiproses() : Resep
 Operasi ini untuk mengambil data Resep dari database Resep yang memiliki status resep Belum diproses.

+getResepAllSedangDiproses() : Resep
 Operasi ini untuk mengambil data Resep dari database Resep yang memiliki status resep sedang diproses.

+getResepAllSelesaiDiproses() : Resep
 Operasi ini untuk mengambil data Resep dari database Resep yang memiliki status resep selesai diproses.

1.6.3.53 Spesifikasi Deskripsi Kelas Kasir_M

Kasir_M	<<Model>>
<p>-Kasir: kasir Atribut ini digunakan untuk menyimpan data Kasirpada database Kasir.</p>	
<p>+Kasir_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+displayDetilPerawatan(Pasien) Operasi ini untuk menampilkan data Detil Perawatan yang akan dimasukkan ke database Detil Perawatan.</p>	

```

+totalBiayaKecantikan(Pasien)
Operasi ini untuk menghitung biaya kecantikan dari
Customer.
+PembayaranKecantikan(Pasien): Pasien
Operasi ini untuk mengelola pembayaran perawatan
kecantikan Pasien dari database Pembayaran.
+PembayaranRumah Sakit(Pasien): Pasien
Operasi ini untuk mengelola pembayaran pemeriksaan
Pasien di Rumah Sakit dari database Pembayaran.
+pembelianObat(Pasien): Pasien
Operasi ini untuk mengelola pembelian obat dari
pelanggan non pasien Rumah Sakit dari database
Layanan.
+getPembayaranRumah Sakit(Pasien): Pasien
Operasi ini untuk mengambil data Pembayaran dari
database Pembayaran Rumah Sakit.
+penghitunganUangKembali(Pasien): Pasien
Operasi ini untuk melakukan penghitungan biaya dari
pembayaran perawatan maupun pemeriksaan dari Pasien
jika uang yang diberikan melebihi biaya tercantum.
+getAllObat() : Obat
Operasi ini untuk mengambil data Obat dari database
Obat.

```

1.6.3.54 Spesifikasi Deskripsi Kelas LayananLab_M

LayananLab_M	<<Model>>
-LayananLab: LayananLab	
Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.	

Program Studi Teknik Informatika	DPPL – SIRS	118/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

```

+ LayananLab_M()
Default konstruktor, digunakan untuk inisialisasi
semua attribute dari kelas ini.
+EntryLayananLab(LayananLab)
Operasi ini untuk membuat data LayananLab yang akan
dimasukkan ke databaseLayananLab.
+EditLayananLab (LayananLab)
Operasi ini untuk memperbaharuidataLayananLab dari
database LayananLab.
+DeleteLayananLab(LayananLab): LayananLab
Operasi ini untuk menghapusdata LayananLab dari
database LayananLab.
+DisplayLayananLab (LayananLab): LayananLab
Operasi ini untuk menampilkan data LayananLab dari
database LayananLab.
+getLayananLab () : LayananLab
Operasi ini untuk mengambil data LayananLab dari
database LayananLab.

```

1.6.3.55 Spesifikasi Deskripsi Kelas HasilLab_M

HasilLab_M	<<Model>>
<p>-HasilLab: HasilLab</p> <p>Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.</p>	
<pre> + HasilLab_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +EntryHasilLab(HasilLab) Operasi ini untuk membuat data HasilLab yang akan dimasukkan ke databaseHasilLab. </pre>	


```

+EditHasilLab (HasilLab)
Operasi ini untuk memperbaharuidataHasilLab dari
database HasilLab.
+DeleteHasilLab(HasilLab): HasilLab
Operasi ini untuk menghapusdata HasilLab dari
database HasilLab.
+DisplayHasilLab (HasilLab): HasilLab
Operasi ini untuk menampilkan data HasilLab dari
database HasilLab.
+getHasilLab () : HasilLab
Operasi ini untuk mengambil data HasilLab dari
database HasilLab.

```

1.6.3.56 Spesifikasi Deskripsi Kelas Radiologi_M

Radiologi_M	<<Model>>
<p>-Radiologi: Radiologi</p> <p>Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.</p>	
<p>+ Radiologi_M()</p> <p>Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+EntryRadiologi(Radiologi)</p> <p>Operasi ini untuk membuat data Radiologi yang akan dimasukkan ke databaseRadiologi.</p> <p>+EditRadiologi (Radiologi)</p> <p>Operasi ini untuk memperbaharuidataRadiologi dari database Radiologi.</p> <p>+DeleteRadiologi(Radiologi): Radiologi</p> <p>Operasi ini untuk menghapusdata Radiologi dari database Radiologi.</p>	

```

+DisplayRadiologi (Radiologi): Radiologi
Operasi ini untuk menampilkan data Radiologi dari
database Radiologi.
+getRadiologi () : Radiologi
Operasi ini untuk mengambil data Radiologi dari
database Radiologi.

```

1.6.3.57 Spesifikasi Deskripsi Kelas HasilRadiologi_M

HasilRadiologi_M	<<Model>>
<p>-HasilRadiologi: HasilRadiologi Atribut ini digunakan untuk menyimpan data JadwalKaryawan pada database JadwalKaryawan.</p>	
<p>+ HasilRadiologi_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.</p> <p>+EntryHasilRadiologi (HasilRadiologi) Operasi ini untuk membuat data HasilRadiologi yang akan dimasukkan ke databaseHasilRadiologi.</p> <p>+EditHasilRadiologi (HasilRadiologi) Operasi ini untuk memperbaharuidataHasilRadiologi dari database HasilRadiologi.</p> <p>+DeleteHasilRadiologi (HasilRadiologi): HasilRadiologi Operasi ini untuk menghapusdata HasilRadiologi dari database HasilRadiologi.</p> <p>+DisplayHasilRadiologi (HasilRadiologi): HasilRadiologi Operasi ini untuk menampilkan data HasilRadiologi dari database HasilRadiologi.</p> <p>+getHasilRadiologi () : HasilRadiologi Operasi ini untuk mengambil data HasilRadiologi dari</p>	

database HasilRadiologi.

Perancangan Data

1.7 Dekomposisi Data

1.7.1 Deskripsi Entitas Data ICD10

Nama	Tipe	Panjang	Keterangan
kode	int	11	kode , primary key
NamaPenyakit	string	50	Nama penyakit

1.7.2 Deskripsi Entitas Data Detil Layanan

Nama	Tipe	Panjang	Keterangan
Iddetail	int	11	Id detail layanan, primary key
Idlayanan	int	11	Id layanan
Tanggal_layanan	date		Tanggal layanan
Idrawatinap	int	11	Id rawatinap
tanggalLayanan	date		Tanggal dilakukan layanan
biayalayanan	float		Harga setiap layanan

1.7.3 Deskripsi Entitas Data rawatinap

Nama	Tipe	Panjang	Keterangan
------	------	---------	------------

Program Studi Teknik Informatika	DPPL – SIRS	122/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

idrawatinap	int		Id rawatinap primary key
idkamar	int		Id kamar yang digunakan
nomr	int		No medicalrecord pasien
tanggalmasuk	datetime		Tanggal masuk rawat inap
total	float		Total biaya rawat inap
statusRI	string	20	Status rawat inap

1.7.4 Deskripsi Entitas Data Kamar

Nama	Tipe	Panjang	Keterangan
idkamar	int		Id kamar primary key
idkelas	int		Id kelas dari kamar
namakamar	string	40	Nama kamar
Jumlah_bad	int		Jumlah tempat tidur

1.7.5 Deskripsi Entitas Data kelas

Nama	Tipe	Panjang	Keterangan
idkelas	int		Id kelas primary key
Namakelas	string	40	Nama kelas
harga	float		Harga dari

Program Studi Teknik Informatika	DPPL – SIRS	123/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

			kamar dari kelas tersebut
--	--	--	---------------------------

1.7.6 Deskripsi Entitas Data Pegawai

Nama	Tipe	Panjang	Keterangan
idpegawai	int		Id pegawai, primary key
idrole	int		Id role dari pegawai
idpoliklinik	int		Id dari poliklinik pegawai
namapegawai	string	40	Nama pegawai
jeniskelamin	string	20	Jenis kelamin setiap pegawai
alamat	string	250	Alamat dari pegawai
jenispegawai	string		Jenis pegawai dari setiap pegawai
username	string	10	Username dari pegawai
password	string	20	Password dari setiap pegawai

1.7.7 Deskripsi Entitas Data Jadwal Dokter

Nama	Tipe	Panjang	Keterangan
------	------	---------	------------

Program Studi Teknik Informatika	DPPL – SIRS	124/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Idjadwal	int		Id jadwal untuk dokter, primary key
IdPoliklinik	int		Id Poliklinik yang dimiliki dokter
Iddokter	int		Id dokter
Hari	string	10	Hari kerja untuk dokter
day	string	10	Hari kerja dengan format waktu
jammulai	string	10	Jam mulai jadwal
jammulai	string	10	Jam selesai jadwal
sesi	int		Sesi jadwal

1.7.8 Deskripsi Entitas Data Detail layanan

Nama	Tipe	Panjang	Keterangan
Iddetail	int		Id detail layanan primary key
idlayanan	int		Id layanan
Idrawatinap	int		Jenis rawat inap setiap detail
tanggallayanan	datetime		Tanggal layanan

Program Studi Teknik Informatika	DPPL – SIRS	125/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

biayalayanan	float		Biaya layanan rawat inap
--------------	-------	--	--------------------------

1.7.9 Deskripsi Entitas Dataukuran film

Nama	Tipe	Panjang	Keterangan
Idfilm	int		Id film primary key
idkategori	int		Id kategori film
kodofilm	int		Kode film yang digunakan
stock	int		Stock film yang digunakan
ukuran			Ukuran film

1.7.10 Deskripsi Entitas Data Medical Record

Nama	Tipe	Panjang	Keterangan
nomr	int		No medical record, primary key
Tanggal_periksa	date		Tanggal periksa
Iddokter	int	11	Id dokter yang melakukan pemeriksaan
Idresep	int		Id resep
Idpasien	string	20	Id pasien yang akan periksa
dianogsa	string	250	dianogsa yang

Program Studi Teknik Informatika	DPPL – SIRS	126/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

			di berikan kepada pasien dari dokter
Tindakandokter	string	150	Tindakan yang do lakukan oleh dokter
keluhan	string	150	Keluhan yang di rasakan oleh pasien
Cektensi	string	40	Cek tensi untuk pasien
Beratbadan	string	20	Berat badan untuk pasien
Tinggi	string	29	Tinggi badan untuk pasien
Rujuk	string	150	Rujukan rumah sakit yang akan di ajak kerjasama.
Suhu_badan	string	40	Suhu badan untuk pasien
nadi	string	50	Detak nadi pasien
Riwayatoperasi	string	150	Riwayat operasi pasien
Alergiobat	string	80	Alergi obat pasien
statusMr	string	20	Status medicalrecord
Konsultasi	string	80	Konsultasi pasien ke dokter

1.7.11 Deskripsi Entitas DataAntrian

Nama	Tipe	Panjang	Keterangan
Nourut	int		No urut untuk antrian, primary key
Iddokter	int		Id dokter
Idpasien	int		Id pasien yang ada di antrian
IdPoliklinik	int		Id Poliklinik
tanggalpemeriksaan	datetime		Tanggal antrian hari itu
Noantrian	int		No antrian untuk pasien
Status	string	20	Status untuk pasien yaitu antri atau masuk.

1.7.12 Deskripsi Entitas Data Obat

Nama	Tipe	Panjang	Keterangan
Idobat	int		Id obat, primary key
Namaobat	string	50	Untuk nama

Program Studi Teknik Informatika	DPPL – SIRS	128/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

			obat
Keterangan	string	200	Keterangan dari obat tersebut
Harga	float		Harga untuk obat
Stok	int		Jumlah stok obat
Jenisobat	string	50	Jenis obat misalnya injeksi atau tablet

Deskripsi Entitas Data Pasien

Nama	Tipe	Panjang	Keterangan
Idpasien	int	11	Id pasien, primary key
Namapasien	string	50	Nama pasien
Alamat	string	150	Alamat pasien
Jeniskelamin	string	20	Jenis kelamin pasien laki-laki atau perempuan
Tempatlahir	string	50	Tempat pasien lahir
Tanggallahir	datetime	11	Tanggal lahir pasien
Goldarah	string	10	golongan

			darah pasien
Umur	int	11	Umur pasien
Agama	String	20	Agama pasien
Status_nikah	String	20	Status nikah pasien
Pekerjaan	String	45	Pekerjaan pasien
Warganegara	String	10	Warganegara pasien
Telp_pasien	String	20	No telpon pasien
Noktp	String	50	No kartu tanda penduduk
status	String	50	Status pasien

1.7.13 Deskripsi Entitas Data Pembayaran

Nama	Tipe	Panjang	Keterangan
Idpembayaran	Int	11	Id pembayaran primary key
Idpasien	String	20	Id pasien yang akan membayar
Tanggalpembayaran	Datetime		Tanggal pembayaran pasien

Biayalab	Float		Biaya lab
Biayaperawatan	Float		Biaya perawatan
Biayakamar	Float		Biaya kamar
biayapemeriksaan	Float		Biaya pemeriksaan
total	float		Biaya total
Totalbayar	Float		Total pembayaran pemeriksaan yang sudah di lakukan pasien
biayaobat	Float		biayaobat
Jumlah_uang	Float		Jumlah uang cash yang di bayar pasien
Kembalian	Float		Uang kembalian pembayaran
status	string	50	Status pembayaran, Lunas

1.7.14 Deskripsi Entitas Data Pemeriksaan

Nama	Tipe	Panjang	Keterangan
Idpemeriksaan	int		Id pemeriksaan
Idpasien	String	20	Id pasien yang periksa

Program Studi Teknik Informatika	DPPL – SIRS	131/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Tanggalperiksa	Datetime		Tanggal pemeriksaan pasien
Totalbiaya	Float		Total biaya yang harus dibayar oleh pasien
Jenispemeriksaan	String	50	Jenis pemeriksaan yang dilakukan pasien
StatusPemeriksaan	String	20	Status pemeriksaan

1.7.15 Deskripsi Entitas DataPegawai

Nama	Tipe	Panjang	Keterangan
Idpegawai	int	11	Id untuk user, primary key
Namapegawai	string	50	Nama user
Username	string	40	Username untuk user masuk ke system
Password	string	40	Password
Role	string	35	Role user sebagai apa, administrator, perawat, dokter, karyawan, kasir atau apoteker

Program Studi Teknik Informatika	DPPL – SIRS	132/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.7.16 Deskripsi Entitas Data Poliklinik

Nama	Tipe	Panjang	Keterangan
IdPoliklinik	int		Id Poliklinik, primary key
NamaPoliklinik	string	50	Nama Poliklinik
biaya	float		Biaya poliklinik

1.7.17 Deskripsi Entitas Data Resep

Nama	Tipe	Panjang	Keterangan
Idresep	int	11	Id resep, primary key
nomr	int		nomedicalrecord
Total	float		Total bayar obat yang harus di bayar pasien .

1.7.18 Deskripsi Entitas Data Jenis Pemeriksaan

Nama	Tipe	Panjang	Keterangan
idjenispemeriksaan	int		Id jenis pemeriksaan, primary key
idkategori	int		Id kategori layanan
tarif	float		Biaya layanan lab
Nilainormal	string	25	Nilai normal

			pemeriksaan
Keteranganpem	string	250	Keterangan pemeriksaan

1.7.19 Deskripsi Entitas DataKategori layanan

Nama	Tipe	Panjang	Keterangan
Idkategori	int	11	Id kategori, primary key
kategori	string	25	Nama kategori
Biaya	float		Biaya dari pemeriksaan Radiologi
status	string	20	Status kategori

1.8 Deskripsi Perancangan UI

1.8.1 Antarmuka Login

Program Studi Teknik Informatika	DPPL – SIRS	134/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

The image shows a login window with a title bar labeled 'Image'. Below the title bar, the word 'Login' is centered. There are two text input fields: one labeled 'Username' and one labeled 'Password', both containing the text 'Text'. Below these fields is a button labeled 'Login'. The window has a light beige background and a dark grey title bar and footer bar.

Gambar 1.62 Rancangan Antarmuka Login

Antarmuka ini digunakan untuk proses login ke aplikasi desktop dalam system SIPAS. Untuk dapat masuk ke dalam suatu sistem ini user harus melakukan input *username* dan *password* terlebih dahulu. Di pastikan *username* dan *password* harus benar. Setelah user mengisi *username* dan *password* dan kemudian menekan button "*Login*" maka sistem akan mengecek terlebih dahulu inputan *username* dan *password* valid atau tidak. Jika benar *user* akan masuk ke aplikasi ini dan dapat menggunakan menu-menu yang disediakan dalam aplikasi ini. Jika salah *user* akan menginputkan ulang sampai benar.

Program Studi Teknik Informatika	DPPL – SIRS	135/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.8.2 Antarmuka Manu Utama



Gambar 1.63 Rancangan Antarmuka Menu Utama

Antarmuka ini akan muncul ketika kita login menggunakan username administrator. Gambar - gambar tersebut merupakan menu-menu untuk pengelolaan yang di kelola oleh user admin. User bisa menggunakan sesuai dengan kebutuhan user.

1.8.3 Antaramuka Edit Password



The image shows a web form titled "Ubah Password". It contains three text input fields: "Username", "Password Baru", and "Konfirmasi Password". Below these fields are two buttons: "Clear" and "Save". The form is overlaid on a watermark of a university logo with the motto "servians in lumine veritatis".

Gambar 1.64 Rancangan Antarmuka Edit Password

Antarmuka ini muncul ketika kita memilih user memilih link "Change Password". antarmuka ini digunakan untuk mengedit password user, dimana yang bisa melakukan edit password ini adalah user yang memiliki username itu sendiri. User diminta menginputkan password lama, password baru, dan confirm password baru. kemudian setelah selesai mengisi itu klik button "Ubah" untuk mengganti passwordnya.

Program Studi Teknik Informatika	DPPL – SIRS	137/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.8.4 Antarmuka Pengelolaan Data Pegawai



Gambar 1.65 Rancangan Antarmuka Pengelolaan Data Pegawai

Antarmuka ini merupakan antarmuka dari pengelolaan data pegawai, antarmuka ini akan muncul ketika user memilih pada menu utama "Pengelolaan Data Pegawai". Tampilan awal dalam pengelolaan data pegawai ini adalah menampilkan semua data-data pegawai antara lain *nama user, username, role dan aksi(edit,delete)*. Pada antarmuka ini terdapat beberapa perintah yang berbentuk link antara lain fungsi untuk "tambah data, edit, dan hapus" fungsi-fungsi tersebut bisa di gunakan sesuai dengan kebutuhan user yang mengelola. Ada menu "kembali"

pada tabmenu yang ada di atas itu berfungsi untuk kembali ke menu utama. Dan setiap form akan ada link pada pojok kanan atas "change password dan logout" untuk mengubah password dan keluar dari system ini.

1.8.5 Antarmuka Tambah Data Pengelolaan Data Pegawai

Pengelolaan Pegawai

Log Out

Username

Nama Pegawai

Gender L

Alamat Pegawai

No Tlpn

Jenis Pegawai

< Back

Gambar 1.66 Rancangan Antarmuka Tambah Data Pengelolaan Data Pegawai

Antarmuka ini akan muncul ketika user akan menambah pegawai dengan memilih link "tambah data" pada antarmuka pengelolaan data pegawai awal maka akan menuju ke antarmuka ini. User akan memasukkan "nama user, username, dan memilih role pada combobox". Untuk password user tidak menginputkan karena password akan mengeset secara otomatis sama

Program Studi Teknik Informatika	DPPL – SIRS	139/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

dengan *username*. Setelah user menginputkan data-datanya kemudian klik button add, data akan tersimpan dan kemudian langsung akan menuju antarmuka pengelolaan data pegawai awal, dan data yang baru saja di inputkan akan ikut ditampilkan bersama data-data yang sebelumnya, dan akan muncul pemberitahuan "1 data berhasil ditambahkan".

1.8.6 Antarmuka Edit Data Pengelolaan Data Pegawai



The screenshot shows a web interface titled "Pengelolaan Pegawai". At the top right is a "Log Out" button. The main content area contains a form with the following fields: "Username" (text input), "Nama Pegawai" (text input), "Gender" (dropdown menu with "L" selected), "Alamat Pegawai" (text input), "No Tlpn" (text input), and "Jenis Pegawai" (text input). Below the form are two buttons: "Update" and "Cancel". At the bottom left of the form area is a "< Back" button.

Gambar 1.67 Rancangan Antarmuka Edit Data Pengelolaan Data Pegawai

Antarmuka ini akan muncul ketika user akan mengedit data pegawai yang diinginkan dengan memilih link "edit" pada data yang akan di edit yang ada pada antarmuka pengelolaan data pegawai awal maka akan menuju ke antarmuka ini, dan data yang di pilih

Program Studi Teknik Informatika	DPPL – SIRS	140/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

untuk di edit akan masuk ke dalam textbox dan combobox. User akan mengedit "nama user, username, dan role pada combobox". Untuk password user tidak bisa mengeditnya lewat form ini, tetapi password diubah dengan menggunakan perintah "change password" yang ada di pojok kanan atas. Setelah user mengedit data-datanya kemudian klik button ubah, data akan terupdate dan kemudian langsung akan menuju antarmuka pengelolaan data pegawai awal, dan data yang baru saja di update akan ikut ditampilkan bersama data-data yang sebelumnya dengan data yang baru sesuai, dan akan muncul pemberitahuan "1 data berhasil diubah".

1.8.7 Antarmuka Delete Data Pengelolaan Data Pegawai



Gambar 1.68 Rancangan Antarmuka Delete Data Pengelolaan Data Pegawai

Program Studi Teknik Informatika	DPPL – SIRS	141/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Antarmuka ini akan muncul ketika user menghapus data pegawai yang diinginkan dengan memilih link "delete" pada data yang akan di hapus yang ada pada antarmuka pengelolaan data pegawai awal maka data yang dipilih akan dihapus dan akan hilang dari data-data yang di tampilkan dan akan muncul pemberitahuan "1 data berhasil dihapus".

1.8.8 Antarmuka Pengelolaan Poliklinik



Gambar 1.69 Rancangan Antarmuka Pengelolaan Poliklinik

Antarmuka ini merupakan antarmuka dari pengelolaan data Poliklinik, antarmuka ini akan muncul ketika user memilih pada menu utama "Pengelolaan Data Poliklinik". Tampilan awal dalam pengelolaan data Poliklinik ini adalah menampilkan

semua data-data Poliklinik antara lain *id_* Poliklinik, *nama_* Poliklinik, *waktu layanan* dan *aksi(edit,delete)*. Pada antarmuka ini terdapat beberapa perintah yang berbentuk link antara lain fungsi untuk "*tambah data, edit , dan hapus*" fungsi-fungsi tersebut bisa di gunakan sesuai dengan kebutuhan user yang mengelola. Ada menu "*kembali*" pada tabmenu yang ada di atas itu berfungsi untuk kembali ke menu utama. Dan setiap form akan ada link pada pojok kanan atas "*change password dan logout*" untuk mengubah password dan keluar dari sistem ini.

1.8.9 Antarmuka Tambah Data Pengelolaan Poliklinik



The screenshot displays a web interface titled "Pengelolaan Data Poliklinik". At the top right, there is a "Log Out" button. The main content area contains a form with two input fields: "Nama Poliklinik" and "Waktu Layanan". Below these fields are two buttons: "Tambah" and "Cancel". In the bottom left corner of the form area, there is a "< Back" button.

Gambar 1.70 Rancangan Antarmuka Tambah Data Pengelolaan Poliklinik

Program Studi Teknik Informatika	DPPL – SIRS	143/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Antarmuka ini akan muncul ketika user akan menambah data Poliklinik dengan memilih link "tambah data" pada antarmuka pengelolaan data Poliklinik awal maka akan menuju ke antarmuka ini. User akan memasukkan "nama Poliklinik, waktu layanan". Setelah user menginputkan data-datanya kemudian klik button add, data akan tersimpan dan kemudian langsung akan menuju antarmuka pengelolaan data Poliklinik, dan data yang baru saja diinputkan akan ikut ditampilkan bersama data-data yang sebelumnya, dan akan muncul pemberitahuan "1 data berhasil ditambahkan".

1.8.10 Antarmuka Edit Data Pengelolaan Poliklinik

The image shows a web interface titled "Pengelolaan Data Poliklinik". At the top right, there is a "Log Out" button. The main content area contains two input fields: "Nama Poliklinik" and "Waktu Layanan". Below these fields are two buttons: "Update" and "Cancel". At the bottom left of the form area, there is a "< Back" link. The interface is overlaid on a large, faint watermark of a university logo.

Gambar 1.71 Rancangan Antarmuka Edit Data Pengelolaan Poliklinik

Program Studi Teknik Informatika	DPPL – SIRS	144/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Antarmuka ini akan muncul ketika user akan mengedit data Poliklinik yang diinginkan dengan memilih link "edit" pada data yang akan di edit yang ada pada antarmuka pengelolaan data Poliklinik awal maka akan menuju ke antarmuka ini, dan data yang dipilih untuk diedit akan masuk ke dalam textbox dan combobox. User akan mengedit "nama Poliklinik, waktu layanan". Setelah user mengedit data-datanya kemudian klik button ubah, data akan terupdate dan kemudian langsung akan menuju antarmuka pengelolaan data Poliklinik awal, dan data yang baru saja di update akan ikut ditampilkan bersama data-data yang sebelumnya dengan data yang baru sesuai, dan akan muncul pemberitahuan "1 data berhasil diubah".

Program Studi Teknik Informatika	DPPL – SIRS	145/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.8.11 Antarmuka Delete Data Pengelolaan Poliklinik



Gambar 1.72 Rancangan Antarmuka Delete Data Pengelolaan Poliklinik

Antarmuka ini akan muncul ketika user menghapus data Poliklinik yang diinginkan dengan memilih link "delete" pada data yang akan di hapus yang ada pada antarmuka pengelolaan data Poliklinik awal maka data yang dipilih akan dihapus dan akan hilang dari data-data yang di tampilkan dan akan muncul pemberitahuan "1 data berhasil dihapus".

1.8.12 Antarmuka Pengelolaan Jadwal Dokter



Gambar 1.73 Rancangan Antarmuka Pengelolaan Jadwal Dokter

Antarmuka ini merupakan antarmuka dari pengelolaan jadwal dokter, antarmuka ini akan muncul ketika user memilih pada menu utama "Pengelolaan jadwal Dokter dan karyawan". Tampilan awal dalam pengelolaan jadwal dokter ini adalah menampilkan semua data-data Poliklinik antara lain *id_jadwal*, *nama_dokter*, *nama Poliklinik*, *hari*, *waktu layanan* dan *aksi(edit,delete)*. Pada antarmuka ini terdapat beberapa perintah yang berbentuk link antara lain fungsi untuk "tambah data, edit, dan hapus" fungsi-fungsi tersebut bisa di gunakan sesuai dengan kebutuhan user yang mengelola. Ada menu "kembali" pada tabmenu yang ada di atas itu berfungsi untuk

Program Studi Teknik Informatika	DPPL – SIRS	147/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

kembali ke menu utama. Dan setiap form akan ada link pada pojok kanan atas "change password dan logout" untuk mengubah password dan keluar dari system ini

1.8.13 Antarmuka Tambah Data Pengelolaan Jadwal Dokter

The screenshot shows a web form titled "Pengelolaan Jadwal Dokter". At the top right, there is a "Log Out" button. The form contains the following fields and controls:

- Nama Dokter:** A dropdown menu with "Selection..." and a downward arrow.
- Hari:** A text input field.
- Waktu:** A text input field.
- Nama Poliklinik:** A dropdown menu with "Selection..." and a downward arrow.
- Waktu Layanan:** A dropdown menu with "Selection..." and a downward arrow.
- Buttons:** "Tambah" and "Cancel" buttons are located below the input fields.
- Navigation:** A "< Back" button is located at the bottom left of the form area.

Gambar 1.74 Rancangan Antarmuka Tambah Data Pengelolaan Jadwal Dokter

Antarmuka ini akan muncul ketika user akan menambah jadwal dokter dengan memilih link "tambah data" pada antarmuka pengelolaan jadwal dokter awal maka akan menuju ke antarmuka ini. User akan memasukkan "nama dokter, nama Poliklinik, hari dan waktu layanan". Setelah user menginputkan data-

Program Studi Teknik Informatika	DPPL – SIRS	148/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

datanya kemudian klik button add, data akan tersimpan dan kemudian langsung akan menuju antarmuka pengelolaan jadwal dokter, dan data yang baru saja di inputkan akan ikut ditampilkan bersama data-data yang sebelumnya, dan akan muncul pemberitahuan "1 data berhasil ditambahkan".

1.8.14 Antarmuka Edit Data Pengelolaan Jadwal Dokter

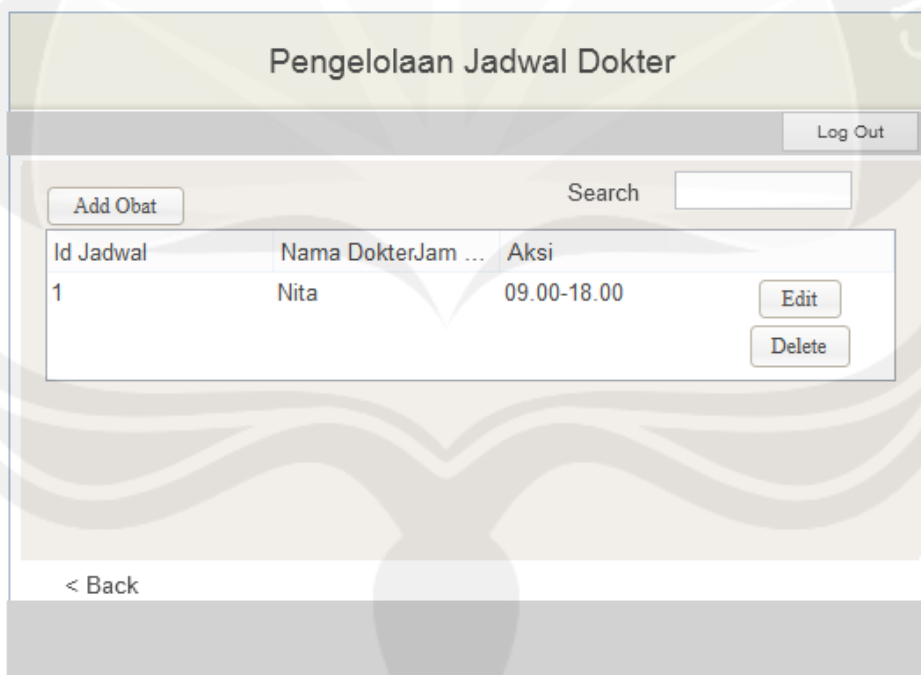
Gambar 1.75 Rancangan Antarmuka Edit Data Pengelolaan Jadwal Dokter

Antarmuka ini akan muncul ketika user akan mengedit jadwal dokter yang diinginkan dengan memilih link "edit" pada data yang akan di edit yang ada pada antarmuka pengelolaan jadwal dokter awal maka akan menuju ke antarmuka ini, dan data yang

Program Studi Teknik Informatika	DPPL – SIRS	149/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

di pilih untuk di edit akan masuk ke dalam textbox dan combobox. User akan mengedit "nama dokter, nama Poliklinik, hari, waktu layanan". Setelah user mengedit data-datanya kemudian klik button ubah, data akan terupdate dan kemudian langsung akan menuju antarmuka pengelolaan jadwal dokter awal, dan data yang baru saja di update akan ikut ditampilkan bersama data-data yang sebelumnya dengan data yang baru sesuai, dan akan muncul pemberitahuan "1 data berhasil diubah".

1.8.15 Antarmuka Delete Pengelolaan Jadwal Dokter



Gambar 1.76Rancangan Antarmuka Delete Data Pengelolaan Jadwal Dokter

Antarmuka ini akan muncul ketika user menghapus jadwal dokter yang diinginkan dengan memilih link "delete" pada data yang akan di hapus

Program Studi Teknik Informatika	DPPL – SIRS	150/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

yang ada pada antarmuka pengelolaan jadwal dokter awal maka data yang dipilih akan dihapus dan akan hilang dari data-data yang di tampilkan akan muncul pemberitahuan "1 data berhasil dihapus".

1.8.16 Antarmuka Pengelolaan Data Obat



Gambar 1.77 Rancangan Antarmuka Pengelolaan Data Obat

Antarmuka ini merupakan antarmuka dari pengelolaan data obat, antarmuka ini akan muncul ketika user memilih pada menu utama "Pengelolaan Data Obat". Tampilan awal dalam pengelolaan data obat ini adalah menampilkan semua data-data obat antara lain *id_obat*, *nama_obat*, *jenis_obat*, *harga_obat*, *stock_obat*, *dll* dan *aksi(edit,delete)*. Pada antarmuka ini terdapat beberapa perintah yang

berbentuk link antara lain fungsi untuk "tambah data, edit, dan hapus" fungsi-fungsi tersebut bisa di gunakan sesuai dengan kebutuhan user yang mengelola. Ada menu "kembali" pada tabmenu yang ada di atas itu berfungsi untuk kembali ke menu utama. Dan setiap form akan ada link pada pojok kanan atas "change password dan logout" untuk mengubah password dan keluar dari system ini.

1.8.17 Antarmuka Tambah Data Pengelolaan Data Obat

The image shows a web-based interface for drug management. The main heading is "Pengelolaan Obat". A "Log Out" button is located in the top right corner. The central part of the interface is a form with the following fields: "Nama Obat", "Jenis Obat", "Harga", "Stock", "Exp", and "Keterangan". Below the form are two buttons: "Tambah" and "Cancel". At the bottom left, there is a "< Back" button.

Gambar 1.78 Rancangan Antarmuka Tambah Data Obat

Antarmuka ini akan muncul ketika user akan menambah data obat dengan memilih link "tambah data" pada antarmuka pengelolaan data obat awal maka akan menuju ke antarmuka ini. User akan memasukkan

Program Studi Teknik Informatika	DPPL – SIRS	152/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

"nama_ obat, jenis obat, harga obat, stock obat, dll". Setelah user menginputkan data-datanya kemudian klik button add, data akan tersimpan dan kemudian langsung akan menuju antarmuka pengelolaan data obat, dan data yang baru saja di inputkan akan ikut ditampilkan bersama data-data yang sebelumnya , dan akan muncul pemberitahuan "1 data berhasil ditambahkan".

1.8.18 Antarmuka Edit Data Pengelolaan Data Obat

Pengelolaan Obat

Log Out

Nama Obat

Jenis Obat

Harga

Stock

Exp

Keterangan

Tambah Cancel

< Back

Gambar 1.79 Rancangan Antarmuka Edit Data Obat

Antarmuka ini akan muncul ketika user akan mengedit data obat yang diinginkan dengan memilih link "edit" pada data yang akan di edit yang ada

Program Studi Teknik Informatika	DPPL – SIRS	153/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

pada antarmuka pengelolaan data obat awal maka akan menuju ke antarmuka ini, dan data yang di pilih untuk di edit akan masuk ke dalam textbox. User akan mengedit "nama_obat, jenis obat, harga obat, stock obat, dll". Setelah user mengedit data-datanya kemudian klik button ubah, data akan terupdate dan kemudian langsung akan menuju antarmuka pengelolaan data obat awal, dan data yang baru saja di update akan ikut ditampilkan bersama data-data yang sebelumnya dengan data yang baru sesuai, dan akan muncul pemberitahuan "1 data berhasil diubah".

1.8.19 Antarmuka Delete Data Pengelolaan Data Obat

The screenshot shows a web interface titled "Pengelolaan Obat". At the top right is a "Log Out" button. Below it is a search bar with the label "Search" and an "Add Obat" button. The main content is a table with the following data:

Id Obat	Nama Obat	Jenis Obat	Aksi
1	Paramek	pil	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

At the bottom left of the interface is a "< Back" button.

Gambar 1.80 Rancangan Antarmuka Delete Data Obat

Antarmuka ini akan muncul ketika user menghapus data obat yang diinginkan dengan memilih link "delete" pada data yang akan di hapus yang ada pada antarmuka pengelolaan data obat awal maka data yang dipilih akan dihapus.

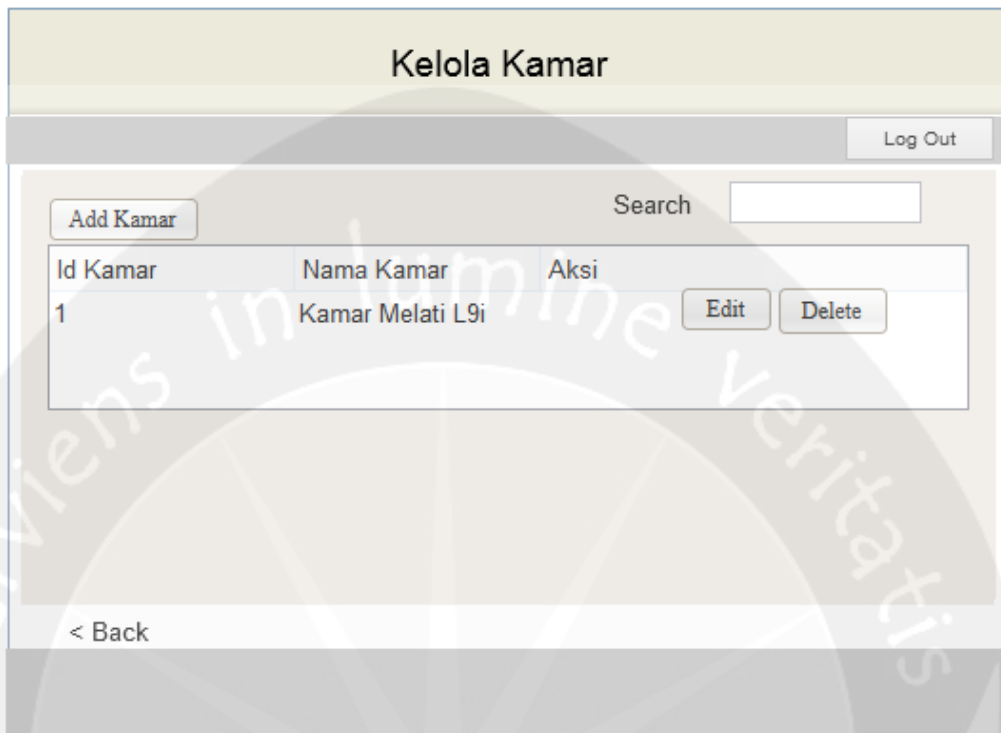
1.8.20 Antarmuka Search Data Obat



Gambar 1.81 Rancangan Antarmuka Search Data Obat

Antarmuka ini di gunakan untuk mencari data obat yang di inginkan. Setelah kita memasukkan nama obat yang akan di cari maka data yang akan tampil hanya data yang di cari saja.

1.8.21 Antarmuka Pengelolaan Kamar



Gambar 1.82 Rancangan Antarmuka Pengelolaan Kamar

Antarmuka ini merupakan antarmuka dari pengelolaan data kamar, antarmuka ini akan muncul ketika user memilih pada menu utama "Pengelolaan Kamar". Tampilan awal dalam pengelolaan data bidang medis ini adalah menampilkan semua data-data bidang medis antara lain *id_kamar*, *nama_kamar*, *biaya* dan *aksi(edit,delete)*. Pada antarmuka ini terdapat beberapa perintah yang berbentuk link antara lain fungsi untuk "tambah data, edit, dan hapus" fungsi-fungsi tersebut bisa di gunakan sesuai dengan kebutuhan user yang mengelola. Ada menu "kembali" pada tabmenu yang ada di atas itu berfungsi untuk kembali ke menu utama. Dan setiap form akan ada link

pada pojok kanan atas "change password dan logout" untuk mengubah password dan keluar dari system ini.

1.8.22 Antarmuka Tambah Data Kamar

The screenshot displays a web interface titled "Pengelolaan Data Kamar". At the top right, there is a "Log Out" button. The main content area contains three input fields: "Nama Kamar" (text input), "Jumlah Tempat Tidur" (text input), and "Kelas" (dropdown menu with "Selection..." and a downward arrow). Below these fields are two buttons: "Tambah" and "Cancel". At the bottom left of the form area, there is a "< Back" link.

Gambar 1.83 Rancangan Antarmuka Tambah Data Kamar

Antarmuka ini akan muncul ketika user akan menambah pegawai dengan memilih link "tambah data" pada antarmuka pengelolaan data bidang medis awal maka akan menuju ke antarmuka ini. User akan memasukkan "nama kamar dan biaya, diskripsi". Setelah user menginputkan data-datanya kemudian klik button add, data akan tersimpan dan kemudian langsung akan menuju antarmuka pengelolaan data kamar awal, dan data yang baru saja di inputkan akan ikut ditampilkan bersama data-data yang sebelumnya, dan akan muncul pemberitahuan "1 data berhasil ditambahkan".

Program Studi Teknik Informatika	DPPL – SIRS	157/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

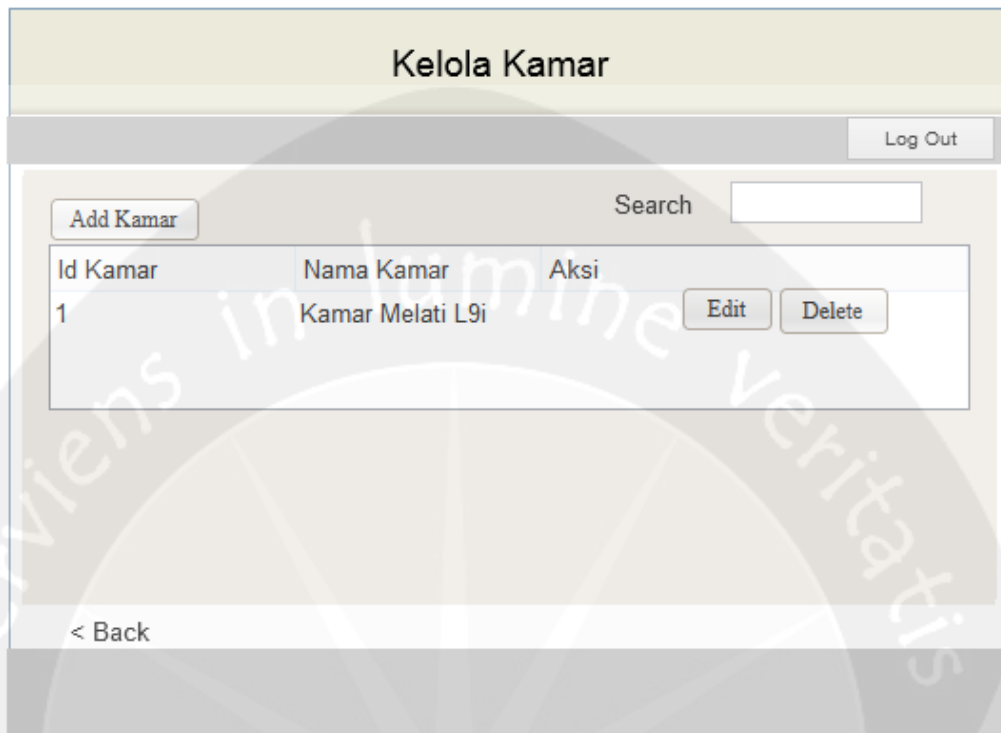
1.8.23 Antarmuka Edit Data Kamar

Gambar 1.84 Rancangan Antarmuka Edit Data Kamar

Antarmuka ini akan muncul ketika user akan mengedit data pegawai yang diinginkan dengan memilih link "edit" pada data yang akan di edit yang ada pada antarmuka pengelolaan data kamar awal maka akan menuju ke antarmuka ini, dan data yang di pilih untuk di edit akan masuk ke dalam textbox. User akan mengedit "nama kamar dan biaya, diskripsi". Setelah user mengedit data-datanya kemudian klik button ubah, data akan terupdate dan kemudian langsung akan menuju antarmuka pengelolaan data kamar, dan data yang baru saja di update akan ikut ditampilkan bersama data-data yang sebelumnya dengan data yang baru sesuai, dan akan muncul pemberitahuan "1 data berhasil diubah".

Program Studi Teknik Informatika	DPPL – SIRS	158/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.8.24 Antarmuka Delete Data Kamar



Gambar 1.85 Rancangan Antarmuka Delete Data Kamar

Antarmuka ini akan muncul ketika user menghapus data bidang Medis yang diinginkan dengan memilih link "delete" pada data yang akan di hapus yang ada pada antarmuka pengelolaan data bidang medis awal maka data yang dipilih akan dihapus dan akan hilang dari data-data yang di tampilkan dan akan muncul pemberitahuan "1 data berhasil dihapus"

1.8.25 Antarmuka Tambah Data Pasien

Pendaftaran Pemeriksaan

Tambah Pasien Daftar Pemeriksaan cetak bukti daftar Log Out

No MR

Nama Pasien

Jenis Kelamin Se... v No Tlpn

Tanggal Daftar date TTL date

Pekerjaan

Golongan Darah

Alamat Pasien

Update Cancel

< Back

1.86 Rancangan Antarmuka Input Data Data Pasien

Antarmuka ini digunakan oleh user resepsionis untuk menginputkan data pasien baru yang akan melakukan pemeriksaan. Data pasien yang diinputkan antara lain adalah *nama pasien, jenis kelamin, tanggal lahir, tempat lahir, alamat, no telp dll* sesuai data yang ada di atas. Setelah mengisi data pasien maka klik Button "submit". Data pasien akan di simpan.

1.8.26 Antarmuka Pendaftaran Pemeriksaan

The screenshot shows a web interface for 'Pendaftaran Pemeriksaan'. The title bar is 'Pendaftaran Pemeriksaan'. Below it is a navigation bar with buttons: 'Tambah Pasien', 'Daftar Pemeriksaan', 'cetak bukti daftar', and 'Log Out'. The main form area contains input fields for 'No MR' and 'Tanggal', a text field for 'Nama Pasien', and a 'Dokter' field with a dropdown menu. Below the 'Dokter' field is a 'Poliklinik' label and another dropdown menu. At the bottom of the form are 'Daftar' and 'Cancel' buttons. A '< Back' button is located at the bottom left of the form area.

Gambar 1.87 Rancangan Antarmuka daftar pemeriksaan pilih Poliklinik

Antarmuka ini akan muncul jika user memilih "daftar pemeriksaan". Daftar pemeriksaan ini digunakan untuk mendaftarkan pasien sesuai dengan Poliklinik dan pilihan dokter sesuai dengan kebutuhan dan keinginan pasien. Saat pasien akan mendaftar ke antrian sebelumnya pasien memilih Jadwal Dokter terlebih dahulu kemudian akan muncul form seperti diatas kemudian memilih Poliklinik terlebih dahulu. Kemudian setelah memilih Poliklinik Dokter yang akan memeriksa setelah itu masukan antrian dengan menekan tombol daftar secara otomatis akan masuk ke antrean.

Program Studi Teknik Informatika	DPPL – SIRS	161/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.8.27 Antarmuka Pendaftaran Rawat Inap

The screenshot shows a web interface for inpatient registration. At the top, there is a header with the title "Pendaftaran Rawat Inap". Below the header is a navigation bar with four buttons: "Daftar Rawat Inap", "Tambah Layanan", "Daftar Pasien Keluar", and "Log Out". The main content area contains a registration form with the following fields: "No MR" (text input), "Tanggal" (text input), "Nama Pasien" (text input), "Nama Kamar" (text input with a dropdown menu), "Kelas Kamar" (text input with a dropdown menu), and "Harga" (text input). Below the form are two buttons: "Daftar" and "Cancel". At the bottom left of the form area, there is a "< Back" link.

Gambar 1.89 Rancangan Antarmuka Pendaftaran Rawat Inap

Antarmuka ini akan muncul ketika user Melakukan Pendaftaran Rawat Inap pertama akan muncul form untuk memilih pasien kemudian bisa akan muncul form seperti diatas disitu kita akan memilih nama kamar dan kelas kamar sehingga akan muncul harga kemudian untuk memasukan dalam daftar pasien rawat inap langsung saja melakukan pendaftaran dengan menekan tombol daftar.

1.8.28 Antarmuka Tambah Layanan

The screenshot shows a web application interface for inpatient registration. The main heading is "Pendaftaran Rawat Inap". A navigation bar at the top contains four buttons: "Daftar Rawat Inap", "Tambah layanan", "Daftar Pasien Keluar", and "Log Out". The central form area includes the following elements:

- "No MR" and "Tanggal" (Date) input fields.
- "Nama Layanan" (Service Name) dropdown menu with "Selection..." as the current selection.
- "Harga" (Price) input field.
- "Add Layanan" and "Cancel" buttons.

At the bottom left of the form area, there is a "< Back" button.

Gambar 1.90 Rancangan Antarmuka Daftar Pasien Periksa

Antarmuka ini muncul ketika user memilih untuk menambah layanan untuk pasien rawat inap jadi pertama perawat akan memilih pasien yang akan ditambah layanannya kemudian akan muncul form diatas kemudian memilih layanan yang akan digunakan beserta harganya setelah semua layanan dipilih maka akan muncul form baru untuk semua layanan kemudian pilih tombol save untuk menyimpan layanan yang diinginkan.

Program Studi Teknik Informatika	DPPL – SIRS	163/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1.8.29 Antarmuka Cetak Bukti Pendaftaran

The screenshot shows a web interface for 'Pendaftaran Pemeriksaan'. At the top, there's a header with the title 'Pendaftaran Pemeriksaan'. Below the header is a navigation bar with four buttons: 'Tambah Pasien', 'Daftar Pemeriksaan', 'cetak bukti daftar', and 'Log Out'. The main content area is titled 'Pencarian' and contains a search form. The search form has a dropdown menu labeled 'Cari Berdasarkan' with a value of 'Sel...' and a 'Cari' button. Below the search form are two input fields: 'No MR' and 'Nama Pasien'. A 'Cetak' button is positioned below the 'Nama Pasien' field. At the bottom left of the main content area, there is a '< Back' link.

Gambar 1.91 Rancangan Antarmuka Cetak Bukti Pendaftaran

Antarmuka ini muncul ketika user memilih untuk cetak bukti pendaftaran. Masukkan no pasien yang akan di cetak atau berdasarkan no medical record.

1.8.30 Antarmuka Pemeriksaan Awal

Pertama akan muncul form untuk memilih pasien seperti sebelumnya kemudian yang akan menampilkan data pasien yang statusnya masuk di hari itu. Kemudian perawat melakukan pemeriksaan awal dengan pilih link "Pemeriksaan Awal" kemudian akan menuju ke form berikut :

Program Studi Teknik Informatika	DPPL – SIRS	164/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

Pemeriksaan Awal

Pemeriksaan Awal Edit Pemeriksaan Log Out

No MR

Nama Pasien

Tensi Berat Badan Tinggi Badan

Suhu Alergi

Keluhan

Save Cancel

< Back

Gambar 1.92 Rancangan Antarmuka Input Pemeriksaan Awal

Perawat kemudian memasukkan data pemeriksaan pasien setelah melakukan pemeriksaan awal. Setelah data terisi maka klik button "Save" kemudian tersimpan jika masih ingin melakukan edit data pemeriksaan awal maka klik tab edit pemeriksaan maka akan muncul antar muka seperti berikut:

1.8.31 Antarmuka Ubah Pemeriksaan Awal

Pemeriksaan Awal

Pemeriksaan Awal Edit Pemeriksaan Log Out

No MR

Nama Pasien

Tensi Berat Badan Tinggi Badan

Suhu Alergi

Keluhan

Ubah Cancel

< Back

Gambar 1.93 Rancangan Antarmuka Ubah Hasil Pemeriksaan

Antarmuka ini muncul jika perawat atau dokter melakukan perubahan pemeriksaan awal. Pertama masukkan data yang diubah sesuai dengan kebutuhan setelah data di ubah klik button Ubah data. Pemeriksaan awal selesai dilakukan.

1.8.32 Antarmuka Pemeriksaan Dokter

The screenshot shows a web application interface for a doctor's examination. The main heading is "Pemeriksaan Dokter". Below the heading is a navigation menu with four items: "Pemeriksaan Awal", "Pemeriksaan Dokter" (which is highlighted), "Kelola Resep", and "Log Out". The main content area contains a form with the following fields and buttons:

- Input fields for "No MR", "Tanggal Periksa", and "Jenis Pemeriksaan".
- Input fields for "Nama Pasien" and "Rujuk ke".
- Input fields for "Diagnosa" and "Konsultasi".
- Input fields for "Tindakan Dokter" and "Anjuran".
- An input field for "Catatan".
- Buttons for "Save", "Cancel", and "Kelola Resep".
- A "< Back" button at the bottom left.

Gambar 1.94 Rancangan Antarmuka Pemeriksaan Dokter

Antarmuka ini muncul ketika user login sebagai dokter. Muncul form memilih pasien yang akan diperiksa sesuai urutan kemudian akan muncul form diatas kemudian dokter akan mengisikan data pemeriksaan seperti diagnosa, tindakan yang dilakukan, anjuran, konsultasi dan memilih resep jika resep dipilih akan muncul form sebagai berikut:

1.8.33 Antarmuka Pengelolaan Resep

The screenshot shows a web application interface for managing prescriptions. The title is "Pengelolaan Resep". There are four navigation buttons at the top: "Pemeriksaan Awal", "Edit Pemeriksaan", "Kelola Resep", and "Log Out". The main content area contains several input fields: "No Resep", "Nama Obat", "Jenis Obat", "Harga", "Keerangan", and "Aturan Pakai". Below these are two more input fields: "Jumlah yang dibeli" and "Stock". At the bottom of the form are two buttons: "Add" and "Cancel". A "< Back" button is located at the bottom left of the form area.

Gambar 1.95 Rancangan Antarmuka Inputan Pemeriksaan Dokter

Disini dokter akan memilih obat kemudian mengisikan jumlah obat yang dibeli kemudian jika akan menambahkan obat kembali pilih add kemudian setelah semua dipilih akan muncul form untuk melihat semua data obat yang dipilih setelah itu klik save maka akan kembali ke form pemilihan obat dan klik save maka akan tersimpan di database resep yang telah dipilih dan juga pemeriksaan yang telah dilakukan dokter.

1.8.33.1 Antarmuka Pembayaran Rumah Sakit

Antarmuka ini akan muncul saat user login sebagai kasir dan memilih menu Pembayaran Rumah Sakit. Pertama akan tampil data pasien dengan total pembayaran yang sudah melakukan pemeriksaan Rumah Sakit, kemudian klik link "Lakukan Pembayaran" untuk transaksi pembayaran, yang akan muncul antarmuka sebagai berikut :

The screenshot shows a web interface for hospital payments. At the top, there's a title 'Pembayaran Rumah Sakit' and three buttons: 'Pembayaran', 'Cetak Nota', and 'Log Out'. Below this, there are two columns of input fields. The left column lists various cost categories: 'No Kwitansi', 'Biaya Obat', 'Biaya Dokter', 'Biaya Lab', 'Biaya Layanan', 'Biaya Kamar', and 'Total Biaya'. The right column lists payment-related fields: 'Pembayaran', 'Dibayar', 'Biaya Total', and 'Kembalian'. A 'Hitung' button is positioned below the 'Kembalian' field. At the bottom left of the form area, there is a '< Back' button.

Gambar 1.96 Rancangan Antarmuka Detail Pembayaran

User akan melakukan transaksi. Pada antarmuka ini langsung ada jumlah total dan rincian biaya yang digunakan dan harus di bayar pasien. User memasukkan uang cash yang di beri pasien. Kemudian klik "Hitung" maka uang kembalian langsung tampil ke textbox kembalian. Pembayaran Rumah Sakit selesai di lakukan.

Program Studi Teknik Informatika	DPPL – SIRS	169/161
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		