

BAB 6

KESIMPULAN DAN SARAN

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

8.1 Kesimpulan

Berdasarkan pembahasan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan yaitu:

1. Sistem Informasi Klinik Gigi Dental Smile (SIGIDES) Menggunakan Framework CodeIgniter berhasil dibangun dengan tools pengembang Sublime Text 2, Notepad++, dan Adobe Dreamweaver CS3 dan dapat diakses oleh banyak pengguna yang memiliki hak akses berbeda meskipun data terpusat.
2. Sistem Informasi Klinik Gigi ini dapat membantu proses bisnis dan administrasi Klinik Gigi Dental Smile.

8.2 Saran

Saran yang dapat diambil dari proses analisis hingga pembuatan tugas akhir ini adalah:

1. Pengembangan lebih lanjut diharapkan SIGIDES ada pengembangan untuk versi *mobile web* yang dapat diakses pasien.
2. Penambahan basisdata obat yang pengelolaannya terpisah dari tanggung jawab dokter.
3. Pemberian fitur untuk *backup* data.

SKPL

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

SIGIDES

(Sistem Informasi Klinik Gigi Dental Smile)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Bernadus Brian Budiawan / 09 07 05834

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halaman
		SKPL-SIGIDES		1/42
		Revisi		

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	

INDEX TGL	-	A	B	C	D	E	F	G

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi
			

Daftar Isi

1. Pendahuluan.....	6
1.1.Tujuan.....	6
1.2.Lingkup Masalah.....	6
1.3.Definisi, Akronim, dan Singkatan.....	7
1.4.Referensi.....	8
1.5.Deskripsi Umum (Overview).....	8
2. Deskripsi Kebutuhan.....	9
2.1.Perspektif Produk.....	9
2.2.Fungsi Produk.....	11
2.3.Karakteristik Pengguna.....	17
2.4.Batasan-Batasan.....	18
2.5.Asumsi dan Ketergantungan.....	18
3. Kebutuhan Khusus.....	18
3.1.Kebutuhan antarmuka eksternal.....	18
3.1.1. Antarmuka pemakai.....	18
3.1.2. Antarmuka perangkat keras.....	18
3.1.3. Antarmuka perangkat lunak.....	19
3.1.4. Antarmuka komunikasi.....	20
3.2.Kebutuhan fungsionalitas Perangkat Lunak.....	21
3.2.1. Use Case Diagram.....	21
4. Spesifikasi Rinci Kebutuhan.....	22
4.1.Spesifikasi Kebutuhan Fungsionalitas.....	22
4.1.1. Use Case Specification: Login.....	22
4.1.2. Use Case Specification: Ubah Password.....	23
4.1.3. Use Case Specification: Pengelolaan Data Pasien.....	24
4.1.4. Use Case Specification: Pengelolaan Data Pegawai.....	26
4.1.5. Use Case Specification: Pengelolaan Data Tindakan.....	28
4.1.6. Use Case Specification: Pengelolaan Data Diagnosa.....	30
4.1.7. Use Case Specification: Pengelolaan Data Rekam Medis.....	33
4.1.8. Use Case Specification: Pembayaran Klinik.....	35
4.1.9. Use Case Specification: Pengelolaan Jadwal Praktek Dokter.....	36
4.1.10. Use Case Specification: Pengelolaan Antrian Pasien.....	38
4.1.11. Use Case Specification: Menampilkan Laporan Klinik.....	39
5. Entity Relationship Diagram (ERD).....	42

Daftar Gambar

Gambar 1.	Arsitektur Perangkat Lunak SIGIDES	10
Gambar 2.	<i>Use Case Diagram</i> untuk SIGIDES	21
Gambar 3.	<i>Entity Relationship Diagram</i> untuk SIGIDES	42



1 Pendahuluan

1.1 Tujuan

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini merupakan dokumen spesifikasi kebutuhan perangkat lunak SIGIDES (Sistem Informasi Klinik Gigi Dental Smile) untuk mendefinisikan kebutuhan perangkat lunak yang meliputi antarmuka eksternal (antarmuka antara sistem dengan sistem lain perangkat lunak dan perangkat keras, dan) 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-SIGIDES ini juga mendefinisikan batasan perancangan perangkat lunak.

1.2 Lingkup Masalah

Perangkat Lunak SIGIDES dikembangkan dengan tujuan untuk :

1. Menangani pengelolaan data diagnosa dan tindakan medis dari dokter untuk pasien.
2. Menangani pengelolaan data pengguna yang memiliki hak akses secara khusus terhadap fungsi-fungsi yang ada dalam SIGIDES.
3. Menangani pengelolaan jadwal praktek dokter yang akan digunakan dalam proses bisnis Klinik Gigi Dental Smile.
4. Menangani pendaftaran pasien baru dan cetak kartu pemeriksaan pasien.
5. Menangani pendaftaran antrean pasien pemeriksaan klinik.

Program Studi Teknik Informatika	SKPL – SIGIDES	6/42
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. Menangani pemeriksaan oleh dokter untuk pasien yang berupa menampilkan riwayat diagnosa dan pemeriksaan terakhir, menerima keluhan dari pasien, memberikan anamnesa oleh dokter, memberikan pilihan diagnosa beserta posisi giginya untuk dokter, memberikan pilihan tindakan medis beserta posisi giginya untuk dokter.
7. Menangani penghitungan biaya dan pencetakan nota atas tindakan medis yang telah dilakukan dokter kepada pasien dengan pengguna kasir.
8. Menampilkan laporan kepada manager klinik berupa laporan pasien atau pengunjung, laporan diagnose dan tindakan medis, dan laporan pemasukan.

Dan berjalan pada lingkungan dengan platform PC yang terhubung dengan internet.

1.3 Definisi, Akronim dan Singkatan

Beberapa definisi, akronim, singkatan yang digunakan dalam SKPL SIGIDES ini antara lain :

Keyword/Phrase	Definisi
SKPL	Merupakan spesifikasi kebutuhan dari perangkat lunak yang akan dikembangkan.
SKPL-SIGIDES-XXX	Kode yang merepresentasikan kebutuhan pada SIGIDES (Sistem Informasi Klinik Gigi Dental Smile) dimana XXX merupakan nomor fungsi

	produk.
SIGIDES	Perangkat lunak untuk pengelolaan proses bisnis dalam Klinik Gigi Dental Smile.
Role	Hak akses yang dimiliki pengguna untuk menggunakan sistem
Web-Based System	Sebuah sistem yang berbasis jaringan, sehingga pengaksesannya bisa dilakukan melalui internet dengan terhubung ke database sistem.

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. MSDN Library-October 2005, Microsoft, 2005.
4. Wibisono, Aryo. SKPL SIAMA: Sistem Informasi Akademik Mahasiswa. 2010.
5. Budiawan, Bernadus Brian. SKPL SPP: Sistem Penjualan Perabotan. 2012.
6. Novitasari, Vitalis Dian. SKPL SIPAS: Sistem Informasi Poliklinik Asy-Syifa. 2012.

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

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

2 Deskripsi Kebutuhan

2.1 Perspektif Produk

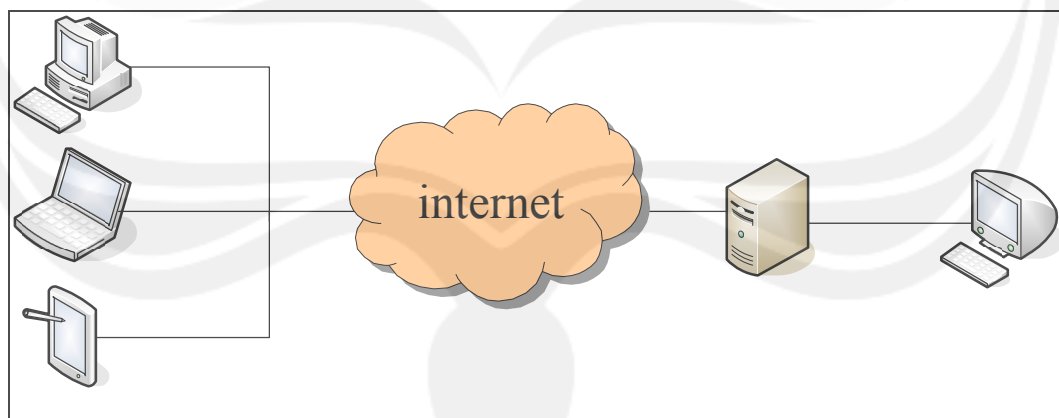
SIGIDES merupakan perangkat lunak yang dikembangkan untuk membantu pengelolaan proses bisnis Klinik Gigi Dental Smile. Sistem ini menangani pendaftaran pasien baru. Kemudian saat dilakukan pemeriksaan sistem akan membantu untuk mencatat dalam data rekam medis. Kemudian setelah pemeriksaan selesai yang dilakukan sistem adalah menghitung total biaya yang harus dibayarkan oleh pasien. Untuk Manager, sistem ini dapat memberikan laporan pengunjung klinik, laporan pendapatan klinik dan laporan penyakit.

Perangkat lunak SIGIDES ini berjalan pada komputer *desktop* ataupun komputer tablet yang memiliki aplikasi *browser* dan terhubung dengan internet. Untuk

Program Studi Teknik Informatika	SKPL – SIGIDES	9/42
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		

pembangunan SIGIDES penulis menggunakan Framework CodeIgniter (CI), bahasa pemrograman PHP, dan menggunakan database MySQL. Sedangkan untuk lingkungan pemrogramannya menggunakan Notepad++ dan Adobe DreamWeaver CS3.

Pengguna akan berinteraksi dengan sistem melalui antarmuka GUI (Graphical User Interface). Untuk masuk ke dalam sistem yang pertama dilakukan adalah memasukkan alamat situs sistemnya terlebih dahulu. Kemudian setelah pemanggilan alamat situs, dari sistem akan menampilkan GUI untuk login sistem. Setelah masuk ke dalam sistem sesuai dengan role masing-masing pengguna sistem. Untuk arsitektur perangkat lunaknya bisa dilihat pada Gambar 1. Arsitektur perangkat lunak yang digunakan berupa *client server*, di mana semua data disimpan di server dan bisa diakses oleh pengguna sistem sebagai kliennya



Gambar 1. Arsitektur Perangkat Lunak SIGIDES

Pengguna dapat mengakses data yang ada di server dengan sistem *login* yang memiliki koneksi internet. Dari *login* tersebut kemudian data dikirim ke server. Proses berikutnya tetap menggunakan jaringan internet untuk berhubungan dengan Server, DBMS, dan Klien.

Data yang dimasukkan akan disimpan dalam *database server*. Jika ada pencarian data, maka data yang diinginkan akan dicari ke *database server* yang selanjutnya dikirimkan ke klien yang mengirim permintaan melalui komputer *desktop* atau komputer *tablet*.

2.2 Fungsi Produk

Fungsi produk perangkat lunak SIGIDES adalah sebagai berikut :

1. Fungsi *Login* (**SKPL-SIGIDES-001**) .

Merupakan fungsi yang digunakan oleh pengguna untuk dapat masuk dalam sistem yang akan digunakan sesuai dengan *role* atau aturan pengguna.

2. Fungsi *Ubah Password*(**SKPL-SIGIDES-002**) .

Merupakan fungsi yang digunakan oleh pengguna untuk dapat memperbaharui *password* lama dengan *password* baru.

3. Fungsi *Pengelolaan Data Pasien* (**SKPL-SIGIDES-003**) .

Merupakan fungsi yang digunakan untuk mengelola data pasien di dalam sistem.

Fungsi pengelolaan pasien dalam sistem meliputi:

a. Fungsi *Entry Data Pasien*(**SKPL-SIGIDES-003-01**) .

Merupakan fungsi yang digunakan untuk menambah data pasien baru dalam sistem.

b. Fungsi *Edit Data Pasien*(**SKPL-SIGIDES-003-02**) .

Merupakan fungsi yang digunakan untuk mengubah data pasien yang sudah ada dalam sistem.

c. Fungsi *Display Data Pasien*(**SKPL-SIGIDES-003-03**) .

Program Studi Teknik Informatika	SKPL – SIGIDES	11/42
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 untuk menampilkan data pasien.

d. Fungsi *Non-Aktif Data Pasien*(**SKPL-SIGIDES-003-04**) .

Merupakan fungsi yang digunakan untuk menonaktifkan data pasien dari sistem bukan menghapusnya.

e. Fungsi *Cetak Kartu Periksa*(**SKPL-SIGIDES-003-05**) .

Merupakan fungsi yang digunakan untuk mencetak kartu periksa pasien sebagai alat untuk pendaftaran pemeriksaan

4. Fungsi *Pengelolaan Data Pegawai* (**SKPL-SIGIDES-004**) .

Merupakan fungsi yang digunakan untuk mengelola data pegawai di dalam sistem.

Fungsi pengelolaan pegawai dalam sistem meliputi:

a. Fungsi *Entry Data Pegawai*(**SKPL-SIGIDES-004-01**) .

Merupakan fungsi yang digunakan untuk menambah data pegawai baru dalam sistem.

b. Fungsi *Edit Data Pegawai*(**SKPL-SIGIDES-004-02**) .

Merupakan fungsi yang digunakan untuk mengubah data pegawai.

c. Fungsi *Display Data Pegawai*(**SKPL-SIGIDES-004-03**) .

Merupakan fungsi yang digunakan untuk menampilkan data pegawai.

d. Fungsi *Non-Aktif Data Pegawai* (**SKPL-SIGIDES-004-04**) .

Merupakan fungsi yang digunakan untuk menonaktifkan data pegawai dari sistem bukan menghapusnya.

5. Fungsi *Pengelolaan Data Tindakan* (**SKPL-SIGIDES-005**) .

Merupakan fungsi yang digunakan untuk mengelola data tindakan medis dokter di dalam sistem.

Fungsi pengelolaan data tindakan medis dokter dalam sistem meliputi:

a. Fungsi *Entry Data Tindakan* (**SKPL-SIGIDES-005-01**) .

Merupakan fungsi yang digunakan untuk menambah data tindakan medis baru dalam sistem.

b. Fungsi *Edit Data Tindakan* (**SKPL-SIGIDES-005-02**) .

Merupakan fungsi yang digunakan untuk mengubah data tindakan medis yang sudah ada di dalam sistem.

c. Fungsi *Display Data Tindakan* (**SKPL-SIGIDES-005-03**) .

Merupakan fungsi yang digunakan untuk menampilkan data tindakan medis.

d. Fungsi *Non-Aktif Data Tindakan* (**SKPL-SIGIDES-005-04**) .

Merupakan fungsi yang digunakan untuk menonaktifkan data tindakan medis dari sistem bukan menghapusnya

6. Fungsi *Pengelolaan Data Diagnosa* (**SKPL-SIGIDES-006**) .

Merupakan fungsi yang digunakan untuk mengelola data diagnosa di dalam sistem.

Fungsi pengelolaan data diagnosa dalam sistem meliputi:

a. Fungsi *Entry Data Diagnosa* (**SKPL-SIGIDES-006-01**) .

Merupakan fungsi yang digunakan untuk menambah data diagnosa baru dalam sistem.

b. Fungsi *Edit Data Diagnosa* (**SKPL-SIGIDES-006-02**) .

Merupakan fungsi yang digunakan untuk mengubah data diagnosa yang ada dalam sistem.

c. Fungsi *Display Data Diagnosa* (**SKPL-SIGIDES-006-03**) .

Merupakan fungsi yang digunakan untuk menampilkan data diagnosa.

d. Fungsi *Non-Aktif Data Diagnosa* (**SKPL-SIGIDES-006-04**) .

Merupakan fungsi yang digunakan untuk menonaktifkan data diagnosa dari sistem tanpa menghapusnya

7. Fungsi *Pengelolaan Data Rekam Medis* (**SKPL-SIGIDES-007**) .

Merupakan fungsi yang digunakan untuk mengelola data Rekam Medis di dalam sistem yang berupa pencatatan hasil pemeriksaan pasien.

Fungsi pengelolaan data rekam medis dalam sistem meliputi:

Program Studi Teknik Informatika	SKPL – SIGIDES	14/42
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 *Generate No Rekam Medis* (**SKPL-SIGIDES-007-01**) .

Merupakan fungsi yang digunakan untuk melakukan penomoran otomatis ketika ada penambahan data pemeriksaan dalam sistem.

b. Fungsi *Entry Data Pemeriksaan* (**SKPL-SIGIDES-007-02**) .

Merupakan fungsi yang digunakan untuk memasukkan data pemeriksaan pasien yang baru ke dalam Rekam Medis.

c. Fungsi *Update Data Pemeriksaan* (**SKPL-SIGIDES-007-03**) .

Merupakan fungsi yang digunakan untuk mengubah data pemeriksaan yang sudah tersimpan dalam sistem seperti mengubah status, mengisikan diagnosa, mengisikan tindakan medis, mengisikan total biaya, diskon, pembayaran dan kembalian yang dibayarkan pasien.

d. Fungsi *Search Data Pemeriksaan* (**SKPL-SIGIDES-007-04**) .

Merupakan fungsi yang digunakan untuk mencari data pemeriksaan pasien dalam rekam medis yang ada di sistem dan menampilkannya

8. Fungsi *Pembayaran Klinik* (**SKPL-SIGIDES-008**) .

Merupakan fungsi yang digunakan untuk melakukan pembayaran klinik dan mencetak bukti pembayaran oleh pasien.

Fungsi pembayaran klinik meliputi:

a. Fungsi *Pembayaran* (**SKPL-SIGIDES-008-01**)

Fungsi ini digunakan untuk melakukan penghitungan pembayaran pemeriksaan pasien mulai dari penjumlahan total dari tindakan, pemberian potongan harga dalam persen, penghitungan total setelah dikurangi diskon, pemasukan pembayaran dari pasien, dan penghitungan kembalian.

b. Fungsi *Cetak Pembayaran* (**SKPL-SIGIDES-008-02**)

Fungsi ini digunakan untuk mencetak bukti pembayaran pemeriksaan pasien beserta detail tindakannya.

9. Fungsi *Cetak Laporan* (**SKPL-SIGIDES-009**) .

Merupakan fungsi yang digunakan untuk melakukan pencetakan laporan yang dilakukan oleh manager. Laporan yang dicetak meliputi laporan pengunjung atau pasien, laporan diagnosa dan tindakan medis, dan laporan pemasukan harian, mingguan, serta bulanan.

10. Fungsi *Pengelolaan Antrean Pasien*(**SKPL-SIGIDES-010**) .

Merupakan fungsi yang digunakan oleh resepsionis untuk melakukan pendaftaran antrean bagi pasien yang akan diperiksa berdasarkan jadwal praktek yang tersedia pada hari dan jam itu.

11. Fungsi *Pengelolaan Jadwal Praktek Dokter*(**SKPL-SIGIDES-011**) .

Merupakan fungsi yang dilakukan oleh admin untuk melakukan pembuatan jadwal praktek untuk dokter.

Program Studi Teknik Informatika	SKPL – SIGIDES	16/42
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 pengelolaan jadwal praktek dokter meliputi:

a. Fungsi *Entry Jadwal Praktek Dokter*(**SKPL-SIGIDES-011-01**)

Merupakan fungsi yang digunakan untuk menambah data jadwal praktek baru dalam sistem.

b. Fungsi *Edit Jadwal Praktek Dokter*(**SKPL-SIGIDES-011-02**)

Merupakan fungsi yang digunakan untuk mengubah data jadwal praktek dokter dalam sistem.

c. Fungsi *Display Jadwal Praktek Dokter*(**SKPL-SIGIDES-011-03**)

Merupakan fungsi yang digunakan untuk melihat data jadwal praktek dokter dalam sistem.

d. Fungsi *Non-Aktif Jadwal Praktek Dokter* (**SKPL-SIGIDES-011-04**)

Merupakan fungsi yang digunakan untuk menonaktifkan jadwal praktek dokter yang ada di sistem tanpa menghapusnya.

2.3 Karakteristik Pengguna

Karakteristik dari pengguna perangkat lunak SIGIDES adalah sebagai berikut :

1. Pengguna harus memahami pengoperasian komputer *desktop*.
2. Pengguna harus memahami internet dan juga *web browser* untuk menjalankan sistem.
3. Pengguna juga diharapkan mampu memahami pengoperasian SIGIDES.

Program Studi Teknik Informatika	SKPL – SIGIDES	17/42
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		

2.4 Batasan-Batasan

Batasan-batasan dalam pengembangan perangkat lunak SIGIDES tersebut adalah :

1. Kebijaksanaan Umum

Berpedoman pada tujuan dari pengembangan perangkat lunak SIGIDES.

2. Keterbatasan perangkat keras

Dapat diketahui kemudian setelah sistem ini berjalan (sesuai dengan kebutuhan).

2.5 Asumsi dan Ketergantungan

Sistem ini dapat digunakan untuk aplikasi yang diakses melalui web browser.

3 Kebutuhan Khusus

3.1 Kebutuhan antarmuka eksternal

Kebutuhan antarmuka eksternal pada perangkat lunak SIGIDES meliputi kebutuhan antarmuka pemakai, antarmuka 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

Perangkat keras yang digunakan dalam sistem ini antara lain adalah :

Program Studi Teknik Informatika	SKPL – SIGIDES	18/42
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. Personal Computer (PC)
2. Mouse
3. Keyboard
4. Server : Web Server dan Database Server

3.1.3 Antarmuka perangkat lunak

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

1. Nama : MySQL
Sumber : phpMyAdmin
Sebagai database management system (DBMS) yang digunakan untuk menyimpan data di sisi server.
2. Nama : Windows XP/Vista/7/8
Sumber : Microsoft.
Sebagai sistem operasi untuk pengguna desktop dimana aplikasi akan dijalankan.
3. Nama : Apache
Nomor Versi : 2.0
Sumber : Open Source (XAMPP)
Sebagai web server lokal untuk pembangunan aplikasi.
4. Nama : Framework CodeIgniter
Nomor Versi : 2.1.3
Sumber : Open Source Ellislab Inc.
Sebagai framework untuk membangun aplikasi.
5. 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 dan komputer pengguna.

6. Nama : PHP

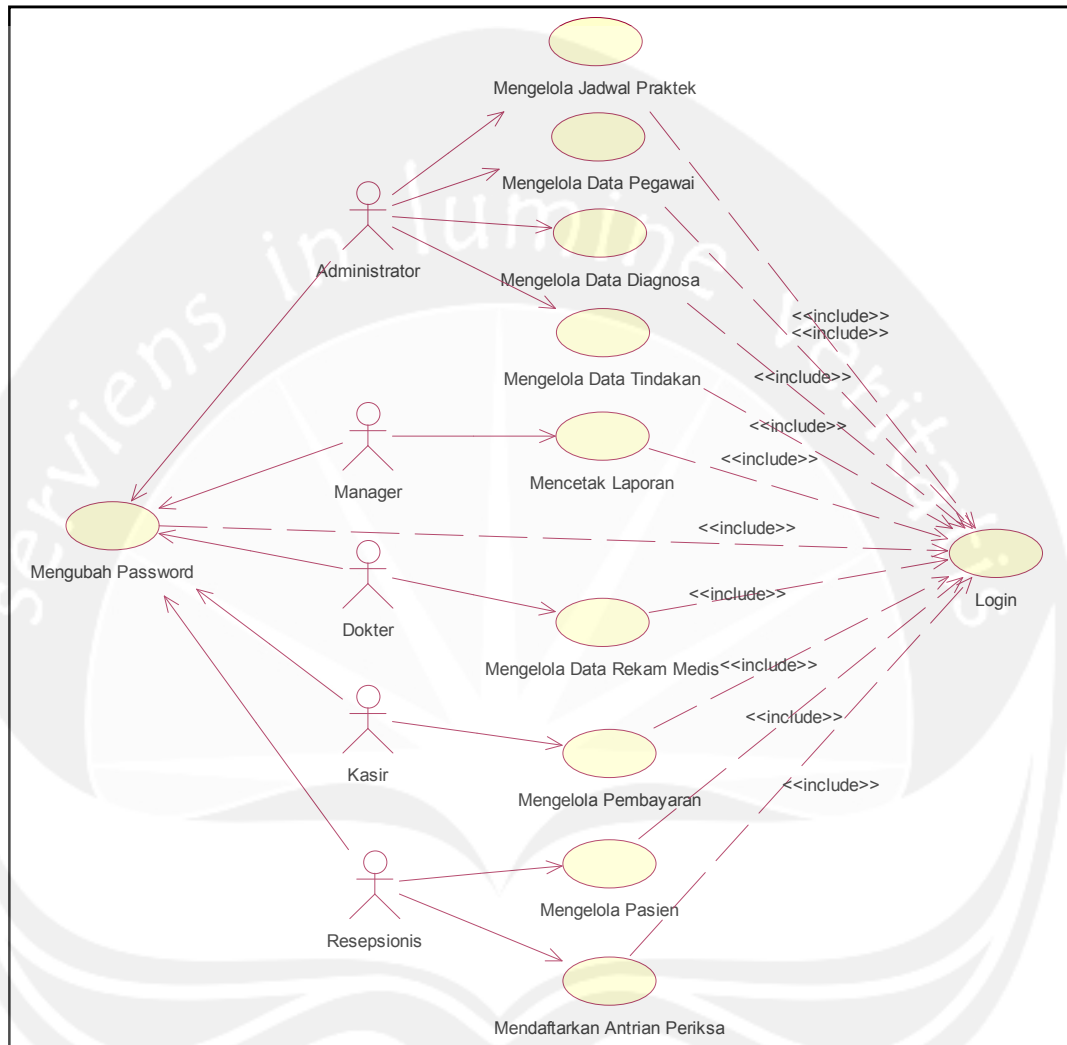
Nomor Versi : 5.4.7

Sumber : PHP

Sebagai bahasa pemrograman untuk membangun sistem.

3.1.4 Antarmuka Komunikasi

Antarmuka komunikasi perangkat lunak SIGIDES menggunakan protocol HTTP.



4 Spesifikasi Rinci Kebutuhan

4.1 Spesifikasi Kebutuhan Fungsionalitas

4.1.1 Use case Specification : Login

1. Brief Description

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

2. Primary Actor

1. Administrator
2. Manager
3. Dokter
4. Kasir
5. Resepsionis

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan login ke dalam sistem
2. Sistem menampilkan antarmuka untuk login
3. Aktor memasukkan id dan password
4. Sistem memeriksa id dan password yang diinputkan aktor

E-1 Password atau id user tidak sesuai

5. Sistem memberikan akses sistem kepada aktor
6. Use Case ini selesai

5. Alternative Flow

none

6. Error Flow

E-1 Password atau nama user tidak sesuai

1. Sistem menampilkan peringatan bahwa id user atau password tidak sesuai

Program Studi Teknik Informatika	SKPL – SIGIDES	22/42
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. Kembali ke Basic Flow langkah ke 3

7. PreConditions

none

8. PostConditions

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

4.1.2 Use case Specification : Ubah Password

1. Brief Description

Use Case ini digunakan oleh administrator, dokter, karyawan, kasir, manager untuk mengubah password.

2. Primary Actor

1. Administrator
2. Manager
3. Dokter
4. Kasir
5. Resepsionis
6. Pasien

3. Supporting Actor

none

4. Basic Flow

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

Program Studi Teknik Informatika	SKPL – SIGIDES	23/42
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 ini selesai dilakukan.

5. Alternative Flow

None.

6. Error Flow

E-1 Password lama yang dimasukkan salah

1. Sistem memberikan pesan peringatan bahwa password lama yang diinputkan salah.
2. Kembali ke Basic Flow langkah ke 3

7. PreConditions

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

8. PostConditions

1. Data password user di database telah diperbaharui.

4.1.3 Use case Specification :Pengelolaan Data Pasien

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data pasien. Aktor dapat melakukan entry data pasien, ubah data pasien, non-aktif 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, ubah data pasien, non-aktif data pasien.
3. Aktor memilih melakukan entry data pasien.

- A-1 Aktor memilih untuk melakukan ubah data pasien
- A-2 Aktor memilih untuk melakukan non-aktif data pasien
4. Sistem menampilkan form untuk menginputan data pasien baru.
5. Aktor memasukkan data pasien baru.
- E-1 Form idPasien kosong.
- E-2 Form namaPasien kosong.
6. Aktor meminta system untuk menyimpan data pasien yang baru diinputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah data pasien.
1. Aktor memilih data pasien yang akan di ubah.
2. Sistem menampilkan form untuk mengubah data pasien.
3. Aktor mengubah data pasien yang sudah di tampilkan.
- E-1 Form idPasien kosong.
- E-2 Form namaPasien kosong.
4. Aktor meminta system untuk menyimpan data pasien yang telah diubah.
5. Sistem menyimpan data pasien yang baru yang telah diubah.
6. Berlanjut ke Basic Flow langkah 7
- A-2 Aktor memilih untuk melakukan non-aktif data pasien
1. Sistem menampilkan data-data pasien.

2. Aktor memilih data pasien yang akan dinonaktifkan.
3. Sistem menonaktifkan data pasien yang sudah dipilih.
4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form Nama idPasien kosong.

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

E-2 Form namaPasien kosong.

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

7. PreConditions

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

8. PostConditions

1. Data Pasien di dalam database telah terupdate.

4.1.4 Use case Specification : Pengelolaan Data Pegawai

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data pegawai yang meliputi entry data pegawai, ubah data pegawai, menonaktifkan data pegawai.

2. Primary Actor

1. Administrator

3. Supporting Actor

Program Studi Teknik Informatika	SKPL – SIGIDES	26/42
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		

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, ubah data pegawai, non-aktif data pegawai.
3. Aktor memilih melakukan entry data pegawai.
 - A-1 Aktor memilih untuk melakukan ubah data pegawai
 - A-2 Aktor memilih untuk melakukan non-aktif data pegawai
4. Sistem menampilkan form untuk menginputan data pegawai baru.
5. Aktor memasukkan data pegawai baru.
 - E-1 Form pegawai harus terisi semua
6. Aktor meminta system untuk menyimpan data pegawai yang baru di inputkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah data pegawai
 1. Aktor memilih data yang akan diubah.
 2. Sistem menampilkan form untuk mengubah data pegawai
 3. Aktor mengubah data pegawai yang sudah ditampilkan.
 - E-1 Form pegawai harus terisi semua
 4. Aktor meminta system untuk menyimpan data pegawai yang telah diubah.

Program Studi Teknik Informatika	SKPL – SIGIDES	27/42
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. Sistem menyimpan data pegawai yang baru yang telah diubah.

6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan non-aktif data pegawai

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

6. Error Flow

E-1 Form ada yang kosong

1. Sistem akan error karena data pegawai harus dilengkapi.
2. Berlanjut ke basic flow no 5.

7. PreConditions

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

8. PostConditions

1. Data pegawai di dalam database telah diperbaharui

4.1.5 Use case Specification :Pengelolaan Data Tindakan

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data tindakan medis dokter. Aktor dapat melakukan entry jadwal data tindakan baru, ubah data tindakan, non-aktifkan data tindakan.

Program Studi Teknik Informatika	SKPL – SIGIDES	28/42
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

1. Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data tindakan.
2. Sistem memberikan pilihan untuk melakukan entry data tindakan baru, ubah data tindakan, non-aktif data tindakan.
3. Aktor memilih melakukan entry data tindakan.
 - A-1 Aktor memilih untuk melakukan ubah data tindakan.
 - A-2 Aktor memilih untuk melakukan non-aktif data tindakan.
4. Sistem menampilkan form untuk memasukkan data tindakan baru.
5. Aktor memasukkan data tindakan baru.
 - E-1 Form harus diisi semua kecuali gambar.
6. Aktor meminta system untuk menyimpan data tindakan yang baru dimasukkan
7. Use Case ini selesai dilakukan

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah data tindakan.
 1. Aktor memilih data yang akan diubah.
 2. Sistem menampilkan form untuk mengubah data tindakan.
 3. Aktor mengubah data tindakan yang sudah ditampilkan.

E-1 Form harus diisi semua kecuali gambar.

4. Aktor meminta system untuk menyimpan data tindakan yang telah diubah.
 5. Sistem menyimpan data tindakan yang baru yang telah diubah.
 6. Berlanjut ke Basic Flow langkah 7
- A-2 Aktor memilih untuk melakukan non-aktif data tindakan.
1. Sistem menampilkan data tindakan.
 2. Aktor memilih data tindakan yang akan dinon-aktifkan.
 3. Sistem menghapus data tindakan yang sudah dipilih.
 4. Berlanjut ke Basic Flow langkah 7

6. Error Flow

E-1 Form harus diisi semua kecuali gambar

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

7. PreConditions

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

8. PostConditions

1. Data tindakan dalam database telah terupdate

4.1.6 Use case Specification : Pengelolaan Data Diagnosa

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data diagnosa, meliputi entry data

Program Studi Teknik Informatika	SKPL – SIGIDES	30/42
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		

diagnosa, non-aktif data diagnosa dan ubah data diagnosa.

2. Primary Actor

1. Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data diagnosa.
2. Sistem memberikan pilihan untuk melakukan entry data diagnosa, ubah data diagnosa, delete data diagnosa.
3. Aktor memilih melakukan entry data diagnosa.
 - A-1 Aktor memilih untuk melakukan ubah data diagnosa
 - A-2 Aktor memilih untuk melakukan non-aktif data diagnosa.
4. Sistem menampilkan form untuk memasukkan data diagnosa baru.
5. Aktor memasukkan data diagnosa baru.
 - E-1 Form idDiagnosa harus diisi
 - E-2 Form namaDiagnosa harus diisi
6. Aktor meminta system untuk menyimpan data diagnosa yang baru dimasukkan
7. Use Case ini selesai dilakukan.

5. Alternative Flow

- A-1 Aktor memilih untuk melakukan ubah data diagnosa.
 1. Aktor memilih data yang akan diubah.

2. Sistem menampilkan form untuk mengubah data diagnosa.

3. Aktor mengubah data diagnosa yang sudah ditampilkan.

E-1 Form idDiagnosa harus diisi

E-2 Form namaDiagnosa harus diisi

4. Aktor meminta system untuk menyimpan data diagnosa yang telah diubah.

5. Sistem menyimpan data diagnosa yang baru yang telah diubah.

6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan non-aktif data diagnosa

1. Sistem menampilkan data-data diagnosa

2. Aktor memilih data diagnosa yang akan dinonaktifkan.

3. Sistem menonaktifkan data diagnosa yang sudah dipilih.

4. Berlanjut ke Basic Flow langkah 7.

6. Error Flow

E-1 Form idDiagnosa harus diisi

1. Sistem memberikan peringatan bahwa form idDiagnosa data diagnosa harus diisi.

2. Berlanjut ke basic flow no 5.

E-2 Form namaDiagnosa harus diisi

1. Sistem memberikan peringatan bahwa form namaDiagnosa data diagnosa harus diisi.

2. Berlanjut ke basic flow no 5.

7. PreConditions

1. Use Case Login telah dilakukan

2. Aktor telah memasuki sistem

Program Studi Teknik Informatika	SKPL – SIGIDES	32/42
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 diagnosa di database telah terupdate

4.1.7 Use case Specification : Pengelolaan Data Rekam Medis

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola data rekam medis, meliputi entry data pemeriksaan, cari data pemeriksaan, dan ubah data pemeriksaan.

2. Primary Actor

1. Dokter

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan data rekam medis.
2. Sistem memberikan pilihan untuk melakukan entry data pemeriksaan, cari data pemeriksaan, atau ubah data pemeriksaan.
3. Aktor memilih melakukan entry data pemeriksaan
 - A-1 Aktor memilih untuk melakukan cari data pemeriksaan
 - A-2 Aktor memilih untuk melakukan ubah data pemeriksaan
4. Sistem menampilkan form untuk menginputan data pemeriksaan rekam medis baru.
5. Aktor memasukkan data pemeriksaan.
 - E-1 Form pemeriksaan harus diisi semua kecuali keluhan.

6. Aktor meminta system untuk menyimpan data pemeriksaan yang baru dimasukkan

7. Use Case ini selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan cari data pemeriksaan.

1. Sistem menampilkan semua data pemeriksaan.
2. Aktor memasukkan kata kunci untuk mencari data pemeriksaan yang akan dilihat.
3. Sistem menampilkan data yang dicari sesuai dengan kata kunci yang dimasukkan.
4. Aktor memilih untuk melihat detail pemeriksaan.
5. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan ubah data pemeriksaan

1. Sistem menampilkan data-data rekam medis
2. Aktor memilih data pemeriksaan yang akan diubah.
3. Sistem menampilkan data pemeriksaan yang akan diubah berserta form untuk pengubahan data.
4. Aktor melakukan pengubahan data pemeriksaan.

E-1 Form pemeriksaan harus diisi semua kecuali keluhan.

5. Berlanjut ke Basic Flow langkah 6.

6. Error Flow

Program Studi Teknik Informatika	SKPL – SIGIDES	34/42
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-1 Form pemeriksaan terdapat data yang belum terisi kecuali keluhan.

1. Sistem memberikan pesan peringatan bahwa data pemeriksaan wajib diisi semua.
2. Kembali ke Basic Flow Langkah ke 5

7. PreConditions

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

8. PostConditions

1. Data rekam medis pemeriksaan pasien di database telah terupdate

4.1.8 Use case Specification : Pembayaran Klinik

1. Brief Description

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

2. Primary Actor

1. Kasir

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pembayaran klinik.
2. Sistem menampilkan data pasien yang sudah melakukan pemeriksaan klinik.
3. Aktor melakukan pembayaran dan memasukkan uang cash yang di berikan pasien
4. Aktor melakukan cetak pembayaran
5. Use Case selesai

5. Alternative Flow

Program Studi Teknik Informatika	SKPL – SIGIDES	35/42
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		

none

6. Error Flow

none

7. PreConditions

1. Use Case Login telah dilakukan
2. Aktor telah memasuki system
3. Pasien sudah diperiksa dan belum melakukan pembayaran

8. PostConditions

1. Pasien telah melakukan pembayaran pemeriksaan.

4.1.9 Use case Specification :Pengelolaan Jadwal Praktek Dokter

1. Brief Description

Use Case ini digunakan oleh aktor untuk mengelola jadwal praktek dokter. Aktor dapat melakukan entry jadwal praktek dokter, ubah jadwal praktek dokter, non-aktif jadwal praktek dokter.

2. Primary Actor

Administrator

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pengelolaan jadwal praktek dokter.
2. Sistem memberikan pilihan untuk melakukan entry jadwal praktek dokter, ubah jadwal praktek dokter, non-aktif jadwal praktek dokter.

3. Aktor memilih melakukan entry jadwal praktek dokter.

A-1 Aktor memilih untuk melakukan ubah jadwal praktek dokter

A-2 Aktor memilih untuk melakukan non-aktif jadwal praktek dokter.

4. Sistem menampilkan form untuk memasukkan jadwal praktek dokter.

5. Aktor memasukkan jadwal praktek dokter baru.

E-1 Form ada yang kosong.

6. Use case telah selesai dilakukan.

5. Alternative Flow

A-1 Aktor memilih untuk melakukan ubah jadwal praktek dokter.

1. Aktor memilih jadwal praktek dokter yang akan di ubah.

2. Sistem menampilkan form untuk mengubah jadwal praktek dokter.

3. Aktor mengubah jadwal praktek dokter yang sudah ditampilkan.

E-1 Form ada yang kosong.

4. Aktor meminta system untuk menyimpan jadwal praktek dokter yang telah diubah.

5. Sistem menyimpan jadwal praktek dokter yang baru yang telah diubah.

6. Berlanjut ke Basic Flow langkah 7

A-2 Aktor memilih untuk melakukan non-aktif data pasien

1. Sistem menampilkan data-data pasien.

2. Aktor memilih data pasien yang akan dinonaktifkan.

3. Sistem menonaktifkan data pasien yang sudah dipilih.

4. Berlanjut ke Basic Flow langkah 7.

6. Error Flow

E-1 Form ada yang kosong.

1. Sistem memberikan peringatan bahwa form harus diisi semua.

2. Berlanjut ke basic flow ke 5

7. PreConditions

1. Use Case Login telah dijalankan

2. Aktor telah masuk ke dalam system

8. PostConditions

1. Jadwal dokter telah masuk dalam basis data.

4.1.10 Use case Specification :Pengelolaan Antrian Pasien

1. Brief Description

Use Case ini digunakan oleh aktor untuk melakukan pengelolaan antrian berupa pendaftaran antrian untuk pemeriksaan.

2. Primary Actor

1. Resepsionis

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan pendaftaran antrian pemeriksaan klinik.

2. Sistem memberikan form nomor pasien dan password yang harus diisi oleh aktor.

3. Aktor memasukkan nomor pasien dan password.

Program Studi Teknik Informatika	SKPL – SIGIDES	38/42
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 jadwal yang tersedia untuk melakukan pemeriksaan.

5. Aktor memilih jadwal yang akan didaftarkan kemudian meminta system untuk menyimpan pilihan tersebut.

6. Sistem menyimpan hasil pilihan dari actor.

5. Use case selesai.

5. Alternative Flow

none

6. Error Flow

none

7. PreConditions

1. Use case login sudah dilakukan

2. Aktor sudah memasuki sistem

8. PostConditions

1. Pendaftaran antrian ke dalam system oleh actor telah selesai.

4.1.11 Use case Specification : Menampilkan Laporan Klinik

1. Brief Description

Use Case ini digunakan oleh aktor untuk menampilkan laporan pembayaran klinik, laporan penyakit, dan laporan pengunjung.

2. Primary Actor

Manager

3. Supporting Actor

none

4. Basic Flow

1. Use Case ini dimulai ketika aktor memilih untuk melakukan penyediaan laporan klinik.

Program Studi Teknik Informatika	SKPL – SIGIDES	39/42
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 menampilkan laporan kunjungan, laporan pendapatan dari pembayaran pasien periksa dan laporan penyakit yang telah ditangani klinik.

3. Aktor memilih untuk menampilkan laporan kunjungan dalam periode tertentu.

A-1. Aktor memilih agar sistem menampilkan laporan pendapatan klinik.

A-2. Aktor memilih agar system menampilkan laporan penyakit.

4. Sistem menyediakan laporan klinik periode tertentu.

5. Use case selesai

5. Alternative Flow

A-1. Aktor memilih agar sistem menampilkan laporan pendapatan klinik.

1. Aktor meminta sistem untuk menyediakan laporan pendapatan klinik pada periode tertentu.

2. Sistem menyediakan laporan pendapatan klinik pada periode tertentu.

3. Kembali pada Basic flow langkah ke 6.

A-2. Aktor memilih agar system menampilkan laporan penyakit.

1. Aktor meminta sistem untuk menyediakan laporan penyakit.

2. Sistem menyediakan laporan penyakit.

3. Kembali pada Basic flow langkah ke 6.

6. Error Flow

none

7. PreConditions

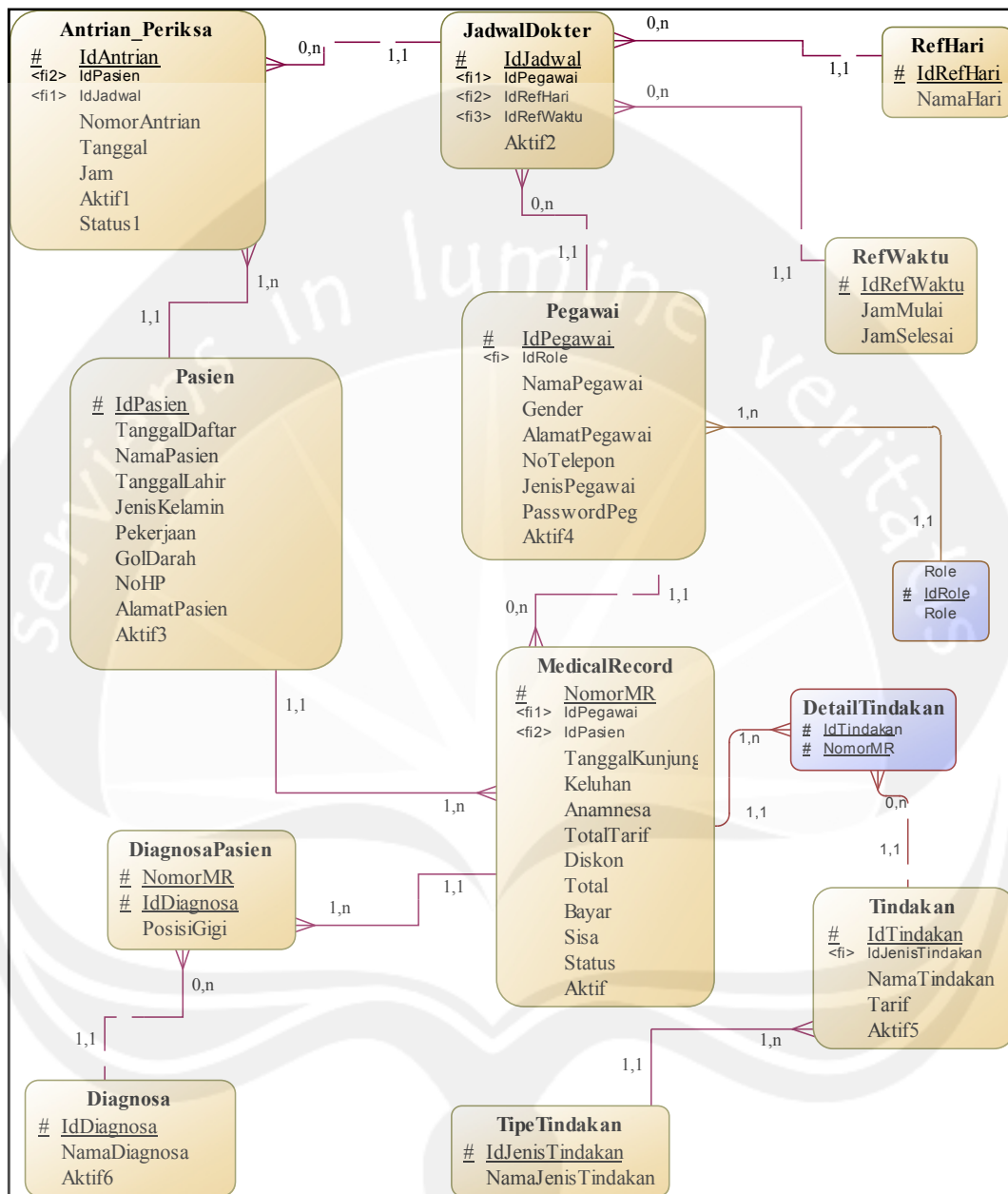
Program Studi Teknik Informatika	SKPL – SIGIDES	40/42
Dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-UAJY dan bersifat rahasia. Dilarang untuk me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika		

1. Use case login sudah dilakukan
2. Aktor sudah memasuki sistem

8. PostConditions

1. Penyediaan laporan klinik oleh aktor telah berhasil





DPPL

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

SIGIDES

(Sistem Informasi Klinik Gigi Dental Smile)

Untuk :

Universitas Atma Jaya Yogyakarta

Dipersiapkan oleh:

Bernadus Brian Budiawan / 09 07 05834

Program Studi Teknik Informatika - Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik Informatika Fakultas Teknologi Industri	Nomor Dokumen		Halama
		DPPL-SIGIDES		1/106
		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								
Disetu- jui oleh								

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

1. Pendahuluan.....	7
1.1. Tujuan.....	7
1.2. Ruang Lingkup.....	7
1.3. Definisi dan Akronim.....	8
1.4. Referensi.....	9
2. Perancangan Sistem.....	10
2.1. Diagram Arsitektur.....	10
2.2. Perancangan Detail.....	11
2.2.1. Sequence Diagram.....	11
2.2.1.1. Login.....	11
2.2.1.2. Ubah Passwor.....	12
2.2.1.3. Kelola Data Pasien.....	13
2.2.1.4. Kelola Data Pegawai.....	17
2.2.1.5. Kelola Data Tindakan.....	21
2.2.1.6. Kelola Data Diagnosa.....	25
2.2.1.7. Kelola Data Rekam Medis.....	29
2.2.1.8. Pembayaran Klinik.....	32
2.2.1.9. Kelola Cetak Laporan.....	33
2.2.1.10. Kelola Antrian Pasien.....	34
2.2.1.11. Kelola Jadwal Praktek Dokter.....	35
2.2.2. Class Diagram.....	39
2.2.3. Spesifikasi Deskripsi Kelas Diagram.....	41
2.2.3.1. Spesifikasi Deskripsi Kelas Login_V.....	41
2.2.3.2. Spesifikasi Deskripsi Kelas UbahPassword_V.....	41
2.2.3.3. Spesifikasi Deskripsi Kelas PengelolaanPegawai_V.....	41
2.2.3.4. Spesifikasi Deskripsi Kelas PengelolaanDiagnosa_V.....	42
2.2.3.5. Spesifikasi Deskripsi Kelas PengelolaanTindakanMedis_V.....	43
2.2.3.6. Spesifikasi Deskripsi Kelas PengelolaanJadwalDokter_V.....	44
2.2.3.7. Spesifikasi Deskripsi Kelas PengelolaanPasien_V.....	44
2.2.3.8. Spesifikasi Deskripsi Kelas PengelolaanAntrian_V.....	45
2.2.3.9. Spesifikasi Deskripsi Kelas PengelolaanRekamMedis_V.....	46
2.2.3.10. Spesifikasi Deskripsi Kelas PembayaranKlinik_V.....	46
2.2.3.11. Spesifikasi Deskripsi Kelas CetakLaporan_V.....	46
2.2.3.12. Spesifikasi Deskripsi Kelas Login_C.....	47
2.2.3.13. Spesifikasi Deskripsi Kelas UbahPassword_C.....	48
2.2.3.14. Spesifikasi Deskripsi Kelas PengelolaanPegawai_C.....	48
2.2.3.15. Spesifikasi Deskripsi Kelas PengelolaanDiagnosa_C.....	49

2.2.3.16.	Spesifikasi Deskripsi Kelas PengelolaanTindakanMedis_C.....	50
2.2.3.17.	Spesifikasi Deskripsi Kelas PengelolaanJadwalDokter_C.....	51
2.2.3.18.	Spesifikasi Deskripsi Kelas PengelolaanPasien_C.....	52
2.2.3.19.	Spesifikasi Deskripsi Kelas PengelolaanAntrian_C.....	53
2.2.3.20.	Spesifikasi Deskripsi Kelas PengelolaanRekamMedis_C.....	54
2.2.3.21.	Spesifikasi Deskripsi Kelas PembayaranKlinik_C.....	56
2.2.3.22.	Spesifikasi Deskripsi Kelas CetakLaporan_C.....	57
2.2.3.23.	Spesifikasi Deskripsi Kelas Pegawai_M.....	57
2.2.3.24.	Spesifikasi Deskripsi Kelas Diagnosa_M.....	58
2.2.3.25.	Spesifikasi Deskripsi Kelas TindakanMedis_M.....	59
2.2.3.26.	Spesifikasi Deskripsi Kelas JadwalDokter_M.....	60
2.2.3.27.	Spesifikasi Deskripsi Kelas Pasien_M.....	62
2.2.3.28.	Spesifikasi Deskripsi Kelas AntrianPeriksa_M.....	63
2.2.3.29.	Spesifikasi Deskripsi Kelas RekamMedis_M.....	64
3.	Perancangan Data.....	66
3.1.	Dekomposisi Data.....	66
3.1.1.	Deskripsi Entitas Data Master Pegawai.....	66
3.1.2.	Deskripsi Entitas Data Master Diagnosa.....	67
3.1.3.	Deskripsi Entitas Data Master Tindakan Medis.....	67
3.1.4.	Deskripsi Entitas Data Master Jadwal Pemeriksaan untuk Dokter.....	68
3.1.5.	Deskripsi Entitas Data Master Pasien.....	68
3.1.6.	Deskripsi Entitas Data Master Antrian Periksa...	69
3.1.7.	Deskripsi Entitas Data Master Rekam Medis.....	69
3.2.	Physical Data Model (PDM).....	71
4.	Perancangan Antar Muka.....	72
4.1.	Antarmuka Pengguna dan Deskripsinya.....	72
4.1.1.	Login View.....	72
4.1.2.	Change Password.....	73
4.1.3.	Halaman Admin.....	74
4.1.4.	Pengelolaan Pegawai.....	75
4.1.5.	Pengelolaan Tindakan Medis.....	79
4.1.6.	Pengelolaan Diagnosa.....	82
4.1.7.	Pengelolaan Jadwal Dokter.....	85
4.1.8.	Halaman Resepsionis.....	88
4.1.9.	Pengelolaan Pasien.....	89
4.1.10.	Pengelolaan Antrian.....	92
4.1.11.	Halaman Dokter.....	96
4.1.12.	Halaman Kasir.....	102
4.1.13.	Cetak Laporan.....	105

Daftar Gambar

1. Perancangan Arsitektur Perangkat Lunak SPP.....10
2. Class Diagram40



1 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 SIGIDES dikembangkan dengan tujuan untuk :

1. Menangani pengelolaan data diagnosa dan tindakan medis dari dokter untuk pasien.
2. Menangani pengelolaan data pengguna yang memiliki hak akses secara khusus terhadap fungsi-fungsi yang ada dalam SIGIDES.
3. Menangani pengelolaan jadwal praktek dokter yang akan digunakan dalam proses bisnis Klinik Gigi Dental Smile.
4. Menangani pendaftaran pasien baru dan cetak kartu pemeriksaan pasien.
5. Menangani pendaftaran antrean pasien pemeriksaan klinik.
6. Menangani pemeriksaan oleh dokter untuk pasien yang berupa menampilkan riwayat diagnosa dan pemeriksaan terakhir, menerima keluhan dari pasien, memberikan anamnesa oleh dokter, memberikan pilihan diagnosa beserta posisi giginya untuk dokter, memberikan pilihan

Program Studi Teknik Informatika	DPPL – SIGIDES	7/106
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		

tindakan medis beserta posisi giginya untuk dokter.

7. Menangani penghitungan biaya dan pencetakan nota atas tindakan medis yang telah dilakukan dokter kepada pasien dengan pengguna kasir.
8. Menampilkan laporan kepada manager klinik berupa laporan pasien atau pengunjung, laporan diagnose dan tindakan medis, dan laporan pemasukan.

Dan berjalan pada lingkungan dengan platform PC yang terhubung dengan internet.

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.
SIGIDES	Sistem Informasi Klinik Gigi Dental Smile, merupakan sistem informasi yang digunakan untuk proses bisnis Klinik Gigi Dental Smile.
GUI	Graphic User Interface, tampilan antarmuka sistem untuk pengguna
Database	Kumpulan data terkait yang diorganisasikan dalam struktur

	tertentu dan dapat diakses dengan cepat
Role	Hak akses yang dimiliki pengguna untuk menggunakan sistem
Web-Based System	Sebuah sistem yang berbasiskan jaringan, sehingga pengaksesannya bisa dilakukan melalui internet dengan terhubung ke database sistem.

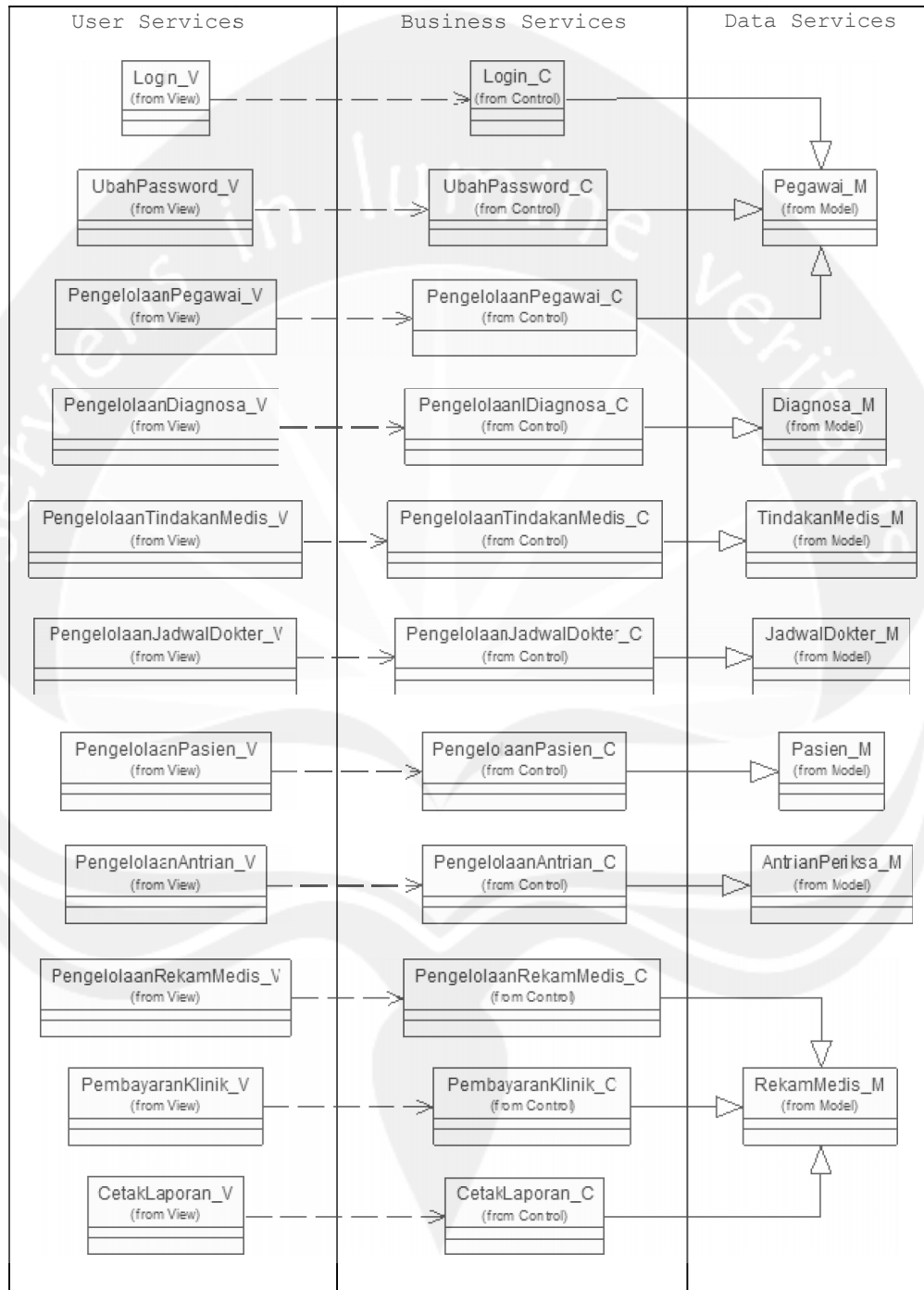
1.4 Referensi

Referensi yang digunakan pada perangkat lunak tersebut adalah:

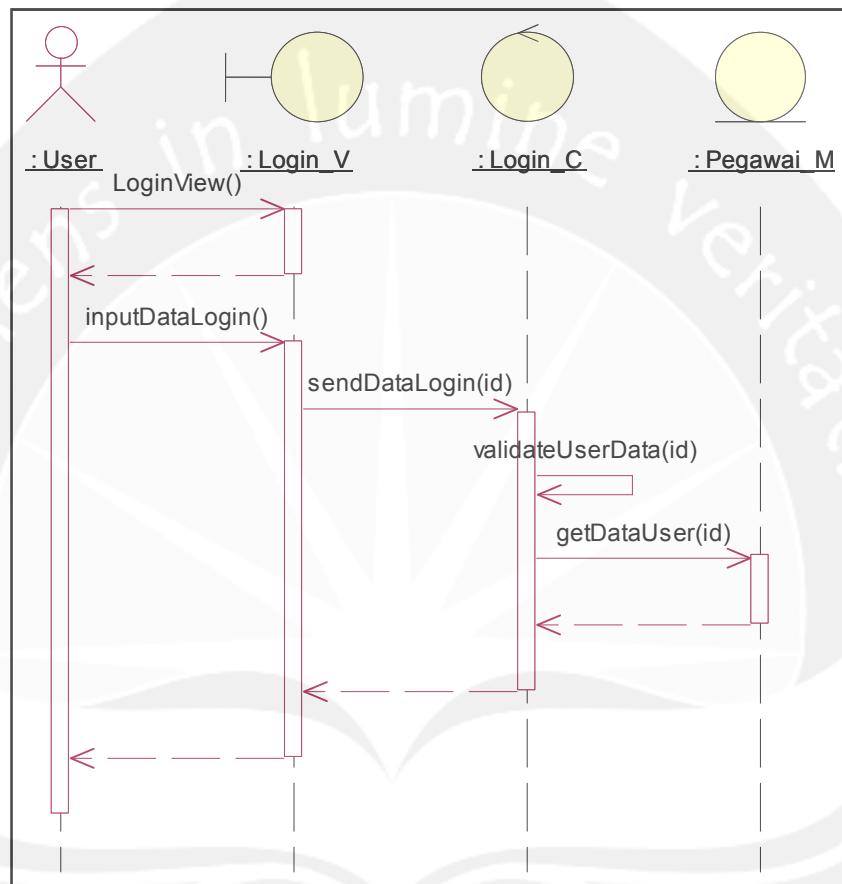
1. Bennet Simon, McRobb Steve, Farmer Ray, *Object-Oriented System Analysis and Design Using UML*, McGraw-Hill Companies, 2002.
2. Boggs Wendy, Boggs Michael, *Mastering UML with Rational Rose 2002*, SYBEX Inc, 2002.
3. *MSDN Library*-October 2005, Microsoft, 2005.
4. Wibisono, Aryo. SKPL SIAMA: Sistem Informasi Akademik Mahasiswa. 2010.
5. Brian Budiawan, Bernadus. *SKPL SPP (Sistem Penjualan Perabotan)*. Universitas Atma Jaya Yogyakarta. 2012
6. Novitasari, Vitalis Dian. SKPL SIPAS: Sistem Informasi Poliklinik Asy-Syifa. 2012.

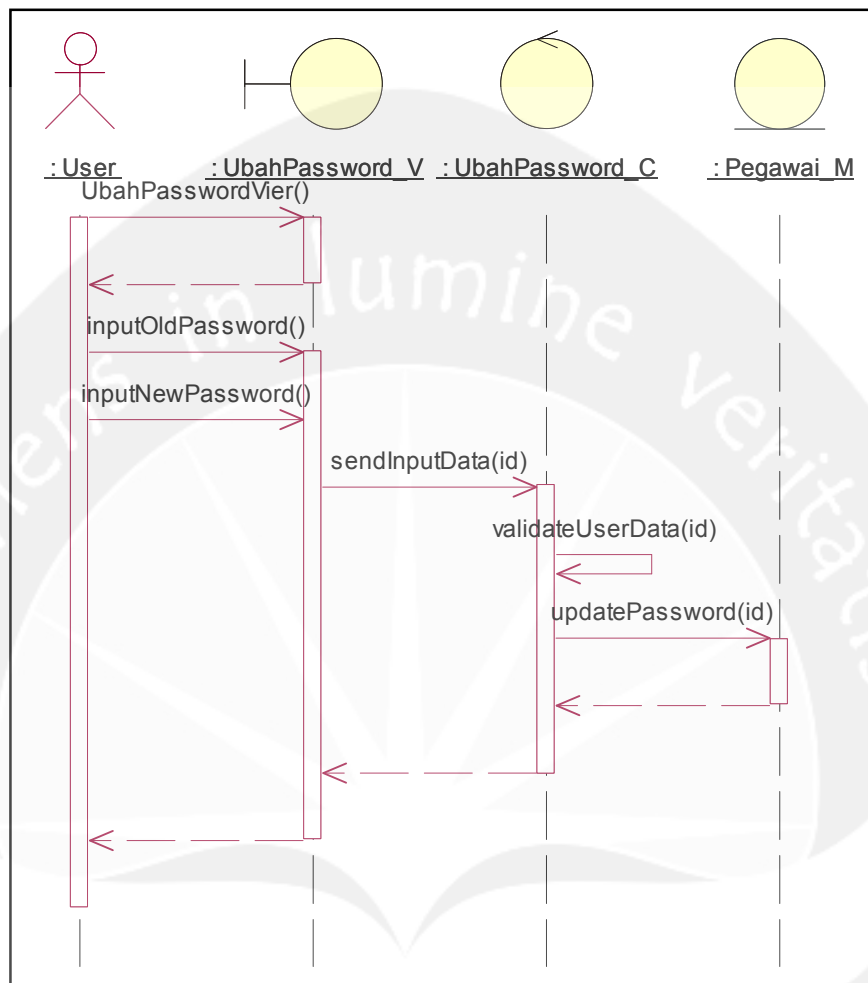
2 Perancangan Sistem

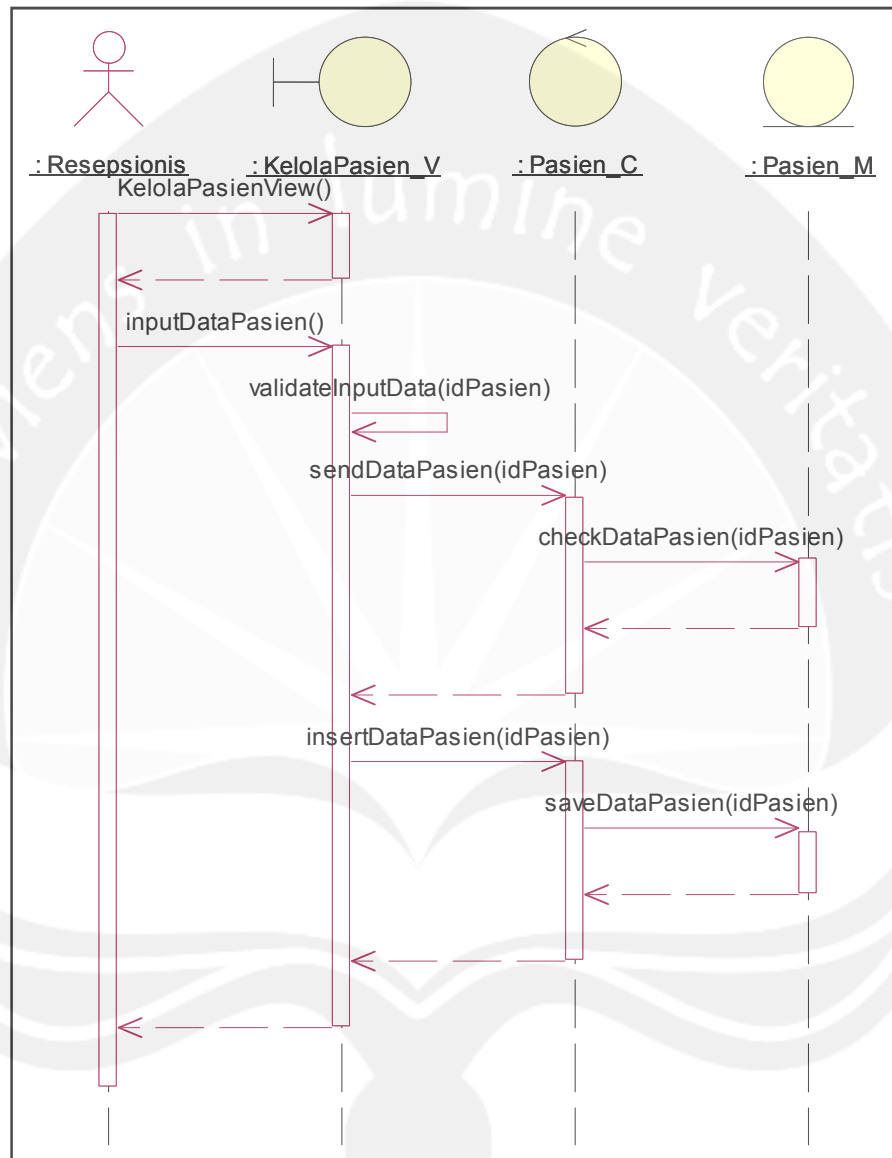
2.1 Diagram Arsitektur

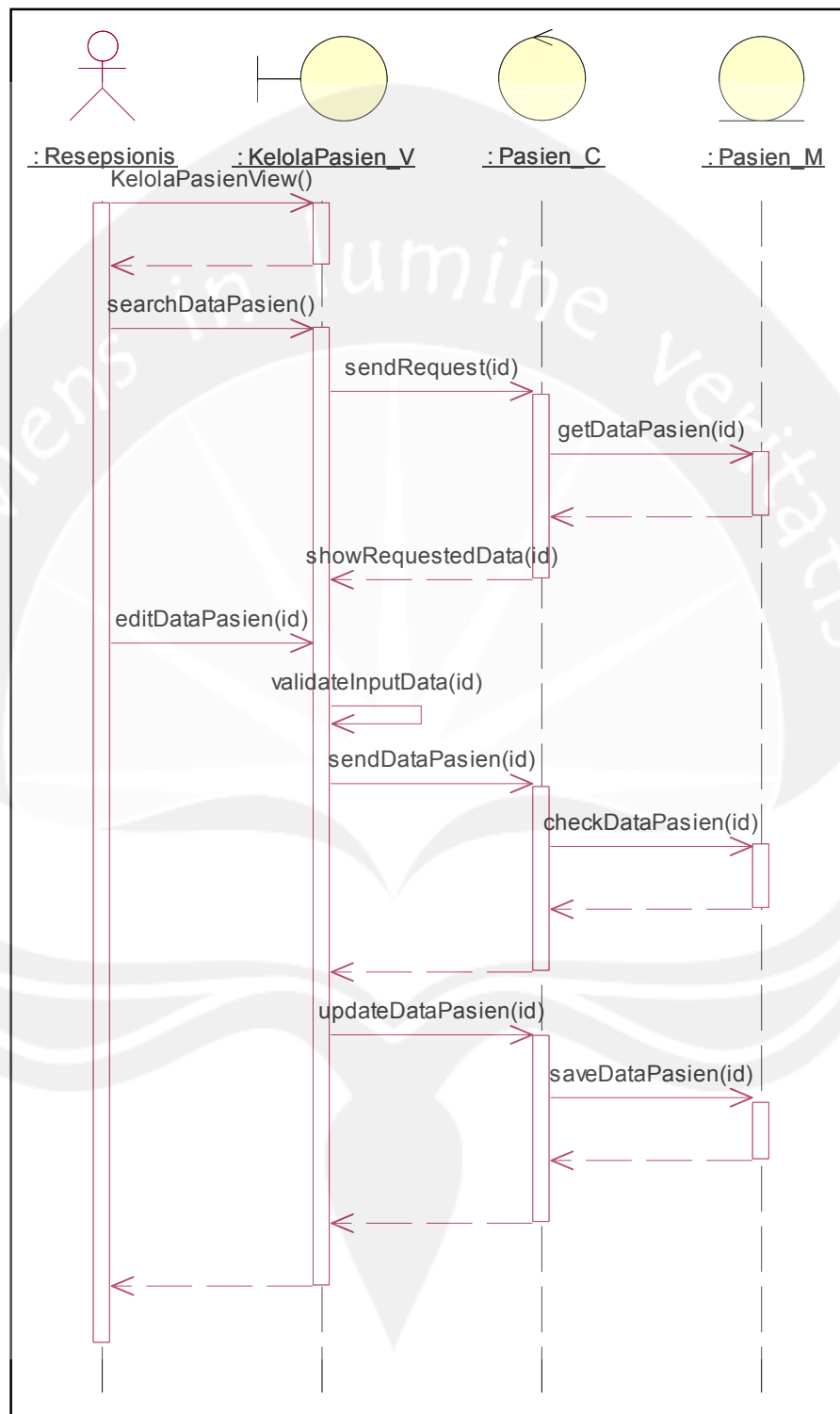


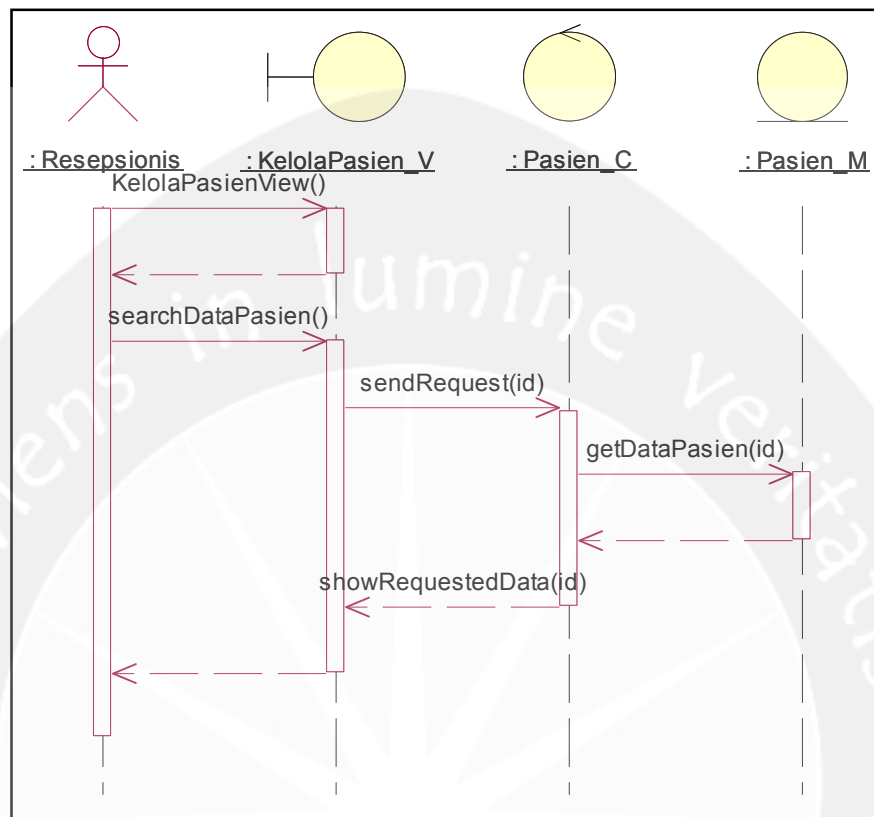
Gambar 2.1 Rancangan Arsitektur SIGIDES

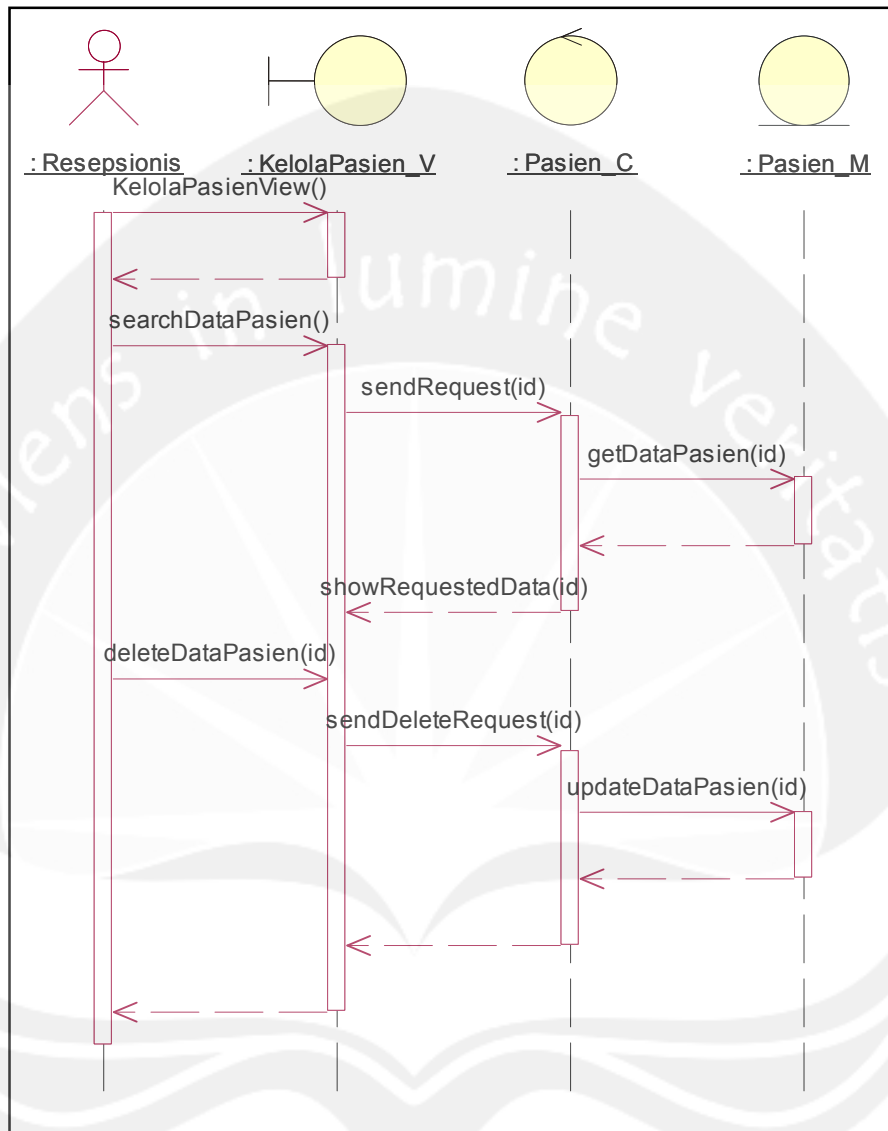


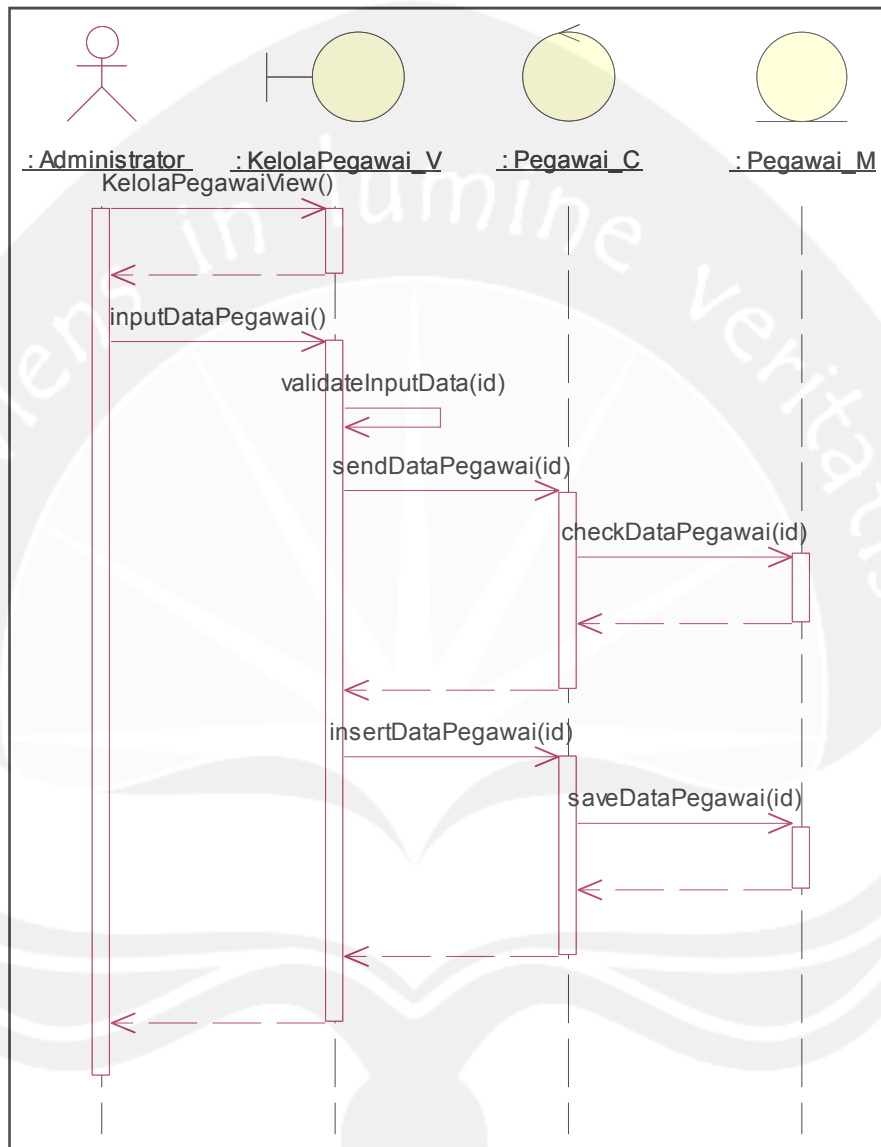


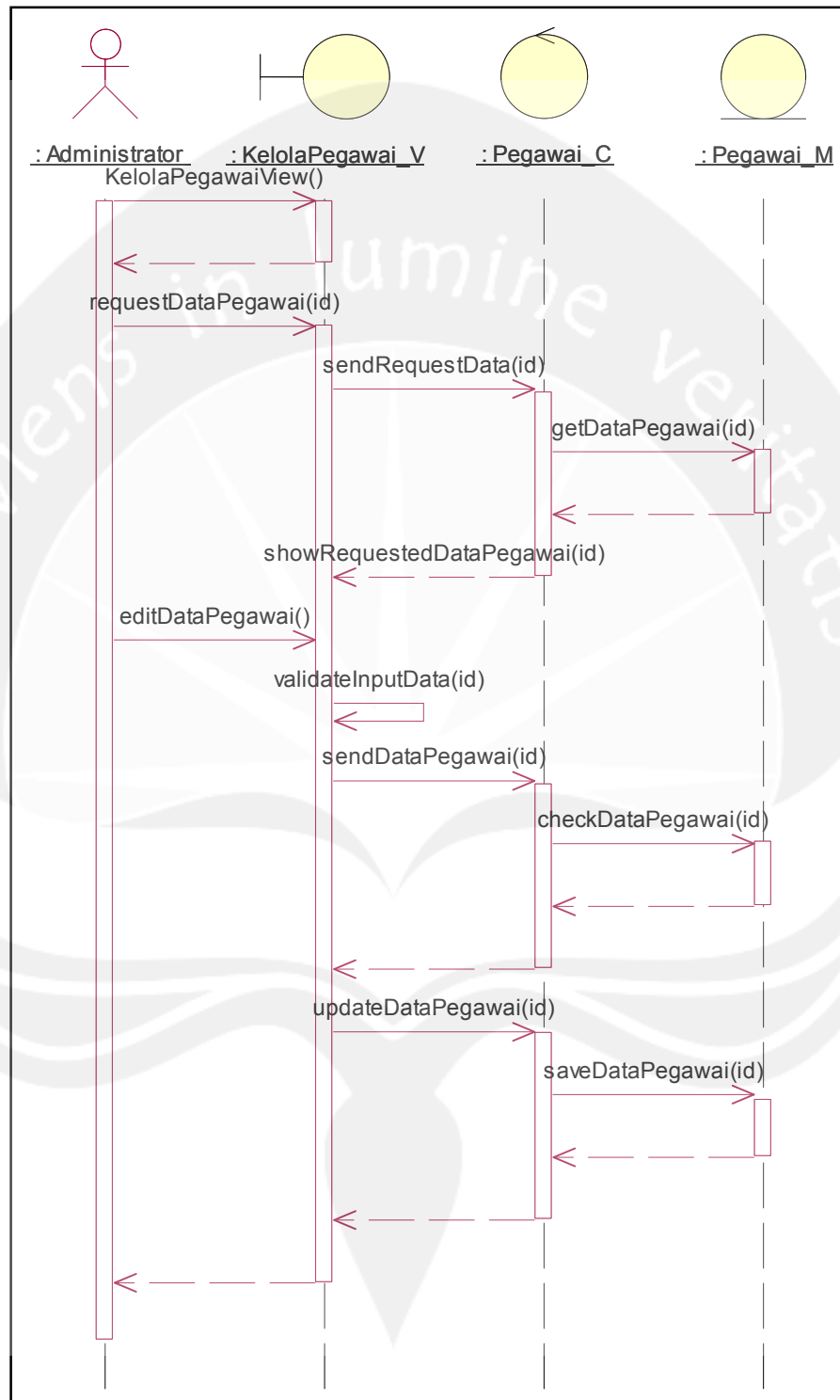


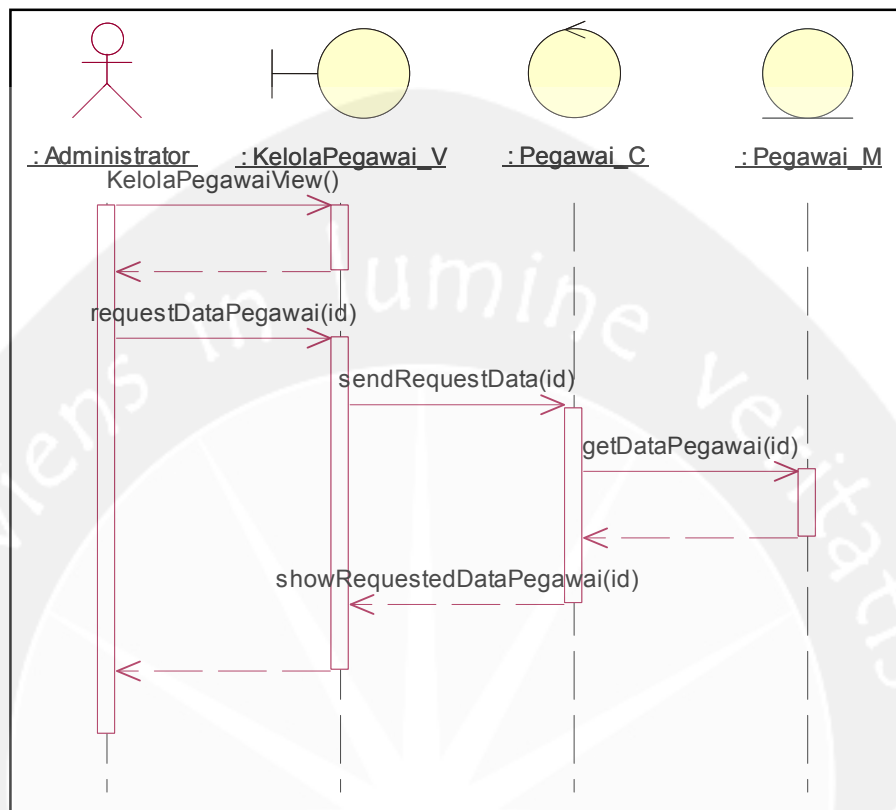




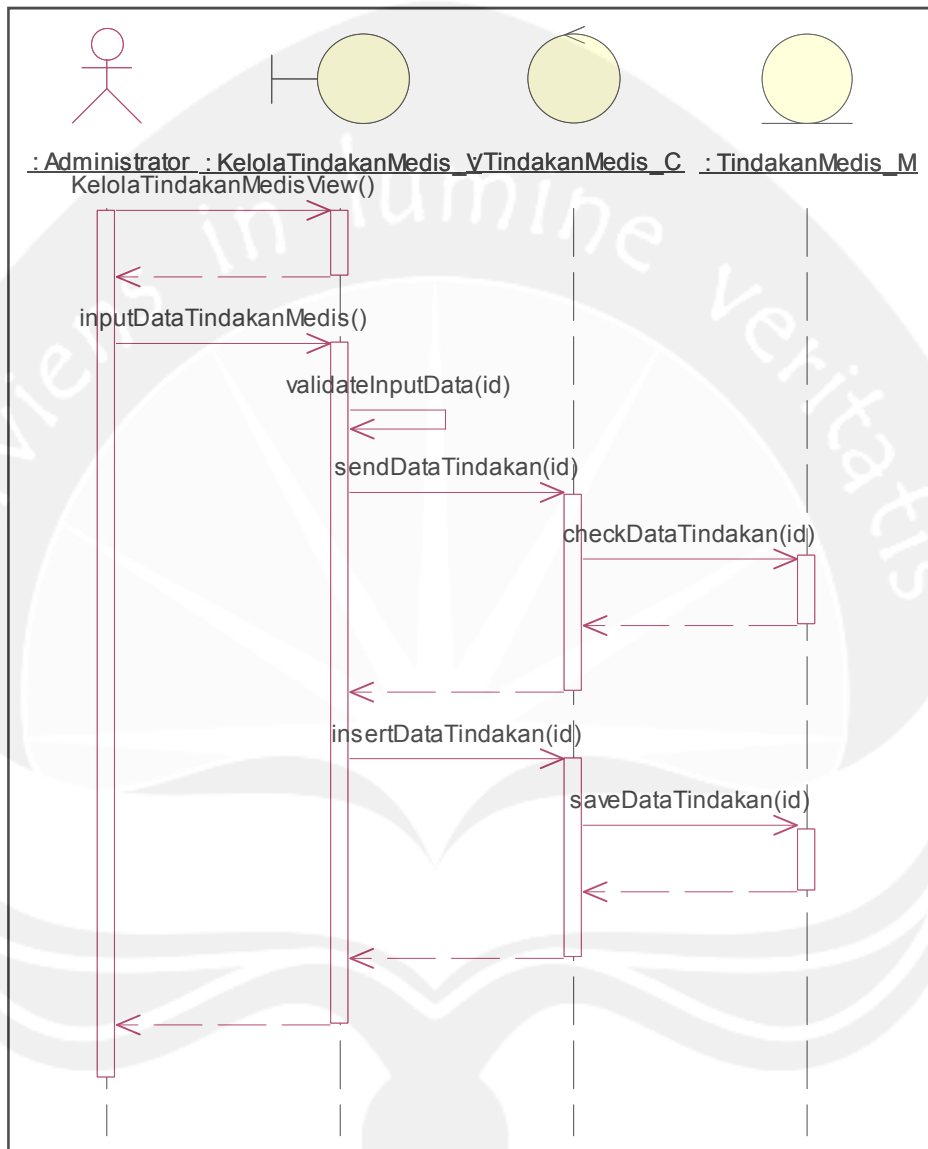


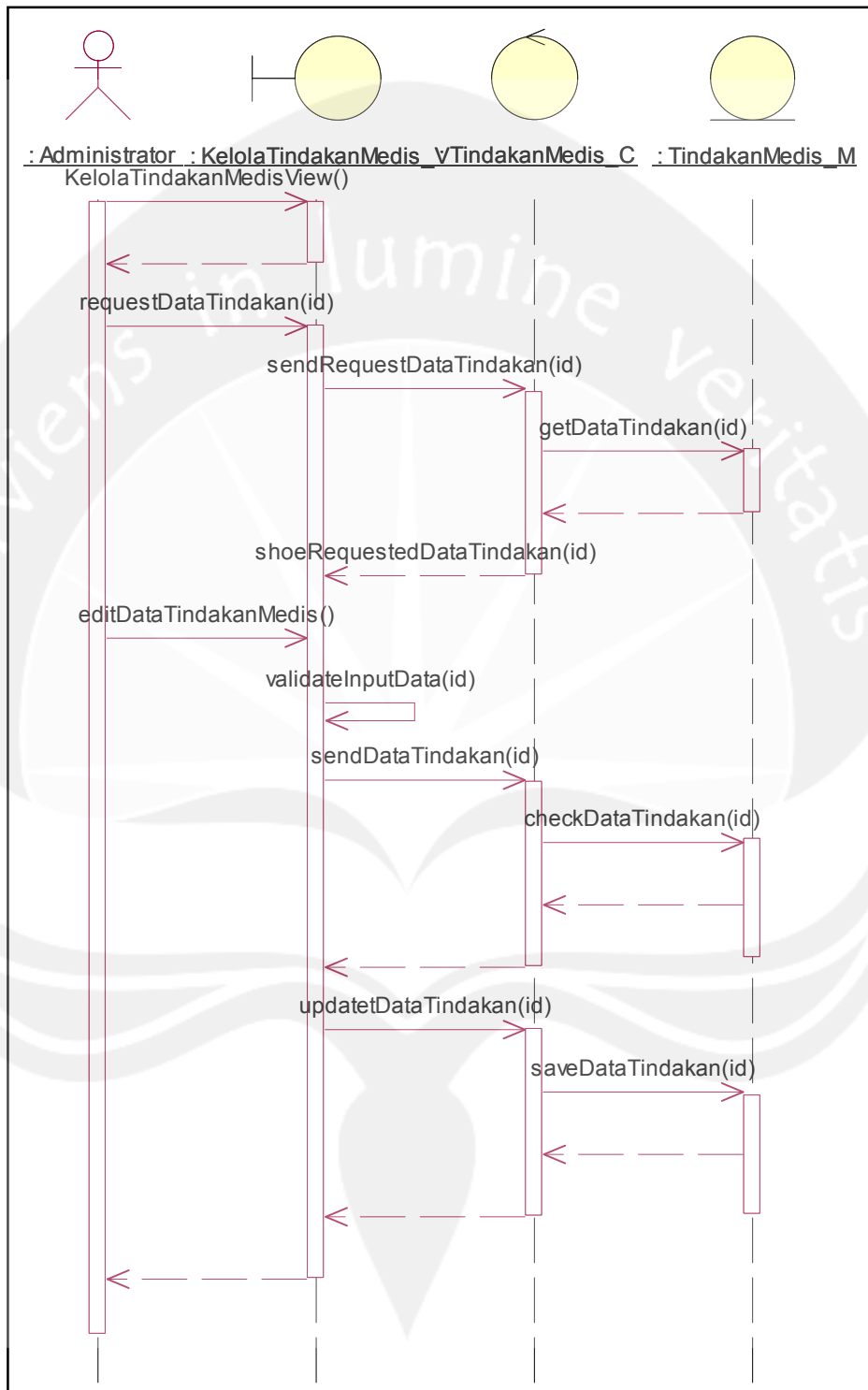


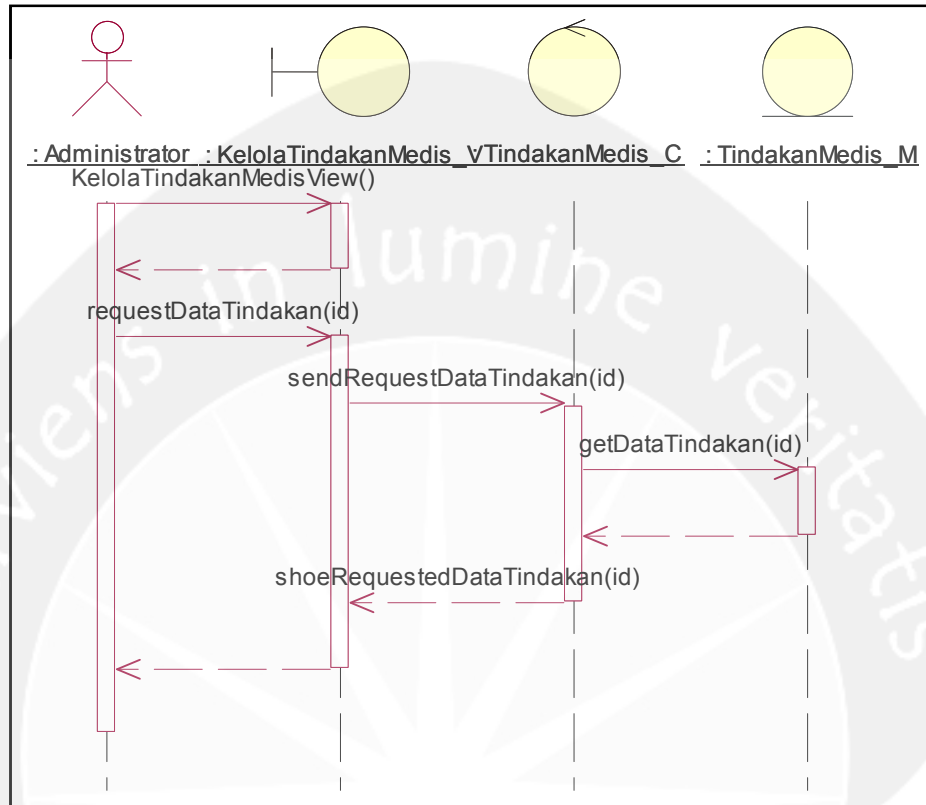


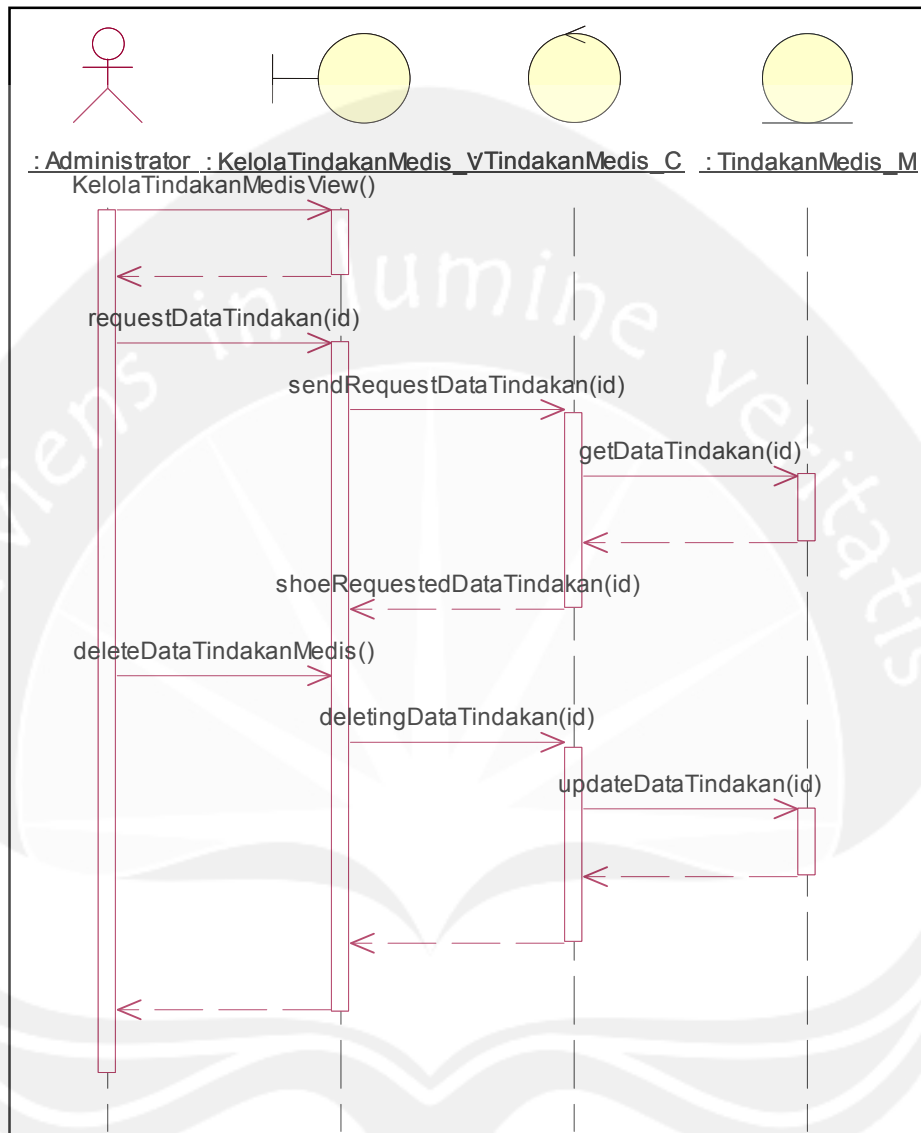






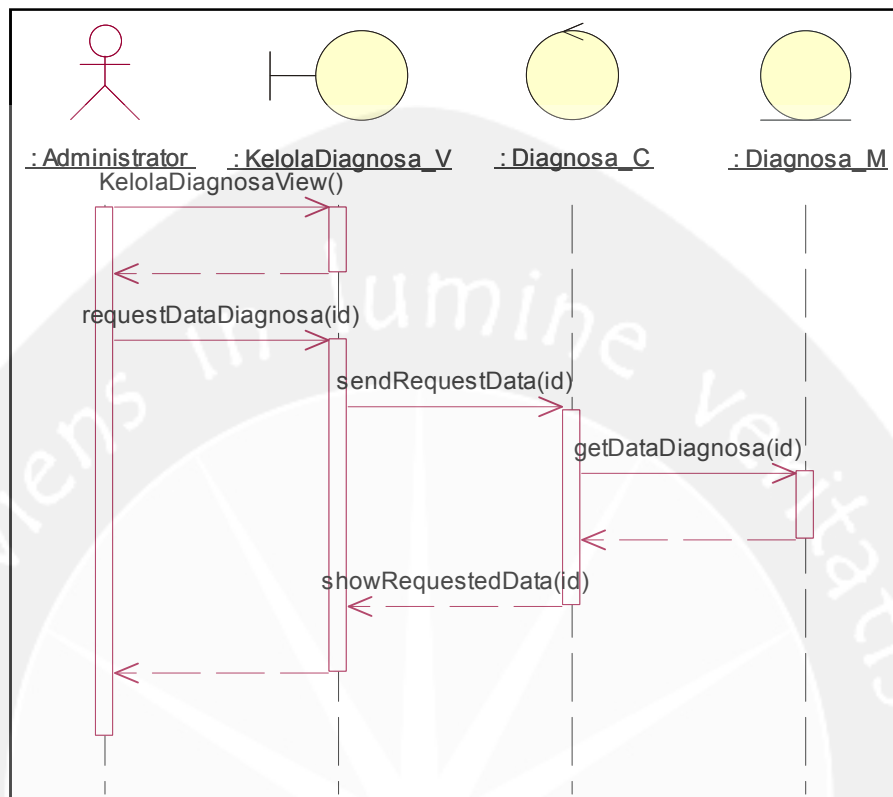


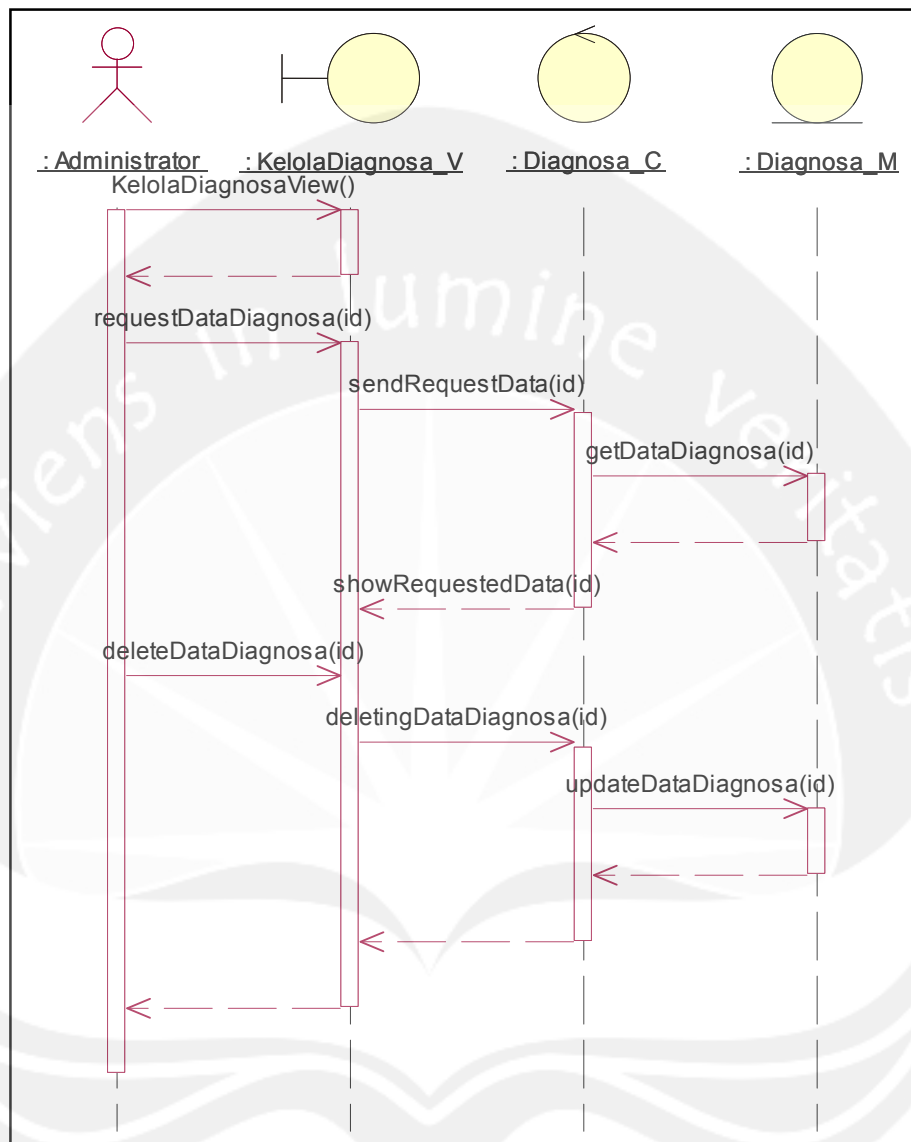


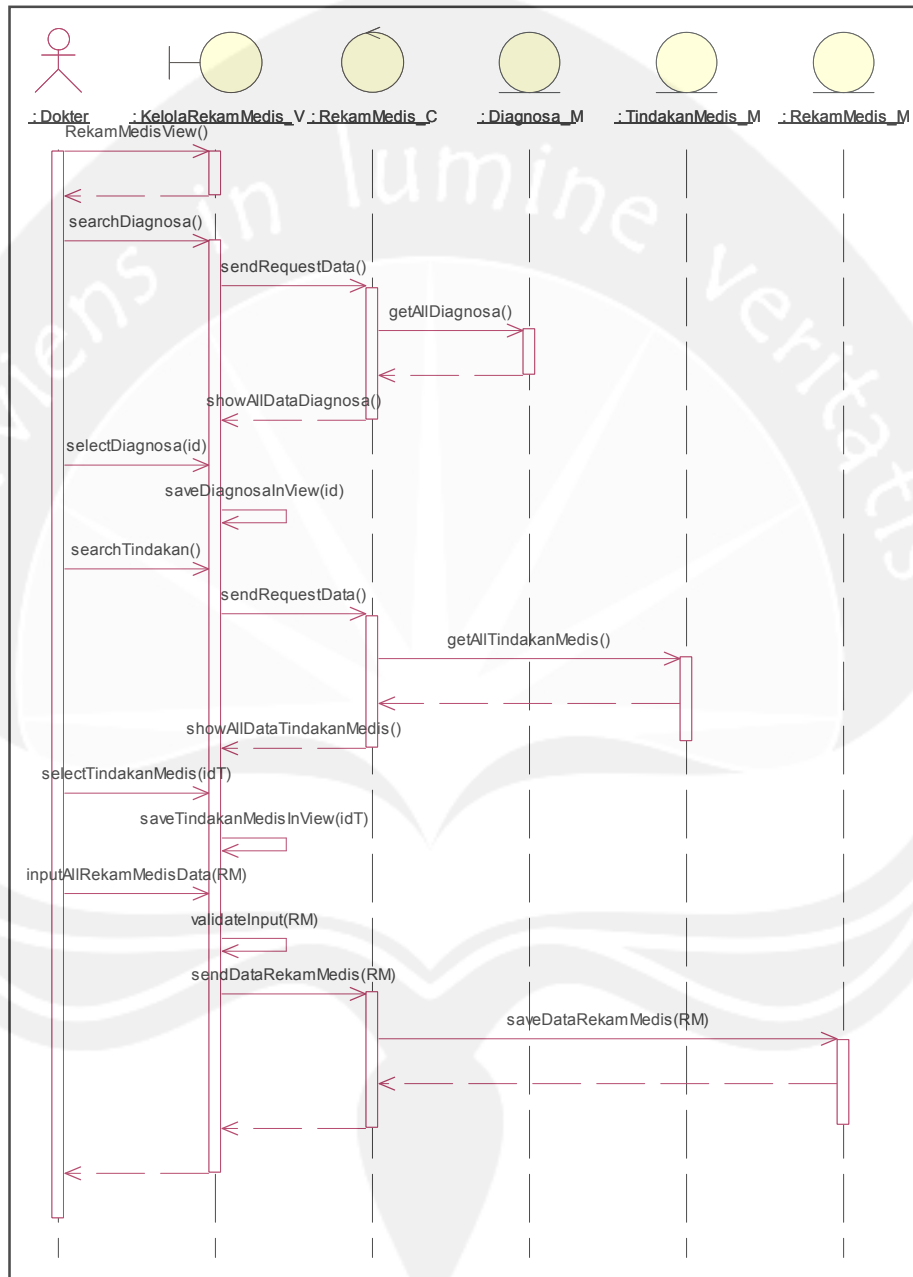


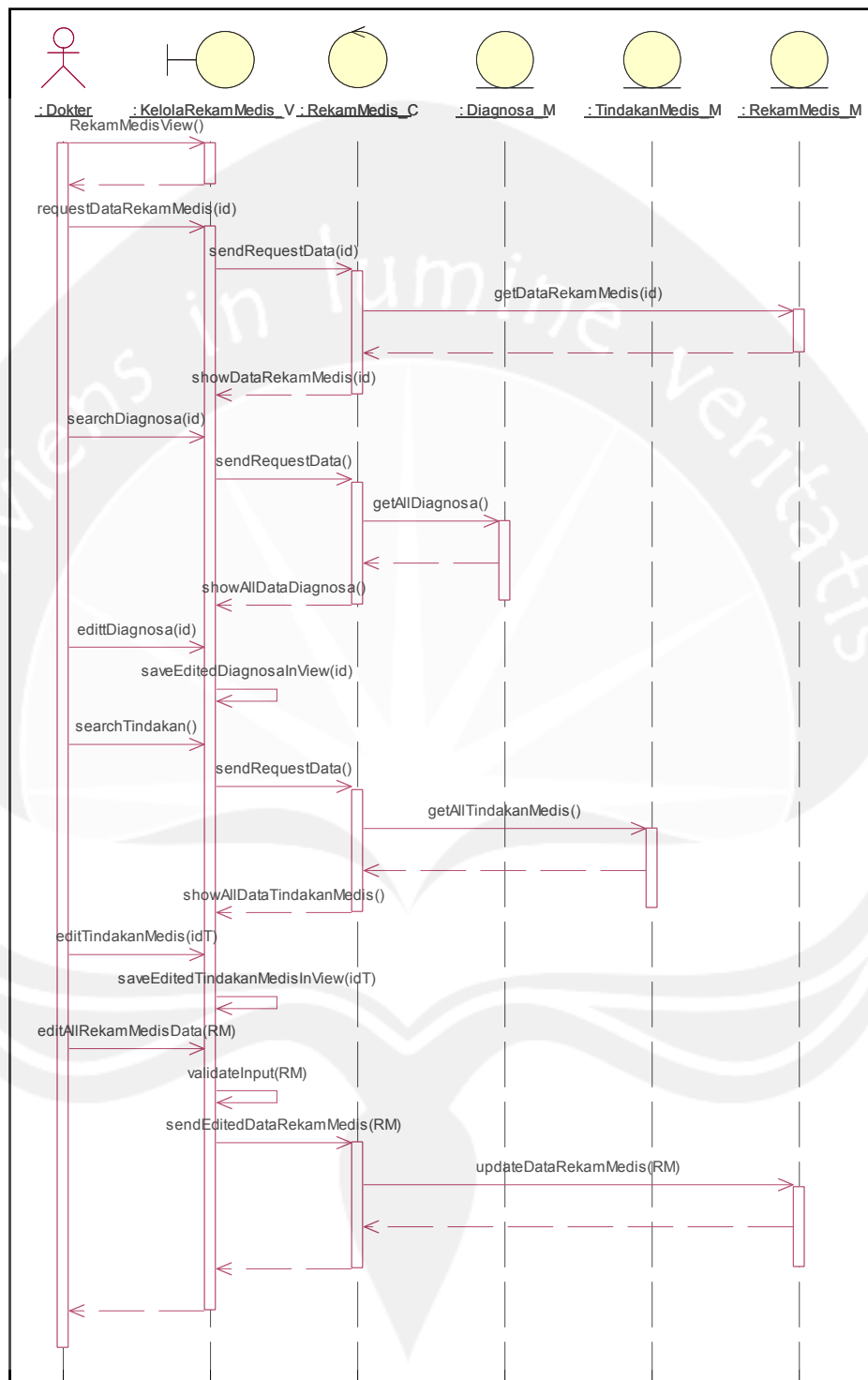


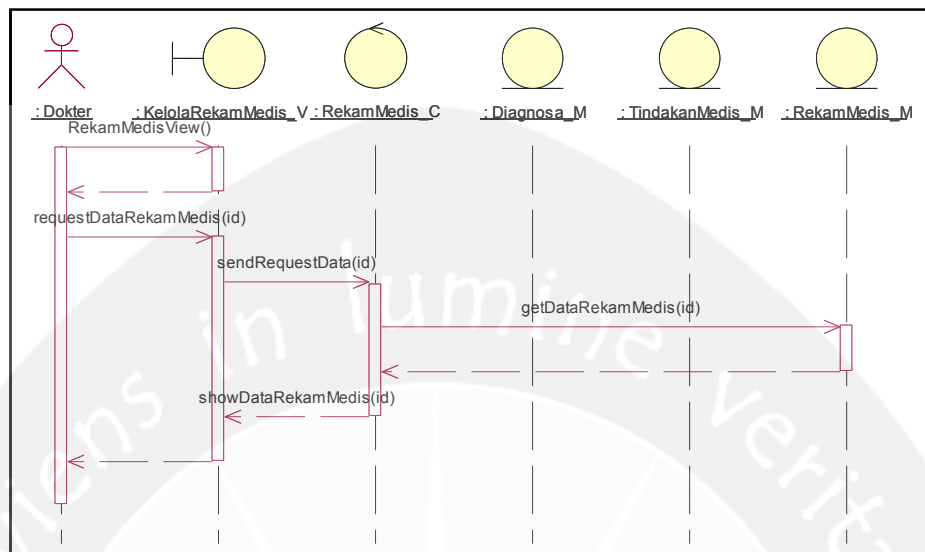


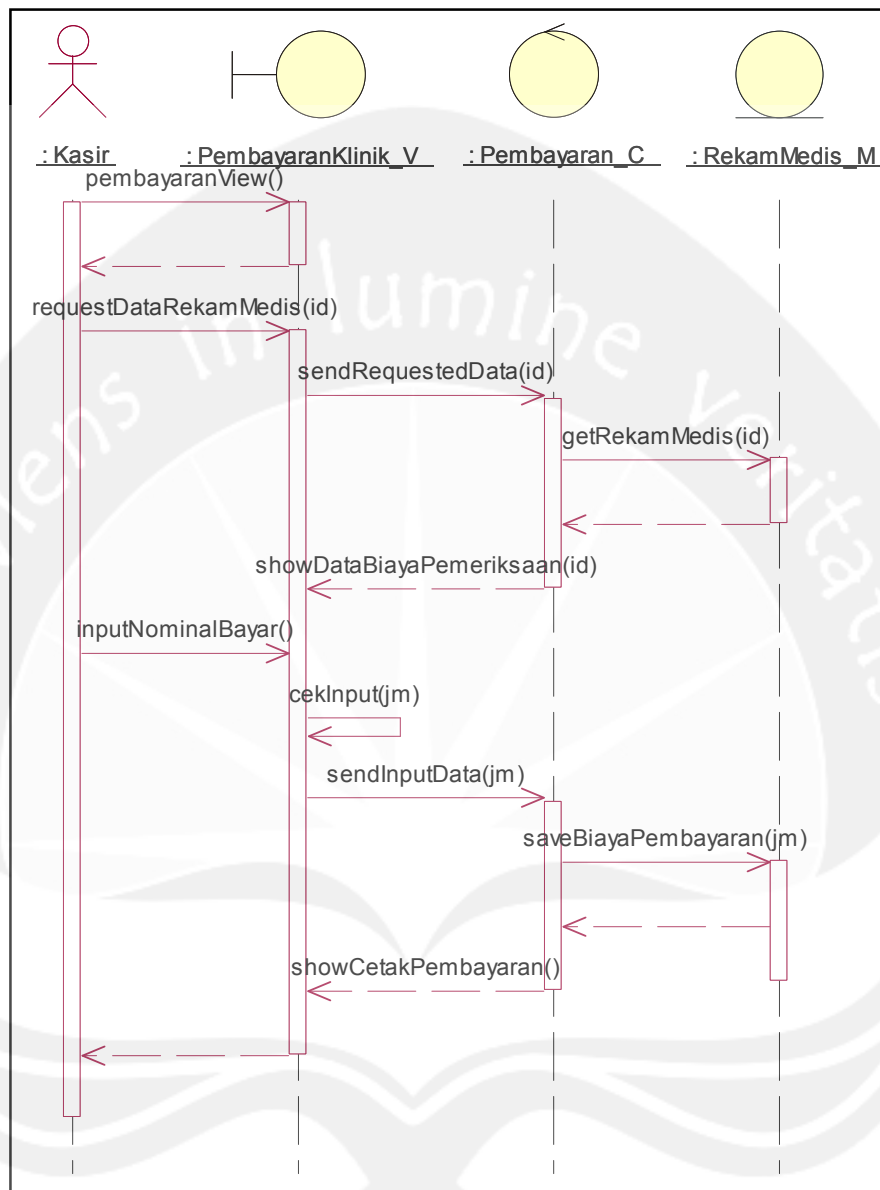


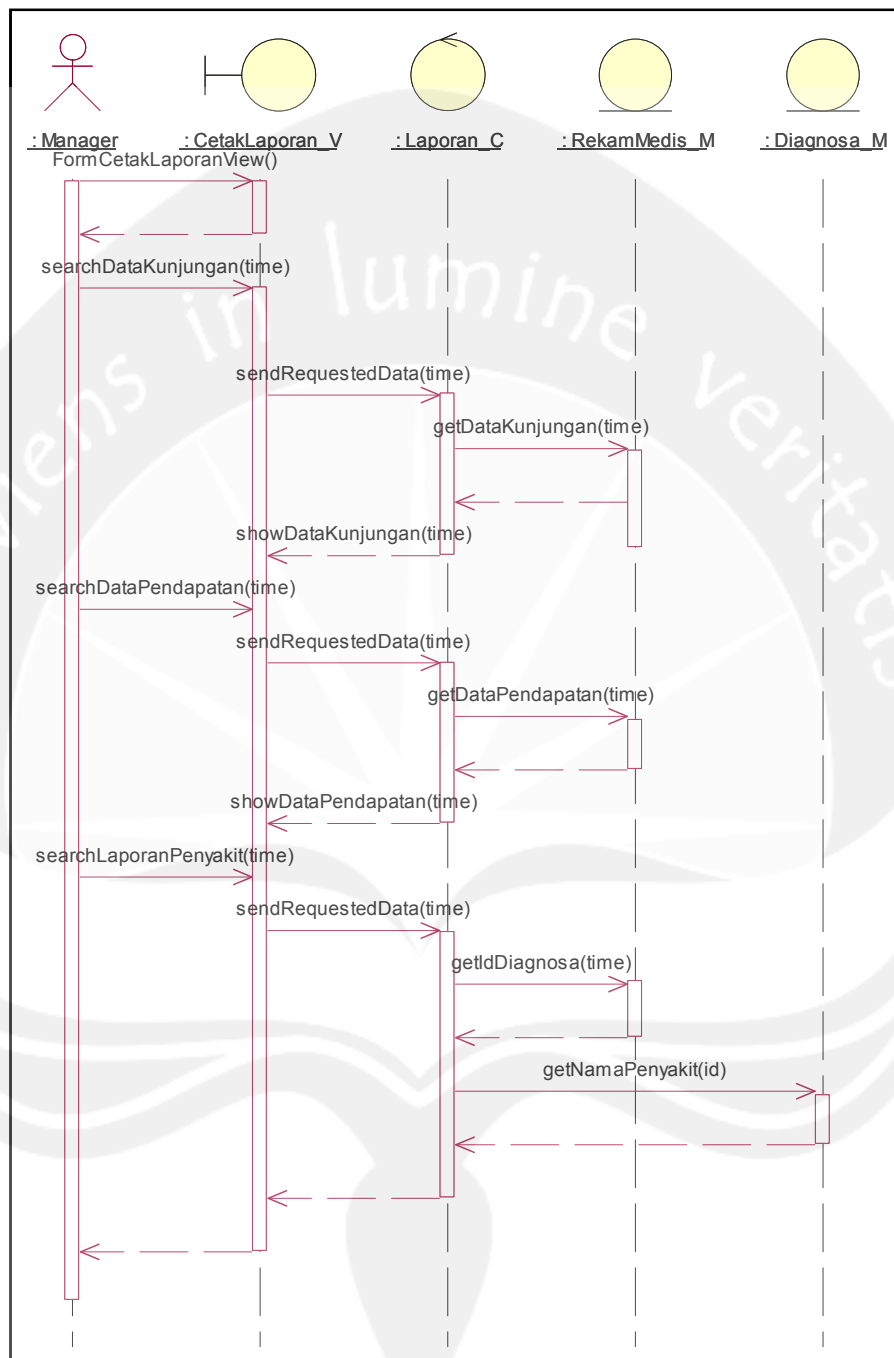


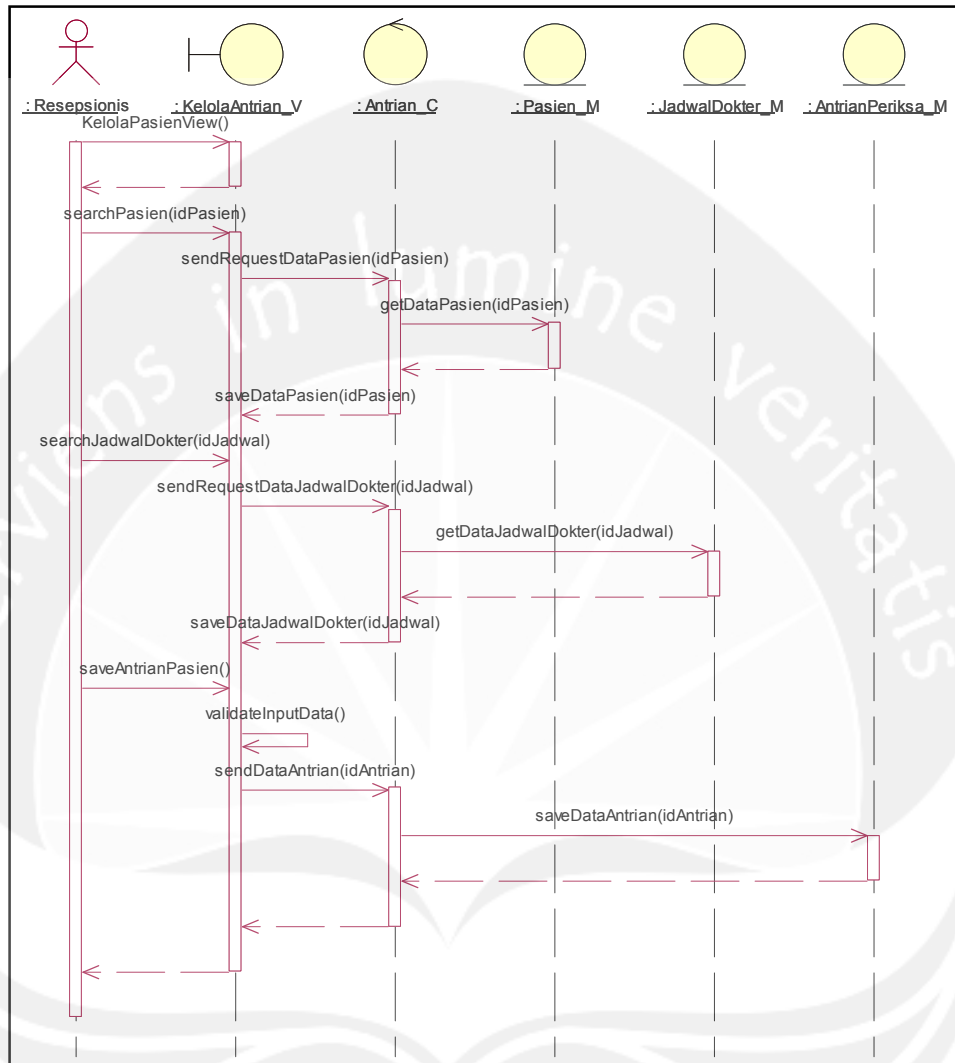


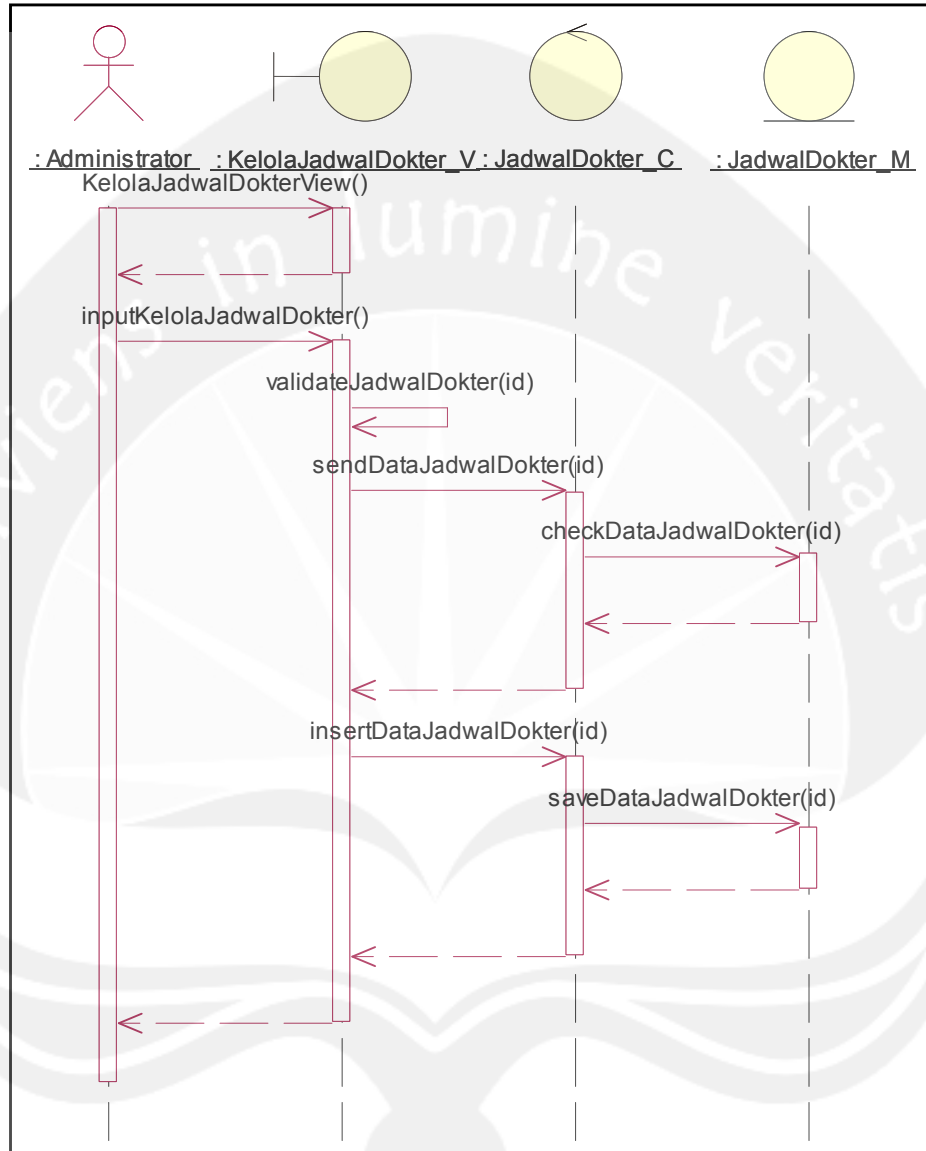


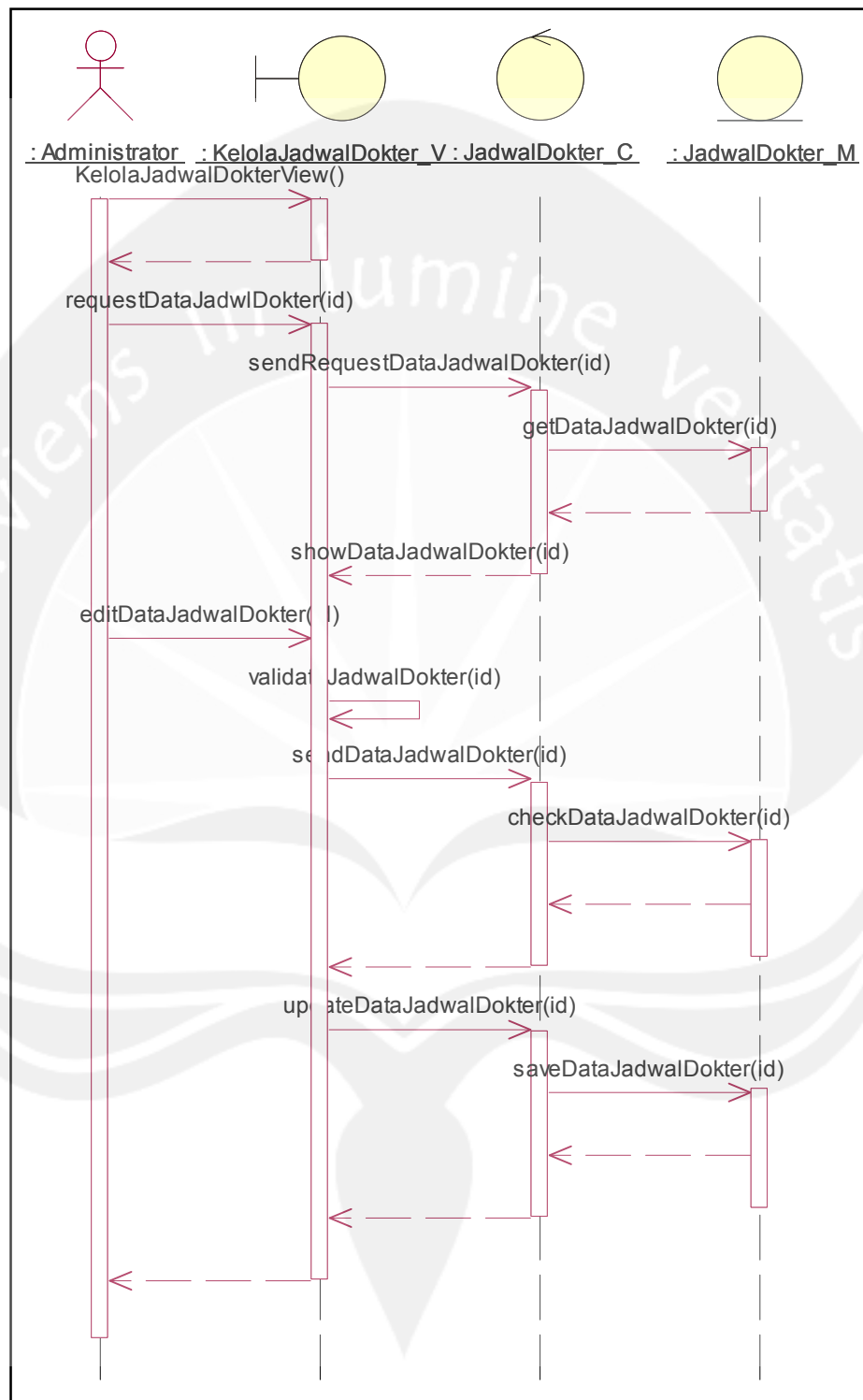


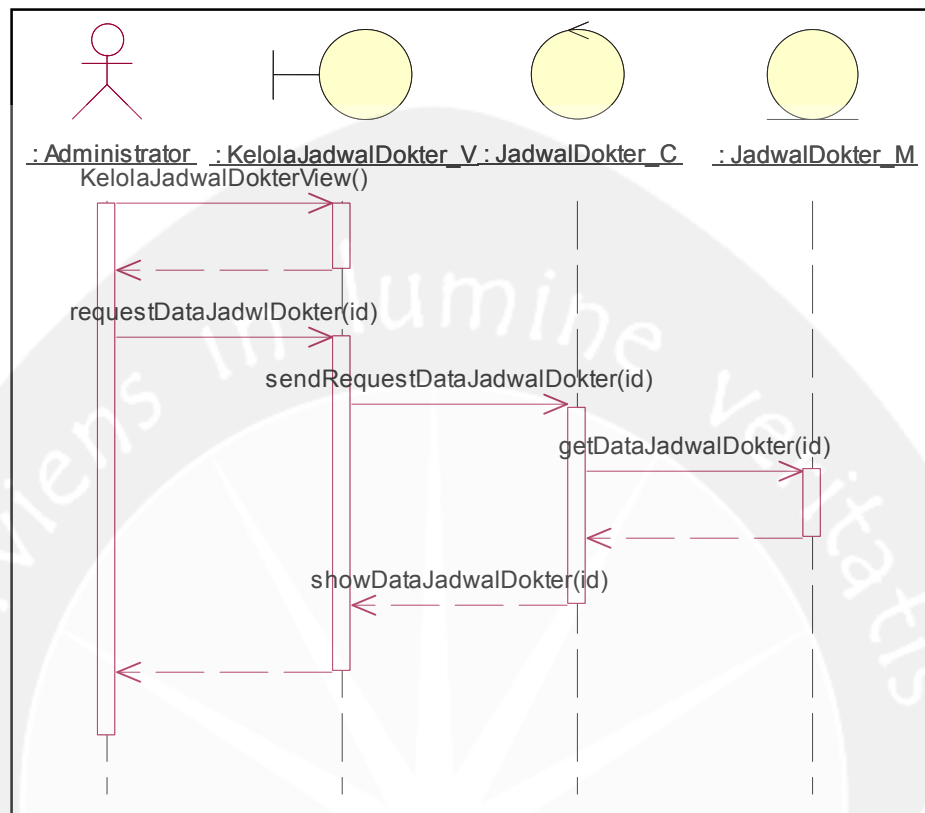






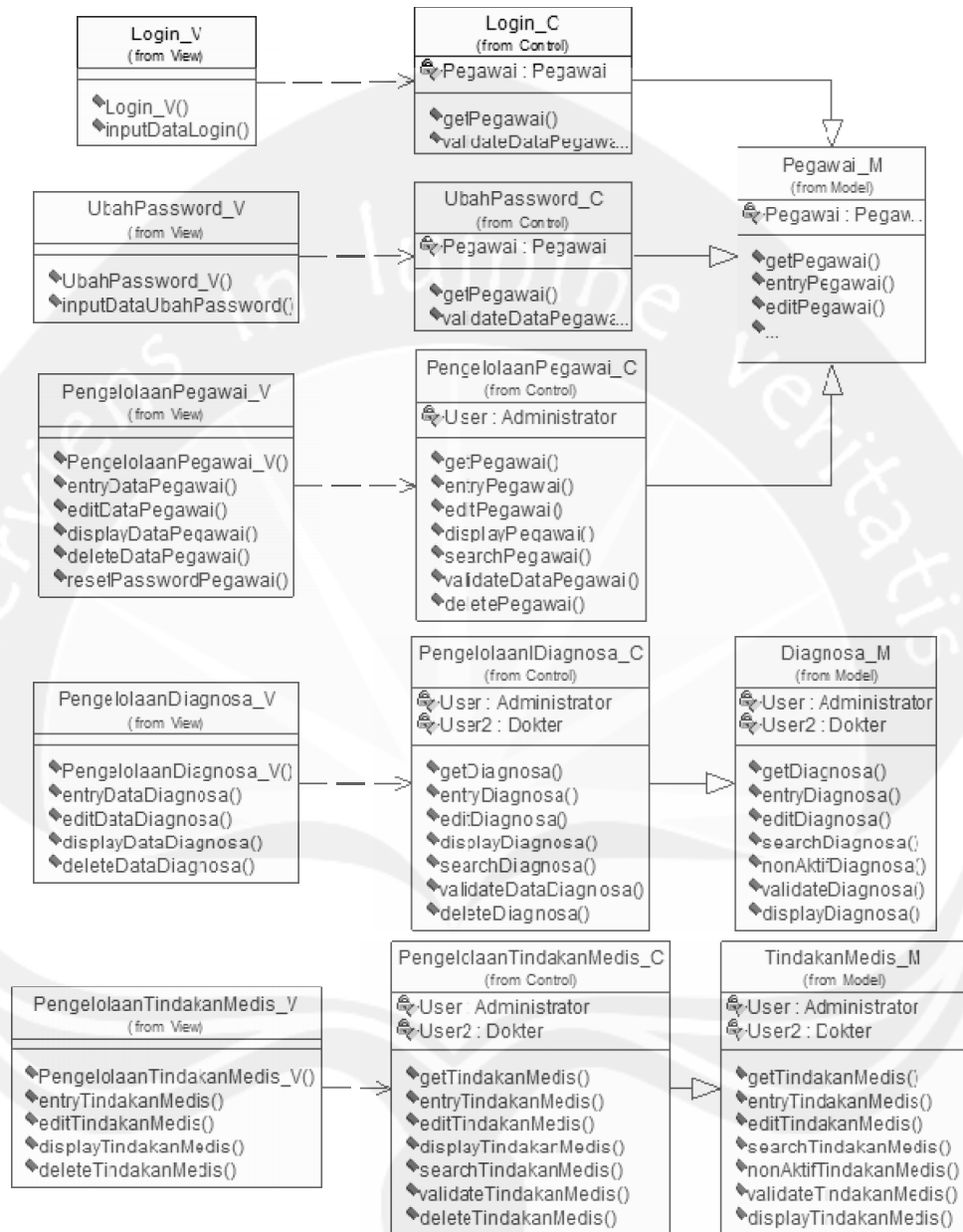


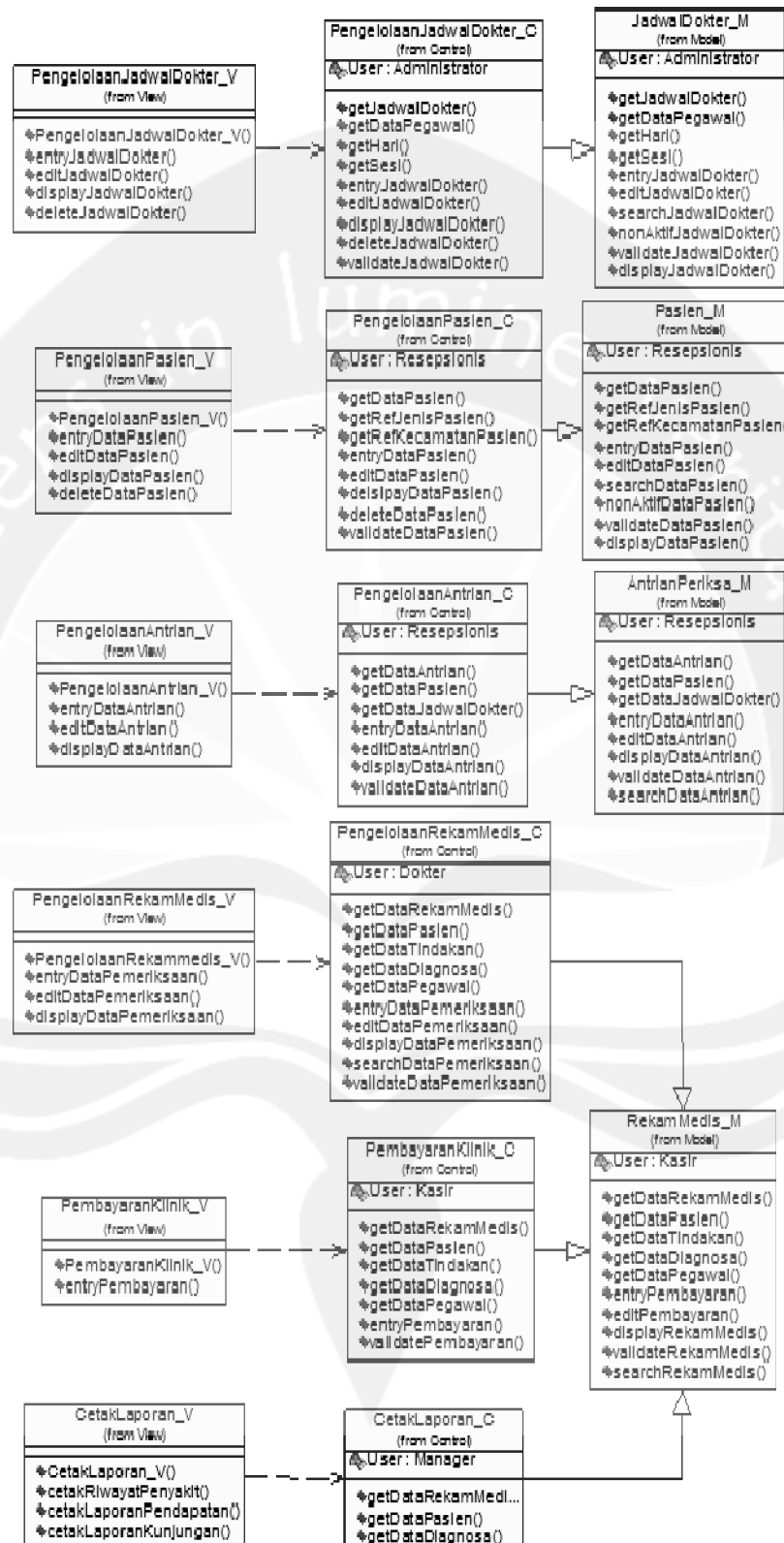






2.2.2 Class Diagram





Gambar 2.30 Class Diagram Perangkat Lunak SIGIDES

2.2.3 Spesifikasi Deskripsi Kelas Diagram

2.2.3.1 Spesifikasi Deskripsi Kelas Login_V

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

2.2.3.2 Spesifikasi Deskripsi Kelas UbahPassword_V

UbahPassword_V	<<View>>
<pre>+ UbahPassword_V() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +InputDataUbahPassword(Pegawai) Operasi ini digunakan untuk mengecek data yang dimasukkanpegawai ke basisdata dan digunakan untuk mengubah password lama dengan password baru ke basisdata.</pre>	

2.2.3.3 Spesifikasi Deskripsi Kelas

PengelolaanPegawai_V

PengelolaanPegawai_V	<<View>>
<pre>+PengelolaanPegawai_V() Default konstruktor, digunakan untuk inisialisasi</pre>	

```

semua atribut dari kelas ini.
+entryDataPegawai(Pegawai)
Operasi ini untuk membuat data Pegawai yang akan
dimasukkan ke basisdataPegawai.
+editDataPegawai(Pegawai)
Operasi ini untuk mengubahdataPegawaidari
basisdataPegawai.
+displayDataPegawai(Pegawai) : Pegawai
Operasi ini untuk menampilkandata Pegawaidari
basisdataPegawai.
+deleteDataPegawai(Pegawai): Pegawai
Operasi ini untuk menghapusdata Pegawaidari basisdata
Pegawai, namun dengan cara mengubah status aktif
menjadi tidak aktif.
+ resetPasswordPegawai(Pegawai) : Pegawai
Operasi ini untuk mereset password Pegawaidari
basisdata Pegawai.

```

2.2.3.4 Spesifikasi Deskripsi Kelas

PengelolaanDiagnosa_V

PengelolaanDiagnosa_V	<<View>>
<pre> +PengelolaanDiagnosa_V() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryDataDiagnosa(diagnosa) Operasi ini untuk membuat data Diagnosayang akan dimasukkan ke basisdata Diagnosa. +editDataDiagnosa(diagnosa) Operasi ini untuk mengubahdata Diagnosadari </pre>	

```

basisdataDiagnosa.
+displayDataDiagnosa(diagnosa) : Diagnosa
Operasi ini untuk menampilkan data Diagnosa dari
basisdataDiagnosa.
+deleteDataDiagnosa(diagnosa): Diagnosa
Operasi ini untuk menghapus data Diagnosa dari
database Diagnosa, namun dengan cara mengubah status
aktif menjadi tidak aktif.

```

2.2.3.5 Spesifikasi Deskripsi Kelas PengelolaanTindakanMedis_V

PengelolaanTindakanMedis_V	<< View>>
<pre> +PengelolaanTindakanMedis_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryTindakanMedis(tindakan) Operasi ini untuk membuat data Tindakan Medis yang akan dimasukkan ke basisdataTindakanMedis. +editTindakanMedis(tindakan) Operasi ini untuk mengubah data Tindakan Medis dari basisdataTindakanMedis. +displayTindakanMedis(tindakan) : Tindakan Operasi ini untuk menampilkan data Tindakan Medis dari basisdataTindakanMedis. +deleteTindakanMedis(tindakan): Tindakan Operasi ini untuk menghapus data Tindakan Medis dari basisdata TindakanMedis, namun dengan cara mengubah status aktif menjadi tidak aktif. </pre>	

2.2.3.6 Spesifikasi Deskripsi Kelas

PengelolaanJadwalDokter_V

PengelolaanJadwalDokter_V	<< View>>
<pre>+PengelolaanJadwalDokter_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryJadwalDokter (JadwalDokter) Operasi ini untuk membuat data Jadwal Dokter yang akan dimasukkan ke basisdataJadwalDokter. +editJadwalDokter(JadwalDokter) Operasi ini untuk mengubahdata Jadwal Dokter dari basisdataJadwalDokter. +displayJadwalDokter(JadwalDokter) : JadwalDokter Operasi ini untuk menampilkandata JadwalDokter dari basisdataJadwalDokter. +deleteJadwalDokter(JadwalDokter): JadwalDokter Operasi ini untuk menghapusdata Jadwal Dokter dari basisdataJadwalDokter, namun dengan cara mengubah status aktif menjadi tidak aktif.</pre>	

2.2.3.7 Spesifikasi Deskripsi Kelas

PengelolaanPasien_V

PengelolaanPasien_V	<< View>>
<pre>+PengelolaanPasien_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryDataPasien(Pasien)</pre>	

Operasi ini untuk membuat data Pasien yang akan dimasukkan ke basisdata Pasien.

+editDataPasien(Pasien)

Operasi ini untuk mengubah data Pasien dari basisdata Pasien.

+displayDataPasien(Pasien) : Pasien

Operasi ini untuk menampilkan data Pasien dari basisdata Pasien.

+deleteDataPasien(Pasien) : Pasien

Operasi ini untuk menghapus data Pasien dari basisdata Pasien.

2.2.3.8 Spesifikasi Deskripsi Kelas

PengelolaanAntrian_V

PengelolaanAntrian_V	<< View>>
+PengelolaanAntrian_V() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryDataAntrian(Antrian) Operasi ini untuk membuat data Antrian yang akan dimasukkan ke basisdata Antrian. +editDataAntrian(Antrian) Operasi ini untuk mengubah data Antri dari basisdata Antrian. +displayDataAntrian(Antrian) : Antrian Operasi ini untuk menampilkan data Antri dari basisdata Antrian.	

2.2.3.9 Spesifikasi Deskripsi Kelas

PengelolaanRekamMedis_V

PengelolaanRekamMedis_V	<< View>>
<pre>+PengelolaanRekamMedis_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryDataPemeriksaan(RekamMedis) Operasi ini untuk membuat data Pemeriksaayang akan dimasukkan ke basisdata RekamMedis. +editDataPemeriksaan(RekamMedis) Operasi ini untuk mengubahdata Pemeriksaan dari basisdata RekamMedis. +displayDataPemeriksaan(RekamMedis) : RekamMedis Operasi ini untuk menampilkandata Pemeriksaandari basisdata RekamMedis.</pre>	

2.2.3.10 Spesifikasi Deskripsi Kelas

PembayaranKlinik_V

PembayaranKlinik_V	<< View>>
<pre>+PembayaranKlinik_V () Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +entryPembayaran(Pembayaran) Operasi ini untuk membuat data Pembayaranyang akan dimasukkan ke databaseRekamMedis.</pre>	

2.2.3.11 Spesifikasi Deskripsi Kelas CetakLaporan_V

CetakLaporan_V	<< View>>
----------------	-----------

Program Studi Teknik Informatika	DPPL – SIGIDES	46/106
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		

+CetakLaporan_V () Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +cetakRiwayatPenyakit(Diagnosa) Operasi ini untuk mencetak daftar penyakit apa yang paling sering dialami pasien selama periode tertentu melalui basisdata RekamMedis dan Diagnosa. +cetakLaporanPendapatan(RekamMedis) Operasi ini untuk mencetak laporan pendapatan klinik selama periode tertentu yang tercatat dalam basisdata RekamMedis. +cetakLaporanKunjungan(RekamMedis) Operasi ini untuk mencetak total pengunjung atau pasien selama periode tertentu yang tercatat dalam basisdata RekamMedis.

2.2.3.12 Spesifikasi Deskripsi Kelas Login_C

Login_C	<<Control>>
+Login_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getPegawai() : Pegawai Operasi ini untuk mengambil data Pegawai dari basisdata Pegawai. +validateDataPegawai() : Pegawai Operasi ini untuk mengecek kebenaran id Pegawai dan password yang dimasukkan oleh Pegawai berdasarkan basisdata Pegawai.	

2.2.3.13 Spesifikasi Deskripsi Kelas UbahPassword_C

UbahPassword_C	<<Control>>
<pre>+ UbahPassword_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getPegawai() Operasi ini untuk mengubah password lama dengan yang baru ke dalam basisdata Pegawai. +ValidateDataPegawai() : Pegawai Operasi ini untuk mengecek kebenaran id pegawai dan password yang dimasukkan oleh Pegawai berdasarkan basisdata Pegawai.</pre>	

2.2.3.14 Spesifikasi Deskripsi Kelas

PengelolaanPegawai_C

PengelolaanPegawai_C	<<Control>>
<pre>+PengelolaanPegawai_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getPegawai() : Pegawai Operasi ini untuk mengambil data Pegawai dari basisdata Pegawai. +entryPegawai(Pegawai) Operasi ini untuk membuat data Pegawai yang akan dimasukkan ke basisdata Pegawai. +editPegawai(Pegawai) Operasi ini untuk mengubah data Pegawai dari basisdata Pegawai.</pre>	

```

+displayPegawai(Pegawai)
Operasi ini untuk menampilkan data Pegawaidari
basisdata Pegawai.
+searchPegawai() : Pegawai
Operasi ini untuk mencari data Pegawaidari
basisdataPegawai dengan nilai tertentu.
+validateDataPegawai() : Pegawai
Operasi ini untuk mengecek kebenaran nilaiPegguna
yang akan dimasukkan apakah sesuai dengan format pada
basisdata Pegawai.
+deletePegawai(Pegawai): Pegawai
Operasi ini untuk menghapusdata Pegawaidari
basisdataPegawai.

```

2.2.3.15 Spesifikasi Deskripsi Kelas PengelolaanDiagnosa_C

PengelolaanDiagnosa_C	<<Control>>
<pre> +PengelolaanDiagnosa_C() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDiagnosa() : Diagnosa Operasi ini untuk mengambil data Diagnosa dari basisdataDiagnosa. +entryDiagnosa(Diagnosa) Operasi ini untuk membuat data Diagnosayang akan dimasukkan ke basisdataDiagnosa. +editDiagnosa(Diagnosa) Operasi ini untuk mengubahdata Diagnosadari basisdata Diagnosa. </pre>	

+displayDiagnosa(Diagnosa)

Operasi ini untuk menampilkan data Diagnosa dari basisdata Diagnosa.

+searchDiagnosa() : Diagnosa

Operasi ini untuk mencari data Diagnosa dari basisdata Diagnosa dengan nilai tertentu.

+validateDiagnosa() : Diagnosa

Operasi ini untuk mengecek kebenaran nilai Diagnosa yang akan dimasukkan apakah sesuai dengan format pada basisdata Diagnosa.

+deleteDiagnosa(Diagnosa) : Diagnosa

Operasi ini untuk menghapus data Diagnosa dari basisdata Diagnosa.

2.2.3.16 Spesifikasi Deskripsi Kelas

PengelolaanTindakanMedis_C

PengelolaanTindakanMedis_C	<<Control>>
----------------------------	-------------

+PengelolaanTindakanMedis_C()

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

+getTindakanMedis() : TindakanMedis

Operasi ini untuk mengambil data TindakanMedis dari basisdataTindakanMedis.

+entryTindakanMedis(TindakanMedis)

Operasi ini untuk membuat data TindakanMedis yang akan dimasukkan ke basisdataTindakanMedis.

+editTindakanMedis(TindakanMedis)

Operasi ini untuk mengubah data TindakanMedis dari basisdataTindakanMedis.

+displayTindakanMedis(TindakanMedis)

Operasi ini untuk menampilkan data TindakanMedis dari basisdata TindakanMedis.

+searchTindakanMedis() : TindakanMedis

Operasi ini untuk mencari data TindakanMedis dari basisdataTindakanMedis dengan nilai tertentu.

+validateTindakanMedis() : TindakanMedis

Operasi ini untuk mengecek kebenaran nilai TindakanMedis yang akan dimasukkan apakah sesuai dengan format pada basisdata TindakanMedis.

+deleteTindakanMedis(TindakanMedis): TindakanMedis

Operasi ini untuk menghapusdata TindakanMedis dari basisdataTindakanMedis.

2.2.3.17 Spesifikasi Deskripsi Kelas

PengelolaanJadwalDokter_C

PengelolaanJadwalDokter_C	<<Control>>
---------------------------	-------------

+PengelolaanJadwalDokter_C()

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

+getJadwalDokter () : JadwalDokter

Operasi ini untuk mengambil data JadwalDokter dari basisdataJadwalDokter.

+getDataPegawai() : Dokter

Operasi ini untuk mengambil data Pegawai yang berjenis Dokter dari basisdataPegawai.

+getHari() : Hari

Operasi ini untuk mengambil data Hari dari basisdataReferensi Hari.

+getSesi() : Sesi
 Operasi ini untuk mengambil data Sesi pemeriksa dari basisdataReferensi Sesi.

+entryJadwalDokter(JadwalDokter)
 Operasi ini untuk membuat data JadwalDokter yang akan dimasukkan ke basisdataJadwalDokter.

+editJadwalDokter(JadwalDokter)
 Operasi ini untuk mengubah data JadwalDokter dari basisdataJadwalDokter.

+displayJadwalDokter(JadwalDokter)
 Operasi ini untuk menampilkan data JadwalDokter dari basisdata JadwalDokter.

+deleteJadwalDokter(JadwalDokter) : JadwalDokter
 Operasi ini untuk menghapus data JadwalDokter dari basisdataJadwalDokter.

+validateJadwalDokter() : JadwalDokter
 Operasi ini untuk mengecek kebenaran nilai JadwalDokter yang akan dimasukkan apakah sesuai dengan format pada basisdata JadwalDokter.

2.2.3.18 Spesifikasi Deskripsi

KelasPengelolaanPasien_C

PengelolaanPasien_C	<<Control>>
+PengelolaanPasien_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	
+getDataPasien() : Pasien Operasi ini untuk mengambil data Pasien dari basisdataPasien.	

```

+getRefJenisPasien() : JenisPasien
Operasi ini untuk mengambil data Jenis Pasien dari
basisdataRefJenisPasien.
+getRefKecamatanPasien() : Kecamatan
Operasi ini untuk mengambil data Kecamatan, Kabupaten,
dan Propinsi dari basisdataReferensi Kecamatan,
Kabupaten, dan Propinsi.
+entryDataPasien(Pasien)
Operasi ini untuk membuat data Pasien yang akan
dimasukkan ke basisdataPasien.
+editDataPasien(Pasien)
Operasi ini untuk mengubah data Pasien dari
basisdataPasien.
+displayDataPasien(Pasien)
Operasi ini untuk menampilkan data Pasien dari
basisdata Pasien.
+deleteDataPasien(Pasien): Pasien
Operasi ini untuk menghapus data Pasien dari
basisdataPasien.
+validateDataPasien(): Pasien
Operasi ini untuk mengecek kebenaran nilai Pasien yang
akan dimasukkan apakah sesuai dengan format pada
basisdata Pasien.

```

2.2.3.19 Spesifikasi Deskripsi KelasPengelolaanAntrian_C

PengelolaanAntrian_C	<<Control>>
+PengelolaanAntrian_C() Default konstruktor, digunakan untuk inisialisasi	

```

semua atribut dari kelas ini.
+getDataAntrian():Antrian
Operasi ini untuk mengambil data Antriandari
basisdataAntrian.
+getDataPasien(): Pasien
Operasi ini untuk mengambil data Pasien dari
basisdataPasien.
+getDataJadwalDokter(): JadwalDokter
Operasi ini untuk mengambil data Jadwal Dokterdari
basisdataJadwalDokter.
+entryDataAntrian(Antrian)
Operasi ini untuk membuat data Antrian yang akan
dimasukkan ke basisdataAntrian.
+editDataAntrian(Antrian)
Operasi ini untuk mengubahdata Antrian dari
basisdataAntrian.
+displayDataAntrian(Antrian)
Operasi ini untuk menampilkan data Antrian dari
basisdata Antrian.
+validateDataAntrian(): Antrian
Operasi ini untuk mengecek kebenaran nilai
Antrianyang akan dimasukkan apakah sesuai dengan
format pada basisdata Antrian.

```

2.2.3.20 Spesifikasi Deskripsi

KelasPengelolaanRekamMedis_C

PengelolaanRekamMedis_C	<<Control>>
+PengelolaanRekamMedis_C()	
Default konstruktor, digunakan untuk inisialisasi	

semua attribute dari kelas ini.

+getDataRekamMedis() : RekamMedis
 Operasi ini untuk mengambil data Rekam Medis dari basisdata RekamMedis.

+getDataPasien() : Pasien
 Operasi ini untuk mengambil data Pasien dari basisdata Pasien.

+getDataTindakan() : Tindakan
 Operasi ini untuk mengambil data Tindakan Medis dari basisdata Tindakan.

+getDataDiagnosa() : Diagnosa
 Operasi ini untuk mengambil data Diagnosa dari basisdata Diagnosa.

+getDataPegawai() : Pegawai
 Operasi ini untuk mengambil data Pegawai dari basisdata Pegawai.

+entryDataPemeriksaan(Pemeriksaan)
 Operasi ini untuk membuat data Pemeriksaan Pasien yang akan dimasukkan ke basisdata RekamMedis.

+editDataPemeriksaan(Pemeriksaan)
 Operasi ini untuk mengubah data Pemeriksaan dari basisdata RekamMedis.

+displayDataPemeriksaan(Pemeriksaan)
 Operasi ini untuk menampilkan data Pemeriksaan dari database RekamMedis.

+searchDataPemeriksaan() : Pemeriksaan
 Operasi ini untuk mencari data Pemeriksaan dari basisdata RekamMedis dengan nilai tertentu.

+validateDataPemeriksaan() : Pemeriksaan
 Operasi ini untuk mengecek kebenaran nilai Pemeriksaan yang akan dimasukkan apakah sesuai dengan

format pada basisdata RekamMedis.

2.2.3.21 Spesifikasi Deskripsi KelasPembayaranKlinik_C

PembayaranKlinik_C	<<Control>>
<pre>+PembayaranKlinik_C() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDataRekamMedis() : RekamMedis Operasi ini untuk mengambil data Rekam Medisdari basisdataRekamMedis. +getDataPasien() : Pasien Operasi ini untuk mengambil data Pasiendari basisdataPasien. +getDataTindakan() : Tindakan Operasi ini untuk mengambil data Tindakandari basisdataTindakan. +getDataDiagnosa() : Diagnosa Operasi ini untuk mengambil data Diagnosadari basisdataDiagnosa. +getDataPegawai() : Pegawai Operasi ini untuk mengambil data Pegawaidari basisdataPegawai. +entryPembayaran(Pembayaran) Operasi ini untuk membuat data Pembayaranyang akan dimasukkan ke basisdataRekamMedis. +editPembayaran(Pembayaran) Operasi ini untuk mengubahdata Pembayarandari basisdataRekamMedis. +displayPembayaran(Pembayaran)</pre>	

Operasi ini untuk menampilkan data Pembayaran dari basisdataRekamMedis.

+validatePembayaran() : Pembayaran

Operasi ini untuk mengecek kebenaran nilai Pembayaran yang akan dimasukkan apakah sesuai dengan format pada basisdata RekamMedis.

2.2.3.22 Spesifikasi Deskripsi KelasCetakLaporan_C

CetakLaporan_C	<<Control>>
+CetakLaporan_C() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini. +getDataRekamMedis() : RekamMedis Operasi ini untuk mengambil data Rekam Medis dari basisdataRekamMedis. +getDataPasien() : Pasien Operasi ini untuk mengambil data Pasien dari basisdataPasien. +getDataDiagnosa() : Diagnosa Operasi ini untuk mengambil data Diagnosa dari basisdataDiagnosa.	

2.2.3.23 Spesifikasi Deskripsi Kelas Pegawai_M

Pegawai_M	<<Model>>
-Pegawai : Pegawai Atribut ini digunakan untuk menyimpan data Pegawai pada basisdata Pegawai. +Pegawai_M() Default konstruktor, digunakan untuk inisialisasi semua attribute dari kelas ini.	

+getPegawai() : Pegawai

Operasi ini untuk mengambil data Pegawai dari basisdata Pegawai.

+entryPegawai(Pegawai)

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

+editPegawai(Pegawai)

Operasi ini untuk memperbaharui data Pegawai dari basisdata Pegawai.

+displayPegawai(Pegawai): Pegawai

Operasi ini untuk menampilkan data Pegawai dari basisdata Pegawai.

+nonAktifPegawai(Pegawai): Pegawai

Operasi ini untuk menghapus data Pegawai dari basisdata Pegawai namun dengan cara melakukan set status aktif data sehingga ketika status data non-aktif data tidak akan terseleksi dan tidak ditampilkan.

2.2.3.24 Spesifikasi Deskripsi Kelas Diagnosa_M

Diagnosa_M	<<Model>>
-Diagnosa : Diagnosa Atribut ini digunakan untuk menyimpan data Diagnosa pada basisdata Diagnosa.	
+Diagnosa_M() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+getDiagnosa(Diagnosa) : Diagnosa Operasi ini untuk mengambil data Diagnosa dari basisdata Diagnosa.	

```

+entryDiagnosa(Diagnosa)
Operasi ini untuk membuat data Diagnosayang akan
dimasukkan ke basisdata Diagnosa.
+editDiagnosa(Diagnosa)
Operasi ini untuk memperbaharuidata Diagnosadari
basisdata Diagnosa.
+searchDiagnosa(Diagnosa)
Operasi ini untuk mencaridata Diagnosadari basisdata
Diagnosa.
+nonAktifDiagnosa(Diagnosa): Diagnosa
Operasi ini untuk menghapusdata Diagnosadari
basisdataDiagnosa namun dengan cara melakukan set
status aktif data sehingga ketika status data non-
aktif data tidak akan terseleksi dan tidak
ditampilkan.
+validateDiagnosa(): Diagnosa
Operasi ini untuk mengecek kebenaran nilai Diagnosa
yang akan dimasukkan apakah sesuai dengan format pada
basisdata Diagnosa.
+displayDiagnosa(Diagnosa): Diagnosa
Operasi ini untuk menampilkan data Diagnosa dari
basisdata Diagnosa.

```

2.2.3.25 Spesifikasi Deskripsi Kelas TindakanMedis_M

TindakanMedis_M	<<Model>>
-Tindakan Medis: TindakanMedis Atribut ini digunakan untuk menyimpan data Tindakan Medis pada basisdataTindakanMedis.	
+TindakanMedis_M() Default konstruktor, digunakan untuk inisialisasi	

semua atribut dari kelas ini.

+getTindakanMedis(TindakanMedis) : TindakanMedis
Operasi ini untuk mengambil data Tindakan Medis dari basis data TindakanMedis.

+entryTindakanMedis(TindakanMedis)
Operasi ini untuk membuat data Tindakan Medis yang akan dimasukkan ke basis data TindakanMedis.

+editTindakanMedis(TindakanMedis)
Operasi ini untuk memperbaharui data Tindakan Medis dari basis data TindakanMedis.

+searchTindakanMedis(TindakanMedis)
Operasi ini untuk mencari data Tindakan Medis dari basis data TindakanMedis.

+nonAktifTindakanMedis(TindakanMedis): TindakanMedis
Operasi ini untuk menghapus data Tindakan Medis dari basis data TindakanMedis namun dengan cara melakukan set status aktif data sehingga ketika status data non-aktif data tidak akan terseleksi dan tidak ditampilkan.

+validateTindakanMedis(): TindakanMedis
Operasi ini untuk mengecek kebenaran nilai Tindakan Medis yang akan dimasukkan apakah sesuai dengan format pada basis data TindakanMedis.

+displayTindakanMedis(TindakanMedis): TindakanMedis
Operasi ini untuk menampilkan data Tindakan Medis dari basis data TindakanMedis.

2.2.3.26 Spesifikasi Deskripsi Kelas JadwalDokter_M

JadwalDokter_M	<<Model>>
-JadwalDokter : Jadwal	

Atribut ini digunakan untuk menyimpan data Jadwal Dokter pada basisdata JadwalDokter.

+JadwalDokter_M()

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

+getJadwalDokter(Jadwal)

Operasi ini untuk mengambil data Jadwal Dokter dari basisdata JadwalDokter.

+getDataPegawai(Pegawai)

Operasi ini untuk mengambil data Pegawai dari basisdata Pegawai.

+getHari(Hari)

Operasi ini untuk mengambil data Hari dari basisdata Referensi Hari.

+getSesi(Sesi)

Operasi ini untuk mengambil data Sesi Pemeriksaan dari basisdata Referensi Sesi.

+entryJadwalDokter(Jadwal)

Operasi ini untuk membuat data Jadwal Dokter yang akan dimasukkan ke basisdata JadwalDokter.

+editJadwalDokter(Jadwal)

Operasi ini untuk memperbaharui data Jadwal Dokter dari basisdata JadwalDokter.

+searchJadwalDokter(Jadwal)

Operasi ini untuk mencari data Jadwal Dokter dari basisdata JadwalDokter.

+nonAktifJadwalDokter(JadwalDokter): Jadwal

Operasi ini untuk menghapus data Jadwal Dokter dari basisdata JadwalDokter namun dengan cara melakukan set status aktif data sehingga ketika status data non-aktif data tidak akan terseleksi dan tidak

ditampilkan.

+validateJadwalDokter():JadwalDokter

Operasi ini untuk mengecek kebenaran nilai Jadwal Dokter yang akan dimasukkan apakah sesuai dengan format pada basisdata JadwalDokter.

+displayJadwalDokter(JadwalDokter): JadwalDokter

Operasi ini untuk menampilkan data Jadwal Dokter dari basisdata JadwalDokter.

2.2.3.27 Spesifikasi Deskripsi Kelas Pasien_M

Pasien_M	<<Model>>
-Pasien : Pasien Atribut ini digunakan untuk menyimpan data Pasien pada basisdata Pasien.	
+Pasien_M() Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini. +getDataPasien(Pasien) Operasi ini untuk mengambildata Pasiendari basisdata Pasien. +getRefJenisPasien(Jenis) Operasi ini untuk mengambildata JenisPasiendari basisdata RefJenisPasien. +getRefKecamatanPasien(Kecamatan) Operasi ini untuk mengambildata Kecamatan dari basisdata Referensi Kecamatan. +entryDataPasien(Pasien) Operasi ini untuk membuat data Pasien yang akan dimasukkan ke basisdata Pasien. +editDataPasien(Pasien)	

Operasi ini untuk memperbaharuidata Pasiendari basisdataPasien.

+searchDataPasien(Pasien)

Operasi ini untuk mencaridata Pasien dari basisdata Pasien.

+nonAktifDataPasien(Pasien): Pasien

Operasi ini untuk menghapusdata Pasiendari basisdataPasien namun dengan cara melakukan set status aktif data sehingga ketika status data non-aktif data tidak akan terseleksi dan tidak ditampilkan.

+validateDataPasien(): Pasien

Operasi ini untuk mengecek kebenaran nilai Pasien yang akan dimasukkan apakah sesuai dengan format pada basisdata Pasien.

+displayDataPasien(Pasien): Pasien

Operasi ini untuk menampilkan data Pasien dari basisdata Pasien.

2.2.3.28 Spesifikasi Deskripsi Kelas AntrianPeriksa_M

AntrianPeriksa_M	<<Model>>
-Antrian : Antrian	
Atribut ini digunakan untuk menyimpan data AntrianPeriksa pada basisdataAntrian.	
+AntrianPeriksa_M()	
Default konstruktor, digunakan untuk inisialisasi semua atribut dari kelas ini.	
+getDataAntrian(Antrian)	
Operasi ini untuk mengambildataAntriandari basisdata Antrian.	

```

+getDataPasien(Pasien)
Operasi ini untuk mengambil data Pasien dari basis data Pasien.

+getDataJadwalDokter(Jadwal)
Operasi ini untuk mengambil data Jadwal Dokter dari basis data JadwalDokter.

+entryDataAntrian(Antrian)
Operasi ini untuk membuat data Antrian Pemeriksaan yang akan dimasukkan ke basis data Antrian.

+editDataAntrian(Antrian)
Operasi ini untuk memperbaharui data Antrian Pemeriksaan dari basis data Antrian.

+displayDataAntrian(Antrian): Antrian
Operasi ini untuk menampilkan data Antrian dari basis data Antrian.

+validateDataAntrian(): Antrian
Operasi ini untuk mengecek kebenaran nilai data Antrian Pemeriksaan yang akan dimasukkan apakah sesuai dengan format data pada basis data Antrian.

+searchDataAntrian(Antrian)
Operasi ini untuk mencari data Antrian Pemeriksaan dari basis data Antrian.

```

2.2.3.29 Spesifikasi Deskripsi Kelas RekamMedis_M

RekamMedis_M	<<Model>>
-Rekam Medis : RekamMedis Atribut ini digunakan untuk menyimpan data Rekam Medis pada database RekamMedis.	
+RekamMedis_M() Default konstruktor, digunakan untuk inisialisasi	

semua attribute dari kelas ini.

+getDataRekamMedis() : RekamMedis
 Operasi ini untuk mengambil data Rekam Medis dari basisdata RekamMedis.

+getDataPasien() : Pasien
 Operasi ini untuk mengambil data Pasien dari basisdata Pasien.

+getDataTindakan() : Tindakan
 Operasi ini untuk mengambil data Tindakan Medis dari basisdata Tindakan.

+getDataDiagnosa() : Diagnosa
 Operasi ini untuk mengambil data Diagnosa dari basisdata Diagnosa.

+getDataPegawai() : Pegawai
 Operasi ini untuk mengambil data Pegawai dari basisdata Pegawai.

+entryDataPemeriksaan(Pemeriksaan)
 Operasi ini untuk membuat data Pemeriksaan Pasien yang akan dimasukkan ke basisdata RekamMedis.

+editDataPemeriksaan(Pemeriksaan)
 Operasi ini untuk mengubahdata Pemeriksaan dari basisdataRekamMedis.

+entryPembayaran(Pembayaran)
 Operasi ini untuk membuat data Pembayaran yang akan dimasukkan ke basisdataRekamMedis.

+editPembayaran(Pembayaran)
 Operasi ini untuk mengubahdata Pembayaran dari basisdataRekamMedis.

+displayDataPemeriksaan(Pemeriksaan)
 Operasi ini untuk menampilkan data Pemeriksaan dari database RekamMedis.

```

+validateDataPemeriksaan() : Pemeriksaan
Operasi ini untuk mengecek kebenaran nilai
Pemeriksaan yang akan dimasukkan apakah sesuai dengan
format pada basisdata RekamMedis.
+validatePembayaran() : Pembayaran
Operasi ini untuk mengecek kebenaran nilai Pembayaran
yang akan dimasukkan apakah sesuai dengan format pada
basisdata RekamMedis.
+searchDataPemeriksaan() : Pemeriksaan
Operasi ini untuk mencari data Pemeriksaan dari
basisdataRekamMedis dengan nilai tertentu.

```

3 Perancangan Data

3.1 Dekomposisi Data

3.1.1 Deskripsi Entitas Data Master Pegawai

Nama	Tipe	Panjang	Keterangan
IDPEGAWAI	varchar	50	Nama khusus untuk pegawai yang memiliki hak akses ke dalam sistem, primary key
IDROLE	varchar	2	Foreign key untuk peran pegawai/role pegawai ketika memasuki sistem sebagai apa.
NAMAPEGAWAI	varchar	50	Nama asli dari pegawai
GENDER	varchar	10	Jenis kelamin dari pegawai
Nama	Tipe	Panjang	Keterangan
TANGGALLAHIRPEG	date		Tanggal lahir dari pegawai

ALAMATPEGAWAI	varchar	50	Alamat asal pegawai
NOTELEPON	varchar	20	Nomor telepon dari pegawai
JENISPEGAWAI	varchar	20	Jenis pegawai dimana pegawai tersebut pegawai tetap atau magang
FOTO	longblob(binary)		Foto dari pegawai
PASSWORDPEG	varchar	32	Password dari pegawai untuk masuk ke dalam sistem
AKTIF	int		Status data aktif/non-aktif untuk penghapusan agar tidak terseleksi

3.1.2 Deskripsi Entitas Data Master Diagnosa

Nama	Tipe	Panjang	Keterangan
IDDIAGNOSA	varchar	10	Kode dari diagnose, primary key
NAMADIAGNOSA	varchar	128	Nama dari diagnosa
AKTIF	int		Status data aktif/non-aktif untuk penghapusan agar tidak terseleksi

3.1.3 Deskripsi Entitas Data Master Tindakan Medis

Nama	Tipe	Panjang	Keterangan
IDTINDAKAN	varchar	20	Kodetindakan medis, primary key
IDJENISTINDAKAN	int		Foreign key untuk jenis

			tindakan
NAMATINDAKAN	varchar	160	Nama dari tindakan medis
TARIFUMUM	decimal		Biaya umum tindakan
TARIFKHUSUS	decimal		Biaya khusus tindakan
GAMBAR	longblob(binary)		Gambar logo dari tindakan
AKTIF	int		Status data aktif/non-aktif untuk penghapusan agar tidak terseleksi

3.1.4 Deskripsi Entitas Data MasterJadwal Pemeriksaan untuk Dokter

Nama	Tipe	Panjang	Keterangan
IDJADWAL	varchar	10	Id jadwal pemeriksaan untuk dokter, primary key
IDPEGAWAI	varchar	50	Foreign key dari pegawai
IDREFHARI	varchar	5	Foreign key dari referensi hari
IDREFWAKTU	varchar	5	Foreign key dari referensi sesi
AKTIF	int		Status data aktif/non-aktif untuk penghapusan agar tidak terseleksi

3.1.5 Deskripsi Entitas Data MasterPasien

Nama	Tipe	Panjang	Keterangan
IDPASIEIN	varchar	50	Id pasien, primary key
IDREFJENISPASIEIN	varchar	10	Foreign key dari jenis pasien
IDREFKECAMATAN	varchar	5	Foreign key dari kecamatan untuk

			alamat pasien
TANGGALDAFTAR	date		Tanggal pendaftaran pasien ke dalam sistem
NAMAPASIEN	varchar	25	Nama pasien
TANGGALLAHIR	date		Tanggal lahir pasien
JENISKELAMIN	varchar	10	Jenis kelamin pasien
PEKERJAAN	varchar	25	Pekerjaan pasien
GOLDARAH	varchar	4	Golongan darah pasien
NOHP	varchar	20	Nomor telepon atau handphone pasien
ALAMATPASIEN	varchar	50	Alamat pasien

3.1.6 Deskripsi Entitas Data Master Antrian Periksa

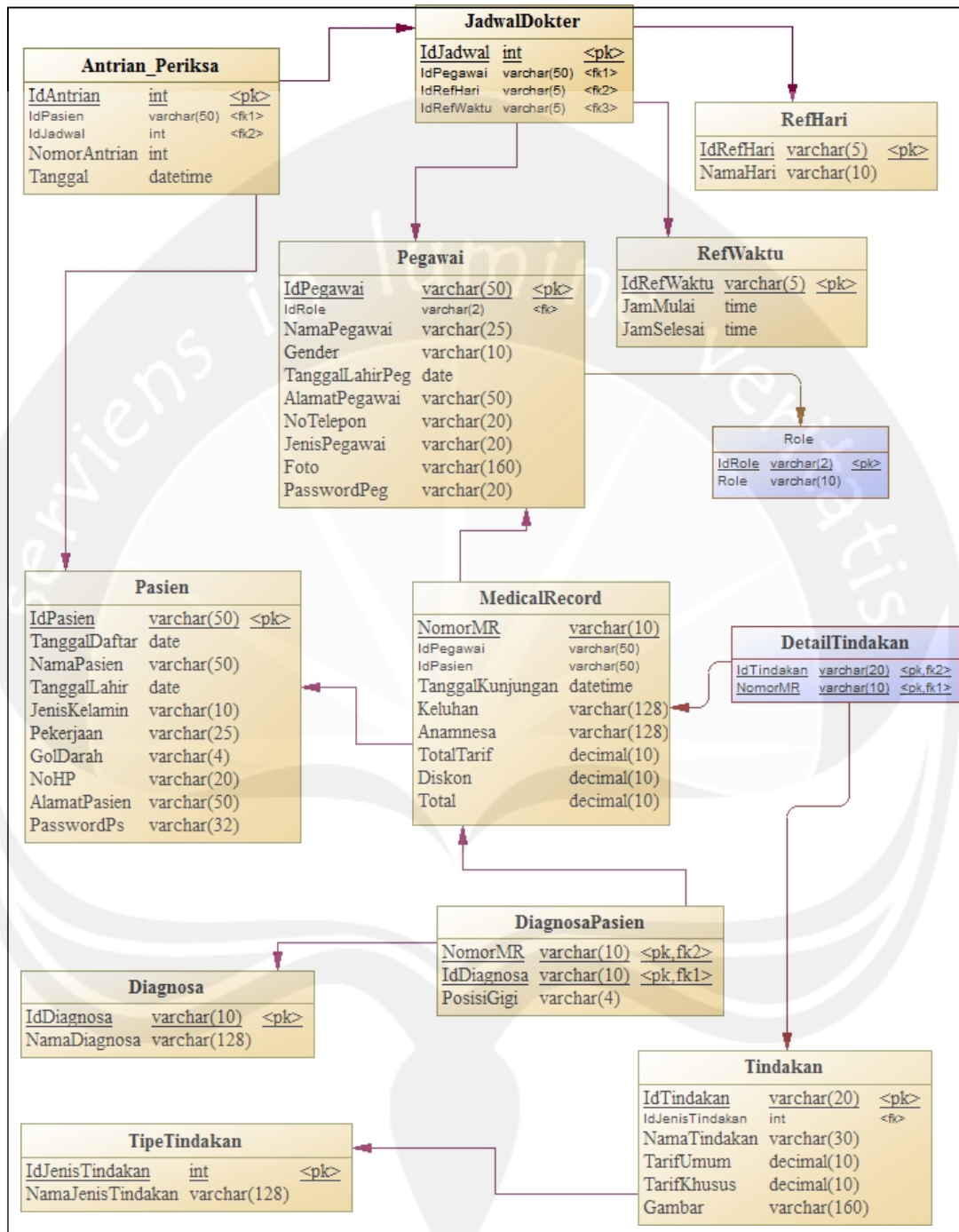
Nama	Tipe	Panjang	Keterangan
IDANTRIAN	varchar	10	Id antrian pemeriksaan untuk pasien
IDPASIEN	varchar	50	Foreign key untuk pasien
IDJADWAL	varchar	10	Foreign key untuk jadwal pemeriksaan yang akan dipilih
NOMORANTRIAN	varchar	30	Nomor antrian yang akan dipakai pasien untuk pemeriksaan
TANGGAL	datetime		Tanggal pemeriksaan

3.1.7 Deskripsi Entitas Data Master Rekam Medis

Nama	Tipe	Panjang	Keterangan
NOMORMR	varchar	10	Nomor rekam medis untuk pemeriksaan pasien, primary

			key
IDPEGAWAI	varchar	50	Foreign key untuk pegawai yang bersangkutan dengan pemeriksaan yaitu dokter.
IDPASIEN	varchar	50	Foreign key untuk pasien yang bersangkutan dengan pemeriksaan.
TANGGALKUNJUNGAN	datetime		Tanggal dan waktu pemeriksaan pasien.
KELUHAN	varchar	128	Keluhan pasien mengenai gigi yang sakit kepada dokter
ANAMNESA	varchar	128	Anamnesa pemeriksaan terhadap pasien
TOTALTARIF	decimal		Total biaya dari tindakan yang diberikan kepada pasien
DISKON	decimal		Potongan harga yang diberikan
TOTAL	decimal		Total biaya setelah dikurangi potongan harga.

3.2 Physical Data Model



Gambar 2.31 Physical Data Model Perangkat Lunak SIGIDES



Header

Ubah Password | Logout | Hi.. (User)

Ubah Password

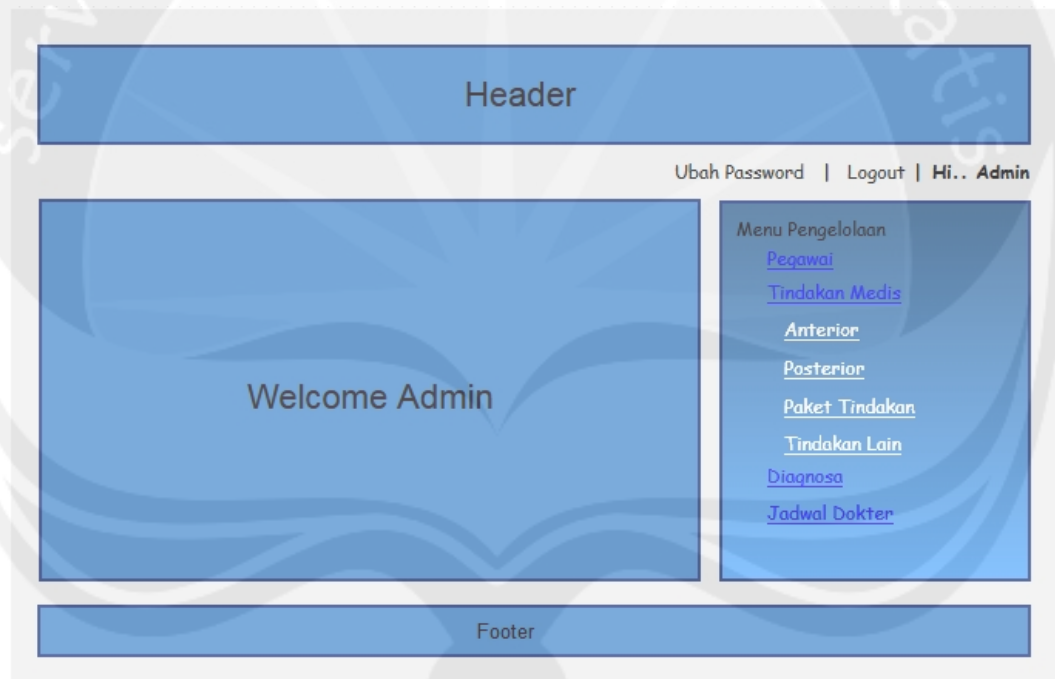
Password Lama

Password Baru

Konfirmasi Password Baru

Menu Kanan

Footer







Header

Ubah Password | Logout | Hi.. Admin

Tambah Pegawai

ID Pegawai

Nama

Role ▼

Gender ▼

Tanggal Lahir

Alamat

No Telepon

Status Pegawai ▼

Menu Pengelolaan

- [Pegawai](#)
- [Tindakan Medis](#)
- [Anterior](#)
- [Posterior](#)
- [Paket Tindakan](#)
- [Tindakan Lain](#)
- [Diagnosa](#)
- [Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Ubah Pegawai

ID Pegawai

admin

Nama

XYZ

Role

ADMINISTRATOR

▼

Gender

PRIA

▼

Tanggal Lahir

1990-12-12

Alamat

Jl. Gayam No. 50 Jogja

No Telepon

+6285729107555

Status Pegawai

TETAP

▼

Simpan

Reset

Kembali

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

dalam basisdata. Namun ketika terdapat kesalahan dalam melakukan pengeditan maka ada tombol Reset untuk mengembalikan data yang ada di form sama seperti sedia kala saat di basisdata. Kemudian ada tombol kembali untuk kembali ke halaman utama pengelolaan pegawai tanpa melakukan pengubahan data.

Kemudian untuk fungsi atur ulang password pegawai dilakukan ketika seorang pegawai mengalami lupa password. *Logical Procedural: UPDATE PEGAWAI SET PASSWORD = \$tanggalahir WHERE IDPEGAWAI=\$idpegawai.* Secara, administrator juga tidak dapat melakukan penggantian password, hanya dapat melakukan penghapusan password dan diganti dengan password baru yang nantinya juga oleh user bersangkutan harus segera diganti. Untuk teknis penggantianannya administrator akan melakukan pengubahan password dengan melakukan ubah data pegawai. Ketika memasukkan data pegawai password diisikan sama seperti tanggal lahir pegawai sehingga ketika terjadi pengubahan sesegera mungkin pegawai yang bersangkutan mengganti password dari tanggal lahir tersebut menjadi password yang sesuai dengan keinginan pegawai.

Header

Ubah Password | Logout | Hi.. Admin

Display Tindakan Medis

Tambah Baru

ID	Nama Tindakan	Tipe	Tarif	Ubah	Hapus
A001	UN-ERUPTED	ANTERIOR	1500	☒	☐
P013	COMPOSITE FILLING	POSTERIOR	7500	☐	☐

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Tambah Tindakan Medis

ID Tindakan

Nama Tindakan

Tipe Tindakan

Tarif Umum

ANTERIOR

Simpan

Reset

Batal

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Ubah Tindakan Medis

ID Tindakan

A001

Nama Tindakan

UN-ERUPTED

Tipe Tindakan

ANTERIOR

▼

Tarif Umum

1500

Simpan

Reset

Batal

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Display Diagnosa

Tambah Baru

ID	Jenis Diagnosa	Ubah	Hapus
D001	GIGI NORMAL	☒	☐
D002	MESIODENS	☐	☐

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Tambah Diagnosa

Kode Diagnosa

Nama Diagnosa

SimpanResetBatal

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Ubah Diagnosa

Kode Diagnosa

A001

Nama Diagnosa

GIGI NORMAL

Simpan

Reset

Batal

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Display Jadwal Dokter

Tambah Baru

ID	Nama Dokter	Hari	Jam Mulai	Jam Selesai	Ubah	Hapus
Sen1	ARBANI	Senin	08.00	10.30	☒	☐
Sen2	MUKTI	Senin	10.30	13.00	☐	☐

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Tambah Jadwal Dokter

Kode Jadwal

Dokter

Hari

Senin

▼

Sesi

S1

▼

Simpan

Reset

Batal

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer

Header

Ubah Password | Logout | Hi.. Admin

Ubah Jadwal Dokter

Kode Jadwal

Sen1

Dokter

ARBANI

Hari

Senin

▼

Sesi

S1

▼

Simpan

Reset

Batal

Menu Pengelolaan

[Pegawai](#)

[Tindakan Medis](#)

[Anterior](#)

[Posterior](#)

[Paket Tindakan](#)

[Tindakan Lain](#)

[Diagnosa](#)

[Jadwal Dokter](#)

Footer



Header

Ubah Password | Logout | Hi.. Resepsionis

Display Pasien

Tambah Baru

ID	Pendaftaran	Nama Pasien	Gender	Ubah	Hapus
00286	2012-11-24	A. FLORETA	WANITA	☒	☐
01016	2012-11-30	ABDUL RAHMAN	PRIA	☐	☐

Pengelolaan Pasien

[Pasien](#)

Antrian Pasien

[Antrian](#)

[Sesi 1](#)

[Sesi 2](#)

[Sesi 3](#)

[Sesi 4](#)

Footer

Header

Ubah Password | Logout | Hi.. **Resepsionis**

Tambah Data Pasien

Kode Pasien

Nama Pasien

Tanggal Lahir

Jenis Kelamin

Pekerjaan

Golongan Darah

Alamat

No Telepon

PRIA

A

Simpan

Reset

Kembali

Pengelolaan Pasien

[Pasien](#)

Antrian Pasien

[Antrian](#)

[Sesi 1](#)

[Sesi 2](#)

[Sesi 3](#)

[Sesi 4](#)

Footer

Header

Ubah Password | Logout | Hi.. Resepsionis

Ubah Data Pasien

Kode Pasien

00286

Nama Pasien

A. FLORETA

Tanggal Lahir

1991-07-05

Jenis Kelamin

WANITA

▼

Pekerjaan

MAHASISWA

Golongan Darah

O

▼

Alamat

Jl. Godean Km. 8 Perum Asri No. 34

No Telepon

+62274567789

Simpan

Reset

Batal

Pengelolaan Pasien

[Pasien](#)

Antrean Pasien

[Antrean](#)

[Sesi 1](#)

[Sesi 2](#)

[Sesi 3](#)

[Sesi 4](#)

Footer

Header

[Ubah Password](#) | [Logout](#) | Hi.. **Resepsionis**

Display Antrian
Tanggal Hari Ini: 2013-04-01

Selesai Periksa

Tambah Antrian

No Antri	Kode Pasien	Nama Pasien	Sesi	Nama Dokter	Status	Cek
1	00286	A. FLORETA	1	ARBANI	MASUK	☒
2	01446	ABE	1	ARBANI	ANTRI	☐

Pengelolaan Pasien
[Pasien](#)

Antrian Pasien
[Antrian](#)
[Sesi 1](#)
[Sesi 2](#)
[Sesi 3](#)
[Sesi 4](#)

Footer

Header

[Ubah Password](#) | [Logout](#) | [Hi.. Resepsionis](#)

Antrian > Pilih Pasien

Kode Pasien	Nama Pasien	Jenis Kelamin	Gol Darah	Tambahkan Antrian
00286	A. FLORETA	WANITA	AB	☒
01446	ABE	PRIA	O	☐

Pengelolaan Pasien

[Pasien](#)

Antrian Pasien

[Antrian](#)

[Sesi 1](#)

[Sesi 2](#)

[Sesi 3](#)

[Sesi 4](#)

Footer

Header

Ubah Password | Logout | Hi.. Resepsionis

Antrian> Pilih Pasien> Pilih Sesi

Kode Pasien01446

Nama PasienABE

Tanggal Antri10-06-2013

Jadwal Dokter yang tersedia pada tanggal ini:

Kode Jadwal	Nama Dokter	Sesi	Jam Mulai	Jam Selesai	Daftar
25	SARDJITO	S3	12:30	15:30	daftar
26	INDAH HAJAR	S4	15:00	18:00	daftar

Pengelolaan Pasien

Pasien

Antrian Pasien

Antrian

Sesi 1

Sesi 2

Sesi 3

Sesi 4

Footer

yaitu tanggal hari itu dan daftar dokter yang tersedia pada jam itu. Sehingga pasien dapat mendaftarkan antrean untuk sesi kapanpun yang tersedia dalam daftar dokter yang terjadwal pada hari itu. Pembagian sesi pada jadwal dokter sesi 1 jam 08.00 sampai dengan jam 10.30, kemudian sesi 2 jam 10.00 sampai dengan jam 13.00, kemudian sesi 3 jam 12.30 sampai dengan jam 15.30, kemudian sesi 4 jam 15.00 sampai dengan jam 18.00. Sedangkan untuk sesi 5 dilaksanakan mulai jam 17.30 sampai dengan jam 21.00. Misalkan pasien datang hari itu pada jam 11.00 maka sistem tidak akan menampilkan sesi 1 karena telah melewati jam dan yang ditampilkan adalah sesi 2,3,4, dan 5. Sehingga pasien tersebut dapat mendaftarkan antrian pada sesi 2,3,4, dan 5. Setelah sesi dipilih halaman akan dikembalikan ke halaman antrian, dan pasien yang terpilih telah masuk ke antrian sesuai dengan sesi jadwal yang dipilih dan nomor antrian menyesuaikan dengan data antrian yang sudah ada dalam sesi tersebut.



Header

[Ubah Password](#) | [Logout](#) | **Hi.. Dokter**

Display Data Pemeriksaan: 10-06-2013

No MR	Nama Pasien	Jam	Status	Periksa
96	INDAH DEWI PERMATA	2013-06-10 15:41:47	MASUK	☒
97	EKO BUDI SUSILO	2013-06-10 15:50:05	MASUK	☐

Pemeriksaan Pasien

[Pemeriksaan](#)

Pengelolaan

[Diagnosa](#)

[Tindakan Medis](#)

Footer

Header

Ubah Password | Logout | Hi.. Dokter

Pemeriksaan Pasien

Nama PasienA. FLORETA

Tampil Riwayat

337 x 55

Keluhan

Anamnesa

DiagnosaResetBatal

Pemeriksaan Pasien

[Pemeriksaan](#)

Pengelolaan

[Diagnosa](#)

[Tindakan Medis](#)

Footer

Header

Ubah Password | Logout | Hi.. Dokter

Pemeriksaan Pasien

Nama PasienA. FLORETA

KeluhanGeraham ngilu

AnamnesaA. FLORETA

Tampil Riwayat

Riwayat Diagnosa dan Pemeriksaan
337 x 55

DiagnosaGIGI NORMAL▼Posisi Gigi-pilih-▼

Tabel Diagnosa Terpilih
337 x 55

SimpanResetBatal

Pemeriksaan Pasien

[Pemeriksaan](#)

Pengelolaan

[Diagnosa](#)

[Tindakan Medis](#)

Footer

Program Studi Teknik Informatika

DPPL – SIGIDES

99/106

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

\$nomormr, \$iddiagnosa, \$posisigigi. Setelah tombol diagnosa ditekan maka sistem akan menampilkan halaman diagnosa, seperti pada perancangan pada gambar 4.27. Ditampilkan informasi mengenai pasien beserta riwayat pemeriksaannya dan keluhan serta anamnesa yang pada halaman sebelumnya diisikan. Untuk pemilihan diagnosa yang akan ditentukan oleh dokter berada di paling bawah. Diagnosa akan dipilih menggunakan *combobox/dropdown list* yang isinya diambil dari tabel diagnosa yang dikelola oleh Administrator dan Dokter. Setelah memilih diagnosa dokter menentukan pada posisi mana gigi yang mengalami sakit. Setelah diagnosa dan posisi dipilih maka untuk menyimpan diagnosa tersebut dokter harus menekan tombol simpan yang berada di bagian bawah dari panel utama. Maka daftar diagnosa akan terisi oleh diagnosa yang telah dipilih dokter. Setelah selesai penambahan diagnosa maka selanjutnya halaman berikutnya adalah halaman tindakan medis.

Header

Ubah Password | Logout | Hi.. Dokter

Pemeriksaan Pasien

Nama Pasien

A. FLORETA

Keluhan

Geraham ngilu

Anamnesa

A. FLORETA

Tampil Diagnosa

Menampilkan Diagnosa Terbaru

337 x 55

Tindakan

ANTERIOR

Posisi Gigi

-pilih-

UN-ERUPTED

Kondisi Gigi

337 x 55

Selesai

Reset

Batal

Pemeriksaan Pasien

[Pemeriksaan](#)

Pengelolaan

[Diagnosa](#)

[Tindakan Medis](#)

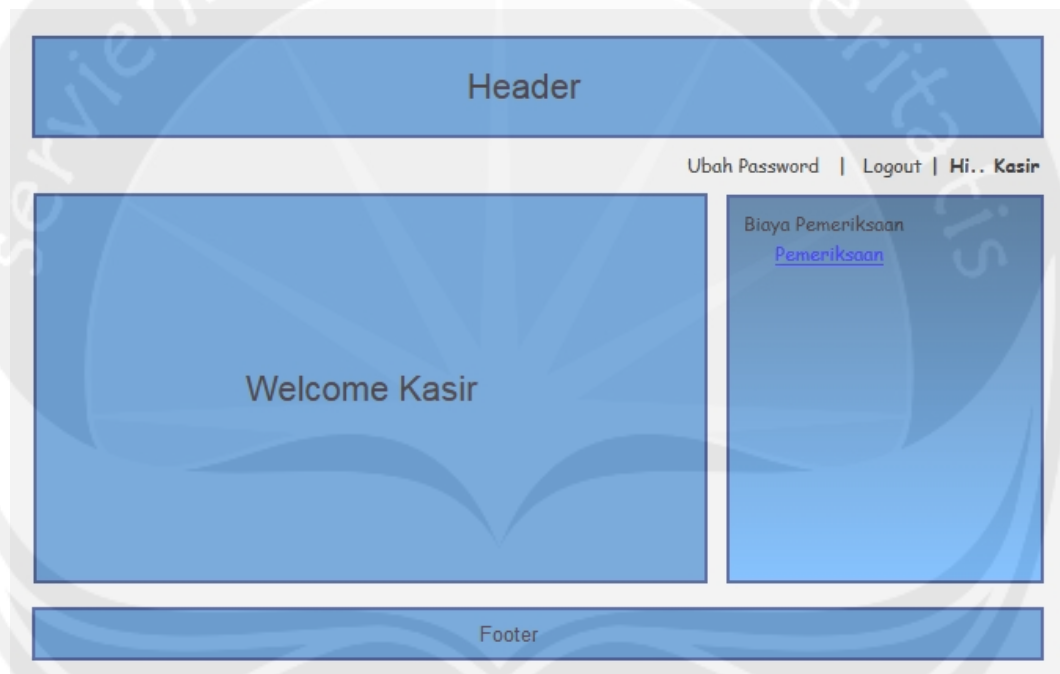
Footer

Program Studi Teknik Informatika

DPPL – SIGIDES

101/106

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



Header

Ubah Password | Logout | Hi.. Kasir

Display Data Pasien Selesai Periksa

No MR	ID Pasien	Pasien	Dokter	Jam Periksa	Detail
96	0130105	INDAH DEWI	SARDJITO	15:41:47	☒
97	0130108	DEWI NURAENI	SARDJITO	15:45:09	☐

Biaya Pemeriksaan

[Pemeriksaan](#)

Footer

Header

Ubah Password | Logout | Hi.. Kasir

Display Detil Pasien

No Rekam Medis

130403Sen101

Nama Pasien

A. FLORETA

Diagnosa

D

Tindakan Medis

T

Total Tarif

Potongan (%)

TOTAL

BAYAR

SISA

Simpan dan Cetak Bukti

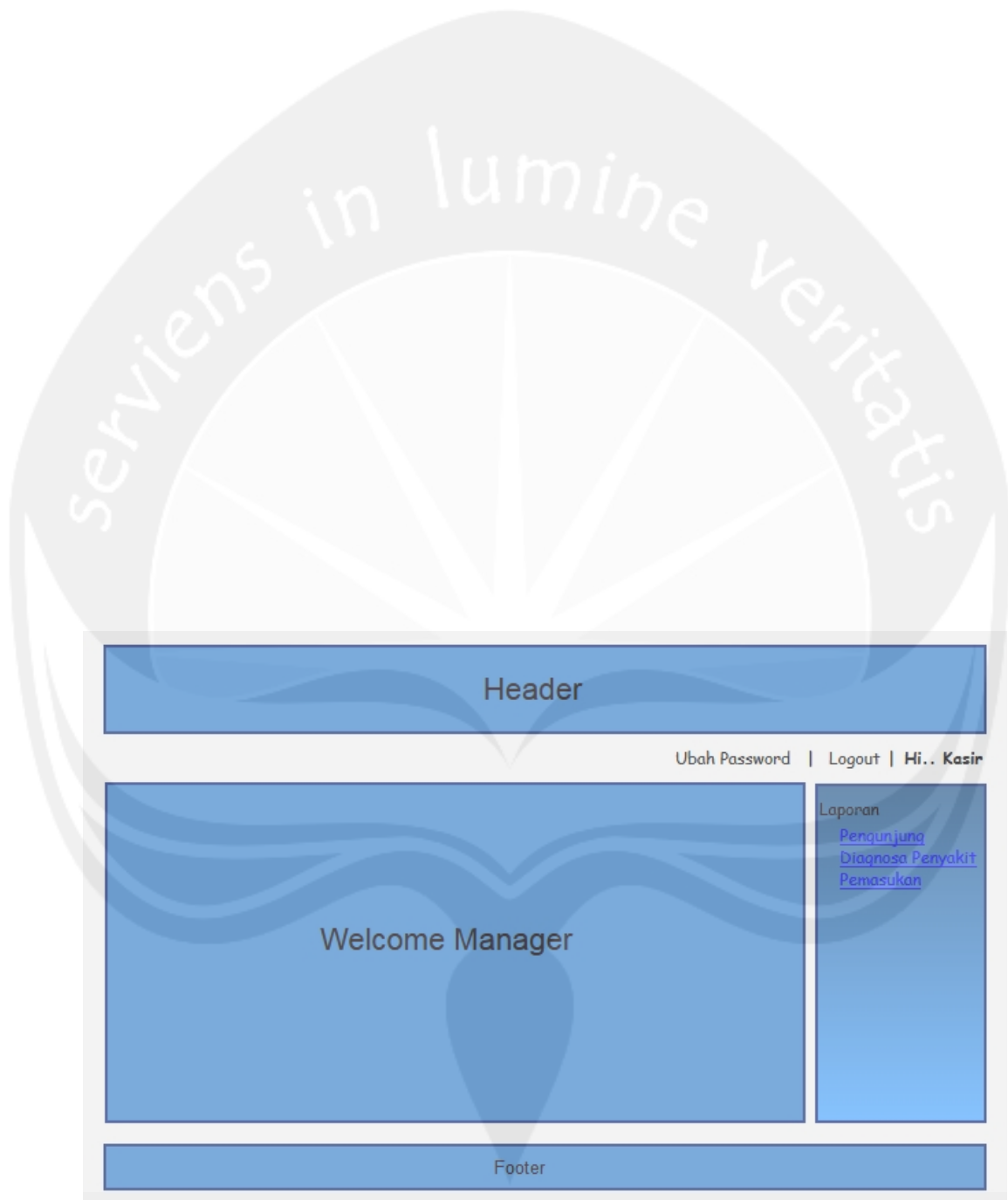
Reset

Kembali

Biaya Pemeriksaan

[Pemeriksaan](#)

Footer



menu dari manager pada panel sidebar diberikan laporan dalam 3 jenis. Ada laporan mengenai pasien atau pengunjung, laporan mengenai diagnosa penyakit dan tindakan medis, serta laporan pemasukan klinik.

Untuk laporan pengunjung akan disampaikan mengenai total seluruh kunjungan, rata-rata usia pengunjung, pengunjung dengan usia muda, pengunjung dengan usia, dan 10 pengunjung yang paling sering berkunjung ke Klinik. Kemudian untuk laporan diagnosa akan disampaikan mengenai diagnosa terbanyak dan tersedikit, tindakan medis terbanyak dan tersedikit, 5 besar diagnosa penyakit, dan 5 besar tindakan medis. Sedangkan laporan yang terakhir adalah laporan mengenai pemasukan. Laporan ini akan memberikan informasi mengenai total pemasukan hari ini, total pemasukan 7 hari terakhir, total pemasukan 1 bulan terakhir, dan total pemasukan per bulan selama 1 tahun.