

DOKUMEN PEMBANGUNAN PERANGKAT LUNAK

Website Manajemen Praktek Dokter

untuk:

Ujian Akhir Semester Manajemen Pengembangan Aplikasi


Dipersiapkan oleh:

Inkreswari Retno Hardini 23518015

Program Studi Teknik Informatika

Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Program Studi Teknik Informatika ITB	Nomor Dokumen		Halaman
		IF-ITB-DOC-2019		
		Revisi	0	10 Mei 2019

DAFTAR ISI

DAFTAR ISI.....	1
DAFTAR GAMBAR	4
I PENDAHULUAN.....	5
I.1 Tujuan Penulisan Dokumen	5
I.2 Lingkup Masalah	5
I.3 Aturan Penomoran	5
I.4 Referensi	5
I.5 Deskripsi umum Dokumen	6
II KEBUTUHAN PERANGKAT LUNAK.....	7
II.1 Deskripsi Umum Sistem	7
II.2 Fitur Utama Perangkat Lunak.....	7
II.2.1 Kebutuhan Fungsional	7
II.2.2 Kebutuhan Non Fungsional	7
II.3 Model <i>Usecase</i>	8
II.3.1 Usecase Package All.....	8
II.3.2 Dokter dan Asisten Dokter Login	8
II.3.3 Dokter dan Asisten Dokter Manajemen Pasien	9
II.3.4 Dokter Manajemen Data Diri.....	9
II.3.5 Dokter dan Asisten Dokter Manajemen Riwayat Kesehatan Pasien.....	9
II.3.6 Dokter dan Asisten Dokter Manajemen Data Resep Pasien	10
II.3.7 <i>Dokter dan Asisten Dokter Manajemen Jadwal Dokter</i>	10
II.4 Definisi Aktor	10
II.5 Definisi Usecase.....	11
II.6 Skenario Usecase	11
II.6.1 <i>Login</i>	11
II.6.2 <i>Manajemen Pasien</i>	12

II.6.3 <i>Manajemen Data Diri</i>	12
II.6.4 <i>Manajemen Riwayat Kesehatan Pasien</i>	13
II.6.5 <i>Manajemen Data Resep Pasien</i>	14
II.6.6 <i>Manajemen Jadwal Dokter</i>	14
II.6.7 <i>Logout</i>	15
II.7 <i>Spesifikasi Tambahan</i>	15
II.8 <i>Glossary</i>	15
III <i>MODEL ANALISIS</i>	16
III.1 <i>Realisasi Usecase Tahap Analisis</i>	16
III.1.1 <i>Login</i>	16
III.1.2 <i>Lihat Data Pasien</i>	17
III.1.3 <i>Edit Data Pasien</i>	17
III.1.4 <i>Tambah Data Pasien</i>	18
III.1.5 <i>Hapus Data Pasien</i>	18
III.1.6 <i>Edit Data Diri Dokter</i>	19
III.1.7 <i>Edit Riwayat Kesehatan</i>	19
III.1.8 <i>Hapus Riwayat Kesehatan</i>	20
III.1.9 <i>Menambah Riwayat Kesehatan</i>	20
III.1.10 <i>Melihat Riwayat Kesehatan</i>	21
III.1.11 <i>Melihat Data Resep</i>	21
III.1.12 <i>Edit Resep</i>	22
III.1.13 <i>Membuat Resep</i>	22
III.1.14 <i>Menghapus Resep</i>	23
III.1.15 <i>Melihat Jadwal</i>	23
III.1.16 <i>Membuat Jadwal</i>	24
III.1.17 <i>Mengedit Jadwal</i>	24
III.1.18 <i>Menghapus Jadwal</i>	25
III.2 <i>Diagram Kelas Keseluruhan</i>	25

III.3 Kelas Analisis	27
III.4 Deskripsi Arsitektur	28
III.5 Perancangan Detil Kelas.....	28
III.5.1 Kelas User	28
III.6.2 Kelas Dokter	29
III.6.3 Kelas AsDok	29
III.6.4 Kelas Pasien.....	30
III.6.5 Kelas Resep	30
III.6.6 Kelas Medical record	31
III.6.7 Kelas Jadwal.....	31
IV IMPLEMENTASI.....	33
IV.1 Lingkungan Implementasi	33
IV.1.1 Hardware yang digunakan untuk implementasi sistem:	33
IV.1.2 Software yang digunakan untuk implementasi sistem:	33

DAFTAR GAMBAR

Gambar 1 Model Usecase Diagram Package All	8
Gambar 2 Dokter dan Asisten Dokter Login	9
Gambar 3 Dokter dan Asisten Dokter Manajemen Pasien	9
Gambar 4 Dokter Manajemen Data Diri	9
Gambar 5 Dokter dan Asisten Dokter Manajemen Riwayat Kesehatan Pasien	9
Gambar 6 Dokter dan Asisten Dokter Manajemen Data Resep Pasien	10
Gambar 7 Dokter dan Asisten Dokter Manajemen Jadwal Dokter	10
Gambar 8 Sequence Login.....	16
Gambar 9 Lihat Data Pasien	17
Gambar 10 Edit Data Pasien	17
Gambar 11 Tambah Data Pasien.....	18
Gambar 12 Hapus Data Pasien.....	18
Gambar 13 Edit Data Diri Dokter	19
Gambar 14 Edit Riwayat Kesehatan	19
Gambar 15 Hapus Riwayat Kesehatan	20
Gambar 16 Tambah Riwayat Kesehatan.....	20
Gambar 17 Melihat Riwayat Kesehatan	21
Gambar 18 Melihat Data Resep	21
Gambar 19 Edit Resep	22
Gambar 20 Membuat Resep	22
Gambar 21 Menghapus Resep	23
Gambar 22 Melihat Jadwal.....	23
Gambar 23 Membuat Jadwal	24
Gambar 24 Edit Jadwal	24
Gambar 25 Hapus Jadwal	25
Gambar 26 Class Diagram	26

I PENDAHULUAN

I.1 Tujuan Penulisan Dokumen

Dokumen *Software Requirement Specification* (SRS) merupakan dokumentasi dari spesifikasi perangkat lunak untuk membangun "Website Manajemen Proyek Dokter". Aplikasi ini dibangun untuk memudahkan dokter serta asisten dokter untuk mendata pasien, mengetahui riwayat kesehatan pasien, memberikan resep secara online kepada pasien serta penjadwalan dokter. Sehingga dokumen ini dapat dijadikan acuan teknis untuk membangun perangkat lunak "WMPD".

I.2 Lingkup Masalah

Website Manajemen Proyek Dokter (WMPD), merupakan aplikasi yang dibangun untuk memudahkan dokter serta asisten dokter untuk mendata pasien, mengetahui riwayat kesehatan pasien, memberikan resep secara online kepada pasien serta penjadwalan dokter.

Aplikasi ini akan memperlihatkan aktivitas yang dilakukan dokter atau asisten dokter terkait pemberian layanan kepada pasien.

I.3 Aturan Penomoran

Pola penomoran yang digunakan pada dokumen ini sebagai berikut :

1. WMPD-SRS-F-### → digunakan untuk penomoran kebutuhan Fungsional
2. WMPD-SRS-NF-### → digunakan untuk penomoran kebutuhan Non Fungsional

Contoh :

1. WMPD-SRS-F-001 melihat riwayat kesehatan pasien.
2. WMPD-SRS-NF-001 Setiap pengguna harus melakukan login terlebih dahulu.

I.4 Referensi

Beberapa daftar referensi yang menjadi panduan dalam menyusun dokumentasi perangkat lunak ini, diantaranya:

1. Roger S Pressman, *Software Engineering: A Practitioner's Approach*, 6/e. R.S. Pressman and Associates.
2. Munawar, *Pemodelan Visual dengan UML*", Penerbit Graha Ilmu.
3. Template Dokumentasi Perancangan Perangkat Lunak PP IF-ITB-UP-DOC-2009.

I.5 Deskripsi umum Dokumen

Dokumentasi SRS ini dibagi menjadi tiga bagian, yaitu:

1. Pendahuluan

Pada bagian ini berisi penjelasan tentang dokumen SRS yang mencakup: (i) Tujuan Pembuatan Dokumen, (ii) Lingkup Masalah yang diselesaikan oleh Perangkat Lunak (PL), (iii) Penomoran, (iv) Referensi dan (v) Deskripsi Umum.

2. Kebutuhan Perangkat Lunak

Pada bagian ini berisi penjelasan tentang dokumen SRS yang mencakup: (i) Deskripsi Umum Sistem; (ii) Model *Usecase*; (iii) Spesifikasi Tambahan yang diperlukan oleh aplikasi.

3. Model Analisis

Pada bagian ini berisi penjelasan tentang dokumen SRS yang mencakup: Realisasi *Usecase* Tahap Analisis, Diagram Kelas Keseluruhan, Kelas Analisis, Paket Analisis, Pedoman Perancangan, Perancangan Detil Elemen Logical View, Perancangan Detil Kelas.

II KEBUTUHAN PERANGKAT LUNAK

II.1 Deskripsi Umum Sistem

Website Manajemen Proyek Dokter (WMPD) merupakan aplikasi yang dibangun untuk memudahkan dokter serta asisten dokter untuk mendata pasien, mengetahui riwayat kesehatan pasien, memberikan resep secara online kepada pasien serta penjadwalan dokter.

II.2 Fitur Utama Perangkat Lunak

Beberapa fitur yang terdapat didalam sistem ini dibedakan menjadi 2 bagian, diantaranya secara fungsional (F) dan non fungsional (NF).

II.2.1 Kebutuhan Fungsional

Nomor	Deskripsi Kebutuhan
WMPD-SRS-F-001	Manajemen Pasien
WMPD-SRS-F-002	Manajemen Dokter
WMPD-SRS-F-003	Menampilkan Riwayat Kesehatan Pasien
WMPD-SRS-F-004	Manajemen Riwayat Kesehatan Pasien
WMPD-SRS-F-005	Menampilkan Data Resep Pasien
WMPD-SRS-F-006	Manajemen Data Resep Pasien
WMPD-SRS-F-007	Manajemen Jadwal Dokter

II.2.2 Kebutuhan Non Fungsional

Nomor	Deskripsi Kebutuhan
WMPD-SRS-NF-001	Sistem dilengkapi dengan fasilitas login untuk keperluan autentikasi pengguna.
WMPD-SRS-NF-002	Sistem dilengkapi dengan menu Help untuk membantu apabila terdapat pengguna maupun pengguna baru yang mengalami kesulitan dalam menggunakan sistem. Pengguna akan diarahkan dengan benar sehingga pengguna maupun pengguna baru dapat dengan mudah menggunakan sistem.
WMPD-SRS-NF-003	Sistem memiliki waktu respon kurang dari 30 detik
WMPD-SRS-NF-004	Sistem melakukan enkripsi terhadap password yang dipakai pengguna

WMPD-SRS-NF-005	Sistem menggunakan satuan waktu per 30 menit untuk durasi penjadwalan dokter.
WMPD-SRS-NF-006	Sistem akan <i>stand by</i> 24 jam untuk memudahkan pengguna beraktivitas dan keakuratan datanya baik.

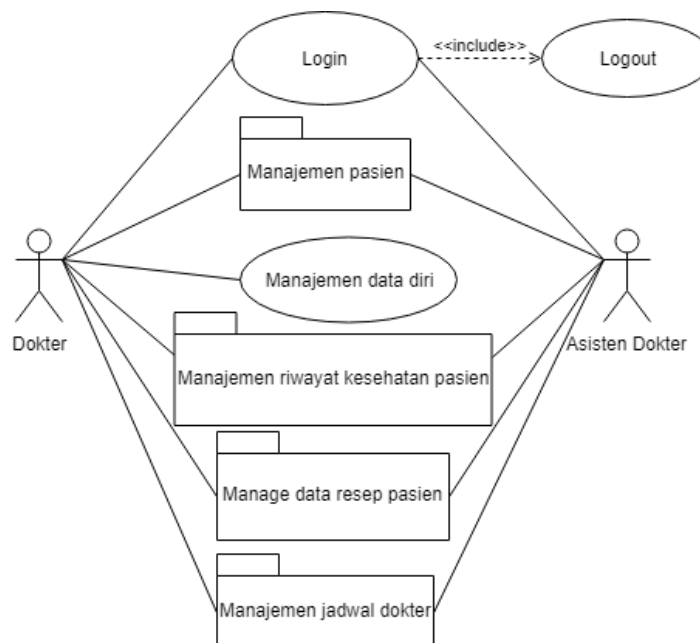
II.3 Model *Usecase*

Model *usecase* ini akan menjelaskan mengenai gambaran aktifitas yang dilakukan oleh tipe aktor terhadap sistem WMPD.

II.3.1 Usecase Package All

Usecase ini akan menggambarkan keseluruhan *package* yang akan digunakan didalam sistem, didalam *package* ini akan dijelaskan lebih detail kedalam *usecase* yang akan didefinisikan berdasarkan aktor yang berinteraksi langsung. Didalam sistem ini aktor dibedakan menjadi dua buah bagian diantaranya:

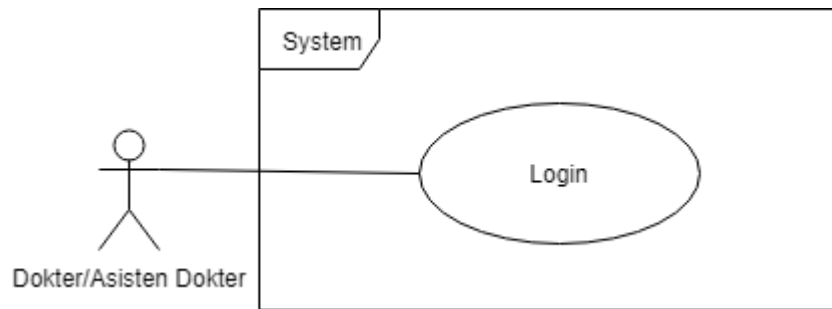
- (1) *Dokter*, Aktor ini termasuk kedalam aktor yang sudah terdaftar didalam sistem dan dapat melakukan *login* untuk mengakses data didalam sistem ;
- (2) *Asisten Dokter*, Aktor ini termasuk kedalam aktor yang sudah terdaftar didalam sistem dan dapat melakukan *login* untuk mengakses data didalam sistem.



Gambar 1 Model Usecase Diagram Package All

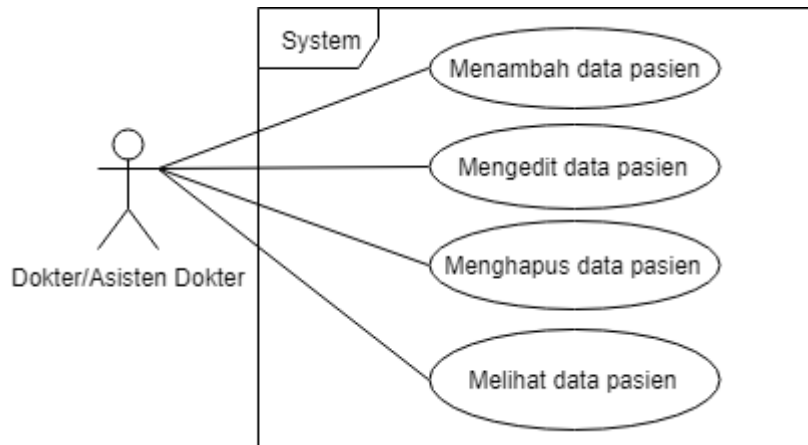
II.3.2 Dokter dan Asisten Dokter Login

Usecase ini akan menjelaskan mengenai keterhubungan aktor terhadap sistem untuk melakukan *login*.



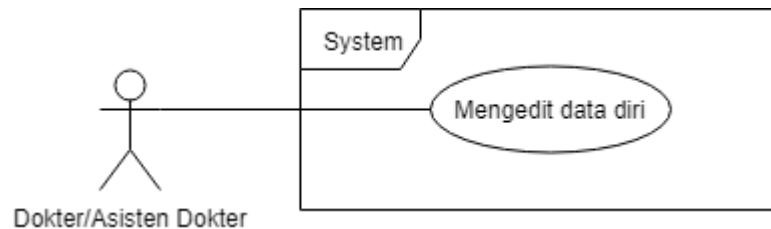
Gambar 2 Dokter dan Asisten Dokter Login

II.3.3 Dokter dan Asisten Dokter Manajemen Pasien



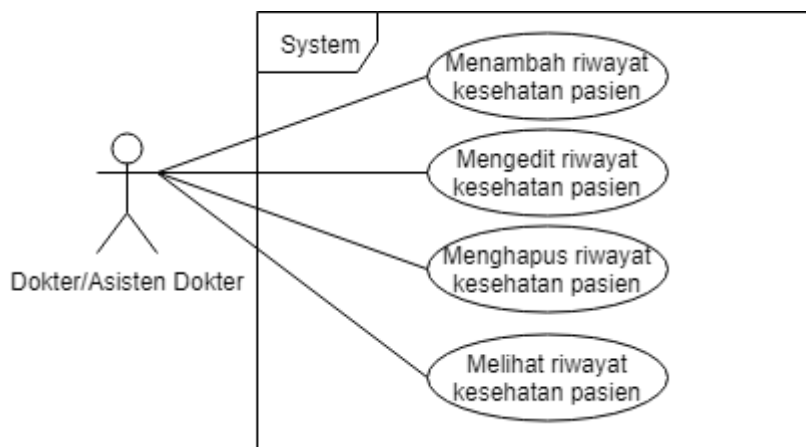
Gambar 3 Dokter dan Asisten Dokter Manajemen Pasien

II.3.4 Dokter Manajemen Data Diri



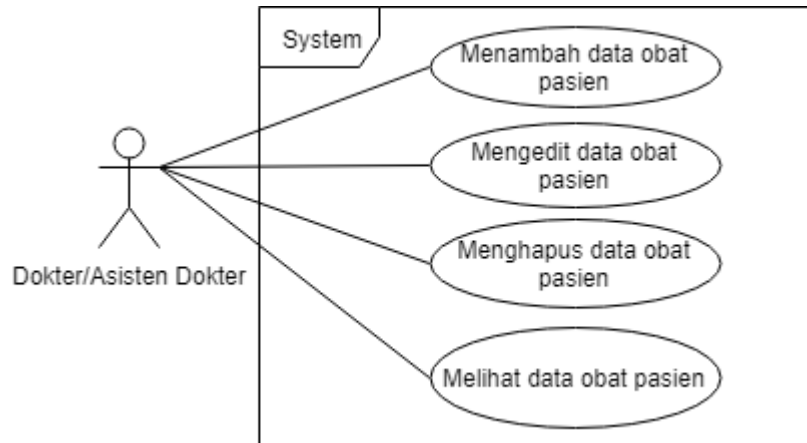
Gambar 4 Dokter Manajemen Data Diri

II.3.5 Dokter dan Asisten Dokter Manajemen Riwayat Kesehatan Pasien



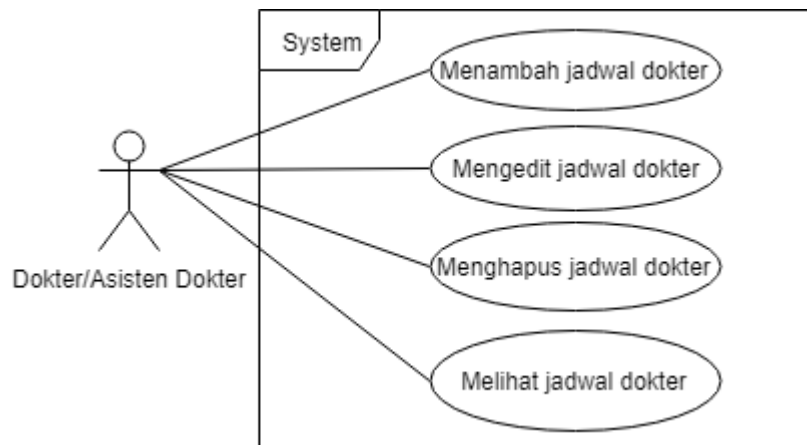
Gambar 5 Dokter dan Asisten Dokter Manajemen Riwayat Kesehatan Pasien

II.3.6 Dokter dan Asisten Dokter Manajemen Data Resep Pasien



Gambar 6 Dokter dan Asisten Dokter Manajemen Data Resep Pasien

II.3.7 Dokter dan Asisten Dokter Manajemen Jadwal Dokter



Gambar 7 Dokter dan Asisten Dokter Manajemen Jadwal Dokter

II.4 Definisi Aktor

Pada bagian ini akan menjelaskan tentang peran dari masing-masing *actor* yang terdapat di dalam sistem.

No	Aktor	Deskripsi
1	Dokter	<ul style="list-style-type: none">▪ <i>Actor</i> ini akan memiliki fungsi untuk manajemen data pasien, manajemen data diri, manajemen riwayat kesehatan pasien, manajemen data resep pasien, manajemen jadwal dokter.
2	Asisten Dokter	<ul style="list-style-type: none">▪ <i>Actor</i> ini akan memiliki fungsi untuk manajemen data pasien, manajemen data diri, manajemen riwayat kesehatan pasien,

manajemen data resep pasien, manajemen jadwal dokter.

II.5 Definisi Usecase

No	Usecase	Deskripsi
1	<i>Login</i>	Fungsi ini digunakan untuk login ke dalam sistem
2	Manajemen Pasien	Fungsi ini digunakan untuk menambah, mengedit dan menghapus data pasien.
3	<i>Manajemen Data Diri</i>	Fungsi ini digunakan untuk mengedit data diri bagi user dokter.
4	<i>Manajemen Riwayat Kesehatan Pasien</i>	Fungsi ini digunakan untuk melihat, mengedit, membuat baru dan menghapus data riwayat kesehatan pasien.
5	<i>Manajemen Data Resep Pasien</i>	Fungsi ini digunakan untuk melihat, mengedit, membuat baru dan menghapus data resep obat pasien.
6	Manajemen Jadwal Dokter	Fungsi ini digunakan untuk mengatur jadwal dokter (membuat, mengedit, melihat, dan menghapus jadwal dokter).

II.6 Skenario Usecase

Bagian ini membahas mengenai skenario setiap *usecase*, beberapa penjelasan tersebut adalah seperti:

II.6.1 Login

Aksi actor	Reaksi system
Skenario normal	
1. Memasukan username dan password menekan submit	
	2. Mengecek kedalam database username dan password cocok.
3. Tampilkan halaman sesuai role dokter dan asisten dokter dan user sudah berhasil login.	
Skenario alternatif	

1. Memasukan username dan password menekan submit	
	2. Mengecek kedalam database username dan password tidak cocok.
3. Tampilkan halaman home dan user gagal login.	

II.6.2 Manajemen Pasien

Aksi actor	Reaksi system
Skenario normal	
1. Memilih <i>menu Pasien</i>	
	2. Menampilkan daftar pasien
3.Membaca data pasien	
4.Mengedit data pasien	
	5. Berhasil menyimpan hasil perbaharuan ke database
Skenario alternatif	
1. Memilih <i>menu Pasien</i>	
	2. Menampilkan daftar pasien
3.Membaca data pasien	
4.Mengedit data pasien	
	5. Gagal menyimpan hasil perbaharuan ke database
6.Tampilkan pesan gagal	

II.6.3 Manajemen Data Diri

Aksi actor	Reaksi system
Skenario normal	
1. Memilih <i>menu Data Diri</i>	
	2. Menampilkan data diri
3.Membaca data diri	
4.Mengedit data diri	
	5. Berhasil menyimpan hasil perbaharuan ke database

Skenario alternatif	
1. Memilih <i>menu Data Diri</i>	
	2. Menampilkan data diri
3.Membaca data diri	
4.Mengedit data diri	
	5. Gagal menyimpan hasil perbaharuan ke database
6.Tampilkan pesan gagal	

II.6.4 Manajemen Riwayat Kesehatan Pasien

Aksi actor	Reaksi sistem
Skenario normal	
1. Memilih <i>menu Riwayat Pasien</i>	
	2. Menampilkan data riwayat kesehatan pasien
3.Membaca data riwayat kesehatan pasien	
4.Mengedit data riwayat kesehatan pasien	
	5. Berhasil menyimpan hasil perbaharuan ke database
Skenario alternatif	
1. Memilih <i>menu Riwayat Pasien</i>	
	2. Menampilkan data riwayat kesehatan pasien
3.Membaca data riwayat kesehatan pasien	
4.Mengedit data riwayat kesehatan pasien	
	5. Gagal menyimpan hasil perbaharuan ke database
6.Tampilkan pesan gagal	

II.6.5 Manajemen Data Resep Pasien

Aksi actor	Reaksi sistem
Skenario normal	
1. Memilih <i>menu Data Resep</i>	
	2. Menampilkan data resep pasien
3.Membaca data resep pasien	
4.Mengedit data resep pasien	
	5. Berhasil menyimpan hasil perbaharuan ke database
Skenario alternatif	
1. Memilih <i>menu Data Resep</i>	
	2. Menampilkan data resep pasien
3.Membaca data resep pasien	
4.Mengedit data resep pasien	
	5. Gagal menyimpan hasil perbaharuan ke database
6.Tampilkan pesan gagal	

II.6.6 Manajemen Jadwal Dokter

Aksi actor	Reaksi sistem
Skenario normal	
1. Memilih <i>menu Jadwal</i>	
	2. Menampilkan data jadwal yang ada di dalam sistem.
3.Membaca jadwal dokter	
4.Mengedit jadwal dokter	
	5. Berhasil menyimpan hasil perbaharuan ke database
Skenario alternatif	
1. Memilih <i>menu Jadwal</i>	
	2. Menampilkan data jadwal yang ada di dalam sistem.
3.Membaca jadwal dokter	
4.Mengedit jadwal dokter	

5. Gagal menyimpan hasil perbaharuan ke database
6. Tampilkan pesan gagal

II.6.7 Logout

Aksi actor	Reaksi sistem
<i>Skenario normal</i>	
1. Memilih <i>menu logout</i>	
	2. Menampilkan <i>form logout</i>
3. Menekan tombol <i>logout</i>	
	4. Berhasil keluar dari sistem dan menampilkan halaman depan aplikasi

II.7 Spesifikasi Tambahan

Mengasumsikan kegiatan yang dilakukan *actor* terhadap sistem aplikasi harus melakukan login terlebih dahulu.

II.8 Glossary

Daftar glossary yang digunakan pada dokumentasi SRS ini sebagai berikut:

1. Akronim

- a. WMPD: Website Manajemen Praktek Dokter
- b. SRS : Software Requirement Specification

2. Definisi

- a. SRS adalah dokumentasi perangkat lunak yang akan dibuat yang merupakan sebagai penyematani komunikasi pengembang dengan pengguna.
- b. *Usecase* adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna.
- c. *Actor* adalah sebuah peran yang bisa dimainkan oleh pengguna dalam interaksinya dengan sistem.

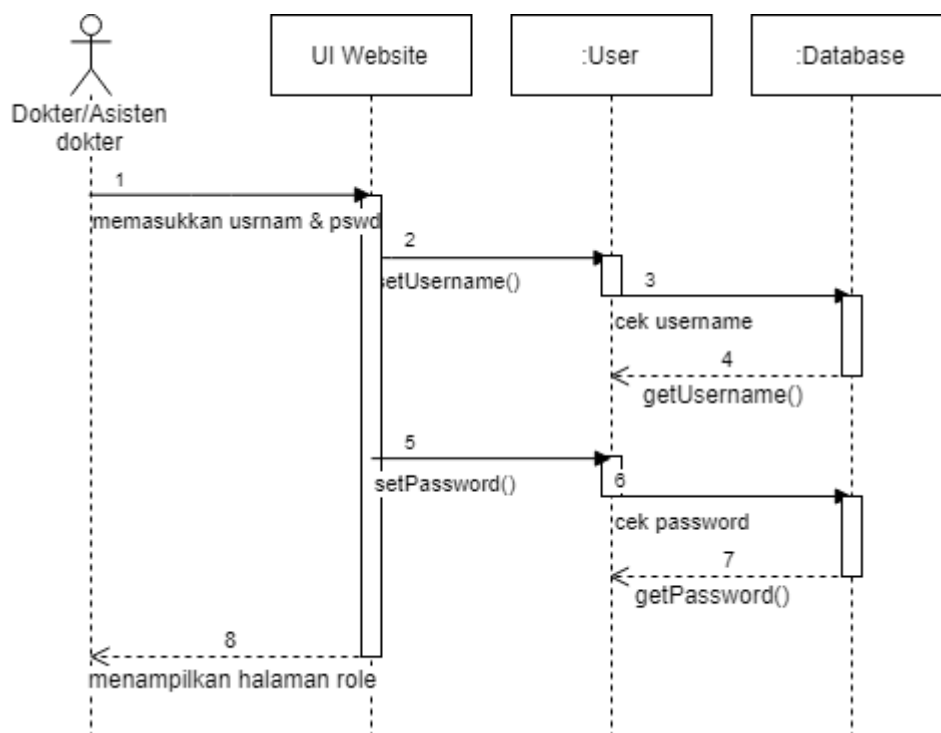
III MODEL ANALISIS

III.1 Realisasi Usecase Tahap Analisis

Bagian ini akan menjelaskan aktivitas dari setiap *user* terhadap sistem. Kegiatan *actor* tersebut akan digambarkan dalam model *sequence diagram* dan selanjutnya akan didefinisikan mengenai kebutuhan sistem untuk memenuhi kegiatan dari masing-masing *actor* kedalam *class diagram*.

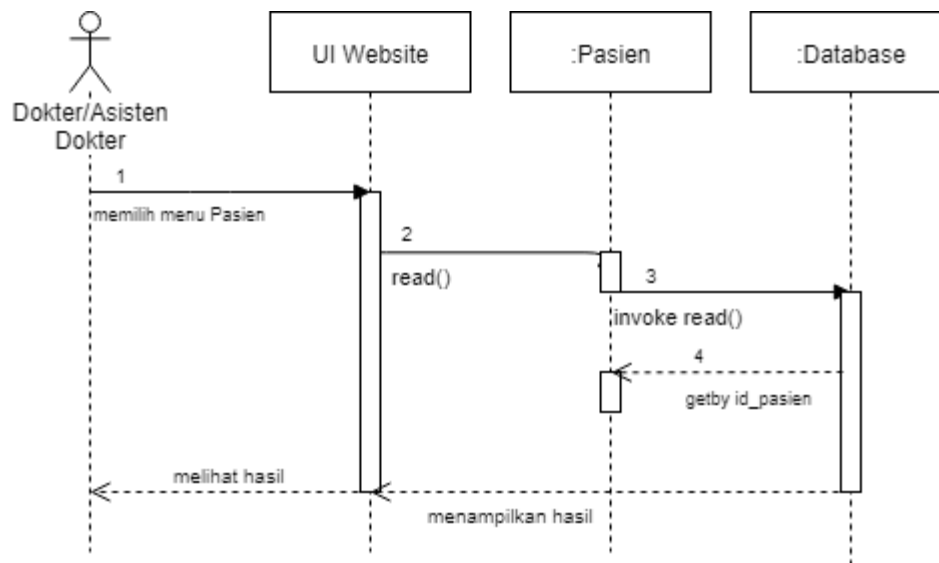
Pada bagian *sequence* ini akan mendefinisikan masing-masing aliran kegiatan *actor* terhadap sistem, diantaranya:

III.1.1 Login



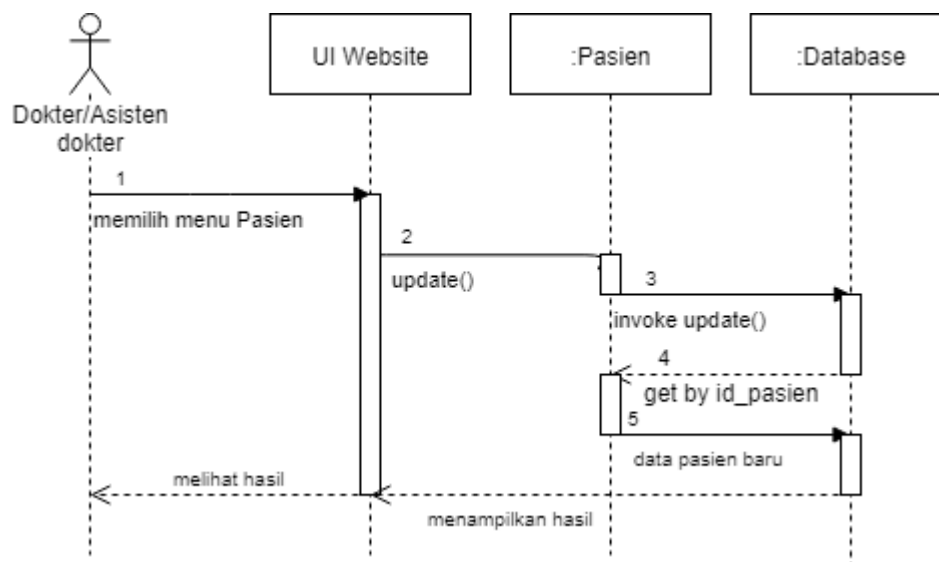
Gambar 8 Sequence Login

III.1.2 Lihat Data Pasien



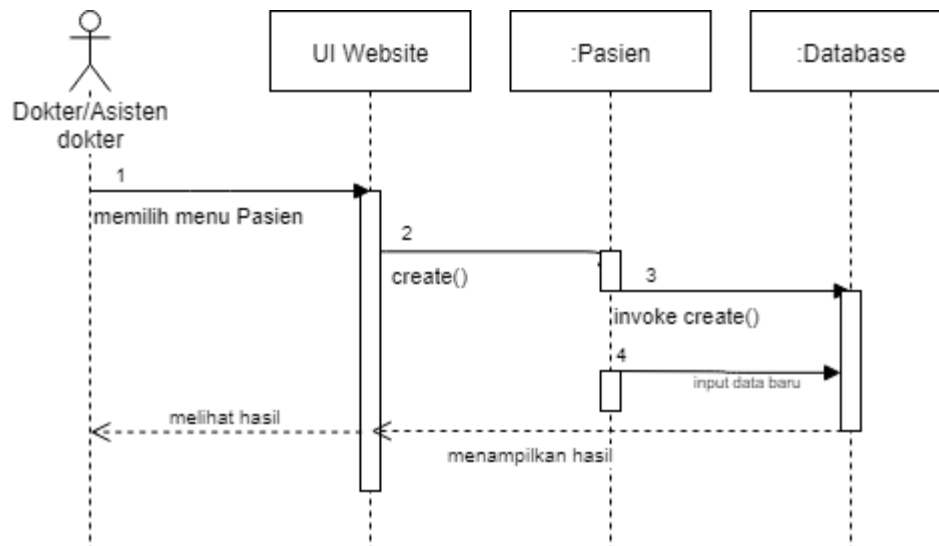
Gambar 9 Lihat Data Pasien

III.1.3 Edit Data Pasien



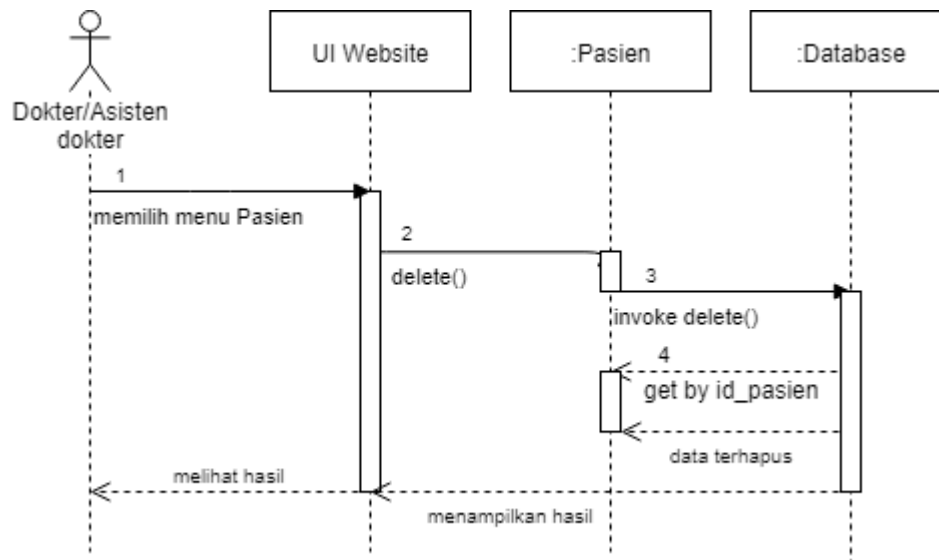
Gambar 10 Edit Data Pasien

III.1.4 Tambah Data Pasien



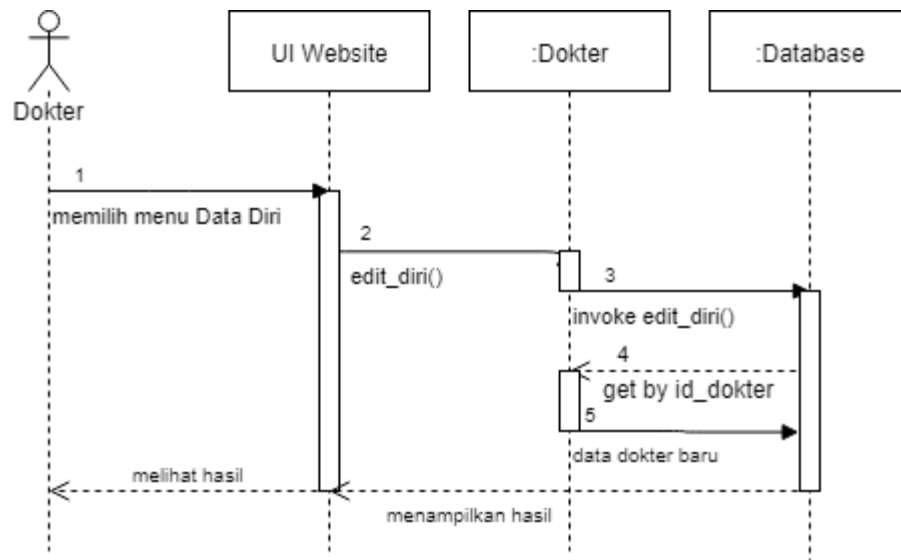
Gambar 11 Tambah Data Pasien

III.1.5 Hapus Data Pasien



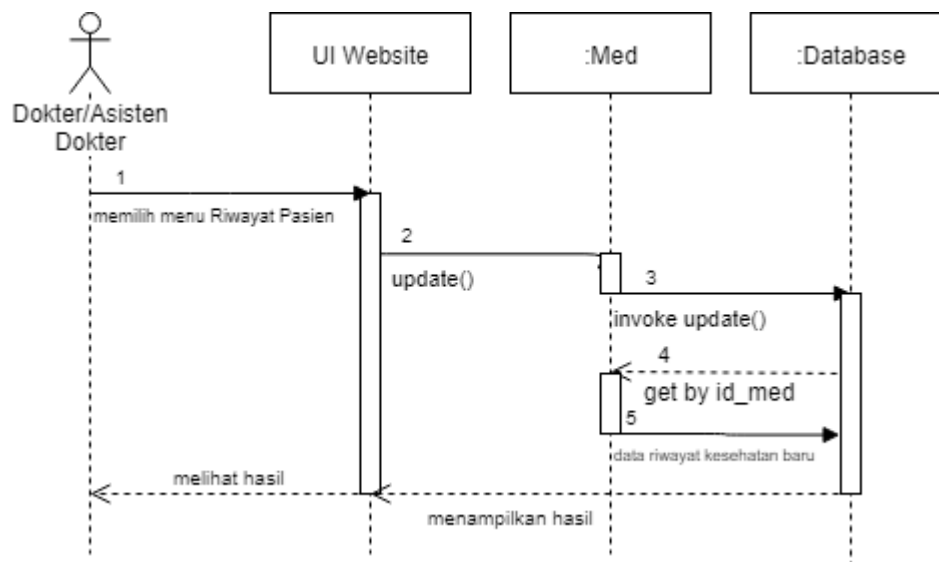
Gambar 12 Hapus Data Pasien

III.1.6 Edit Data Diri Dokter



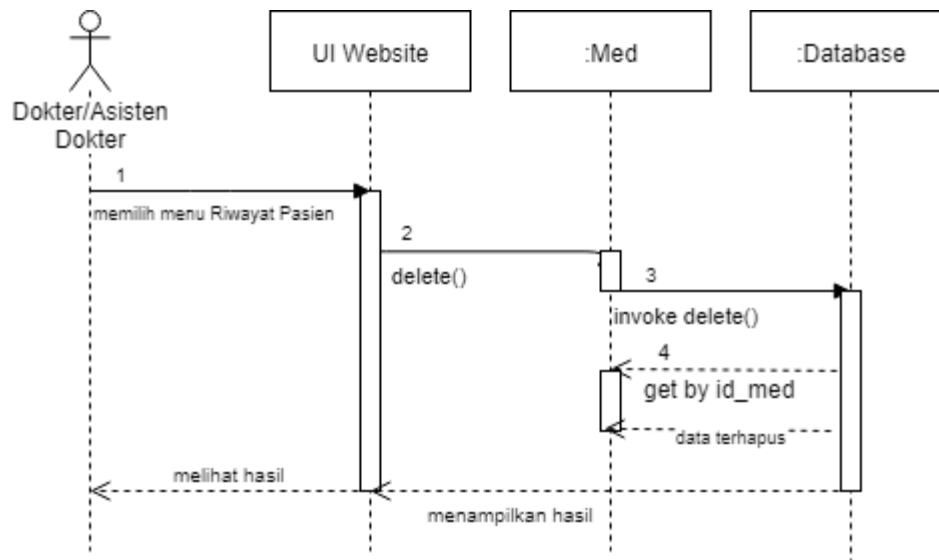
Gambar 13 Edit Data Diri Dokter

III.1.7 Edit Riwayat Kesehatan



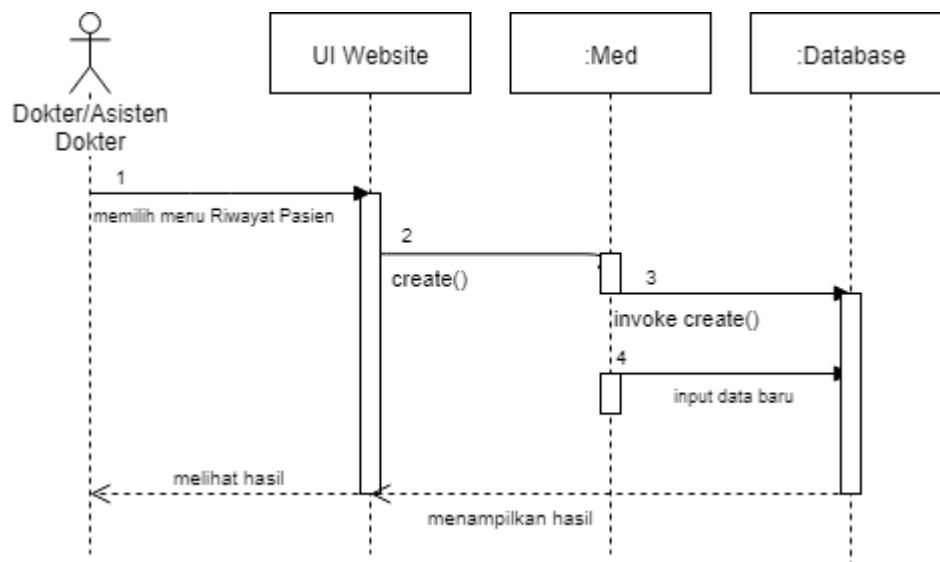
Gambar 14 Edit Riwayat Kesehatan

III.1.8 Hapus Riwayat Kesehatan



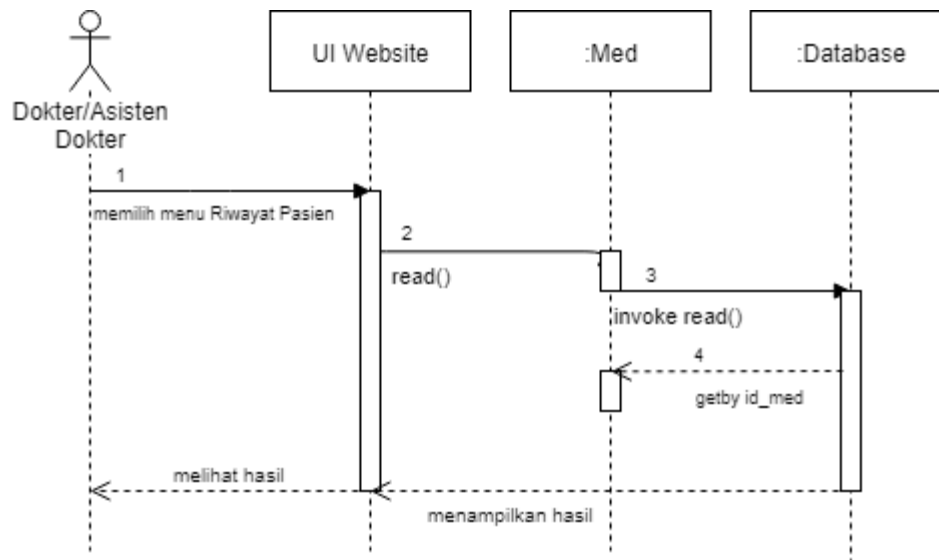
Gambar 15 Hapus Riwayat Kesehatan

III.1.9 Menambah Riwayat Kesehatan



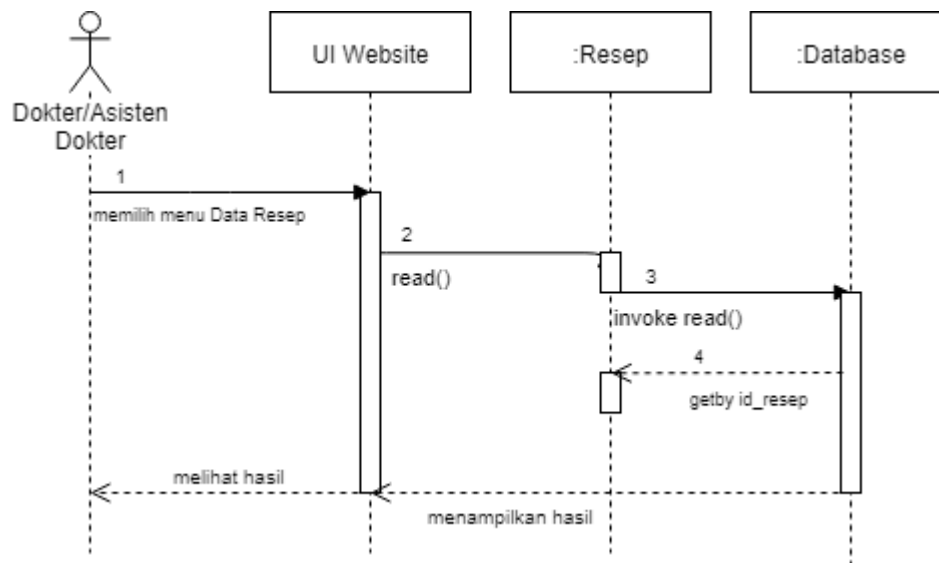
Gambar 16 Tambah Riwayat Kesehatan

III.1.10 Melihat Riwayat Kesehatan



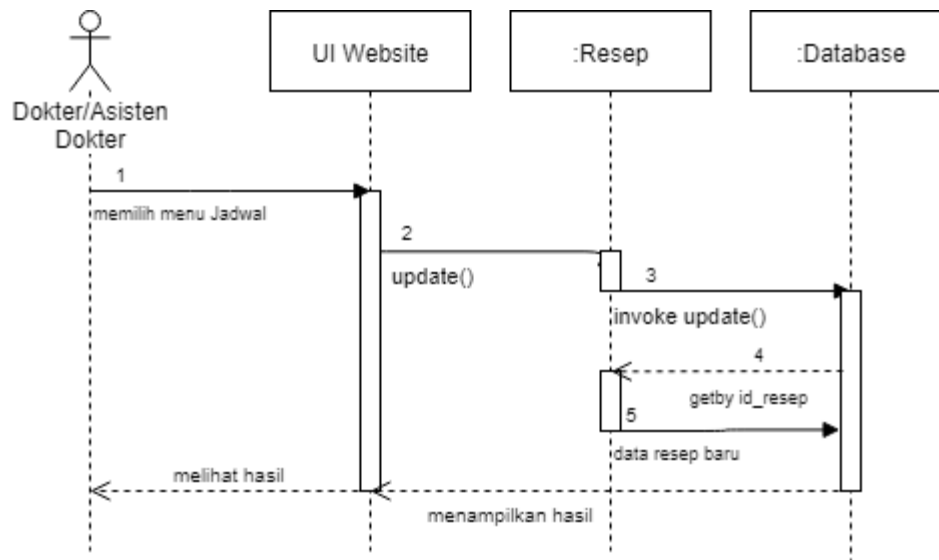
Gambar 17 Melihat Riwayat Kesehatan

III.1.11 Melihat Data Resep



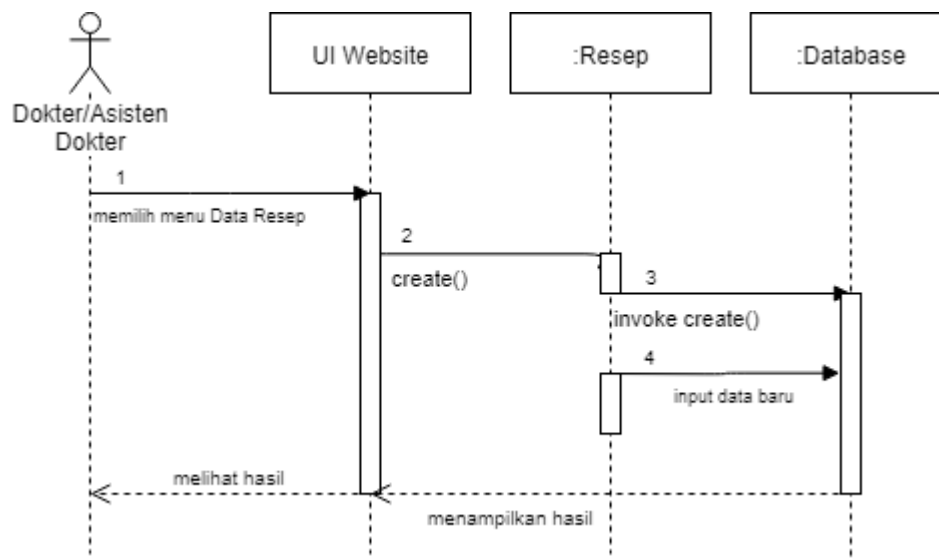
Gambar 18 Melihat Data Resep

III.1.12 Edit Resep



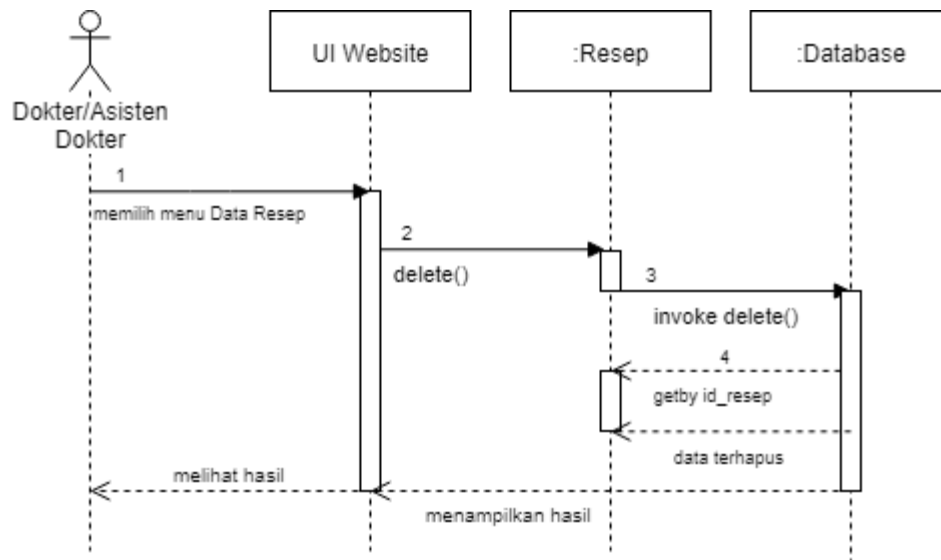
Gambar 19 Edit Resep

III.1.13 Membuat Resep



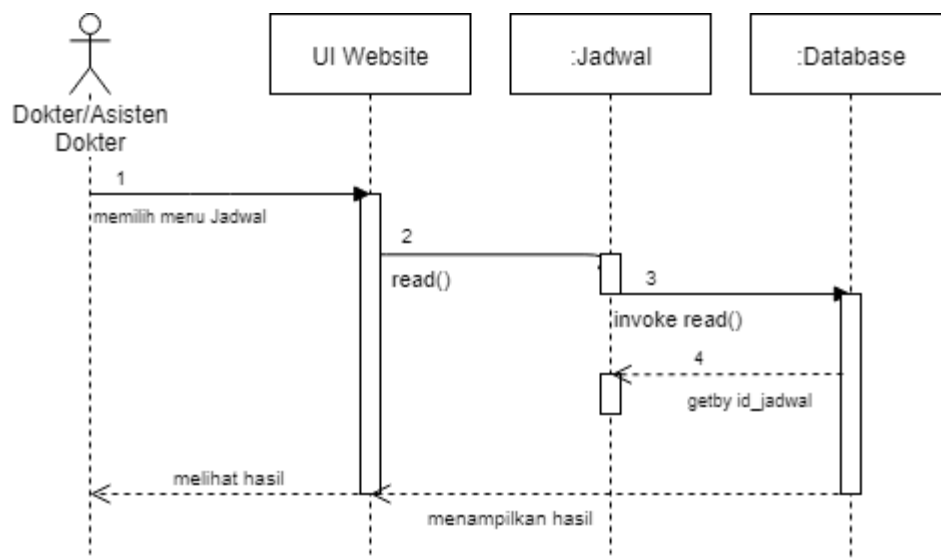
Gambar 20 Membuat Resep

III.1.14 Menghapus Resep



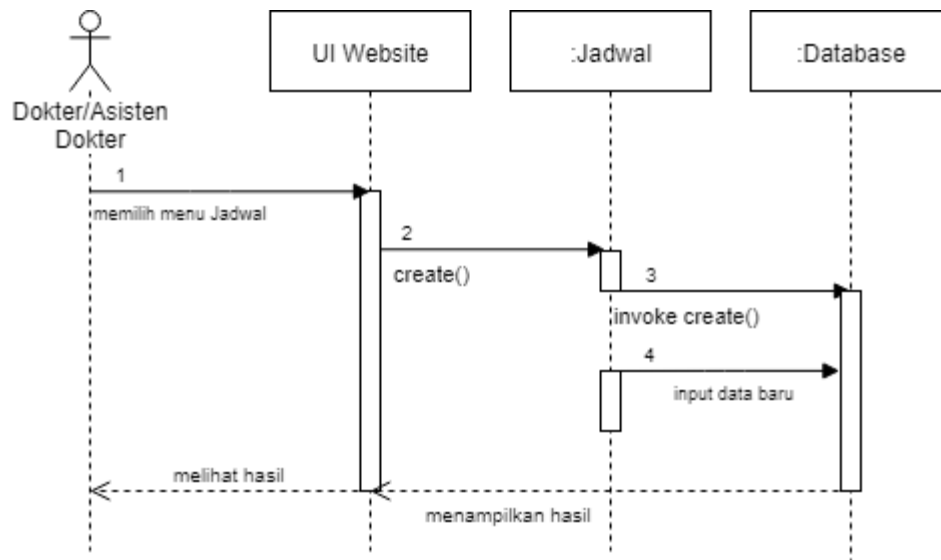
Gambar 21 Menghapus Resep

III.1.15 Melihat Jadwal



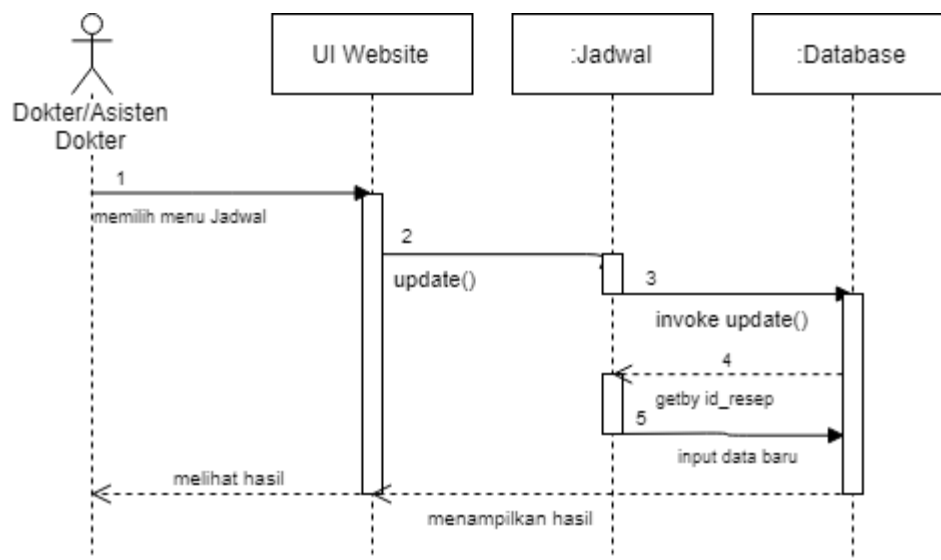
Gambar 22 Melihat Jadwal

III.1.16 Membuat Jadwal



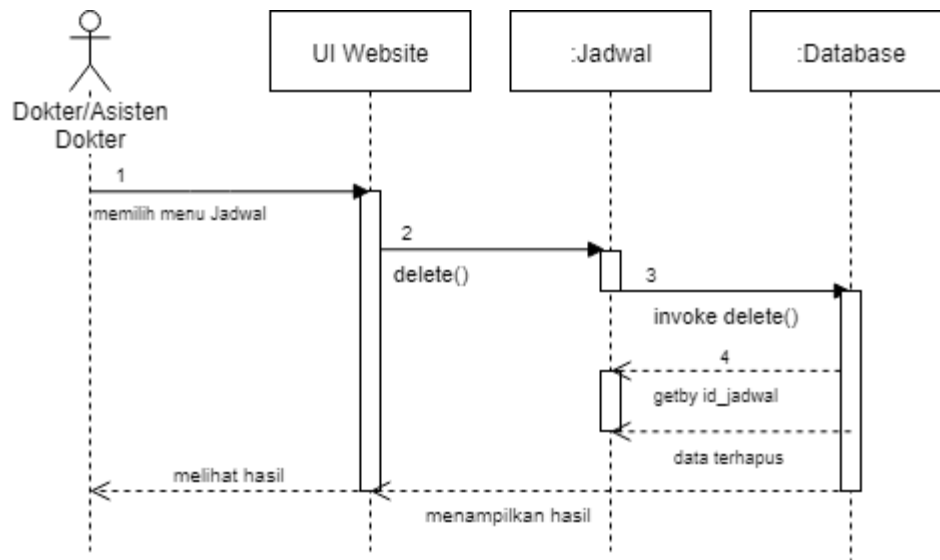
Gambar 23 Membuat Jadwal

III.1.17 Mengedit Jadwal



Gambar 24 Edit Jadwal

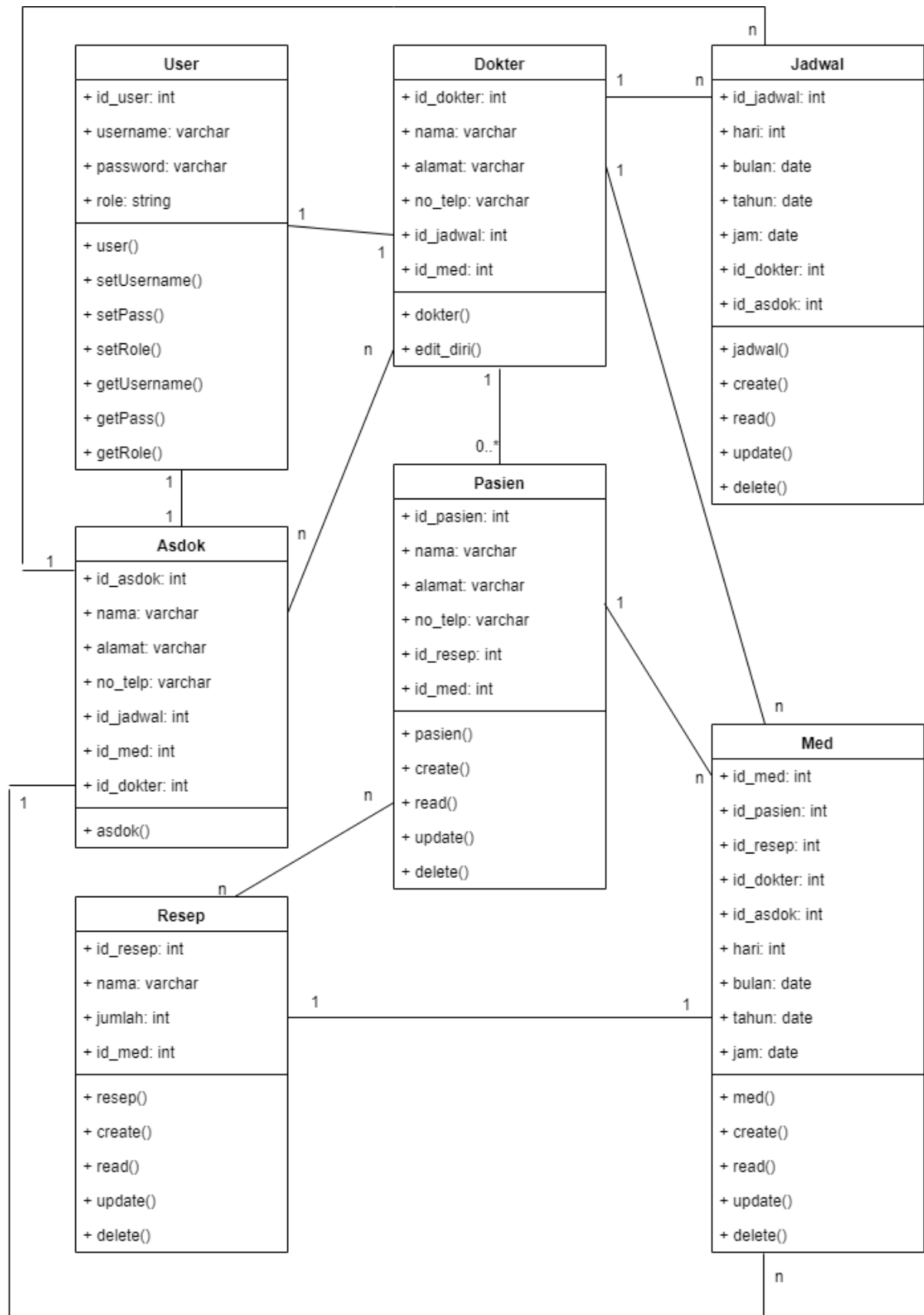
III.1.18 Menghapus Jadwal



Gambar 25 Hapus Jadwal

III.2 Diagram Kelas Keseluruhan

Bagian ini akan menjelaskan desain mengenai class digram secara lebih detail untuk menggambarkan kebutuhan sistem yang akan dibangun. Bentuk class diagram yang dirancang tersebut akan terlihat seperti gambar dibawah ini:



Gambar 26 Class Diagram

III.3 Kelas Analisis

Selanjutnya, setelah melakukan analisis mengenai class entitas maka akan dilengkapi dengan atribut-atribut yang mengikuti kelengkapan dari class diagram tersebut.

Nama Kelas	Daftar Tanggung Jawab	Daftar Atribut
User	1. Mengatur Entitas 2. Sebagai Table Manipulation Object	+ id_user + username + password + role
Dokter	1. Mengatur Entitas 2. Sebagai Table Manipulation Object	+ id_dokter + nama + alamat + no_telp + id_jadwal + id_med
AsDok	1. Mengatur Entitas 2. Sebagai Table Manipulation Object	+ id_asdok + nama + id_user_target + alamat + no_telp + description + id_jadwal + id_med + id_dokter
Pasien	1. Mengatur Entitas 2. Sebagai Table Manipulation Object	+ id_pasien + nama + alamat + no_telp + id_resep + id_med
Resep	1. Mengatur Entitas 2. Sebagai Table Manipulation Object	+ id_resep + nama + jumlah + id_med

Med	1. Mengatur Entitas 2. Sebagai Table Manipulation Object	+ id_med + id_pasien + id_resep + id_dokter + id_asdok + hari + jam + bulan + tahun
Jadwal	1. Mengatur Entitas 2. Sebagai Table Manipulation Object	+ id_jadwal + hari + bulan + tahun + jam + id_dokter + id_asdok

III.4 Deskripsi Arsitektur

Pembangunan aplikasi ini menggunakan konsep Object Oriented (OO) dengan menggunakan bahasa pemrograman yang bersifat *open source*, teknologi yang dibangun berbasis web.

III.5 Perancangan Detil Kelas

Berikut merupakan penjelasan detil kelas yang ada

III.5.1 Kelas User

Nama Kelas : User

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
setUsername	<i>Public</i>	<i>Menuliskan username</i>
setPassword	<i>Public</i>	<i>Menuliskan username</i>
setRole	<i>Public</i>	<i>Menyetel role</i>
getUsername	<i>Public</i>	<i>Mendapatkan data username</i>

getPassword	Mendapatkan data password	
getRole	Mendapatkan data peran	
<i>Nama Atribut</i>	Visibility (private, public)	Tipe
Id_user	Public	Int
Username	Public	Varchar
Password	Public	Varchar
Role	Public	Varchar

III.6.2 Kelas Dokter

Nama Kelas: Dokter

<i>Nama Operasi</i>	Visibility (private, public)	<i>Keterangan</i>
Edit_diri	Public	Mengedit data dokter
<i>Nama Atribut</i>	Visibility (private, public)	Tipe
Id_dokter	Public	Int
Nama	Public	Varchar
Alamat	Public	Varchar
No_telp	Public	Varchar
Id_jadwal	Public	Int
Id_med	Public	Int

III.6.3 Kelas AsDok

Nama Kelas: AsDok

<i>Nama Operasi</i>	Visibility (private, public)	<i>Keterangan</i>
<i>Nama Atribut</i>	Visibility (private, public)	Tipe
Id_asdok	Public	Int
Nama	Public	Varchar
Alamat	Public	Varchar
No_telp	Public	Varchar

<i>Id_jadwal</i>	<i>Public</i>	<i>Int</i>
<i>Id_med</i>	<i>Public</i>	<i>Int</i>
<i>Id_dokter</i>	<i>Public</i>	<i>Int</i>

III.6.4 Kelas Pasien

Nama Kelas: Pasien

<i>Nama Operasi</i>	<i>Visibility</i> (private, public)	<i>Keterangan</i>
Create	<i>Public</i>	<i>Membuat data pasien</i>
Read	<i>Public</i>	<i>Melihat data pasien</i>
Update	<i>Public</i>	<i>Memperbarui data pasien</i>
Delete	<i>Public</i>	<i>Menghapus data pasien</i>
<i>Nama Atribut</i>	<i>Visibility</i> (private, public)	<i>Tipe</i>
<i>Id_pasien</i>	<i>Public</i>	<i>Int</i>
<i>Nama</i>	<i>Public</i>	<i>Varchar</i>
<i>Alamat</i>	<i>Public</i>	<i>Varchar</i>
<i>No_telp</i>	<i>Public</i>	<i>Varchar</i>
<i>Id_resep</i>	<i>Public</i>	<i>Int</i>
<i>Id_med</i>	<i>Public</i>	<i>Int</i>

III.6.5 Kelas Resep

Nama Kelas: Resep

<i>Nama Operasi</i>	<i>Visibility</i> (private, public)	<i>Keterangan</i>
Create	<i>Public</i>	<i>Membuat data resep</i>
Read	<i>Public</i>	<i>Melihat data resep</i>
Update	<i>Public</i>	<i>Memperbarui data resep</i>
Delete	<i>Public</i>	<i>Menghapus data resep</i>
<i>Nama Atribut</i>	<i>Visibility</i> (private, public)	<i>Tipe</i>
<i>Id_resep</i>	<i>Public</i>	<i>Int</i>
<i>Nama</i>	<i>Public</i>	<i>Varchar</i>
<i>Jumlah</i>	<i>Public</i>	<i>Varchar</i>

<i>Id_med</i>	<i>Public</i>	<i>Int</i>
----------------------	---------------	------------

III.6.6 Kelas Medical record

Nama Kelas: Med

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
Create	<i>Public</i>	<i>Membuat data riwayat kesehatan</i>
Read	<i>Public</i>	<i>Melihat data riwayat kesehatan</i>
Update	<i>Public</i>	<i>Memperbarui data riwayat kesehatan</i>
Delete	<i>Public</i>	<i>Menghapus data riwayat kesehatan</i>
<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
<i>Id_med</i>	<i>Public</i>	<i>Int</i>
<i>Id_pasien</i>	<i>Public</i>	<i>Int</i>
<i>Id_resep</i>	<i>Public</i>	<i>Int</i>
<i>Id_dokter</i>	<i>Public</i>	<i>Int</i>
<i>Id_asdok</i>	<i>Public</i>	<i>Int</i>
<i>Hari</i>	<i>Public</i>	<i>Date</i>
<i>Jam</i>	<i>Public</i>	<i>Date</i>
<i>Bulan</i>	<i>Public</i>	<i>Date</i>
<i>Tahun</i>	<i>Public</i>	<i>Date</i>

III.6.7 Kelas Jadwal

Nama Kelas: Jadwal

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
Create	<i>Public</i>	<i>Membuat data jadwal</i>
Read	<i>Public</i>	<i>Melihat data jadwal</i>
Update	<i>Public</i>	<i>Memperbarui data jadwal</i>
Delete	<i>Public</i>	<i>Menghapus data jadwal</i>

<i>Nama Atribut</i>	<i>Visibility</i> <i>(private, public)</i>	<i>Tipe</i>
<i>Id_jadwal</i>	<i>Public</i>	<i>Int</i>
<i>Hari</i>	<i>Public</i>	<i>Date</i>
<i>Bulan</i>	<i>Public</i>	<i>Date</i>
<i>Tahun</i>	<i>Public</i>	<i>Date</i>
<i>Jam</i>	<i>Public</i>	<i>Date</i>
<i>Id_dokter</i>	<i>Public</i>	<i>Int</i>
<i>Id_asdok</i>	<i>Public</i>	<i>Int</i>

IV IMPLEMENTASI

Bagian ini berisi tentang penjelasan implementasi perangkat lunak.

IV.1 Lingkungan Implementasi

IV.1.1 Hardware yang digunakan untuk implementasi sistem:

Server:

- Ram 4Gb
- Harddisk 250Gb
- Monitor dengan seperangkat keyboard dan mouse

Client:

- Ram 2Gb
- Harddisk 160Gb
- Monitor dan seperangkat keyboard dan mouse

IV.1.2 Software yang digunakan untuk implementasi sistem:

Server:

- Apache
- MySQL

Client

- Sistem Operasi
- Web browser (Mozilla Firefox 6.0, Google Chrome 43)