



**PENERAPAN METODE ABC-VEN  
PADA SISTEM INFORMASI MANAJEMEN OBAT  
DI POLIKLINIK POLITEKNIK NEGERI JEMBER**

**SKRIPSI**

oleh

**Andika Widiarso  
122410101056**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS JEMBER  
2019**



**PENERAPAN METODE ABC-VEN  
PADA SISTEM INFORMASI MANAJEMEN OBAT  
DI POLIKLINIK POLITEKNIK NEGERI JEMBER**

**SKRIPSI**

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat untuk menyelesaikan Pendidikan Sarjana (S1) Program Studi Sistem Informasi dan mencapai gelar Sarjana Komputer

oleh

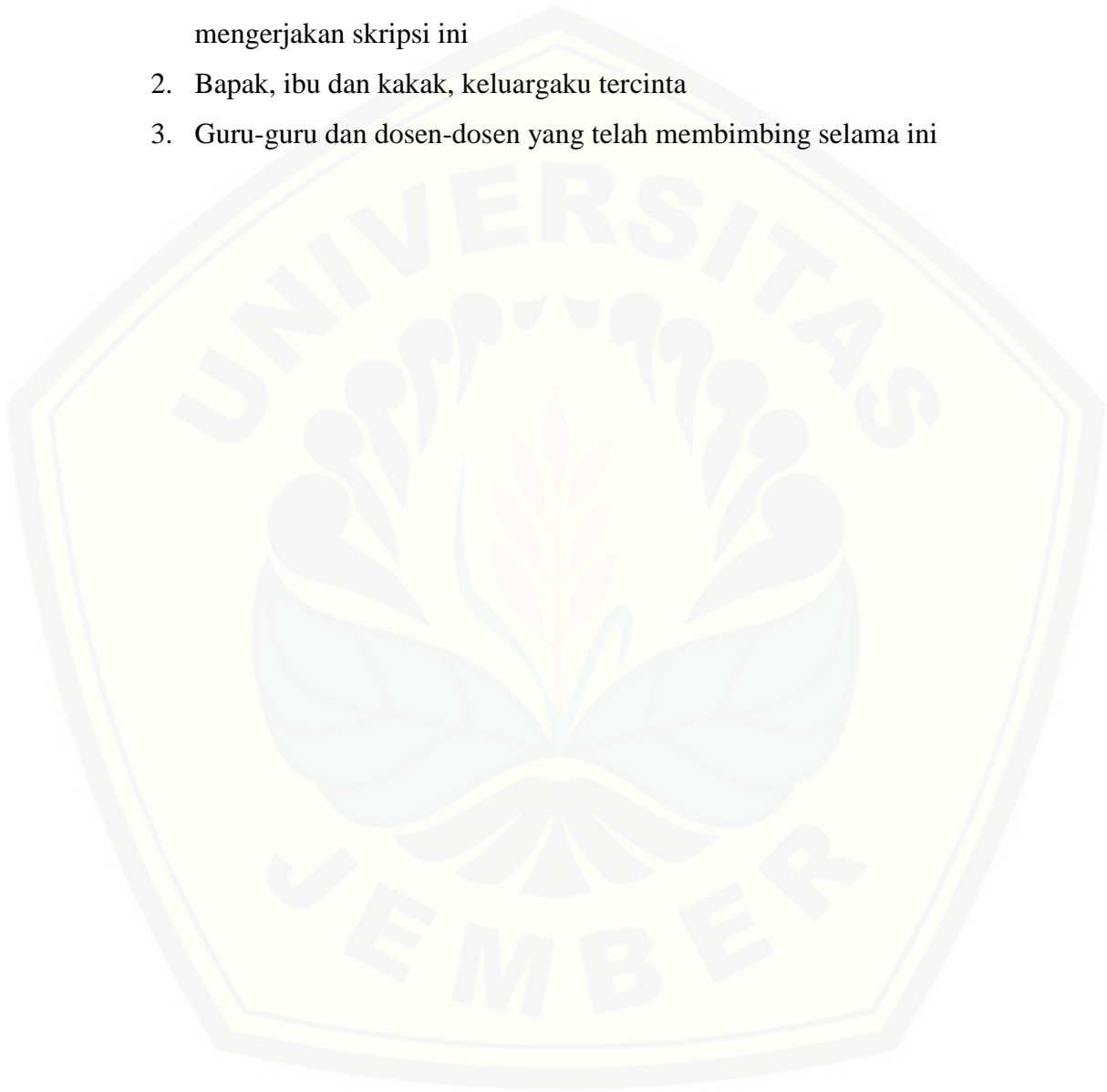
**Andika Widiarso  
122410101056**

**PROGRAM STUDI SISTEM INFORMASI  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS JEMBER  
2019**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT, yang telah memberikan kemudahan dan kelancaran dalam mengerjakan skripsi ini
2. Bapak, ibu dan kakak, keluargaku tercinta
3. Guru-guru dan dosen-dosen yang telah membimbing selama ini



**MOTTO**

Better late than not, better on time than late, better early than on time



**PERNYATAAN**

Saya yang bertanda tangan dibawah ini:

Nama : Andika Widiarso

NIM : 122410101056

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Penerapan Metode ABC-VEN Pada Sistem Informasi Manajemen Obat di Poliklinik Politeknik Negeri Jember”, adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, Juli 2019

Yang menyatakan,

Andika Widiarso

NIM 122410101056

**SKRIPSI**

**PENERAPAN METODE ABC-VEN  
PADA SISTEM INFORMASI MANAJEMEN OBAT  
DI POLIKLINIK POLITEKNIK NEGERI JEMBER**

Oleh

Andika Widiarso

122410101056

**PEMBIMBING**

Dosen Pembimbing Utama : Anang Andrianto ST.,MT.

Dosen pembimbing Pendamping : Yudha Alif Auliya S.Kom., M.Kom.

**PENGESAHAN PEMBIMBING**

Skripsi berjudul "Penerapan Metode ABC-VEN Pada Sistem Informasi Manajemen Obat di Poliklinik Politeknik Negeri Jember", telah diuji dan disahkan pada,  
hari, tanggal : Jum'at, 26 Juli 2019  
tempat : Fakultas Ilmu Komputer Universitas Jember

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Anang Andrianto ST., MT.  
NIP 196906151997021002

Yudha Alif Auliya S.Kom., M.Kom  
NRP 760018031

**PENGESAHAN PENGUJI**

Skripsi berjudul "Penerapan Metode ABC-VEN Pada Sistem Informasi Manajemen Obat di Poliklinik Politeknik Negeri Jember", telah diuji dan disahkan pada,  
hari, tanggal : Jum'at, 26 Juli 2019  
tempat : Fakultas Ilmu Komputer Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

Yanuar Nurdiansyah ST., M.Cs.  
NIP 198201012010124004

Tio Dharmawan S.Kom., M.Kom  
NRP 760016851

Mengesahkan

a. n Dekan

Wakil Dekan I Fakultas Ilmu Komputer,

Drs. Antonius Cahya P, M.App.Sc, Ph.D.

NIP 196909281993021001



## RINGKASAN

**Penerapan Metode ABC-VEN Pada Sistem Informasi Manajemen Obat di Poliklinik Politeknik Negeri Jember;** Andika Widiarso, 122410101056; 2019; 97 halaman; Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Jember.

Kegiatan yang berlangsung pada kamar obat Poliklinik Politeknik Negeri Jember masih dijalankan secara manual dan hanya bergantung pada kontrol internal manusia (asisten apoteker). Kamar obat harus memastikan obat yang dikeluarkan tidak kadaluarsa sehingga perlu ada pengingat untuk stok dan *expired date* (tanggal kadaluarsa), memperhatikan jumlah barang yang masuk sebagai persediaan obat sehingga tidak terjadi persediaan obat berlebih dan meminimalisir kadaluarsanya obat yang tersedia. Untuk meminimalisir terjadinya obat yang kadaluarsa, diperlukan model manajemen yang bisa membantu pemilihan pembelian obat.

Penelitian ini bertujuan untuk menemukan model sistem manajemen kamar obat pada Poliklinik Politeknik Negeri Jember yang optimal.

Jenis dan metode penelitian ini adalah penelitian pengembangan yaitu mengembangkan dan menerapkan metode ABC (Always, Better, Control) atau yang biasa disebut hukum Pareto. Hukum Pareto adalah suatu metode yang dapat digunakan dalam menganalisis pola konsumsi obat, serta dilengkapi dengan metode VEN (Vital, Esensial, Non-Esensial) yang merupakan sistem untuk menentukan pemilihan, pengadaan, dan penggunaan perbekalan farmasi. Dengan demikian, metode gabungan dari analisis ABC dan analisis VEN dapat digunakan untuk mengevaluasi pola pengadaan dengan dasar prioritas.

Hasil penelitian ini adalah terwujudnya sistem informasi manajemen kamar obat dengan metode ABC-VEN di Poliklinik Politeknik Negeri Jember.

Kesimpulan penelitian ini yaitu. *Pertama;* metode ABC dilakukan dengan menghitung nilai pakai dan nilai investasi dari suatu obat pada periode tertentu dengan menghitung variabel yang didapatkan yang berupa jumlah pemakaian dan harga obat per item dari data-data transaksi yang lalu digunakan untuk menghitung

nilai kritis yang dijadikan acuan untuk menentukan kelas dari suatu obat. Pengelompokkan obat dengan menggunakan metode VEN dilakukan oleh apoteker. Metode ABC-VEN baru bisa dilakukan bila pengelompokkan obat dengan menggunakan metode ABC dan VEN telah selesai dilakukan; *ke dua*, Pengembangan sistem metode ABC-VEN dilakukan dengan menggunakan model *waterfall*. Pengembangan sistem model *waterfall* meliputi tahap analisis, desain, implementasi dan pengujian.



## PRAKATA

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul Penerapan Metode ABC-VEN Pada Sistem Informasi Manajemen Obat di Poliklinik Politeknik Negeri Jember. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Prof. Dr Saiful Bukhori, ST., M.Kom., selaku Dekan Fakultas Ilmu Komputer Universitas Jember;
2. Anang Andrianto, S.T., M.T., selaku Dosen Pembimbing Utama dan Yudha Alif Auliya S.Kom., M.Kom., selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
3. Seluruh Bapak dan Ibu dosen beserta staf karyawan di program studi sistem informasi;
4. Seluruh keluarga besar tersayang;
5. Kepala Klinik yang meluangkan waktunya dalam membantu penyelesaian skripsi ini
6. Keluarga besar FORMATION angkatan 2012 yang telah menjadi keluarga selama menempuh pendidikan S1

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, oleh sebab itu penulis mengharapkan adanya masukan yang bersifat membangun dari semua pihak. Penulis berharap skripsi ini dapat bermanfaat bagi semua pihak.

Jember, 26 Juli 2019

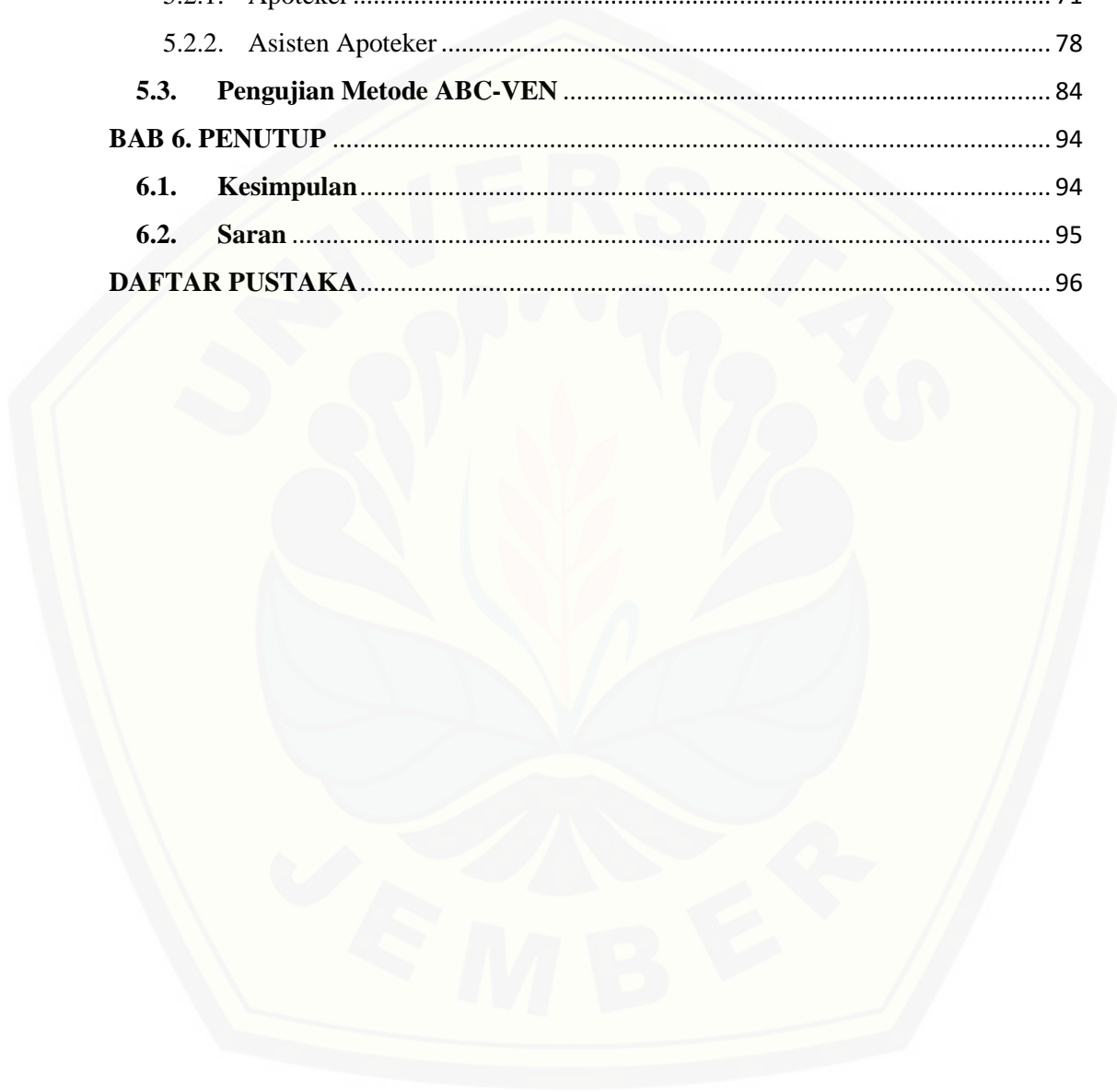
Penulis

**DAFTAR ISI**

<b>PERSEMBAHAN</b> .....	ii
<b>MOTTO</b> .....	iii
<b>PERNYATAAN</b> .....	iv
<b>PEMBIMBING</b> .....	v
<b>PENGESAHAN PEMBIMBING</b> .....	vi
<b>PENGESAHAN PENGUJI</b> .....	vii
<b>RINGKASAN</b> .....	viii
<b>PRAKATA</b> .....	x
<b>DAFTAR ISI</b> .....	xi
<b>DAFTAR GAMBAR</b> .....	xiv
<b>DAFTAR TABEL</b> .....	xvi
<b>DAFTAR LAMPIRAN</b> .....	xvii
<b>BAB 1. PENDAHULUAN</b> .....	1
<b>1.1. Latar Belakang</b> .....	1
<b>1.2. Rumusan Masalah</b> .....	3
<b>1.3. Tujuan</b> .....	3
<b>1.4. Batasan Masalah</b> .....	4
<b>1.5. Sistematika Penulisan</b> .....	4
<b>BAB 2. TINJAUAN PUSTAKA</b> .....	6
<b>2.1. Penelitian Terdahulu</b> .....	6
<b>2.2. Kamar Obat</b> .....	7
<b>2.3. Persediaan</b> .....	8
<b>2.4. Manajemen Persediaan</b> .....	9
<b>2.5. Metode ABC</b> .....	10
<b>2.6. Metode VEN</b> .....	15
<b>2.7. Metode ABC-VEN</b> .....	16
<b>BAB 3. METODE PENELITIAN</b> .....	18
<b>3.1. Jenis Penelitian</b> .....	18
<b>3.2. Tempat dan Waktu Penelitian</b> .....	18

<b>3.3. Tahapan Penelitian</b> .....	18
3.3.1. Analisis dan Definisi Persyaratan.....	19
3.3.2. Perancangan Sistem dan Perangkat Lunak.....	20
3.3.3. Implementasi dan Pengujian Unit .....	22
3.3.4. Pengujian Sistem ( <i>Integration and System Testing</i> ).....	22
3.3.5. Pemeliharaan ( <i>Operation and Maintenance</i> ) .....	23
<b>BAB 4. PENGEMBANGAN SISTEM</b> .....	24
<b>4.1. Analisis Kebutuhan</b> .....	24
4.1.1. Tahap Pengumpulan Data.....	24
4.1.2. <i>Statement of Purpose (SOP)</i> .....	25
4.1.3. Kebutuhan Fungsional.....	26
4.1.4. Kebutuhan Non-fungsional .....	26
<b>4.2. Desain Sistem</b> .....	27
4.2.1. <i>Input-Output Diagram</i> .....	27
4.2.2. <i>Business Process</i> .....	28
4.2.3. <i>Use Case Diagram</i> .....	28
4.2.4. <i>Use Case Scenario</i> .....	32
4.2.5. <i>Activity Diagram</i> .....	35
4.2.6. <i>Sequence Diagram</i> .....	37
4.2.7. <i>Class Diagram</i> .....	41
4.2.8. <i>Entity Relationship Diagram</i> .....	42
<b>4.3. Penulisan Kode Program</b> .....	42
4.3.1. Kode Program Rekomendasi.....	42
4.3.2. Kode Program Pengguna.....	48
4.3.3. Kode Program Data Obat .....	48
4.3.4. Kode Program Stok Obat.....	49
4.3.5. Kode Program Data Transaksi.....	49
4.3.6. Kode Program Pembuangan .....	49
<b>4.4. Pengujian Sistem</b> .....	49
4.4.1. <i>White Box Testing</i> .....	50
4.4.2. <i>Black Box Testing</i> .....	63
<b>BAB 5. HASIL DAN PEMBAHASAN</b> .....	65
<b>5.1. Penerapan Metode ABC-VEN</b> .....	65

5.1.1. Metode ABC.....	65
5.1.2. Metode VEN.....	68
5.1.3. Metode ABC-VEN.....	69
<b>5.2. Hasil Pengembangan Sistem .....</b>	<b>70</b>
5.2.1. Apoteker .....	71
5.2.2. Asisten Apoteker .....	78
<b>5.3. Pengujian Metode ABC-VEN .....</b>	<b>84</b>
<b>BAB 6. PENUTUP .....</b>	<b>94</b>
<b>6.1. Kesimpulan.....</b>	<b>94</b>
<b>6.2. Saran .....</b>	<b>95</b>
<b>DAFTAR PUSTAKA.....</b>	<b>96</b>



**DAFTAR GAMBAR**

Gambar 2. 1. Jenis Manajemen Persediaan (Ma'arif & Tanjung, 2003).....	10
Gambar 2. 2. Grafik Kurva Analisis ABC (Kusnadi, 2009).....	12
Gambar 2. 3. Bagan alur analisis ABC-VEN (Singh, Singh, & Singh, 2015).....	17
Gambar 3. 1 Alur Model <i>Waterfall</i> .....	19
Gambar 4. 1 <i>Input-Output Diagram</i> Sistem Informasi Kamar Obat.....	27
Gambar 4. 2 <i>Business Process</i> Sistem Informasi Kamar Obat.....	28
Gambar 4. 3 <i>Use case Diagram</i> .....	29
Gambar 4. 4 <i>Activity Diagram</i> Rekomendasi.....	37
Gambar 4. 5 <i>Sequence Diagram</i> Rekomendasi.....	40
Gambar 4. 6 <i>Class Diagram</i> .....	41
Gambar 4. 7 <i>Entity Relationship Diagram (ERD)</i> .....	42
Gambar 4. 8 Diagram Alir Fungsi ABC.....	51
Gambar 4. 9 Diagram Alir Fungsi dataABC.....	54
Gambar 4. 10 Diagram Alir Fungsi VENV.....	57
Gambar 4. 11 Diagram Alir Fungsi VENE.....	58
Gambar 4. 12 Diagram Alir Fungsi VENN.....	58
Gambar 4. 13 Diagram Alir Fungsi ABCVEN.....	59
Gambar 4. 14 Diagram Alir Fungsi tahun.....	62
Gambar 5. 1. Bagan alur analisis ABC.....	65
Gambar 5. 2 Halaman ABC.....	68
Gambar 5. 3 Halaman VEN.....	69
Gambar 5. 4 Halaman ABC-VEN.....	70
Gambar 5. 5 Halaman Login.....	71
Gambar 5. 6 Halaman Beranda.....	72
Gambar 5. 7 Halaman Beranda Dengan Modal.....	72
Gambar 5. 8 Halaman Master Obat.....	73
Gambar 5. 9 Halaman Detail Obat.....	73
Gambar 5. 10 Halaman Form Penambahan Obat.....	74
Gambar 5. 11 Halaman Form Perubahan Obat.....	74

Gambar 5. 12 Modal Penghapusan Obat.....	75
Gambar 5. 13 Halaman Daftar Transaksi.....	76
Gambar 5. 14 Halaman Daftar Stok.....	76
Gambar 5. 15 Halaman Form Stok Awal.....	77
Gambar 5. 16 Halaman Pembuangan.....	77
Gambar 5. 17 Modal Form Pembuangan.....	78
Gambar 5. 18 Fitur Login.....	79
Gambar 5. 19 Halaman Beranda.....	79
Gambar 5. 20 Halaman Daftar Obat.....	80
Gambar 5. 21 Halaman Detail Obat.....	80
Gambar 5. 22 Halaman Daftar Transaksi.....	81
Gambar 5. 23 Halaman Form Transaksi.....	82
Gambar 5. 24 Halaman Form Edit Transaksi.....	82
Gambar 5. 25 Modal Penghapusan Transaksi.....	83
Gambar 5. 26 Halaman Pembuangan.....	83
Gambar 5. 27 Modal Form Pembuangan.....	84



**DAFTAR TABEL**

Tabel 2. 1 Tabulasi Silang Analisis ABC dan Analisis VEN .....	17
Tabel 4. 1 Keluar-masuk obat Bulan Maret 2019 .....	24
Tabel 4. 2 Definisi Aktor .....	29
Tabel 4. 3 Deskripsi <i>Use Case</i> .....	30
Tabel 4. 4 Skenario Melihat Rekomendasi .....	33
Tabel 4. 5 Kode Program pada Class Controller Rekomendasi.php.....	42
Tabel 4. 6 Kode Program pada Class Model M_Rekom.php .....	45
Tabel 4. 7 <i>Test case</i> fungsi ABC() .....	51
Tabel 4. 8 <i>Test case</i> fungsi dataABC() .....	55
Tabel 4. 9 <i>Test case</i> fungsi VENV() .....	57
Tabel 4. 10 <i>Test case</i> fungsi VENE().....	58
Tabel 4. 11 <i>Test case</i> fungsi VEN() .....	59
Tabel 4. 12 <i>Test case</i> fungsi ABCVEN().....	60
Tabel 4. 13 <i>Test case</i> fungsi tahun() .....	63
Tabel 4. 14 Hasil Pengujian Black Box .....	63
Tabel 5. 1 Kode perhitungan nilai ABC .....	66
Tabel 5. 2 Kode pemanggilan fungsi dataABC().....	67
Tabel 5. 3 Kode penentuan kelas ABC nilai kritis.....	67
Tabel 5. 4 Kode perhitungan nilai ABC-VEN .....	69
Tabel 5. 5 Keluar masuk obat bulan Maret 2019 .....	84
Tabel 5. 6 Hasil perhitungan nilai pakai .....	86
Tabel 5. 7 Hasil perhitungan nilai investasi .....	88
Tabel 5. 8 Hasil pengelompokkan nilai indeks kritis .....	90
Tabel 5. 9 Hasil klasifikasi menggunakan metode VEN .....	91
Tabel 5. 10 Hasil Pengelompokkan ABC-VEN.....	92

**DAFTAR LAMPIRAN**

<b>LAMPIRAN</b> .....	98
<b>Lampiran A. Skenario</b> .....	98
A.1. Skenario Login .....	98
A.2. Skenario Melihat Obat.....	99
A.3. Skenario Menambah Obat .....	100
A.4. Skenario Mengubah Obat .....	101
A.5. Skenario Menghapus Obat .....	102
A.6. Skenario Melihat Data Stok.....	102
A.7. Skenario Menambah Data Stok .....	103
A.8. Skenario Melihat Data Transaksi .....	104
A.9. Skenario Menambah Data Transaksi.....	105
A.10. Skenario Mengubah Data Transaksi.....	106
A.11. Skenario Menghapus Data Transaksi .....	107
A.12. Skenario Melihat Data Pembuangan Obat .....	108
A.13. Skenario Menambah Data Pembuangan Obat .....	108
A.14. Skenario Logout .....	109
<b>Lampiran B. Activity Diagram</b> .....	98
B.1. Activity Diagram Login .....	110
B.2. Activity Diagram Melihat Obat.....	111
B.3. Activity Diagram Menambah Obat .....	112
B.4. Activity Diagram Mengubah Obat .....	113
B.5. Activity Diagram Menghapus Obat.....	114
B.6. Activity Diagram Melihat Data Stok.....	115
B.7. Activity Diagram Menambah Data Stok .....	116
B.8. Activity Diagram Melihat Data Transaksi.....	117
B.9. Activity Diagram Menambah Data Transaksi .....	118
B.10. Activity Diagram Mengubah Data Transaksi.....	119
B.11. Activity Diagram Menghapus Data Transaksi .....	120
B.12. Activity Diagram Melihat Data Pembuangan Obat.....	121

B.13.	Activity Diagram Menambah Data Pembuangan Obat .....	122
B.14.	Activity Diagram Logout .....	123
<b>Lampiran C. Sequence Diagram .....</b>		<b>123</b>
C.1.	Sequence Diagram Login .....	123
C.2.	Sequence Diagram Melihat Obat.....	124
C.3.	Sequence Diagram Menambah Obat.....	124
C.4.	Sequence Diagram Mengubah Obat.....	125
C.5.	Sequence Diagram Menghapus Obat .....	125
C.6.	Sequence Diagram Melihat Data Stok .....	126
C.7.	Sequence Diagram Menambah Data Stok.....	126
C.8.	Sequence Diagram Melihat Data Transaksi .....	127
C.9.	Sequence Diagram Menambah Data Transaksi.....	127
C.10.	Sequence Diagram Mengubah Data Transaksi.....	128
C.11.	Sequence Diagram Menghapus Data Transaksi .....	128
C.12.	Sequence Diagram Melihat Data Pembuangan Obat .....	129
C.13.	Sequence Diagram Menambah Data Pembuangan Obat .....	129
C.14.	Sequence Diagram Logout .....	130
<b>Lampiran D. Kode Program .....</b>		<b>130</b>
D.1.	Kode Program Pengguna.....	130
D.2.	Kode Program Data Obat .....	132
D.3.	Kode Data Stok Obat.....	136
D.4.	Kode Program Transaksi .....	139
D.5.	Kode Program Pembuangan.....	145
<b>Lampiran E. Pengujian Black Box.....</b>		<b>147</b>

## BAB 1. PENDAHULUAN

Bab ini merupakan langkah awal dari penulisan tugas akhir. Bab ini berisi latar belakang, rumusan masalah, batasan masalah, metodologi penelitian, dan sistematika penulisan.

### 1.1. Latar Belakang

Pelayanan kesehatan adalah suatu alat atau tempat yang digunakan untuk menyelenggarakan upaya pelayanan kesehatan, baik promotif, preventatif, kuratif maupun rehabilitatif yang dilakukan oleh pemerintah, pemerintah daerah, dan masyarakat (Undang-Undang Republik Indonesia Nomor 36 Tahun 2009, 2009). Klinik merupakan fasilitas pelayanan kesehatan yang menyelenggarakan pelayanan kesehatan perorangan, yang menyediakan pelayanan medis dasar dan spesialisik, diselenggarakan oleh lebih dari satu jenis tenaga kesehatan dan dipimpin oleh seorang tenaga medis. (Peraturan Menteri Kesehatan Republik Indonesia Nomor 9, 2014)

Poliklinik Politeknik Negeri Jember merupakan salah satu unit layanan yang ada di Politeknik Negeri Jember. Kegiatan yang dijalankan yaitu meliputi; layanan rawat jalan, tes kesehatan, ataupun layanan konsultasi. Prinsip layanan mengutamakan kepuasan pelanggan yaitu memberikan kecepatan layanan dengan cara menggunakan standar operasional prosedur (SOP). Poliklinik yang secara khusus memiliki tugas untuk mengelola berbagai kegiatan kesehatan yang meliputi pemberian informasi kesehatan, penanganan kesehatan dan meningkatkan kualitas pelayanan kesehatan bagi seluruh keluarga besar pegawai Politeknik Negeri Jember khususnya. Di samping itu pelanggan Poliklinik Politeknik Negeri Jember juga berasal dari masyarakat di sekitar kampus Polije dan banyak juga yang lebih jauh dari itu. Dilihat dari jenis pekerjaan meliputi; pegawai negeri dan mahasiswa. Pelayanan yang tersedia di poliklinik terdiri dari poli umum, poli gigi, pelayanan gawat darurat, pelayanan KIA dan pelayanan KB serta pelayanan obat (kefarmasian).

Pelayanan kefarmasian meliputi pengelolaan sumber daya (SDM, sarana prasarana, sediaan farmasi dan perbekalan kesehatan serta administrasi) dan pelayanan farmasi klinik (penerimaan resep, peracikan obat, penyerahan obat, informasi obat dan pencatatan/penyimpanan resep) dengan memanfaatkan tenaga, dana, prasarana, sarana dan metode tatalaksana yang sesuai dalam upaya mencapai tujuan yang ditetapkan (Peraturan Menteri Kesehatan Republik Indonesia Nomor 9, 2014). Pelayanan kefarmasian di Poliklinik Politeknik Negeri Jember disebut dengan kamar obat. Kamar obat merupakan bagian dari poliklinik yang melakukan pelayanan obat. Kamar obat mengeluarkan obat berdasar resep yang diterima dari poli umum, poli gigi, pelayanan gawat darurat dan pelayanan KIA dan pelayanan KB.

Selama ini, proses yang berlangsung di kamar obat masih dijalankan secara manual dan hanya bergantung pada kontrol internal manusia (asisten apoteker). Kamar obat harus memastikan obat yang dikeluarkan tidak kadaluarsa sehingga perlu ada pengingat untuk stok dan *expired date* (tanggal kadaluarsa), memperhatikan jumlah barang yang masuk sebagai persediaan obat sehingga tidak terjadi persediaan obat berlebih dan meminimalisir kadaluarsanya obat yang tersedia.

Permasalahan yang terjadi di kamar obat adalah pencatatan obat secara manual. Setelah petugas memberikan obat pada pasien, petugas akan merekap obat yang keluar pada hari itu secara manual. Pada proses pencatatan manual inilah sering terjadi kesalahan seperti masih ada obat pada stok yang memiliki tanggal kadaluarsa yang lebih awal dari obat yang telah dikeluarkan, sedangkan obat yang jarang dikeluarkan tidak terekap sehingga bisa ditemukan obat yang mendekati tanggal kadaluarsa.

Berdasarkan permasalahan di atas, untuk memastikan agar obat yang keluar merupakan obat yang memiliki tanggal kadaluarsa yang lebih awal, maka perlu dibangun sistem yang bisa memilih obat mana yang keluar terlebih dahulu. Selain itu, untuk mewujudkan pelayanan yang lebih baik, maka kamar obat perlu menggunakan suatu model manajemen untuk mengendalikan ketersediaan obat. Salah satu model manajemen yang bisa digunakan adalah analisis ABC-VEN.

Analisis ABC (Always, Better, Control) atau yang biasa disebut hukum Pareto adalah suatu analisis yang dapat digunakan dalam menganalisis pola konsumsi obat, sementara analisis VEN (Vital, Esensial, Non-Esensial) adalah suatu sistem untuk menentukan pemilihan, pengadaan, dan penggunaan perbekalan farmasi. Analisis VEN dapat membantu dalam mengontrol stok obat-obatan yang perlu kontrol ketat untuk menghindari stock-out dan memperbesar manfaat dari dana yang tersedia (Devnani, Gupta, & Nigah, 2010). Dengan demikian, analisis gabungan dari analisis ABC dan analisis VEN dapat digunakan untuk mengevaluasi pola pengadaan dengan dasar prioritas. (Quick, Dias, & Rankin, 2012)

Berdasarkan uraian di atas, untuk mewujudkan pelayanan dan manajemen farmasi yang lebih baik, maka perlu dibangun sistem yang mengendalikan dan mencatat persediaan obat di kamar obat. Sistem ini dapat membantu mencatat transaksi yang terjadi dengan tepat sehingga dapat mengetahui keadaan persediaan obat dengan pasti. Selain itu, sistem ini juga dapat membantu untuk menyusun prioritas obat yang seharusnya dibeli selanjutnya dengan menggunakan metode ABC-VEN. Pembangunan sistem ini diharapkan dapat memberikan informasi secara mudah, cepat dan dapat dipercaya kebenarannya.

## **1.2. Rumusan Masalah**

Berdasarkan masalah yang telah diuraikan diatas, maka rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimanakah mengimplementasi metode ABC-VEN pada manajemen persediaan obat?
2. Bagaimanakah mengembangkan sistem informasi manajemen dengan menggunakan metode ABC-VEN?

## **1.3. Tujuan**

Tujuan dari penelitian yang dilakukan dalam skripsi ini adalah sebagai berikut:

1. Menemukan model sistem manajemen kamar obat pada Poliklinik Politeknik Negeri Jember yang optimal.
2. Merancang dan membuat sistem untuk mempermudah manajemen persediaan obat di kamar obat Poliklinik Politeknik Negeri Jember.

#### **1.4. Batasan Masalah**

Pembahasan yang dilakukan dalam skripsi ini memiliki batasan masalah sebagai berikut:

1. Objek yang diteliti adalah data stok obat pada kamar obat Poliklinik Politeknik Negeri Jember.
2. Metode yang digunakan adalah metode ABC-VEN untuk membuat rekomendasi keketatan pengontrolan obat.
3. Aplikasi yang dibangun adalah sistem informasi berbasis *web*.
4. Sistem informasi yang dibangun digunakan untuk menguji efektifitas metode ABC-VEN

#### **1.5. Sistematika Penulisan**

Sistematika penulisan dan urutan skripsi ini disusun sebagai berikut:

1. **Pendahuluan**  
Bab ini menjelaskan tentang latar belakang, perumusan masalah, tujuan dan manfaat, ruang lingkup studi dan sistematika penulisan.
2. **Tinjauan Pustaka**  
Bab ini menjelaskan tentang materi, informasi, tinjauan pustaka, dan studi terdahulu yang menjadi kerangka pemikiran dalam penelitian.
3. **Metodologi Penelitian**  
Bab ini menjelaskan tentang metode penelitian yang digunakan dalam penelitian.
4. **Pengembangan Sistem**  
Bab ini menjelaskan tentang gambaran umum pengembangan sistem, pengujian kinerja, pemeliharaan operasi sistem informasi.

5. Hasil dan Pembahasan

Bab ini menjelaskan tentang hasil dan pembahasan dari penelitian yang dilakukan.

6. Penutup

Bab ini berisi tentang kesimpulan dari penelitian dan saran untuk penelitian selanjutnya.





## BAB 2. TINJAUAN PUSTAKA

Pada bab ini dipaparkan teori-teori dan pustaka yang akan dipakai dalam penelitian. Teori-teori ini berupa teori dari buku literatur dan jurnal. Berikut merupakan teori-teori yang dibahas dalam penelitian.

### 2.1. Penelitian Terdahulu

Penelitian sebelumnya yang pernah dilakukan dengan mengimplementasikan metode ABC yaitu dengan judul “Sistem Pendukung Keputusan Pengadaan Obat Menggunakan Model Pareto ABC dan Optimasi Kualitatif” yang dilakukan oleh Ancelmatini Nurwulandari (2012). Dalam penelitian tersebut, metode analisis pareto ABC diterapkan dalam sistem pengambilan keputusan pengadaan obat. Sistem yang dibuat mampu memberikan informasi tentang prioritas item yang digunakan pada apotek berdasarkan jumlah pemakaian (nilai pakai), jumlah investasi (nilai investasi) dan skor total nilai indeks kritis. Dengan dibangunnya sistem tersebut, pihak apotek dapat melakukan pengendalian pengadaan obat yang tepat dengan biaya yang optimal sehingga mengurangi terjadinya penumpukan obat yang bisa mengakibatkan kadaluarsanya obat.

Berdasarkan pada uraian di atas maka penelitian yang akan dilakukan yaitu dengan judul “Penerapan Metode ABC-VEN pada sistem informasi manajemen obat di Klinik Politeknik Negeri Jember” perlu dilakukan dengan beberapa pertimbangan tersebut;

1. Penelitian ini berbeda dengan dengan penelitian sebelumnya yang telah diuraian di atas sehingga rencana penelitian ini mempunyai kebaruan
2. Penelitian ini belum pernah diterapkan di Klinik Politeknik Negeri Jember
3. Aplikasi metode ABC-VEN memungkinkan dibuat dan selanjutnya diterapkan dalam implementasi sistem informasi manajemen obat

## 2.2. Kamar Obat

Poliklinik Politeknik Negeri Jember merupakan unit layanan kesehatan Politeknik Negeri Jember. Kegiatan pelayanan kesehatan yang dilakukan oleh unit tersebut tentunya tidak bisa jauh dari obat.

Obat adalah suatu bahan yang diolah berdasarkan proses kimiawi ataupun non kimiawi yang terkait dengan fenomena fisiologi dan patologi yang berfungsi untuk tindak lanjut proses diagnosis, pencegahan, penyembuhan, dan peningkatan kesehatan manusia atau hewan. Obat juga dapat diartikan suatu zat atau bahan yang diasumsikan dapat menyembuhkan atau mengurangi rasa sakit. (Anshari, 2009)

Pengelolaan obat pada poliklinik dilakukan oleh kamar obat. Kamar obat bertanggung jawab atas transaksi yang berkaitan dengan obat. Petugas kamar obat akan membeli obat sesuai dengan kebutuhan secara rutin pada awal bulan untuk setiap bulannya. Obat akan dijual kepada pelanggan atau pasien berdasarkan resep obat yang diberikan oleh dokter.

Kamar obat juga memiliki tugas untuk menyimpan obat. Penyimpanan obat dilakukan dengan mengikuti aturan yang telah ada. Pengaturan penyimpanan obat dan persediaan adalah sebagai berikut :

- (a) Simpan obat-obatan yang mempunyai kesamaan secara bersamaan di atas rak. 'Kesamaan' berarti dalam cara pemberian obat (luar, oral, suntikan) dan bentuk ramuannya (obat kering atau cair).
- (b) Simpan obat sesuai tanggal kadaluwarsa dengan menggunakan prosedur FEFO (*First Expiry First Out*). Obat dengan tanggal kadaluwarsa yang lebih dekat ditempatkan di depan obat yang memiliki tanggal kadaluwarsa yang lebih jauh. Bila obat mempunyai tanggal kadaluwarsa sama, tempatkan obat yang baru diterima dibelakang obat yang sudah ada.
- (c) Simpan obat tanpa tanggal kadaluwarsa dengan menggunakan prosedur FIFO (*First In First Out*). Barang yang baru diterima ditempatkan dibelakang barang yang sudah ada.
- (d) Buang obat yang kadaluwarsa dan rusak dengan dibuatkan catatan pemusnahan obat, termasuk tanggal, jam, saksi dan cara pemusnahan.

Kamar obat berfungsi sebagai pengontrol aliran obat pada poliklinik, memiliki fungsi untuk menjadi perantara antara :

- a. Permintaan dan penawaran
- b. Permintaan pelanggan dan barang jadi
- c. Persyaratan untuk suatu operasi dan output dari operasi sebelumnya

Pada kamar obat terdapat dua aktor yang berfungsi untuk menjalankan kegiatan pada kamar obat, yaitu:

- a. Apoteker yang memiliki tugas untuk mengelola data obat yang ada di kamar obat dan juga merencanakan pembelian obat untuk periode berikutnya.
- b. Asisten Apoteker yang memiliki tugas untuk mencatat segala transaksi yang dilakukan oleh kamar obat baik berupa pembelian kebutuhan kamar obat maupun penjualan obat pada pasien.

### **2.3. Persediaan**

Secara umum setiap lembaga, unit ataupun bagian yang dalam kegiatannya memerlukan bahan atau logistik maka diperlukan kebutuhan logistik atau barang yang berfungsi untuk menunjang kegiatan unit tersebut. Kegiatan tersebut penting karena dalam rangka memenuhi atau menyediakan barang, bahan, atau logistik tertentu untuk efektivitas kerja lembaga tersebut.

Pada umumnya, persediaan atau inventory merupakan bahan atau barang yang disimpan oleh perusahaan untuk memenuhi tujuan tertentu. Persediaan juga bisa dianggap sebagai sumber dana yang idle atau menganggur karena sebelum persediaan digunakan dana yang terikat di dalamnya tidak bisa digunakan untuk keperluan lain. (Herjanto, 2015)

Dalam suatu perusahaan, persediaan memiliki beberapa fungsi yang menambah fleksibilitas operasi perusahaan. Persediaan memiliki fungsi penting dalam memenuhi kebutuhan perusahaan, seperti :

1. Mengurangi resiko keterlambatan pengiriman barang yang dibutuhkan
2. Mengurangi resiko pengembalian barang yang berkualitas tidak baik
3. Menghilangkan resiko kenaikan harga barang atau inflasi
4. Menyimpan barang yang dihasilkan secara musiman

5. Mendapat keuntungan dari pembelian berkuantitas besar untuk mendapatkan diskon kuantitas
6. Memberikan pelayanan pada pelanggan dengan tersedianya barang yang diperlukan

Pada organisasi pelayanan kesehatan seperti klinik, sistem persediaan obat adalah suatu sistem yang bertanggung jawab untuk merencanakan dan mengawasi persediaan obat mulai dari tahap penerimaan stok sampai distribusi ke pelanggan. Agar klinik memperoleh keuntungan yang optimum maka sistem persediaan harus bertujuan sebagai berikut :

1. Pelayanan pelanggan yang maksimal
2. Biaya operasional yang rendah
3. Investasi persediaan yang minimal

Karena persediaan disimpan di gudang, maka secara fisik manajemen persediaan dan gudang sangat berkaitan. Dalam kasus ini, persediaan disimpan untuk jangka waktu tertentu. Dalam situasi lain, perputaran persediaan berjalan dengan cepat dan gudang obat berfungsi sebagai pusat distribusi.

## **2.4. Manajemen Persediaan**

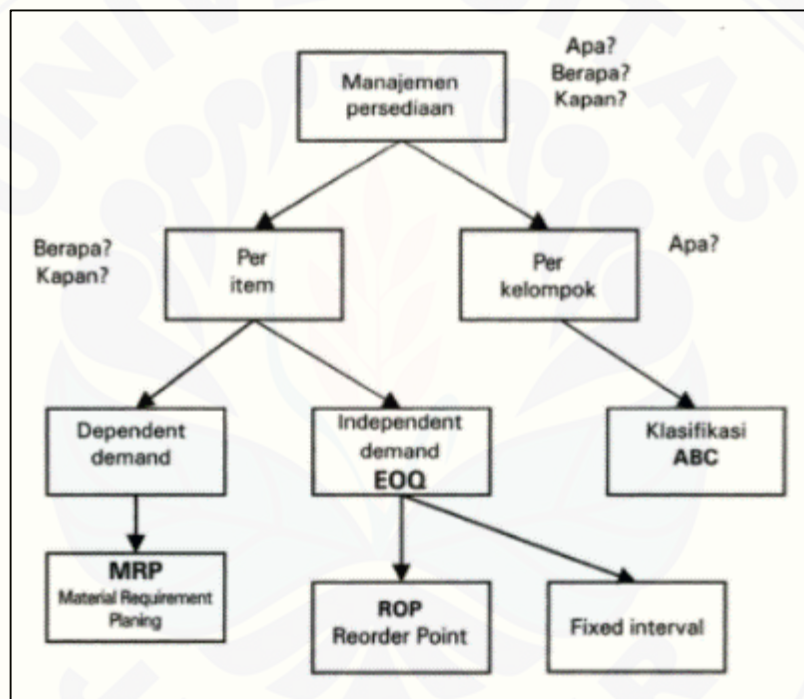
Manajemen persediaan merupakan aktivitas perusahaan atau lembaga untuk mengatur agar barang yang disimpan di dalam lembaga itu tetap tersedia secara optimum. Hal ini penting karena ketersediaan barang akan memperlancar layanan yang seharusnya diberikan.

Persediaan barang dalam perusahaan harus dipastikan dapat memenuhi permintaan barang untuk melayani konsumen. Oleh karena itu perlu disusun rencana persediaan barang di sebuah perusahaan. Apabila persediaan tidak direncanakan dengan baik maka dimungkinkan terjadi kekurangan barang sebagai bahan dasar layanan. Dengan demikian sebaliknya jika perencanaan persediaan dirancang dengan baik maka kepastian ketersediaan barang dapat terjamin. Selanjutnya petugas persediaan barang dapat bekerja secara maksimal.

Manajemen persediaan yang tidak baik dapat menimbulkan resiko akan menurunnya tingkat kepuasan pelanggan serta penghambatan kegiatan perusahaan

sehingga bisa mempengaruhi penjualan. Terlebih lagi kemungkinan terjadinya perubahan nilai uang pada barang yang akan berpengaruh langsung pada keuntungan perusahaan. Selain itu, barang tertentu akan membutuhkan perlakuan khusus pada masa penyimpanannya. Manajemen persediaan berfungsi untuk memastikan ketersediaannya barang dan mengurangi terjadinya stok berlebih atau kehabisan stok.

Manajemen persediaan dapat dibagi menjadi manajemen per item dan manajemen per kelompok sebagaimana terlihat pada Gambar 2.1.



Gambar 2. 1. Jenis Manajemen Persediaan (Ma'arif & Tanjung, 2003)

Manajemen per kelompok merupakan manajemen persediaan yang dilakukan dengan cara mengelompokkan barang sehingga perlakuan terhadap satu kelompok dan yang lain akan berbeda.

## 2.5. Metode ABC

Manajemen persediaan (*inventory management*) merupakan pengendalian dan pengelolaan barang yang bertujuan untuk menyeimbangkan barang yang digunakan dalam sebuah perusahaan atau organisasi yang bergerak di bidang

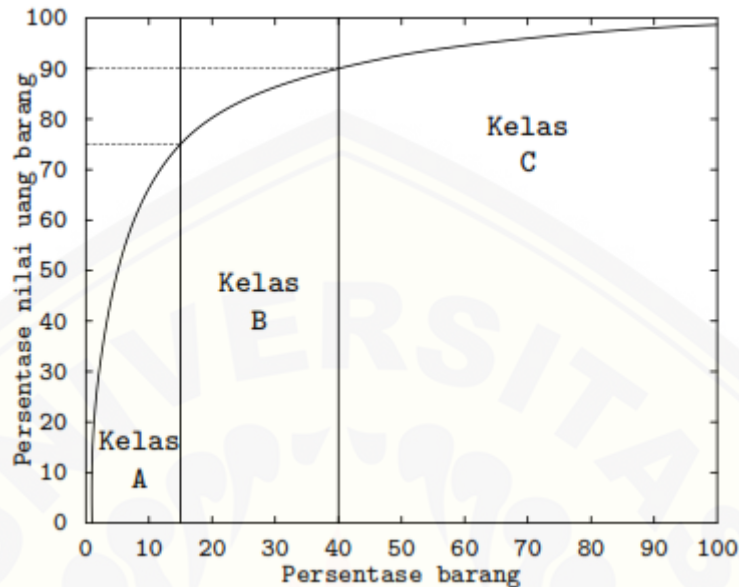
layanan agar selalu tersedia secara optimal. Dengan demikian dalam kasus tersebut tidak akan terjadi kekurangan barang dan juga tidak terjadi penumpukan barang yang mengakibatkan barang rusak yang pada akhirnya menimbulkan kerugian pada perusahaan itu sendiri. Banyak cara atau metode yang dapat digunakan untuk mengembangkan manajemen sistem pengelolaan barang termasuk barang yang berupa obat-obatan di sebuah klinik atau rumah sakit. Metode yang dimaksud antara lain yaitu metode ABC.

Hakikat metode ABC yaitu optimalisasi pengendalian barang yang persentasenya kecil dalam sebuah perusahaan tetapi mempunyai nilai ekonomi dan investasi yang tinggi dalam perusahaan tersebut. Metode atau analisis ABC didasarkan pada hukum pareto (*Ley De Pareto*) yang diambil dari nama Vifredo Pareto (1848-1923) ahli ekonomi Italia yang kemudian dikembangkan oleh Ford Dickie sehingga berkembang sampai sekarang menjadi metode ABC untuk mengefisien dan efektif pengelolaan barang di perusahaan.

Metode ABC merupakan metode yang dapat digunakan untuk mengelompokkan barang menjadi tiga kelompok berdasar nilai dari barang tersebut bila dibandingkan dengan barang lainnya yaitu diurutkan dari nilai tertinggi hingga yang terendah dalam kelompok A, B, dan C. Dalam metode ini, barang dinilai berdasar nilai investasi dari total nilai inventaris. Masing-masing kelas ditentukan dengan besaran sebagai berikut :

1. Kelompok A merupakan kelompok barang yang berjumlah sekitar 15%-20% dari barang yang ada di inventaris tetapi bernilai sekitar 75%-80% dari total nilai inventaris sehingga perlu dikontrol secara ketat karena kelompok ini mempunyai harga yang tinggi.
2. Kelompok B merupakan kelompok barang yang berjumlah sekitar 20%-25% dari barang yang ada di inventaris tetapi bernilai sekitar 10%-15% dari total nilai inventaris sehingga kelompok B memiliki tingkat pengontrolan sedang.
3. Kelompok C merupakan kelompok barang yang berjumlah sekitar 60%-65% dari barang yang ada di inventaris tetapi bernilai sekitar 5%-10% dari total nilai inventaris sehingga memiliki tingkat pengontrolan yang rendah.

Besaran masing-masing kelas di atas akan membentuk suatu kurva sebagaimana terlihat pada Gambar 2.2.



Gambar 2. 2. Grafik Kurva Analisis ABC (Kusnadi, 2009)

Menurut Reddy (2008), dalam membuat analisis ABC ada hal yang harus diperhatikan, yaitu:

1. Dalam mengklasifikasikan menjadi kelompok A, B, dan C yang harus dipertimbangkan adalah total nilai konsumsi, bukan harga barang per unit.
2. Semua barang yang dikonsumsi seharusnya diperhitungkan secara total atau bersama-sama dan sudah tidak dijadikan beberapa kelompok yang berbeda.
3. Periode konsumsi tidak harus selama 1 tahun, dapat disesuaikan dengan kebutuhan seperti misalnya 6 bulan, 4 bulan, dan bahkan 1 bulan.

Analisis ABC dapat dilakukan dengan menghitung 3 nilai, yaitu nilai pakai, nilai investasi dan nilai kritis. Pengelompokan dengan menggunakan analisis ABC bila dilihat dari nilai pakai dan nilai investasi dilakukan dengan mengamati persentase kumulatifnya. Hasil akhir pengelompokan dengan menggunakan analisis ABC ditentukan oleh nilai kritis. (Nurwulandari, 2013)

1. Analisis Nilai Pakai

Nilai pakai adalah nilai yang didapat dari perhitungan persentase pemakaian obat bila dibandingkan dengan jumlah pemakaian dalam satu periode. Untuk memperoleh nilai pakai, pemakaian per obat akan dijumlahkan kemudian diurutkan

dari jumlah pemakaian tertinggi hingga jumlah pemakaian terendah. Setelah data item terurut, kemudian dihitung persentase pemakaiannya dengan perhitungan menggunakan rumus (2.1).

$$\text{Persentase Pemakaian Obat} = \frac{x_N}{\sum x} \times 100\% \dots\dots\dots (2.1)$$

Di mana :

$x_N$  : jumlah pemakaian obat n dalam satu periode

$\sum x$  : jumlah total pemakaian seluruh obat dalam satu periode

Setelah itu, dibuat klasifikasi sediaan sesuai jumlah pemakaiannya menjadi kelompok A, B, dan C berdasarkan persentase kumulatif, dengan kelompok A memiliki persentase kumulatif 0%-75%, kelompok B memiliki persentase kumulatif 75%-95% dan kelompok C memiliki persentase kumulatif 95%-100%.

## 2. Analisis Nilai Investasi

Dalam analisis ini juga dilakukan pengelompokan berdasarkan persentase kumulatif, sama seperti pada analisis nilai pakai tetapi sebelumnya jumlah pemakaian item obat dikalikan dengan harga satuannya. Nilai investasi didapat dengan mengalikan jumlah pemakaian dengan harga satuannya. Perhitungan dilakukan dengan menggunakan rumus (2.2).

$$\text{Jumlah Investasi Obat} = x_N \times h_N \dots\dots\dots (2.2)$$

Di mana:

$x_N$  : jumlah pemakaian obat n dalam satu periode

$h_N$  : harga satuan obat n

Setelah didapat jumlah investasi per item obat selama setahun kemudian diurutkan dari jumlah investasi tertinggi hingga jumlah investasi terendah. Kemudian, dihitung persentase investasi per item obat dengan perhitungan menggunakan rumus (2.3).

$$\text{Persentase Investasi Obat} = \frac{x_N}{\sum x} \times 100\% \dots\dots\dots (2.3)$$

Di mana:

$x_N$  : jumlah investasi obat n dalam satu periode

$\sum x$  : jumlah total investasi seluruh obat dalam satu periode



Dari persentase investasi yang didapat akan dihitung persentase kumulatifnya, kemudian dikelompokkan berdasarkan persentase kumulatifnya.

Kelompok A merupakan obat yang memiliki persentase nilai investasi kumulatif 0%-75%, kelompok B merupakan obat yang memiliki persentase kumulatif 75%-95% sedangkan kelompok C obat yang memiliki persentase kumulatif 95%-100%.

### 3. Analisis Nilai Kritis

Analisis ini dilakukan dengan menggabungkan hasil analisis nilai pakai dan nilai investasi. Masing-masing item yang telah dikelompokkan menjadi kelompok A, B dan C diberi skor berdasarkan hasil analisis nilai pakai dan nilai investasi, dengan kelompok A memiliki nilai 3, kelompok B memiliki nilai 2 dan kelompok C memiliki nilai 1. Nilai indeks kritis dapat dihitung dengan perhitungan menggunakan rumus (2.4).

$$\text{Nilai Indeks Kritis Obat} = \text{Skor Nilai Pakai} + \text{Skor Nilai Investasi} \dots\dots\dots(2.4)$$

Di mana:

Skor Nilai Pakai N : skor kelompok analisis nilai pakai

Skor Nilai Investasi N : skor kelompok analisis nilai investasi

Untuk menentukan kelompok ABC dari analisis nilai indeks kritis, maka harus ditentukan interval skornya. Kelompok A memiliki nilai 3, kelompok B memiliki nilai 2 dan kelompok C memiliki nilai 1, jadi interval skor yang didapat adalah 2-6. Perhitungan interval untuk masing-masing kelompok dengan perhitungan menggunakan rumus (2.5).

$$\text{Interval Skor Nilai Indeks Kritis} = \frac{(X1+X2)-(Y1+Y2)}{(\text{Jumlah Kelas})} \dots\dots\dots(2.5)$$

Di mana :

X1 : nilai tertinggi dari skor nilai pakai

X2 : nilai tertinggi dari skor nilai investasi

Y1 : nilai terendah dari skor nilai pakai

Y2 : nilai terendah dari skor nilai investasi

Berdasarkan perhitungan tersebut, maka sediaan dengan skor NIK 2-3,33 masuk dalam kelompok C, sedangkan sediaan dengan skor NIK 3,33-4,66 masuk

dalam kelompok B dan sediaan dengan skor NIK lebih dari 4,66 masuk dalam kelompok A.

## 2.6. Metode VEN

Metode VEN merupakan metode yang digunakan untuk mengelompokkan obat berdasarkan kepentingannya sebagai obat. Metode ini dinamai metode VEN karena obat menjadi 3 kelompok yang terdiri dari Vital, Esensial dan Non-Esensial. Metode ini juga dikenal sebagai metode VED (Vital, Essential, Desirable).

Kelompok V adalah kelompok obat-obatan yang sangat esensial, yang termasuk dalam kelompok ini adalah obat-obat penyelamat (life saving drugs), obat-obatan untuk pelayanan kesehatan pokok dan obat-obatan untuk mengatasi penyakit-penyakit penyebab kematian terbesar. Kelompok E adalah obat-obatan yang bekerja kausal yaitu obat yang bekerja pada sumber penyebab penyakit. Kelompok N adalah merupakan obat-obatan penunjang yaitu obat-obat yang kerjanya ringan dan bisa dipergunakan untuk menimbulkan kenyamanan atau untuk mengatasi keluhan ringan. (Maimun, 2008)

Berdasarkan toleransi kekosongannya, kelompok V (Vital): sangat esensial untuk memperpanjang hidup, tidak boleh terjadi kekosongan. Kelompok E (Esensial): obat yang bekerja pada sumber penyebab penyakit, kekosongan dapat ditolerir <48 jam. Kelompok N (Non-esensial): obat penunjang, kekosongan dapat ditolerir >48 jam.

Berdasarkan uraian di atas, dapat disimpulkan bahwan kelompok Vital (V) merupakan kelompok jenis obat yang sangat esensial (vital) dan harus ada, yang termasuk dalam kelompok ini antara lain : obat penyelamat (life saving drug), obat-obatan untuk pelayanan kesehatan pokok dan obat-obatan untuk mengatasi penyakit penyebab kematian terbesar. Kekosongan akan obat dari kelompok ini tidak boleh terjadi dan harus selalu ada. Contoh obat yang termasuk jenis obat Vital adalah adrenalin, antitoksin, insulin, obat jantung,

Kelompok Esensial (E) merupakan kelompok jenis obat yang terbukti efektif untuk menyembuhkan penyakit, atau mengurangi penderitaan pasien. Kekosongan akan obat dari kelompok ini dapat ditolerir kurang dari 48 jam. Contoh obat yang

termasuk jenis obat Esensial adalah antibiotic, obat gastrointestinal, NSAID dan lain lain.

Kelompok Non-esensial (N) meliputi aneka ragam obat yang digunakan untuk mengatasi penyakit yang dapat sembuh sendiri (self limiting disease), obat yang diragukan manfaatnya atau obat yang mahal namun tidak mempunyai kelebihan manfaat dibandingkan dengan obat lainnya. Kekosongan akan obat dari kelompok ini dapat ditolerir lebih dari 48 jam. Contoh obat yang termasuk jenis obat Non-esensial adalah vitamin, suplemen dan lain-lain.

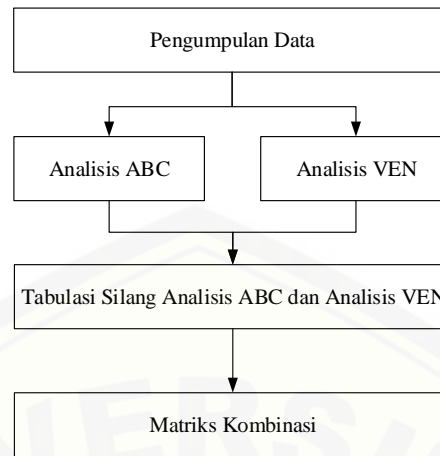
Adapun acuan yang dapat digunakan dalam klasifikasi VEN yaitu WHO Essential Medicines and Health Products Information Portal Jamaica's Drug List, 2008 yang kemudian disesuaikan dengan daftar obat yang ada. Selain itu, Daftar Obat Esensial Nasional (DOEN) juga bisa dijadikan acuan analisis VEN.

## **2.7. Metode ABC-VEN**

Analisis ABC memiliki kelemahan di mana kepentingan sebuah barang berdasarkan pada nilai konsumsinya bukan kegunaannya (Reddy, 2008). Oleh karena itu, pengelompokan tersebut bisa mengabaikan kebutuhan akan barang yang memiliki kekritisian yang tinggi tapi memiliki nilai konsumsi yang rendah. Untuk mengatasi masalah tersebut, analisis VEN digunakan bersamaan dengan analisis ABC.

Metode ABC-VEN merupakan metode gabungan dari metode analisis ABC dan metode analisis VEN. Gabungan yang dimaksud adalah metode analisis ABC digunakan secara bersamaan dengan metode analisis. Metode VEN digunakan untuk memperbaiki kelemahan metode analisis ABC.

Untuk melakukan analisis dengan metode ABC-VEN, beberapa proses harus dilakukan sesuai dengan bagan alur yang bisa dilihat pada Gambar 2.3.



Gambar 2. 3. Bagan alur analisis ABC-VEN (Singh, Singh, & Singh, 2015)

Pada tahap tabulasi silang analisis ABC dan analisis VEN, metode ABC-VEN didapatkan dari tabulasi silang antara analisis ABC dan analisis VEN sehingga didapatkan sembilan kelompok klasifikasi, yaitu AV, AE, AN, BV, BE, BN, CV, CE dan CN. Huruf pertama diambil dari analisis ABC, sedangkan huruf kedua diambil dari analisis VEN (Wandalkar, Pandit, & Zite, 2013). Hasil tabulasi silang dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Tabulasi Silang Analisis ABC dan Analisis VEN

	A ( <i>Always</i> )	B ( <i>Better</i> )	C ( <i>Control</i> )
V (Vital)	AV	BV	CV
E (Esensial)	AE	BE	CE
N(Non- esensial)	AN	BN	CN

Kesembilan klasifikasi yang didapatkan dari tabulasi silang tersebut dapat dikelompokkan menjadi tiga kategori. Kategori I merupakan kelompok obat vital dan mahal terdiri dari AV, BV, CV, AE dan AN. Kategori II terdiri dari BE, CE dan BN. Kategori III yang memiliki prioritas terendah terdiri dari CN saja. (Devnani, Gupta, & Nigah, 2010)

### **BAB 3. METODE PENELITIAN**

Pada bab ini akan dibahas mengenai metode yang akan digunakan dalam penelitian untuk mernancang dan mengembangkan sistem.

#### **3.1. Jenis Penelitian**

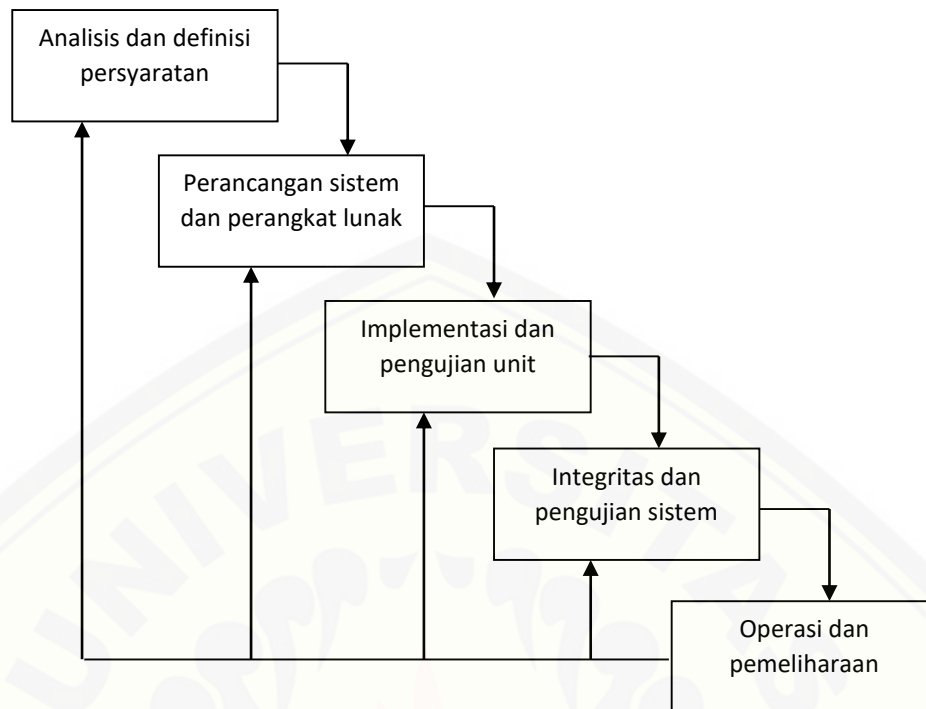
Jenis penelitian yang dilakukan adalah termasuk penelitian pengembangan sistem. Penelitian ini bertujuan untuk memperbaiki sistem manajemen yang sudah ada tetapi belum sesuai dengan tujuan yang diharapkan. Inovasi sistem yang lebih baik selalu diupayakan untuk meningkatkan kualitas layanan pada poliklinik obat di Politeknik. Kegiatan penelitian ini meliputi rencana pengembangan, pelaksanaan pengembangan, serta evaluasi keberhasilan program manajemen kamar obat.

#### **3.2. Tempat dan Waktu Penelitian**

Penelitian ini dilaksanakan di salah satu unit dari lembaga Perguruan Tinggi Politeknik Negeri Jember yaitu Poliklinik Kesehatan Politeknik Negeri Jember. Sedangkan waktu pelaksanaan penelitian selama 4 bulan yaitu mulai Maret s/d Juni 2019.

#### **3.3. Tahapan Penelitian**

Penelitian ini dikembangkan menggunakan metode *Software Development Life Cycle* (SDLC) dengan model *waterfall*. Model *waterfall* merupakan salah satu model yang sistematis dan sekuensial yang mulai pada analisis, desain, kode, tes dan pemeliharaan (Pressman, 2002). Model *waterfall* digunakan karena model ini yang melakukan pendekatan secara sistematis dan urut sehingga sistem yang dihasilkan tidak akan berbeda jauh dari analisis awalnya. Adapun alur kerja *waterfall* ditunjukkan pada gambar 3.1



Gambar 3. 1 Alur Model *Waterfall*

### 3.3.1. Analisis dan Definisi Persyaratan

Analisis kebutuhan merupakan tahap untuk mengumpulkan serta mengolah data dan informasi yang dibutuhkan untuk membangun sistem. Pengumpulan data dapat dilakukan melalui studi literatur atau wawancara pada pihak yang terkait. Data yang terkumpul kemudian diolah dan dikelompokkan menjadi kebutuhan fungsional dan kebutuhan non fungsional.

#### 1. Tahap Pengumpulan Data

Tahap pengumpulan data bertujuan untuk memperoleh data atau informasi yang dibutuhkan dalam mencapai tujuan penelitian. Pengumpulan data pada penelitian ini dilakukan melalui beberapa cara yaitu:

##### a. Studi Literatur

Studi literatur berisi uraian tentang teori, temuan dan bahan penelitian lain yang diperoleh dari bahan acuan untuk dijadikan landasan kegiatan penelitian. Studi literatur berisi ulasan, rangkuman data pemikiran penulisan

tentang beberapa sumber pustaka tentang topik yang dibahas (Hasibuan, 2007).

b. Wawancara

Wawancara merupakan teknik pengumpulan data survei berupa tanya jawab peneliti dengan narasumber. Wawancara tersebut berupa percakapan langsung antar dua pihak atau lebih untuk mendapatkan informasi secara lisan dengan tujuan untuk memperoleh data yang dapat menjelaskan ataupun menjawab suatu permasalahan penelitian (Hasibuan, 2007).

2. Tahap Pengolahan Data

Data yang telah diperoleh pada tahap pengumpulan data diolah untuk menentukan kebutuhan-kebutuhan atau fungsionalitas sistem informasi yang akan dibangun.

a. Kebutuhan Fungsional

Kebutuhan fungsional berisi fitur-fitur apa saja yang nantinya dibutuhkan oleh sistem sehingga sistem bisa berfungsi sesuai kebutuhan pengguna

b. Kebutuhan Non Fungsional

Kebutuhan non-fungsional merupakan fitur-fitur yang berfungsi untuk mendukung fungsionalitas sistem dalam memenuhi kebutuhan pengguna

3.3.2. Perancangan Sistem dan Perangkat Lunak

Desain sistem dirancang dan dikembangkan dengan menggunakan teknik *Unified Modelling Language* (UML) yang dilengkapi dengan konsep *Object Oriented Design*. Langkah kegiatan akan dilakukan dengan rincian berikut.

1. *Input-Output Diagram*

*Business Procces* merupakan gambaran umum sistem yang akan dibangun. *Business Process* menunjukkan *input* yang dibutuhkan dan *output* yang dihasilkan oleh sistem serta tujuan dibangunnya sistem.

2. *Business Process*

*Business Procces* merupakan gambaran umum alur aktivitas pengguna dalam sistem yang akan dibangun.

3. *Use Case Diagram*

*Use Case Diagram* adalah model diagram yang digunakan untuk menggambarkan fitur apa saja yang bisa dilakukan oleh sistem bila dilihat dari sudut pandang pengamatan luar. *Use Case Diagram* menggambarkan interaksi antara aktor dan sistem sehingga dapat diketahui fitur apa yang bisa diakses oleh siapa.

4. *Use Case Scenario*

*Use Case Scenario* merupakan skenario yang digunakan untuk memperjelas alur interaksi antara aktor dan sistem pada suatu fitur atau use case pada *Use Case Diagram*. Pada *Use Case Scenario*, sistem akan memberikan “reaksi” tertentu bila aktor melakukan sebuah “aksi” pada kondisi tertentu sehingga sistem akan memberi “reaksi” yang berbeda bila “aksi” yang dilakukan oleh aktor tidak sesuai dengan kondisi tersebut.

5. *Activity Diagram*

*Activity Diagram* merupakan model diagram yang menggambarkan alur aktivitas yang terjadi pada sistem berdasarkan *Use Case Scenario*. *Activity Diagram* mempermudah pengamatan aktivitas yang terjadi dalam satu fitur.

6. *Sequence Diagram*

*Sequence Diagram* merupakan model diagram yang menggambarkan interaksi antar objek pada sistem sesuai dengan *Use Case Scenario* dan *Activity Diagram* sehingga dapat diketahui objek apa saja yang berinteraksi untuk menjalankan satu fitur.

7. *Class Diagram*

*Class Diagram* merupakan model diagram yang menggambarkan hubungan antar *class* pada sistem secara keseluruhan.

8. *Entity Relationship Diagram (ERD)*

*Entity Relationship Diagram (ERD)* merupakan gambaran dari komponen-komponen yang terdapat dalam database sistem. *Entity Relationship Diagram*



juga menggambarkan relasi yang terbentuk antar table sehingga setiap table dapat saling berhubungan satu sama lain.

### 3.3.3. Implementasi dan Pengujian Unit

Tahap implementasi dilakukan berdasarkan desain sistem yang selanjutnya diubah dalam bentuk program, yaitu :

1. Penulisan program menggunakan bahasa pemrograman *Page Hyper Text Processor (PHP)* dengan framework *Codeigniter (CI)*.
2. *Database Management System (DBMS)* yang digunakan adalah *MySQL* dengan menggunakan jaringan lokal aplikasi *XAMPP*.

### 3.3.4. Pengujian Sistem (*Integration and System Testing*)

Tahap pengujian sistem bertujuan untuk mengetahui sejauh mana sistem ini dapat berjalan. *Integration Testing* berfungsi untuk mengetahui apakah sistem ini dapat berfungsi dengan baik sesuai dengan yang diharapkan, serta untuk mengetahui letak kekurangan pada sistem. Penelitian ini melakukan pengujian sistem dengan cara sebagai berikut:

#### 1. *White Box Testing*

*White box testing* merupakan metode pengujian perangkat lunak dengan melihat modul yang telah dibuat dan program-program yang ada. Pengujian ini dilakukan oleh pembuat program (*developer*). Jika ada modul yang menghasilkan output yang tidak sesuai, maka baris-baris program, variabel dan parameter yang terlibat pada unit tersebut satu persatu akan dicek dan diperbaiki, kemudian di-*compile* ulang.

#### 2. *Black Box Testing*

*Black box testing* merupakan metode pengujian perangkat lunak yang memeriksa fungsionalitas dari aplikasi disesuaikan dengan kebutuhan atau *requirement* yang telah ditentukan. Metode ini memfokuskan pada keperluan fungsional dari *software* dan dilakukan berdasarkan perspektif pengguna.

### 3.3.5. Pemeliharaan (*Operation and Maintenance*)

Tahap pemeliharaan dilakukan ketika sistem memiliki kesalahan yang belum terdeteksi sebelumnya, sehingga kesalahan-kesalahan sistem perlu diperbaiki. Pemeliharaan juga dilakukan apabila sistem mengalami perubahan-perubahan karena permintaan baru dari user.



## BAB 4. PENGEMBANGAN SISTEM

Pada bab ini akan dibahas mengenai pengembangan sistem yang dilakukan dengan menggunakan menggunakan model *waterfall*. Model ini merupakan metodologi pengembangan perangkat lunak yang mengusulkan pendekatan kepada perangkat lunak secara sistematis dan sekuensial yang dimulai dengan analisis, lalu desain, kode, pengujian dan pemeliharaan.

### 4.1. Analisis Kebutuhan

Analisis kebutuhan merupakan tahap awal dalam membangun sistem informasi berdasarkan model *waterfall*. Analisis kebutuhan yang dilakukan sesuai dengan data yang diperoleh pada tahapan yang telah dijelaskan pada bab metodologi penelitian sub bab 3.3.1.

#### 4.1.1. Tahap Pengumpulan Data

Analisis kebutuhan diawali dengan mengidentifikasi persediaan obat yang sering dibutuhkan oleh masyarakat yang ditinjau dari waktu kedaluwarsa obat. Hal ini dilakukan agar terhindar dari kesalahan misalnya obat yang kedaluwarsa

Pengumpulan data dilakukan dengan observasi lapangan serta wawancara kepada kepala poliklinik. Salah satu data yang diperoleh adalah data penjualan obat pada bulan Maret sampai bulan Juni yang dapat dilihat pada tabel 4.1.

Tabel 4. 1 Keluar-masuk obat Bulan Maret 2019

Nama Obat	Stok Awal Maret	Pembelian Maret	Penjualan Maret	Pembelian April
ANACETINE	3	10	5	0
CALORTUSIN TABLET	121	500	315	0
CARGESIK	304	200	51	0
CAVIPLEX KAPLET	282	900	737	800
CAVIPLEX SIRUP	6	0	1	5
CEFADROXIL HEXPHARM	604	200	363	0

CEFIXIME OGB DEXA	116	100	10	0
CURCUMA PLUS	216	0	24	0
DEGIROL	60	250	158	90
DEXCLOSAN	219	400	227	100
DEXTRAL	298	150	70	0
DEXTRAL FORTE	184	0	30	0
DEXYCOL	153	0	25	100
DOLO-LICOBION	259	500	328	500
GENOINT SALEP KULIT	8	5	6	0
GLIMEPIRIDE HEXPHARM	410	0	85	0
GLIMEPIRIDE HEXPHARM	300	0	65	400
HISTIGO	120	100	52	0
METFORMIN HCL OGB DEXA	10	540	195	500
RENADINAC	244	0	16	0
SERUM ANTI TETANUS (KUDA)	1	1	1	0
ZEROPAIN	8	15	7	0

#### 4.1.2. *Statement of Purpose (SOP)*

Sistem Informasi Pengendalian Manajemen Obat Menggunakan Metode ABC-VEN merupakan sebuah sistem yang mampu membantu pengendalian persediaan stok yang ada pada kamar obat. Ada beberapa data yang dibutuhkan agar sistem dapat menghasilkan prioritas pengontrolan obat. Data yang dibutuhkan untuk sistem ini antara lain data transaksi penjualan obat yang telah dilakukan di poliklinik dan data klasifikasi obat berdasarkan metode VEN. Dari data tersebut, dilakukan perhitungan menggunakan metode ABC. Dari hasil penggabungan perhitungan metode ABC dan metode VEN akan diperoleh daftar prioritas pengontrolan obat yang akan dibeli pada periode berikutnya.

Selain itu, sistem ini juga mampu mengelola data obat dan stok yang ada di gudang serta pembuangan yang terjadi pada obat yang telah melampaui tanggal kadaluarsanya.

## 4.1.3. Kebutuhan Fungsional

Kebutuhan fungsional berisi fitur-fitur apa saja yang nantinya dibutuhkan oleh sistem sehingga sistem bisa berfungsi sesuai kebutuhan pengguna. Kebutuhan fungsional dari sistem adalah sebagai berikut :

1. Aktor apoteker mampu mengelola data obat meliputi menambah, mengubah dan menghapus, sedangkan aktor asisten apoteker hanya bisa melihat data obat.
2. Aktor asisten apoteker mampu mengelola data transaksi meliputi menambah, mengubah dan menghapus, sedangkan aktor apoteker hanya bisa melihat data transaksi.
3. Aktor apoteker mampu menambah data stok.
4. Sistem mampu mengubah data stok secara otomatis ketika sebuah transaksi telah berhasil dilakukan.
5. Sistem mampu menentukan item yang keluar saat transaksi penjualan terjadi.
6. Sistem mampu memberi peringatan akan obat yang mendekati atau melewati tanggal kadaluarsanya.
7. Sistem mampu melakukan perhitungan menggunakan metode analisis ABC.
8. Aktor apoteker mampu melakukan pengelompokkan obat menggunakan metode analisis VEN.
9. Sistem mampu melakukan perhitungan menggunakan metode analisis ABC-VEN yang dapat menjadi acuan pengambilan keputusan pembelian obat.
10. Aktor apoteker dan asisten apoteker mampu menambah data pembuangan obat atau penanganan obat yang telah melewati tanggal kadaluarsa

## 4.1.4. Kebutuhan Non-fungsional

Kebutuhan non-fungsional merupakan fitur-fitur yang berfungsi untuk mendukung fungsionalitas sistem dalam memenuhi kebutuhan pengguna. Kebutuhan non-fungsional pada sistem ini adalah sebagai berikut :

1. Sistem menggunakan username dan password untuk autentifikasi akses pengguna pada sistem.

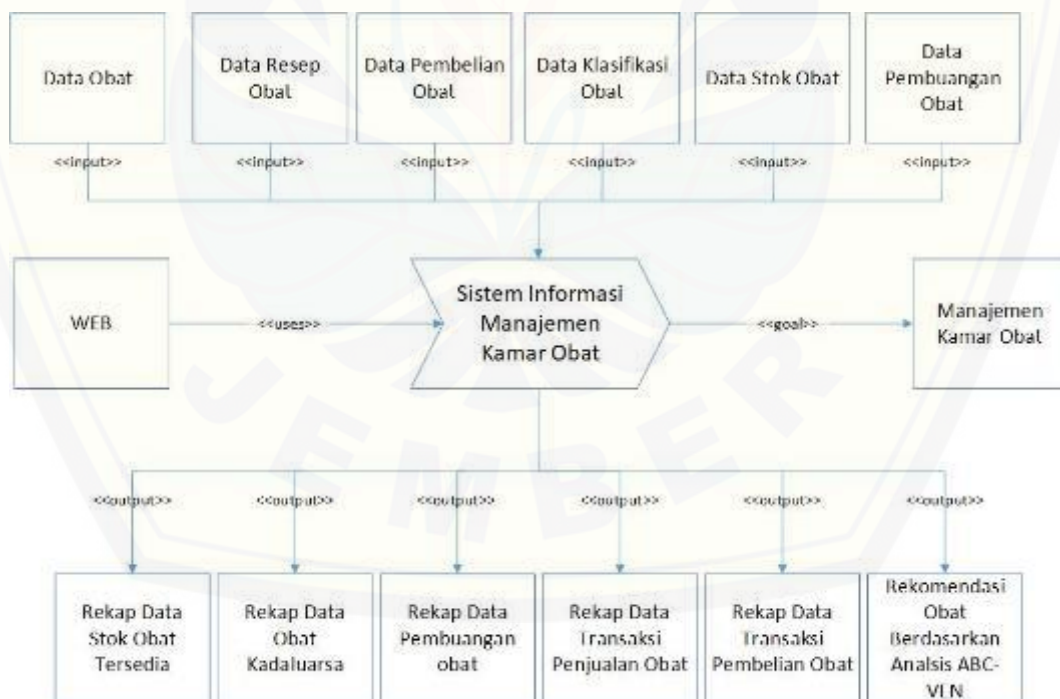
2. Sistem memiliki tampilan yang *user friendly* sehingga mudah untuk dimengerti dan dioperasikan.

#### 4.2. Desain Sistem

Desain sistem dirancang dan dikembangkan dengan menggunakan teknik *Unified Modelling Language* (UML) yang dilengkapi dengan konsep *Object Oriented Design*. Langkah kegiatan akan dilakukan dengan rincian berikut.

##### 4.2.1. Input-Output Diagram

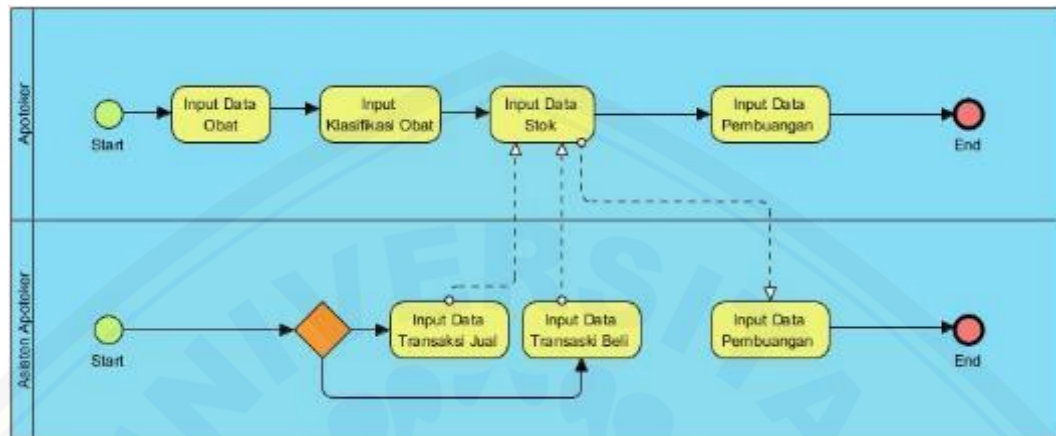
*Input-Output Diagram* merupakan gambaran umum sistem yang akan dibangun. *Input-output Diagram* menunjukkan *input* yang dibutuhkan dan *output* yang dihasilkan oleh sistem serta tujuan dibangunnya sistem. Input-Output diagram digambarkan pada Gambar 4.1.



Gambar 4. 1 *Input-Output Diagram* Sistem Informasi Kamar Obat

#### 4.2.2. Business Process

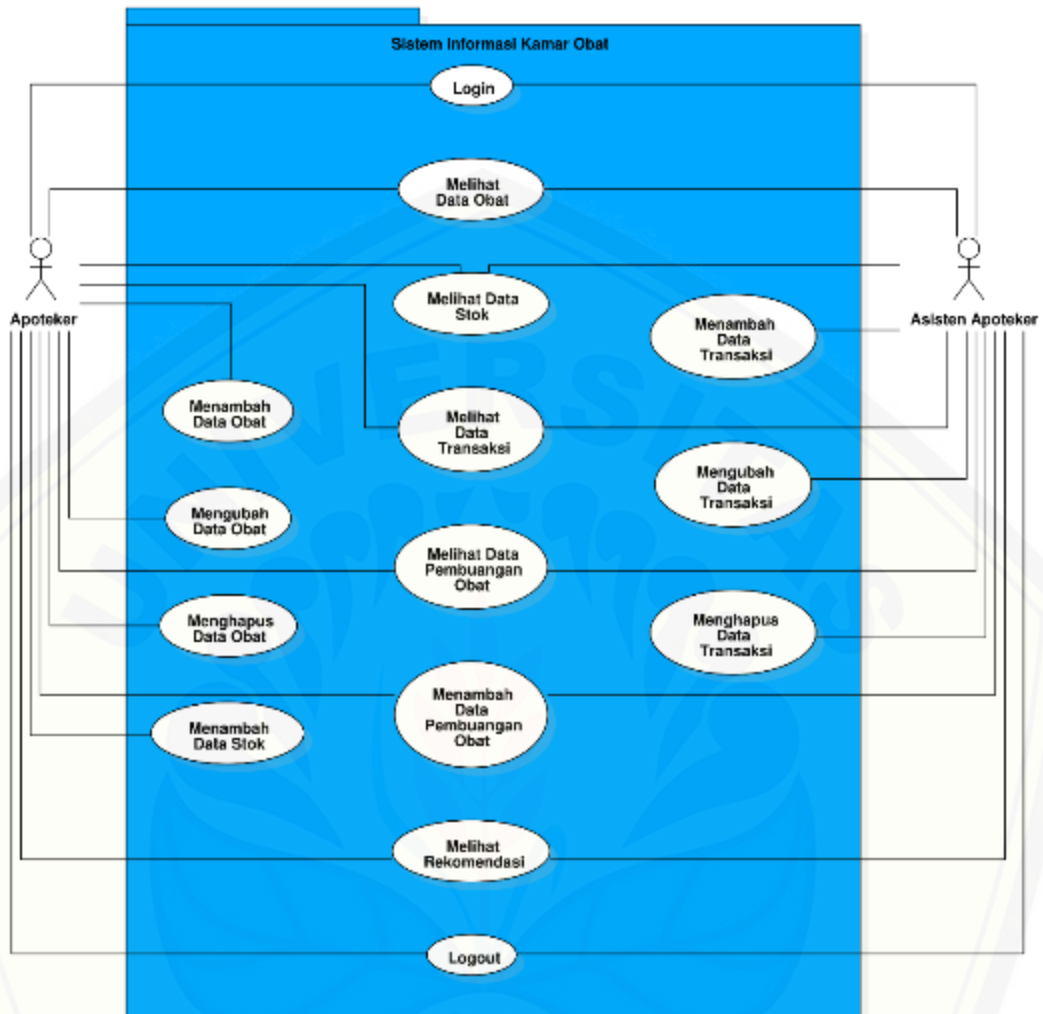
*Business Process* merupakan menggambarkan data yang dibutuhkan oleh sistem. *Business process* digambarkan pada Gambar 4.2



Gambar 4. 2 *Business Process* Sistem Informasi Kamar Obat

#### 4.2.3. Use Case Diagram

*Use Case Diagram* adalah model diagram yang digunakan untuk menggambarkan fitur apa saja yang bisa dilakukan oleh sistem bila dilihat dari sudut pandang pengamatan luar. *Use Case Diagram* menggambarkan interaksi antara aktor dan sistem sehingga dapat diketahui fitur apa yang bisa diakses oleh siapa. *Use case diagram* ditunjukkan pada gambar 4.3 yang terdapat 2 aktor yang dapat mengakses sistem yaitu Apoteker dan Asisten Apoteker.



Gambar 4. 3 Use case Diagram

Fitur-fitur yang terdapat pada sistem beserta aktor yang berhak mengakses fitur tersebut berdasarkan *use case diagram* dijelaskan pada tabel definisi aktor dan deskripsi *use case*. Pada *use case diagram*, terdapat 2 aktor yang bias mengakses sistem. Adapun deskripsi dari masing-masing aktor dapat dilihat pada Tabel

Tabel 4. 2 Definisi Aktor

No.	Aktor	Deskripsi
1	Apoteker	Aktor ini memiliki hak akses untuk : <ul style="list-style-type: none"> <li>• Login</li> <li>• Melihat Data Obat</li> </ul>



- 
- Menambah Data Obat
  - Mengubah Data Obat
  - Menghapus Data Obat
  - Melihat Data Stok
  - Menambah Data Stok
  - Melihat Data Transaksi
  - Melihat Data Pembuangan Obat
  - Menambah Data Pembuangan Obat
  - Melihat Rekomendasi
  - Logout
- 

- |   |                  |  |
|---|------------------|--|
| 2 | Asisten Apoteker | Aktor ini memiliki hak akses untuk : <ul style="list-style-type: none"> <li>• Login</li> <li>• Melihat Data Obat</li> <li>• Melihat Data Stok</li> <li>• Melihat Data Transaksi</li> <li>• Menambah Data Transaksi</li> <li>• Mengubah Data Transaksi</li> <li>• Menghapus Data Transaksi</li> <li>• Melihat Data Pembuangan Obat</li> <li>• Menambah Data Pembuangan Obat</li> <li>• Melihat Rekomendasi</li> <li>• Logout</li> </ul> |
|---|------------------|--|
- 

Dalam *use case diagram*, terdapat 15 *use case* yang ada pada sistem. Adapun deskripsi dari masing-masing *use case* dapat dilihat pada Tabel 4.3

Tabel 4. 3 Deskripsi *Use Case*

No.	Use Case	Penjelasan
1	Login	Merupakan <i>use case</i> yang menggambarkan proses login atau autentifikasi untuk masuk ke sistem

---

2	Melihat Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses menampilkan data obat
3	Menambah Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses menambahkan data obat
4	Mengubah Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses mengubah data obat
5	Menghapus Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses menghapus data obat
6	Melihat Stok Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses menampilkan data stok obat yang tersedia
7	Menambah Stok Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses menambahkan data stok obat yang tersedia
8	Melihat Transaksi	Data	Merupakan <i>use case</i> yang menggambarkan proses menampilkan data transaksi yang telah dilakukan
9	Menambah Transaksi	Data	Merupakan <i>use case</i> yang menggambarkan proses menambahkan data transaksi yang telah dilakukan
10	Mengubah Transaksi	Data	Merupakan <i>use case</i> yang menggambarkan proses mengubah data transaksi yang telah dilakukan
11	Menghapus Transaksi	Data	Merupakan <i>use case</i> yang menggambarkan proses menghapus data transaksi yang telah dilakukan
12	Melihat Pembuangan Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses menampilkan data pembuangan obat
13	Menambah Pembuangan Obat	Data	Merupakan <i>use case</i> yang menggambarkan proses menambahkan data pembuangan obat
14	Melihat Rekomendasi		Merupakan <i>use case</i> yang menggambarkan proses menampilkan data penrotiran obat
15	Logout		Merupakan <i>use case</i> yang menggambarkan proses keluar dari sistem

#### 4.2.4. Use Case Scenario

*Scenario* digunakan untuk menjelaskan cara kerja sistem berdasarkan tugas user yang terdapat pada *use case diagram*. *Scenario* terdiri dari nama *use case*, aksi aktor dan reaksi sistem.

1. *Use Case Scenario Login*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

2. *Use Case Scenario Melihat Data Obat*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

3. *Use Case Scenario Menambah Data Obat*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

4. *Use Case Scenario Mengubah Data Obat*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

5. *Use Case Scenario Menghapus Data Obat*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

6. *Use Case Scenario Melihat Data Stok Obat*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

7. *Use Case Scenario Menambah Data Stok Obat*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

8. *Use Case Scenario Melihat Data Transaksi*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

9. *Use Case Scenario* Menambah Data Transaksi

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

10. *Use Case Scenario* Mengubah Data Transaksi

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

11. *Use Case Scenario* Menghapus Data Transaksi

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

12. *Use Case Scenario* Melihat Data Pembuangan Obat

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

13. *Use Case Scenario* Menambah Data Pembuangan Obat

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

14. *Use Case Scenario* Melihat Rekomendasi

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Tabel 4.2.

Tabel 4. 4 Skenario Melihat Rekomendasi

No Use Case	USC 14
Nama Use Case	Lihat data rekomendasi obat
Aktor	Apoteker, Asisten Apoteker
Prekondisi	Aktor sudah melakukan login
Prakondisi	Aktor sudah melihat data rekomendasi obat
<b>SKENARIO NORMAL</b>	
<b>“Melihat Rekomendasi ABC-VEN”</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Pilih menu “Rekomendasi”	2. Menampilkan submenu
3. Pilih ABC-VEN	4. Menampilkan Halaman Rekomendasi ABC-VEN yang berisi tabel data rekomendasi obat

berdasarkan klasifikasi ABC-VEN yang berisi :

- a. No
- b. Nama
- c. Kelas ABC
- d. Kelas VEN
- e. Kelas ABC-VEN

Serta terdapat pilihan tahun dan bulan untuk memilih periode perhitungan ABC-VEN

**SKENARIO ALTERNATIF  
“Melihat Rekomendasi ABC”**

<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Pilih menu “Rekomendasi”	2. Menampilkan submenu
3. Pilih ABC	4. Menampilkan Halaman Rekomendasi ABC yang berisi tabel data rekomendasi obat berdasarkan klasifikasi ABC yang berisi : <ol style="list-style-type: none"> <li>a. No</li> <li>b. Nama</li> <li>c. Kelas Nilai Pemakaian</li> <li>d. Kelas Nilai Investasi</li> <li>e. Kelas Nilai iKritis</li> </ol> Serta terdapat pilihan tahun dan bulan untuk memilih periode perhitungan ABC

**SKENARIO ALTERNATIF  
“Melihat Rekomendasi VEN”**

<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Pilih menu “Rekomendasi”	2. Menampilkan submenu
3. Pilih VEN	4. Menampilkan tiga tabel data obat yang dikelompokkan berdasarkan klasifikasi VEN yang masing-masing berisi : <ol style="list-style-type: none"> <li>a. No</li> <li>b. Kode Obat</li> <li>c. Nama</li> </ol>

## 15. *Use Case Scenario Logout*

Penjelasan urutan aksi aktor untuk melakukan *use case* dan reaksi sistem pada skenario normal dan skenario alternatif terdapat pada Lampiran A.

### 4.2.5. *Activity Diagram*

*Activity Diagram* merupakan model diagram yang menggambarkan alur aktivitas yang terjadi pada sistem berdasarkan *Use Case Scenario*. *Activity Diagram* mempermudah pengamatan aktivitas yang terjadi dalam satu fitur.

#### 1. *Activity Diagram Login*

*Activity Diagram Login* yang menggambarkan alur aktivitas dalam *use case* Login terdapat pada Lampiran

#### 2. *Activity Diagram Melihat Data Obat*

*Activity Diagram Melihat Data Obat* yang menggambarkan alur aktivitas dalam *use case* Melihat Data Obat terdapat pada Lampiran B.

#### 3. *Activity Diagram Menambah Data Obat*

*Activity Diagram Menambah Data Obat* yang menggambarkan alur aktivitas dalam *use case* Menambah Data Obat terdapat pada Lampiran B.

#### 4. *Activity Diagram Mengubah Data Obat*

*Activity Diagram Mengubah Data Obat* yang menggambarkan alur aktivitas dalam *use case* Mengubah Data Obat terdapat pada Lampiran B.

#### 5. *Activity Diagram Menghapus Data Obat*

*Activity Diagram Menghapus Data Obat* yang menggambarkan alur aktivitas dalam *use case* Menghapus Data Obat terdapat pada Lampiran B.

#### 6. *Activity Diagram Melihat Data Stok Obat*

*Activity Diagram Melihat Data Stok Obat* yang menggambarkan alur aktivitas dalam *use case* Melihat Data Stok Obat terdapat pada Lampiran B.

#### 7. *Activity Diagram Menambah Data Stok Obat*

*Activity Diagram Menambah Data Stok Obat* yang menggambarkan alur aktivitas dalam *use case* Menambah Data Stok Obat terdapat pada Lampiran B.

8. *Activity Diagram* Melihat Data Transaksi

*Activity Diagram* Melihat Data Transaksi yang menggambarkan alur aktivitas dalam *use case* Melihat Data Transaksi terdapat pada Lampiran B.

9. *Activity Diagram* Menambah Data Transaksi

*Activity Diagram* Menambah Data Transaksi yang menggambarkan alur aktivitas dalam *use case* Menambah Data Transaksi terdapat pada Lampiran B.

10. *Activity Diagram* Mengubah Data Transaksi

*Activity Diagram* Mengubah Data Transaksi yang menggambarkan alur aktivitas dalam *use case* Mengubah Data Transaksi terdapat pada Lampiran B.

11. *Activity Diagram* Menghapus Data Transaksi

*Activity Diagram* Menghapus Data Transaksi yang menggambarkan alur aktivitas dalam *use case* Menghapus Data Transaksi terdapat pada Lampiran B.

12. *Activity Diagram* Melihat Data Pembuangan Obat

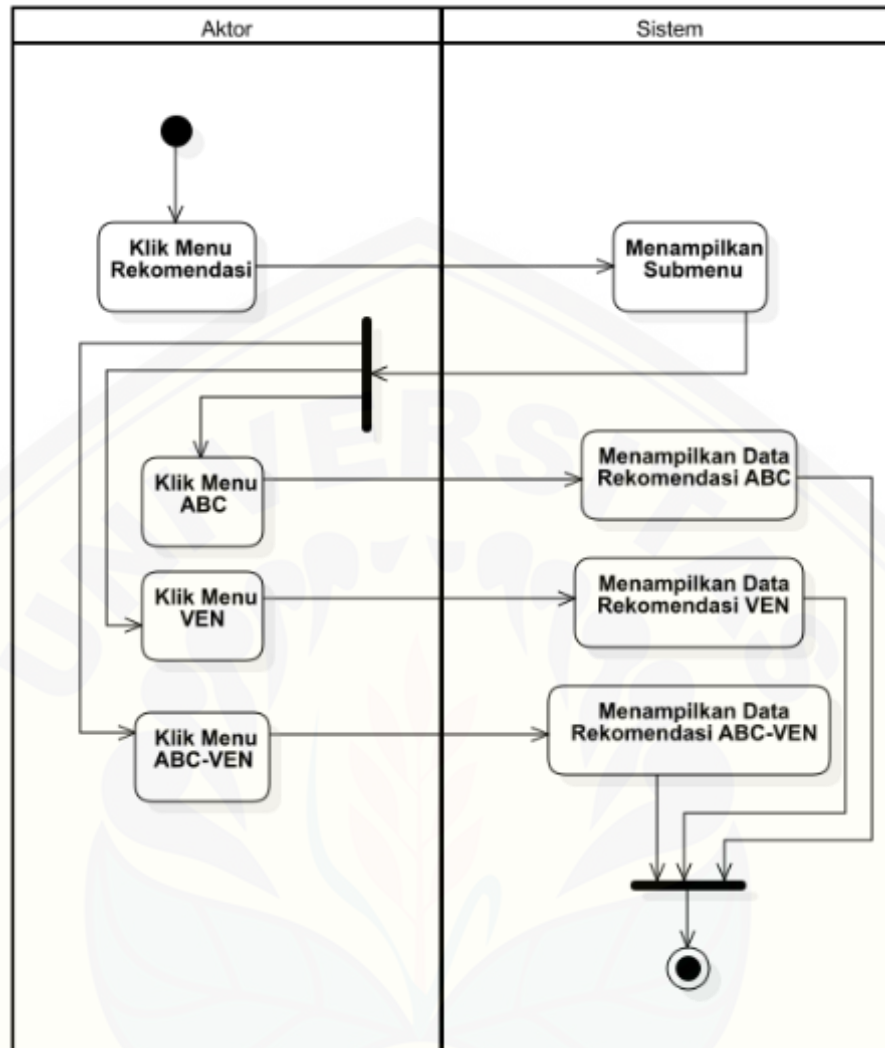
*Activity Diagram* Melihat Data Pembuangan Obat yang menggambarkan alur aktivitas dalam *use case* Melihat Data Pembuangan Obat terdapat pada Lampiran B.

13. *Activity Diagram* Menambah Data Pembuangan Obat

*Activity Diagram* Menambah Data Pembuangan Obat yang menggambarkan alur aktivitas dalam *use case* Menambah Data Pembuangan Obat terdapat pada Lampiran B.

14. *Activity Diagram* Melihat Rekomendasi

*Activity Diagram* Melihat Rekomendasi yang menggambarkan alur aktivitas dalam *use case* Melihat Rekomendasi terdapat pada Gambar 4.4.



Gambar 4. 4 Activity Diagram Rekomendasi

#### 15. Activity Diagram Logout

Activity Diagram Logout yang menggambarkan alur aktivitas dalam use case Logout terdapat pada Lampiran B.

#### 4.2.6. Sequence Diagram

Sequence Diagram merupakan model diagram yang menggambarkan interaksi antar objek pada sistem sesuai dengan Use Case Scenario dan Activity Diagram sehingga dapat diketahui objek apa saja yang berinteraksi untuk menjalankan satu fitur.



1. *Sequence Diagram Login*

*Sequence Diagram Login* yang menggambarkan alur interaksi antar objek dalam *use case* Login terdapat pada Lampiran C.

2. *Sequence Diagram Melihat Data Obat*

*Sequence Diagram Melihat Data Obat* yang menggambarkan alur interaksi antar objek dalam *use case* Melihat Data Obat terdapat pada Lampiran C.

3. *Sequence Diagram Menambah Data Obat*

*Sequence Diagram Menambah Data Obat* yang menggambarkan alur interaksi antar objek dalam *use case* Menambah Data Obat terdapat pada Lampiran

4. *Sequence Diagram Mengubah Data Obat*

*Sequence Diagram Mengubah Data Obat* yang menggambarkan alur interaksi antar objek dalam *use case* Mengubah Data Obat terdapat pada Lampiran C.

5. *Sequence Diagram Menghapus Data Obat*

*Sequence Diagram Menghapus Data Obat* yang menggambarkan alur interaksi antar objek dalam *use case* Menghapus Data Obat terdapat pada Lampiran C.

6. *Sequence Diagram Melihat Data Stok Obat*

*Sequence Diagram Melihat Data Stok Obat* yang menggambarkan alur interaksi antar objek dalam *use case* Melihat Data Stok Obat terdapat pada Lampiran C.

7. *Sequence Diagram Menambah Data Stok Obat*

*Sequence Diagram Menambah Data Stok Obat* yang menggambarkan alur interaksi antar objek dalam *use case* Menambah Data Stok Obat terdapat pada Lampiran

8. *Sequence Diagram Melihat Data Transaksi*

*Sequence Diagram Melihat Data Transaksi* yang menggambarkan alur interaksi antar objek dalam *use case* Melihat Data Transaksi terdapat pada Lampiran C.

9. *Sequence Diagram* Menambah Data Transaksi

*Sequence Diagram* Menambah Data Transaksi yang menggambarkan alur interaksi antar objek dalam *use case* Menambah Data Transaksi terdapat pada Lampiran C.

10. *Sequence Diagram* Mengubah Data Transaksi

*Sequence Diagram* Mengubah Data Transaksi yang menggambarkan alur interaksi antar objek dalam *use case* Mengubah Data Transaksi terdapat pada Lampiran C.

11. *Sequence Diagram* Menghapus Data Transaksi

*Sequence Diagram* Menghapus Data Transaksi yang menggambarkan alur interaksi antar objek dalam *use case* Menghapus Data Transaksi terdapat pada Lampiran C.

12. *Sequence Diagram* Melihat Data Pembuangan Obat

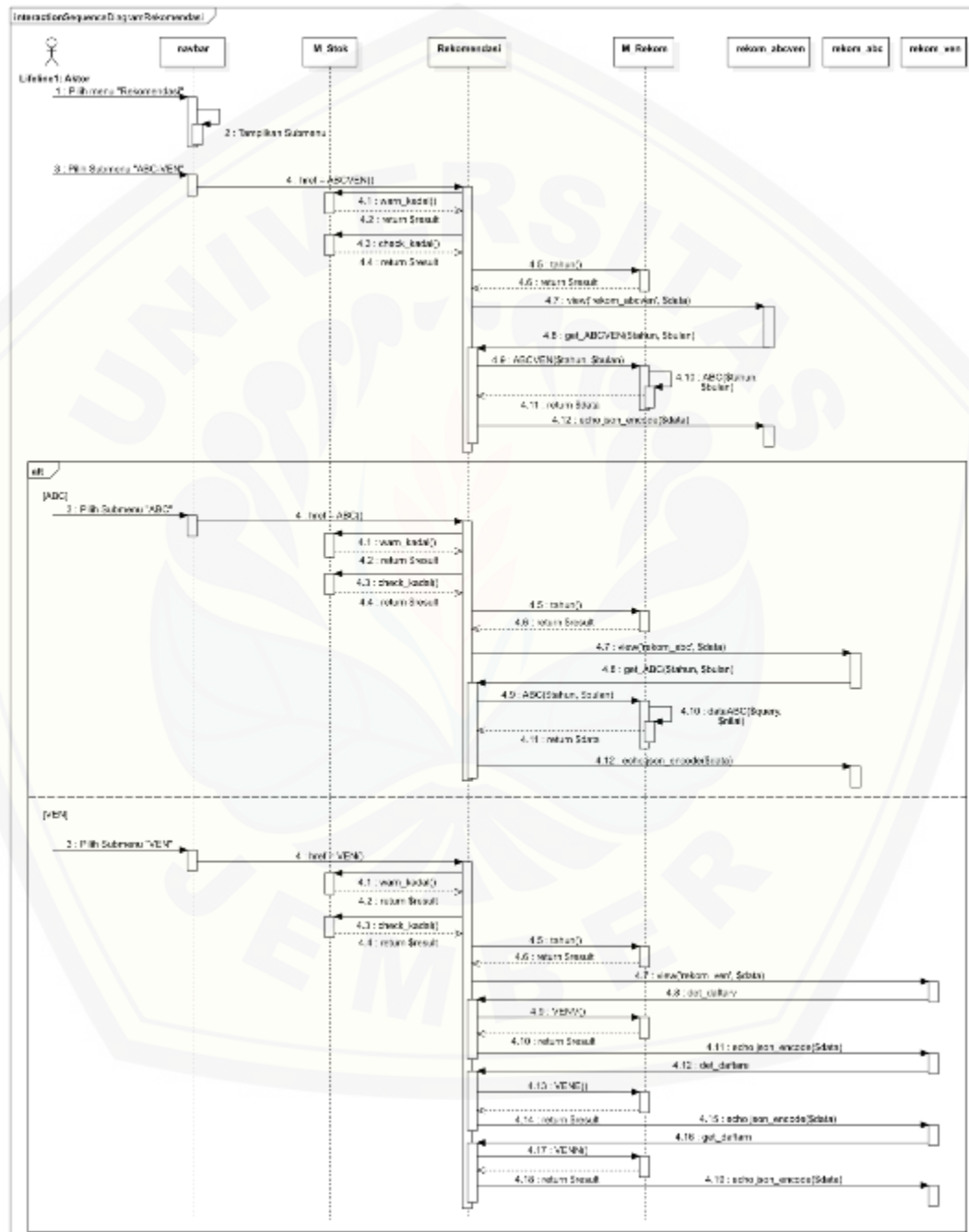
*Sequence Diagram* Melihat Data Pembuangan Obat yang menggambarkan alur interaksi antar objek dalam *use case* Melihat Data Pembuangan Obat terdapat pada Lampiran

13. *Sequence Diagram* Menambah Data Pembuangan Obat

*Sequence Diagram* Menambah Data Pembuangan Obat yang menggambarkan alur interaksi antar objek dalam *use case* Menambah Data Pembuangan Obat terdapat pada Lampiran C.

14. *Sequence Diagram* Melihat Rekomendasi

*Sequence Diagram* Melihat Rekomendasi yang menggambarkan alur interaksi antar objek dalam *use case* Melihat Rekomendasi terdapat pada Gambar 4.5.



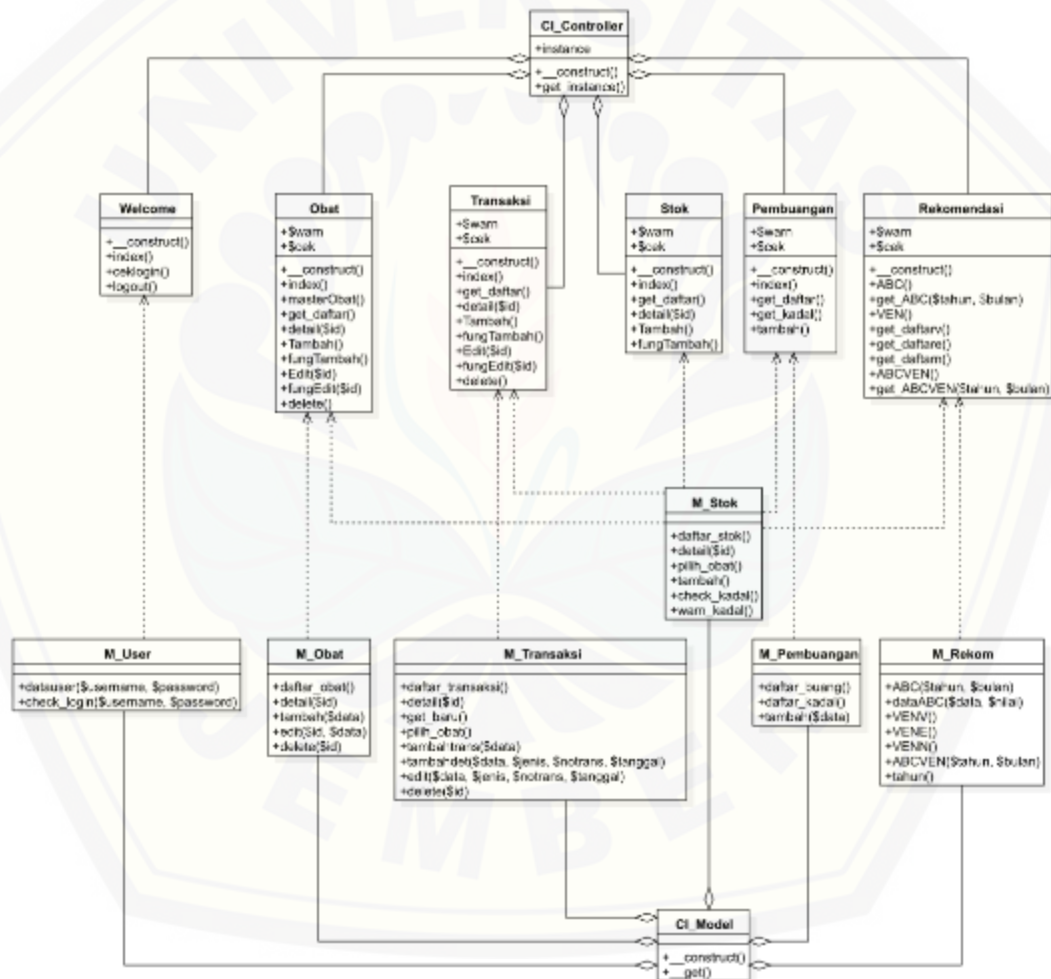
Gambar 4. 5 *Sequence Diagram* Rekomendasi

15. *Sequence Diagram Logout*

*Sequence Diagram Logout* yang menggambarkan alur interaksi antar objek dalam *use case* Logout terdapat pada Lampiran C.

4.2.7. *Class Diagram*

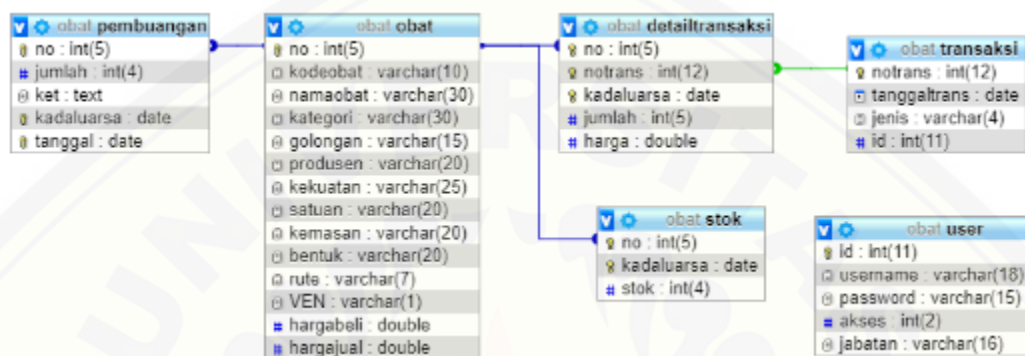
*Class Diagram* merupakan model diagram yang menggambarkan hubungan antar *class* pada sistem secara keseluruhan class diagram dapat dilihat pada gambar 4.6.



Gambar 4. 6 *Class Diagram*

#### 4.2.8. Entity Relationship Diagram

*Entity Relationship Diagram* (ERD) merupakan gambaran dari komponen-komponen yang terdapat dalam database sistem. *Entity Relationship Diagram* juga menggambarkan relasi yang terbentuk antar table sehingga setiap table dapat saling berhubungan satu sama lain.



Gambar 4. 7 Entity Relationship Diagram (ERD)

### 4.3. Penulisan Kode Program

Tahap ini merupakan proses pembuatan aplikasi, yaitu merupakan proses implementasi dari hasil analisa kebutuhan yang dilakukan pada tahap perencanaan, analisa dan desain sistem. Penulisan kode program dilakukan menggunakan bahasa pemrograman PHP dengan *framework Codeigniter*.

#### 4.3.1. Kode Program Rekomendasi

Kode program rekomendasi merupakan kode program yang berkaitan dengan penerapan fitur rekomendasi. Fitur rekomendasi berjalan dengan menggunakan *Class Controller* Rekomendasi dan *Class Model M\_Rekom*.

Tabel 4. 5 Kode Program pada Class Controller Rekomendasi.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Rekomendasi extends CI_Controller {
5

```

---

```

6  /**
7   * Index Page for this controller.
8   *
9   * Maps to the following URL
10  *   http://example.com/index.php/welcome
11  * - or -
12  *   http://example.com/index.php/welcome/index
13  * - or -
14  * Since this controller is set as the default controller in
15  * config/routes.php, it's displayed at http://example.com/
16  *
17  * So any other public methods not prefixed with an underscore will
18  * map to /index.php/welcome/<method_name>
19  * @see https://codeigniter.com/user_guide/general/urls.html
20  */
21 private $warn;
22 private $cek;
23 public function __construct(){
24     parent::__construct();
25     if(!$this->session->userdata('logged')) {
26         redirect(site_url());
27     }
28     $this->load->model('m_rekom', 'mrekom');
29     $this->load->model('M_Stok', 'mstok');
30     $this->warn=$this->mstok->warn_kadal()->result_array();
31     $this->cek=$this->mstok->check_kadal()->result_array();
32     }
33     public function index(){
34         redirect(site_url('rekomendasi/ABCVEN'));
35     }
36     public function ABC(){
37         $data['cek']=$this->cek;
38         $data['warn']=$this->warn;
39         $data['tahun']=$this->mrekom->tahun();
40         $this->load->view('rekom_abc', $data);
41     }
42     public function get_ABC($tahun, $bulan){
43         $data=$this->mrekom->ABC($tahun, $bulan);
44         echo json_encode($data);
45     }
46     public function VEN(){
47         $data['cek']=$this->cek;
48         $data['warn']=$this->warn;
49         $this->load->view('rekom_ven', $data);
50     }
51     public function get_daftarv(){

```

---

---

```

52     $data=$this->mrekom->VENV()->result();
53         echo json_encode($data);
54     }
55     public function get_daftare(){
56     $data=$this->mrekom->VENE()->result();
57         echo json_encode($data);
58     }
59     public function get_daftarn(){
60     $data=$this->mrekom->VENN()->result();
61         echo json_encode($data);
62     }
63     public function ABCVEN(){
64     $data['cek']=$this->cek;
65     $data['warn']=$this->warn;
66     $data['tahun']=$this->mrekom->tahun();
67         $this->load->view('rekom_abcven', $data);
68     }
69     public function get_ABCVEN($tahun, $bulan){
70         $data=$this->mrekom->ABCVEN($tahun, $bulan);
71     echo json_encode($data);
72     }
73 }
74

```

---

Tabel 4.4 merupakan kode program yang terdapat pada Rekomendasi.php yang berisi beberapa fungsi yang digunakan untuk mengakses fitur melihat rekomendasi pada sistem. Fungsi \_\_construct() merupakan fungsi yang dijalankan saat salah satu fungsi pada Rekomendasi.php dipanggil. Fungsi index() merupakan fungsi yang akan dijalankan ketika Rekomendasi dipanggil tanpa memanggil fungsi tertentu. Fungsi ABC() digunakan untuk menampilkan halaman rekomendasi ABC, sedangkan fungsi get\_ABC() digunakan untuk menampilkan hasil perhitungan berdasarkan metode ABC. Fungsi VEN() digunakan untuk menampilkan halaman klasifikasi VEN dan fungsi yang digunakan untuk menampilkan hasil klasifikasi berdasarkan metode VEN adalah fungsi get\_daftarv(), get\_daftare() dan get\_daftarn(). Fungsi ABCVEN digunakan untuk menampilkan halaman rekomendasi ABCVEN, sedangkan fungsi get\_ABCVEN() digunakan untuk menampilkan hasil perhitungan berdasarkan metode ABCVEN.

Tabel 4. 6 Kode Program pada Class Model M\_Rekom.php

```

1 <?php
2
3 if(!defined('BASEPATH')) exit ('No direct script access allowed');
4
5 class M_Rekom extends CI_Model{
6
7     public function ABC($tahun, $bulan){
8         $query1 = "SELECT no, SUM(jumlah) as pakai FROM
detailtransaksi JOIN transaksi on detailtransaksi.notrans =
transaksi.notrans WHERE MONTH(tanggaltrans) = ".$bulan." AND
YEAR(tanggaltrans) = ".$tahun." AND jenis = 'jual' GROUP BY no
ORDER BY pakai DESC";
9         $pakai = $this->dataABC($query1, 'pakai');
10        $query2 = "SELECT no, SUM(harga*jumlah) as inves FROM
detailtransaksi JOIN transaksi on detailtransaksi.notrans =
transaksi.notrans WHERE MONTH(tanggaltrans) = ".$bulan." AND
YEAR(tanggaltrans) = ".$tahun." AND jenis = 'jual' GROUP BY no
ORDER BY inves DESC";
11        $inves = $this->dataABC($query2, 'inves');
12        $query3 = "SELECT obat.no, namaobat, VEN FROM obat join
detailtransaksi on obat.no = detailtransaksi.no JOIN transaksi on
detailtransaksi.notrans = transaksi.notrans WHERE
MONTH(tanggaltrans) = ".$bulan." AND YEAR(tanggaltrans) =
".$tahun." AND jenis = 'jual' GROUP BY obat.no";
13        $result = $this->db->query($query3);
14        $data = array();
15        $h=0;
16        foreach ($result->result_array() as $v) {
17            $data[$h]['no'] = $v['no'];
18            $data[$h]['namaobat'] = $v['namaobat'];
19            $data[$h]['VEN'] = $v['VEN'];
20            $data[$h]['komulatifpakai'] = $pakai[$h]['komulatifpakai'];
21            $data[$h]['persenpakai'] = $pakai[$h]['persenpakai'];
22            $data[$h]['komulatifpersenpakai'] =
$pakai[$h]['komulatifpersenpakai'];
23            $data[$h]['kelaspakai'] = $pakai[$h]['kelaspakai'];
24            $data[$h]['komulatifinves'] = $inves[$h]['komulatifinves'];
25            $data[$h]['perseninves'] = $inves[$h]['perseninves'];
26            $data[$h]['komulatifperseninves'] =
$inves[$h]['komulatifperseninves'];
27            $data[$h]['kelasinves'] = $inves[$h]['kelasinves'];
28            if ($pakai[$h]['nilaipakai']+$inves[$h]['nilaiinves']>4) {
29                $data[$h]['kelaskritis']='A';
30            } else if ($pakai[$h]['nilaipakai']+$inves[$h]['nilaiinves']<4) {
31                $data[$h]['kelaskritis']='C';

```



---

```

32     } else {
33         $data[$h]['kelaskritis']='B';
34     }
35     $h++;
36 }
37 usort($data, function($a, $b) {
38     return $a['kelaskritis'] <=> $b['kelaskritis'];
39 });
40 return $data;
41 }
42 private function dataABC($query, $nilai){
43     $result = $this->db->query($query);
44     $data = array();
45     $h=0;
46     $temp=0;
47     foreach ($result->result_array() as $v) {
48         $data[$h]['no'] = $v['no'];
49         $data[$h][$nilai] = $v[$nilai];
50         $temp+=$v[$nilai];
51         $data[$h]['komulatif'].$nilai] = $temp;
52         $h++;
53     }
54     for ($i=0; $i < count($data); $i++) {
55         $data[$i]['persen'].$nilai] = $data[$i][$nilai] / $temp * 100;
56         $data[$i]['komulatifpersen'].$nilai] = $data[$i]['komulatif'].$nilai] /
57         $temp * 100;
58         if ($data[$i]['komulatifpersen'].$nilai]<=75) {
59             $data[$i]['kelas'].$nilai]='A';
60             $data[$i]['nilai'].$nilai]=3;
61         } else if ($data[$i]['komulatifpersen'].$nilai]>=95) {
62             $data[$i]['kelas'].$nilai]='C';
63             $data[$i]['nilai'].$nilai]=1;
64         } else {
65             $data[$i]['kelas'].$nilai]='B';
66             $data[$i]['nilai'].$nilai]=2;
67         }
68     }
69     usort($data, function($a, $b) {
70         return $a['no'] - $b['no'];
71     });
72     return $data;
73 }
74 public function VENV(){
75     $query = "SELECT kodeobat, namaobat FROM obat WHERE ven =
76     'V'";
77     $result = $this->db->query($query);

```

---

---

```

76     return $result;
77     }
78     public function VENE(){
79         $query = "SELECT kodeobat, namaobat FROM obat WHERE ven =
'E";
80         $result = $this->db->query($query);
81         return $result;
82     }
83     public function VENN(){
84         $query = "SELECT kodeobat, namaobat FROM obat WHERE ven =
'N";
85         $result = $this->db->query($query);
86         return $result;
87     }
88     public function ABCVEN($tahun, $bulan){
89         $result = $this->ABC($tahun, $bulan);
90         $data = array();
91         $h=0;
92         foreach ($result as $v) {
93             $data[$h]['no'] = $v['no'];
94             $data[$h]['namaobat'] = $v['namaobat'];
95             $data[$h]['VEN'] = $v['VEN'];
96             $data[$h]['ABC'] = $v['kelaskritis'];
97             if ($v['VEN']==='V' || $v['kelaskritis']==='A') {
98                 $data[$h]['ABCVEN']='I';
99             } else
100             if ($v['VEN']==='N' && $v['kelaskritis']==='C') {
101                 $data[$h]['ABCVEN']='III';
102             } else {
103                 $data[$h]['ABCVEN']='II';
104             }
105             $h++;
106         }
107         usort($data, function($a, $b) {
108             return $a['ABCVEN'] <=> $b['ABCVEN'];
109         });
110         return $data;
111     }
112     public function tahun(){
113         $query = "SELECT DISTINCT YEAR(tanggaltrans) as tahun FROM
transaksi WHERE jenis = 'jual";
114         $result = $this->db->query($query);
115         return $result;
116     }
117 }
118

```

---

Tabel 4.5 merupakan kode program yang terdapat pada M\_Rekom.php yang berisi beberapa fungsi yang digunakan untuk melakukan perhitungan rekomendasi berdasarkan metode ABC-VEN pada sistem. Fungsi ABC() digunakan untuk melakukan perhitungan nilai kritis dari metode ABC. Fungsi dataABC() digunakan untuk melakukan perhitungan nilai pakai dan nilai investasi dari metode ABC ketika dipanggil oleh fungsi ABC(). Fungsi VENV() digunakan untuk mendapatkan daftar obat yang masuk dalam kelompok V dari Metode VEN. Fungsi VENE() digunakan untuk mendapatkan daftar obat yang masuk dalam kelompok E dari Metode VEN. Fungsi VENN() digunakan untuk mendapatkan daftar obat yang masuk dalam kelompok N dari Metode VEN. Fungsi ABCVEN() digunakan untuk memanggil perhitungan metode ABC dan akhirnya mengklasifikasikan obat berdasarkan metode ABC-VEN. Fungsi tahun() digunakan untuk mendapatkan daftar tahun adanya data transaksi.

#### 4.3.2. Kode Program Pengguna

Kode program pengguna merupakan kode program yang berkaitan dengan penerapan fitur-fitur yang berhubungan dengan pengguna seperti fitur *login* dan *logout*. Kode program pengguna terdiri dari *Class Controller* Welome dan *Class Model* M\_User. Penjelasan lebih lanjut mengenai penulisan kode program pengguna dapat dilihat pada lampiran D.

#### 4.3.3. Kode Program Data Obat

Kode program pengguna merupakan kode program yang berkaitan dengan penerapan fitur-fitur yang berhubungan dengan pengelolaan obat. Kode program pengguna terdiri dari *Class Controller* Obat dan *Class Model* M\_Obat. Penjelasan lebih lanjut mengenai penulisan kode program data obat dapat dilihat pada lampiran D.

#### 4.3.4. Kode Program Stok Obat

Kode program pengguna merupakan kode program yang berkaitan dengan penerapan fitur-fitur yang berhubungan dengan data stok. Kode program pengguna terdiri dari *Class Controller* Stok dan *Class Model* M\_Stok. Penjelasan lebih lanjut mengenai penulisan kode stok obat pengguna dapat dilihat pada lampiran D.

#### 4.3.5. Kode Program Data Transaksi

KoKode program pengguna merupakan kode program yang berkaitan dengan penerapan fitur-fitur yang berhubungan dengan data transaksi. Kode program pengguna terdiri dari *Class Controller* Transaksi dan *Class Model* M\_Transaksi. Penjelasan lebih lanjut mengenai penulisan kode program data transaksi dapat dilihat pada lampiran D.

#### 4.3.6. Kode Program Pembuangan

Kode program pengguna merupakan kode program yang berkaitan dengan penerapan fitur-fitur yang berhubungan dengan pembuangan. Kode program pengguna terdiri dari *Class Controller* Pembuangan dan *Class Model* M\_Pembuangan. Penjelasan lebih lanjut mengenai penulisan kode program pembuangan dapat dilihat pada lampiran D.

### 4.4. Pengujian Sistem

Pengujian dan evaluasi dimaksudkan untuk mengetahui sejauh mana sistem yang dibuat ini dapat berfungsi sesuai dengan proses transaksi dalam optimasi distribusi ini nanti sesuai dengan yang diharapkan oleh pada pengguna. Pada tahap ini dilakukan ujicoba terhadap sistem optimasi untuk menemukan kesalahan-kesalahan yang mungkin terjadi serta melakukan perbaikan untuk lebih menyempurnakan kinerja *website* tersebut. Pengujian dilakukan dengan 2 metode, yaitu *White Box* dan *Black Box*.

4.4.1. *White Box Testing*

Pengujian menggunakan pengujian jalur dasar (*bash path testing*) yang didalamnya terdapat beberapa tahapan pengujian antara lain pembuatan diagram alir, penghitungan kompleksitas siklomatik, dan *test case* dengan penjelasan sebagai berikut

1. Diagram alir merupakan notasi yang digunakan untuk merepresentasikan aliran control yang digambarkan dari hasil penomoran dari listing program. Diagram alir digambarkan dengan node (simpul) yang dihubungkan dengan edge (garis).
2. Kompleksitas siklomatik merupakan metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logis suatu program. Kompleksitas siklomatik mendefinisikan jumlah jalur independen dalam suatu fungsi dengan menggunakan rumus (6).

$$V(G) = E - N + 2 \dots\dots\dots (6)$$

Di mana:

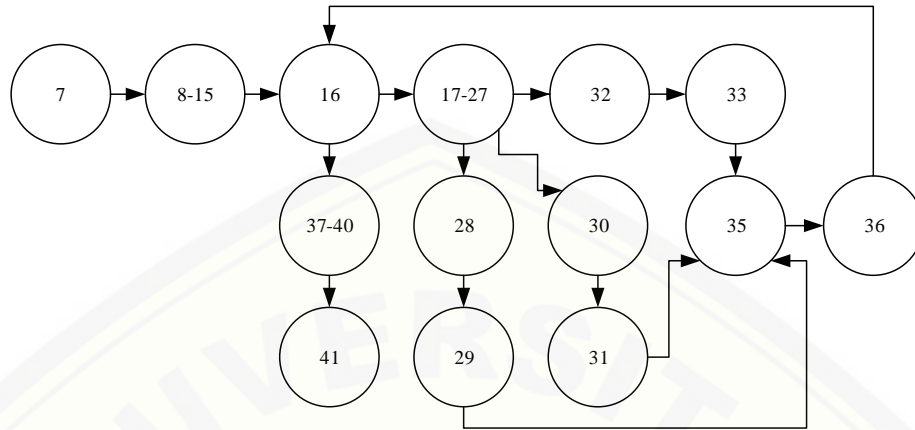
- $V(G)$  : Kompleksitas siklomatik
- $E$  : Jumlah edge (garis)
- $N$  : Jumlah node (simpul)

3. *Basis path* merupakan penentuan jalur independen yang bisa ditempuh
4. *Test case* merupakan pengujian masing-masing jalur independen yang telah didefinisikan

4.4.1.1. Fungsi ABC

Pengujian white-box dari fungsi ABC pada *class M\_Rekom*

- Diagram alir



Gambar 4. 8 Diagram Alir Fungsi ABC

- Kompleksitas siklomatik

$$V(G) = E - N + 2$$

$$V(G) = 16 - 14 + 2$$

$$V(G) = 4$$

- Basis path

Jalur 1:

7-(8-15)-16-(37-40)-41

Jalur 2 :

7-(8-15)-16-(17-27)-28-29-35-36-16-(37-40)-41

Jalur 3:

7-(8-15)-16-(17-27)-30-31-35-36-16-(37-40)-41

Jalur 4:

7-(8-15)-16-(17-27)-32-33-35-36-16-(37-40)-41

- Test case

Tabel 4. 7 Test case fungsi ABC()

Jalur 1

Test Case	Tidak melakukan loop
Target yang diharapkan	Data kosong
Hasil pengujian	Benar
<b>Jalur 2</b>	
Test Case	Bila item memiliki kelas A dari nilai pakai dan kelas A dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas A
Hasil pengujian	Benar
<b>Jalur 2</b>	
Test Case	Bila item memiliki kelas A dari nilai pakai dan kelas B dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas A
Hasil pengujian	Benar
<b>Jalur 2</b>	
Test Case	Bila item memiliki kelas B dari nilai pakai dan kelas A dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas A
Hasil pengujian	Benar
<b>Jalur 3</b>	
Test Case	Bila item memiliki kelas C dari nilai pakai dan kelas C dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas C
Hasil pengujian	Benar
<b>Jalur 3</b>	

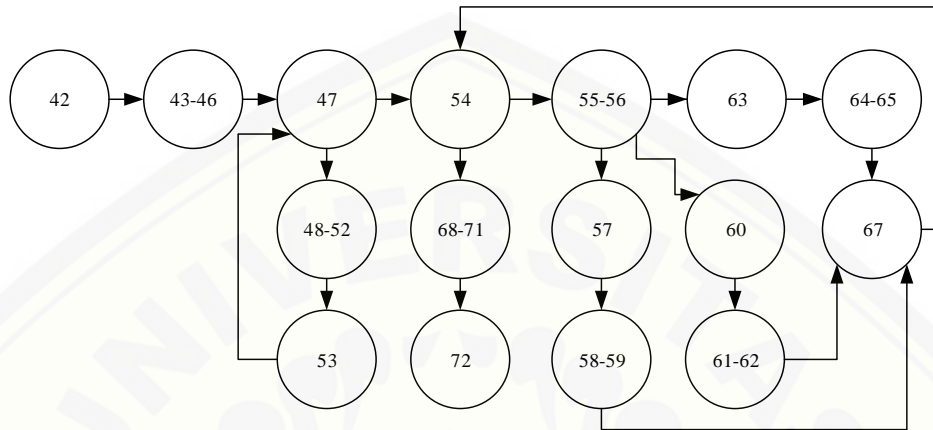
Test Case	Bila item memiliki kelas B dari nilai pakai dan kelas C dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas C
Hasil pengujian	Benar
<b>Jalur 3</b>	
Test Case	Bila item memiliki kelas C dari nilai pakai dan kelas B dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas C
Hasil pengujian	Benar
<b>Jalur 4</b>	
Test Case	Bila item memiliki kelas B dari nilai pakai dan kelas B dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas B
Hasil pengujian	Benar
<b>Jalur 4</b>	
Test Case	Bila item memiliki kelas A dari nilai pakai dan kelas C dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas B
Hasil pengujian	Benar
<b>Jalur 4</b>	
Test Case	Bila item memiliki kelas C dari nilai pakai dan kelas A dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas B
Hasil pengujian	Benar



4.4.1.2. Fungsi dataABC

Pengujian white-box dari fungsi dataABC pada *class* M\_Rekom

- Diagram alir



Gambar 4. 9 Diagram Alir Fungsi dataABC

- Kompleksitas siklomatik

$$V(G) = E - N + 2$$

$$V(G) = 19 - 16 + 2$$

$$V(G) = 5$$

- Basis path

Jalur 1:

42-(43-46)-47-54-(68-71)-72

Jalur 2 :

42-(43-46)-47-(48-52)-53-47-54-(68-71)-72

Jalur 3:

42-(43-46)-47-(48-52)-53-47-54-57-(58-59)-67-54-(68-71)-72

Jalur 4:

42-(43-46)-47-(48-52)-53-47-54-60-(61-62)-67-54-(68-71)-72

Jalur 5:

42-(43-46)-47-(48-52)-53-47-54-63-(64-65)-67-54-(68-71)-72

- Test case

Tabel 4. 8 *Test case* fungsi dataABC()

Jalur 1	
Test Case	Tidak melakukan loop
Target yang diharapkan	Data kosong
Hasil pengujian	Benar
Jalur 2	
Test Case	Bila item memiliki kelas A dari nilai pakai dan kelas A dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas A
Hasil pengujian	Benar
Jalur 2	
Test Case	Bila item memiliki kelas A dari nilai pakai dan kelas B dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas A
Hasil pengujian	Benar
Jalur 2	
Test Case	Bila item memiliki kelas B dari nilai pakai dan kelas A dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas A
Hasil pengujian	Benar
Jalur 3	
Test Case	Bila item memiliki kelas C dari nilai pakai dan kelas C dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas C

Hasil pengujian	Benar
Jalur 3	
Test Case	Bila item memiliki kelas B dari nilai pakai dan kelas C dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas C
Hasil pengujian	Benar
Jalur 3	
Test Case	Bila item memiliki kelas C dari nilai pakai dan kelas B dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas C
Hasil pengujian	Benar
Jalur 4	
Test Case	Bila item memiliki kelas B dari nilai pakai dan kelas B dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas B
Hasil pengujian	Benar
Jalur 4	
Test Case	Bila item memiliki kelas A dari nilai pakai dan kelas C dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas B
Hasil pengujian	Benar
Jalur 4	
Test Case	Bila item memiliki kelas C dari nilai pakai dan kelas A dari nilai investasi
Target yang diharapkan	Kelas ABC nilsi kritis dari item adalah kelas B

---

Hasil pengujian	Benar
-----------------	-------

---

#### 4.4.1.3. Fungsi VENV

Pengujian white-box dari fungsi VENV pada *class* M\_Rekom

- Diagram alir



Gambar 4. 10 Diagram Alir Fungsi VENV

- Kompleksitas siklomatik

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2$$

$$V(G) = 1$$

- Basis path

Jalur 1:

73-(74-76)-77

- Test case

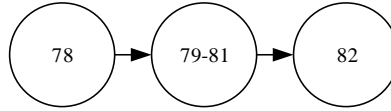
Tabel 4. 9 *Test case* fungsi VENV()

Jalur 1	
Test Case	Pengambilan data kelompok V
Target yang diharapkan	Data kelompok V
Hasil pengujian	Benar

#### 4.4.1.4. Fungsi VENE

Pengujian white-box dari fungsi VENE pada *class* M\_Rekom

- Diagram alir



Gambar 4. 11 Diagram Alir Fungsi VENE

- Kompleksitas siklomatik

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2$$

$$V(G) = 1$$

- Basis path

Jalur 1:

78-(79-81)-82

- Test case

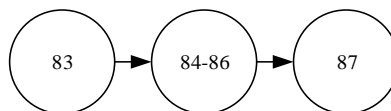
Tabel 4. 10 Test case fungsi VENE()

Jalur 1	
Test Case	Pengambilan data kelompok E
Target yang diharapkan	Data kelompok E
Hasil pengujian	Benar

#### 4.4.1.5. Fungsi VENN

Pengujian white-box dari fungsi VENN pada *class M\_Rekom*

- Diagram alir



Gambar 4. 12 Diagram Alir Fungsi VENN

- Kompleksitas siklomatik

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2$$

$$V(G) = 1$$

- Basis path

Jalur 1:

83-(84-86)-87

- Test case

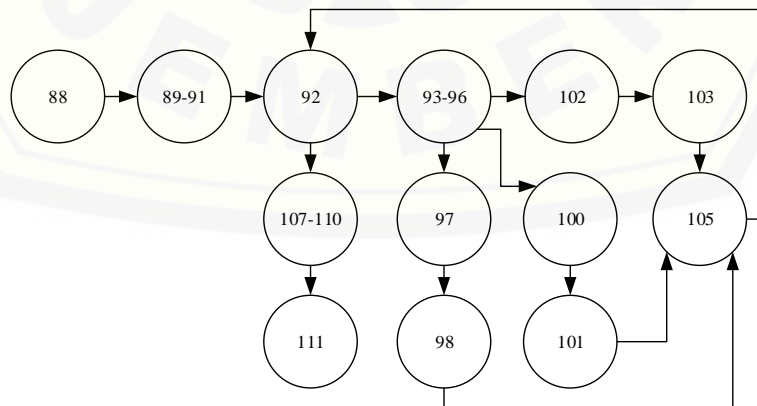
Tabel 4. 11 Test case fungsi VEN()

Jalur 1	
Test Case	Pengambilan data kelompok N
Target yang diharapkan	Data kelompok N
Hasil pengujian	Benar

#### 4.4.1.6. Fungsi ABCVEN

Pengujian white-box dari fungsi ABCVEN pada *class* M\_Rekom

- Diagram alir



Gambar 4. 13 Diagram Alir Fungsi ABCVEN

- Kompleksitas siklomatik

$$V(G) = E - N + 2$$

$$V(G) = 15 - 13 + 2$$

$$V(G) = 4$$

- Basis path

Jalur 1:

88-(89-91)-92-(107-110)-111

Jalur 2 :

88-(89-91)-92-(93-96)-97-98-105-92-(107-110)-111

Jalur 3:

88-(89-91)-92-(93-96)-100-101-105-92-(107-110)-111

Jalur 4:

88-(89-91)-92-(93-96)-102-103-105-92-(107-110)-111

- Test case

Tabel 4. 12 *Test case* fungsi ABCVEN()

Jalur 1	
Test Case	Tidak melakukan loop
Target yang diharapkan	Data kosong
Hasil pengujian	Benar
Jalur 2	
Test Case	Bila item memiliki kelas A dari analisis ABC dan kelas V dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori I
Hasil pengujian	Benar
Jalur 2	

Test Case	Bila item memiliki kelas A dari analisis ABC dan kelas E dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori I
Hasil pengujian	Benar
<b>Jalur 2</b>	
Test Case	Bila item memiliki kelas A dari analisis ABC dan kelas N dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori I
Hasil pengujian	Benar
<b>Jalur 2</b>	
Test Case	Bila item memiliki kelas B dari analisis ABC dan kelas V dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori I
Hasil pengujian	Benar
<b>Jalur 2</b>	
Test Case	Bila item memiliki kelas C dari analisis ABC dan kelas V dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori I
Hasil pengujian	Benar
<b>Jalur 3</b>	
Test Case	Bila item memiliki kelas C dari analisis ABC dan kelas N dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori III
Hasil pengujian	Benar
<b>Jalur 4</b>	



Test Case	Bila item memiliki kelas B dari analisis ABC dan kelas E dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori II
Hasil pengujian	Benar
Jalur 4	
Test Case	Bila item memiliki kelas B dari analisis ABC dan kelas N dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori II
Hasil pengujian	Benar
Jalur 4	
Test Case	Bila item memiliki kelas C dari analisis ABC dan kelas E dari analisis VEN
Target yang diharapkan	Kategori ABC-VEN dari item adalah kategori II
Hasil pengujian	Benar

#### 4.4.1.7. Fungsi tahun

Pengujian white-box dari fungsi tahun pada *class* M\_Rekom

- Diagram alir



Gambar 4. 14 Diagram Alir Fungsi tahun

- Kompleksitas siklomatik

$$V(G) = E - N + 2$$

$$V(G) = 2 - 3 + 2$$

$$V(G) = 1$$

- Basis path

Jalur 1:

112-(113-115)-116

- Test case

Tabel 4. 13 *Test case* fungsi tahun()

Jalur 1	
Test Case	Pengambilan data tahun dari transaksi yang ada
Target yang diharapkan	Daftar data tahun transaksi
Hasil pengujian	Benar

#### 4.4.2. *Black Box Testing*

Pengujian *black box* dilakukan untuk mengetahui apakah *input* dan *output* dari sistem sesuai dengan kebutuhan atau tidak. Pengujian dilakukan pada setiap *use case* berdasarkan sudut pandang pengguna. Hasil pengujian Black Box dapat dilihat pada tabel 4.14 dan Lampiran E.

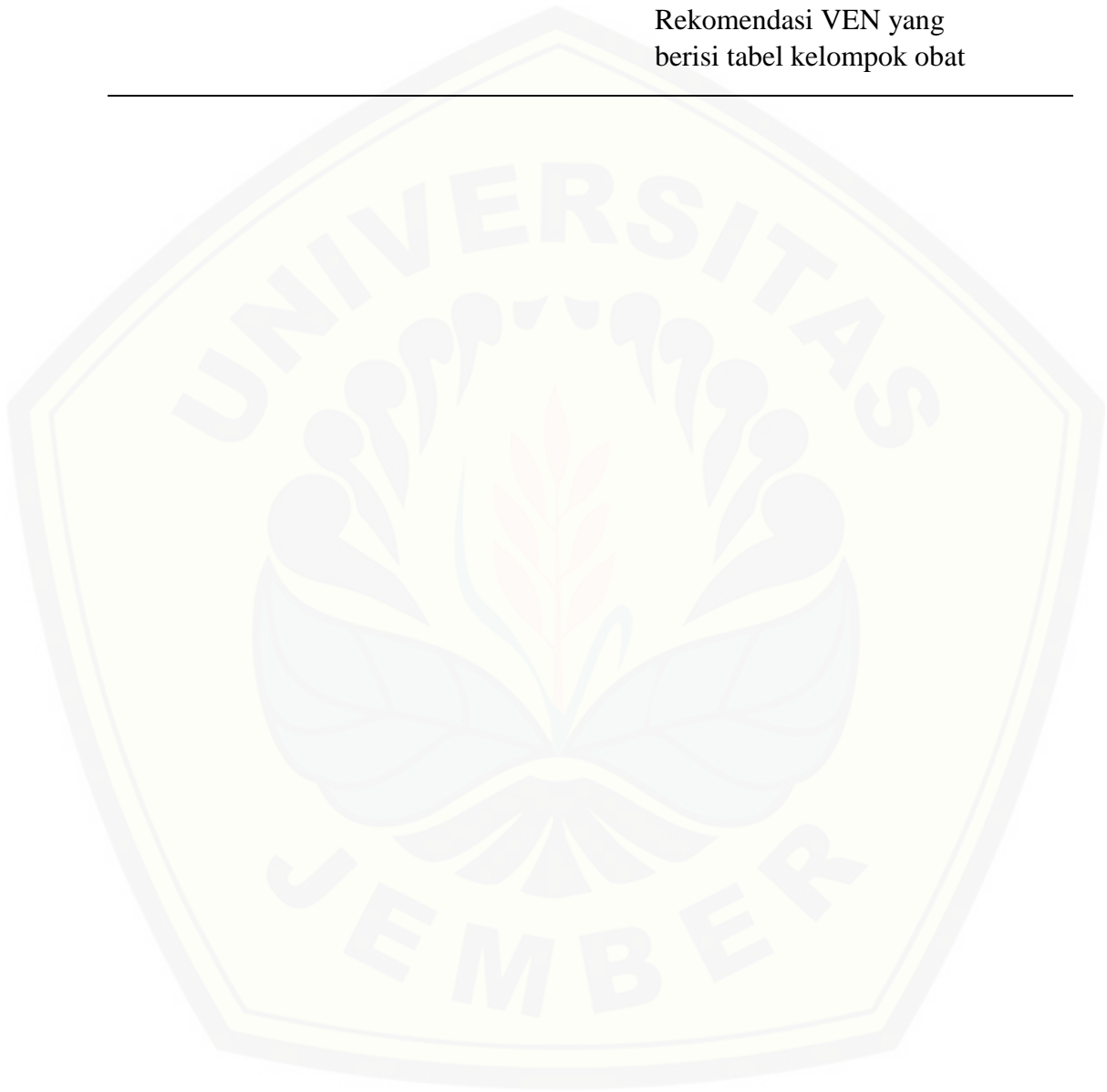
Tabel 4. 14 Hasil Pengujian Black Box

No.	Menu	Aksi	Hasil	Ket
1.	Melihat Rekomendasi	Pilih menu ABC-VEN	Menampilkan Halaman Rekomendasi ABC-VEN berdasarkan perhitungan bulan lalu	√
		Pilih tahun dan bulan pada halaman rekomendasi ABC-VEN	Menampilkan tabel prioritas obat berdasarkan perhitungan pada bulan dan tahun yang dipilih	√
		Pilih menu ABC	Menampilkan Halaman Rekomendasi ABC berdasarkan perhitungan bulan lalu	√

---

Pilih tahun dan bulan pada halaman rekomendasi ABC	Menampilkan tabel prioritas obat berdasarkan perhitungan pada bulan dan tahun yang dipilih	√
Pilih menu VEN	Menampilkan Halaman Rekomendasi VEN yang berisi tabel kelompok obat	√

---



## BAB 6. PENUTUP

### 6.1. Kesimpulan

Berdasarkan pada hasil dan pembahasan penelitian maka dapat disimpulkan sebagai berikut:

1. Metode ABC-VEN diterapkan pada sistem dengan cara melakukan melakukan analisis berdasarkan metode ABC dan metode VEN secara bersamaan. Analisis menggunakan metode ABC dilakukan dengan menghitung nilai pakai dan nilai investasi dari suatu obat pada periode tertentu dengan menghitung variabel yang didapatkan yang berupa jumlah pemakaian dan harga obat per item dari data-data transaksi yang lalu digunakan untuk menghitung nilai kritis yang dijadikan acuan untuk menentukan kelas dari suatu obat. Pengelompokan obat dengan menggunakan metode VEN dilakukan oleh apoteker. Metode ABC-VEN baru bisa dilakukan bila pengelompokan obat dengan menggunakan metode ABC dan VEN telah selesai dilakukan. Berdasarkan pengujian yang telah dilakukan, hasil dari perhitungan menggunakan sistem sama dengan hasil dari perhitungan manual dan dapat disimpulkan bahwa penerapan metode pada sistem telah berhasil.
2. Pengembangan sistem metode ABC-VEN dilakukan dengan menggunakan model *waterfall*. Pengembangan sistem model *waterfall* meliputi tahap analisis, desain, implementasi dan pengujian. Tahap analisis merupakan tahap di mana kebutuhan sistem, baik fungsional maupun non fungsional, didefinisikan. Tahap desain dilakukan dengan melakukan perancangan sistem dengan membuat *input-ouput diagram*, *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *entity relationship diagram*. Tahap implementasi merupakan penerapan rancangan sistem menjadi kode program dengan menggunakan bahasa pemrograman *PHP*, *framework CodeIgniter* dan *database MySQL*. Sistem yang telah dibuat memiliki beberapa fitur yang

dapat diakses oleh pengguna, yaitu melihat data obat, menambah data obat, mengubah data obat, menghapus data obat, melihat data transaksi, menambah data transaksi, mengubah data transaksi, menghapus data transaksi menambah data stok awal, melihat data pembuangan obat, menambah data pembuangan obat, login dan logout. Tahap pengujian dilakukan dengan menggunakan dua cara yaitu *white box testing* dan *black box testing*.

## 6.2. Saran

Pengembangan sistem dapat dilakukan oleh peneliti selanjutnya dengan membuat perhitungan *safety stock*, *reorder point* ataupun dengan menggunakan metode manajemen persediaan per item seperti EOQ (Economic Order Quantity) karena metode ABC-VEN hanya melakukan pengelompokkan obat sehingga setiap kelompok dapat diperlakukan berbeda.

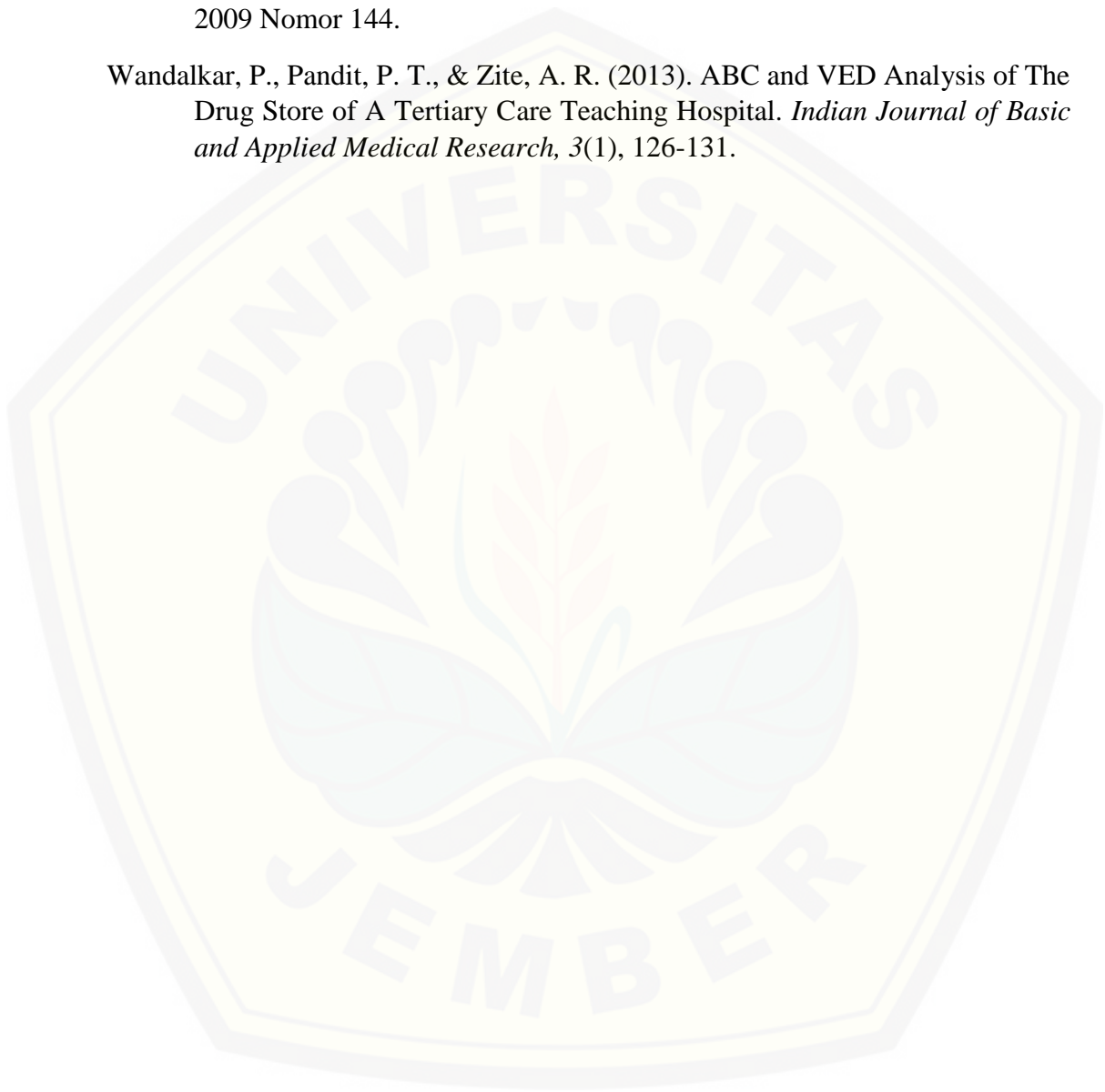
**DAFTAR PUSTAKA**

- Anshari. (2009). *Apilkasi Manajemen Pengelolaan Obat dan Makanan*. Yogyakarta: Nuha Media.
- Devnani, M., Gupta, A. K., & Nigah, R. (2010). ABC and VED Analysis of the Pharmacy Store of a Tertiary Care Teaching, Research and Referral Healthcare Institute of India. *J Young Pharm*, 2(2), 201-205.
- Hasibuan, M. S. (2007). *Manajemen Sumber Daya Manusia Perusahaan*. Bandung: PT. Bumi Aksa.
- Herjanto, E. (2015). *Manajemen Operasi* (Ketiga ed.). Jakarta: Grasindo.
- Kusnadi, E. (2009). *Analisis Produktivitas Terhadap Penyeimbangan Lintasan*. Jakarta: Program Studi Teknik Industri Fakultas Teknologi Universitas Mercu Buana.
- Ma'arif, M. S., & Tanjung, H. (2003). *Manajemen Operasi* (Pertama ed.). Jakarta: Grasindo.
- Maimun, A. (2008). *Perencanaan Obat Antibiotik Berdasarkan Kombinasi Metode Konsumsi dengan Analisis ABC dan Reorder Point terhadap Nilai Persediaan dan Turn Over Ratio di Instalasi Farmasi RS. DarulIstiqomah Kaliwungu Kendal*. Semarang: Universitas Diponegoro.
- Nurwulandari, A. (2013). *Sistem Pendukung Pengambilan Keputusan Pengadaan Obat Menggunakan Model Analisis Pareto ABC dan Optimasi Kualitatif*. Yogyakarta: Program Studi Informatika Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Sanata Dharma.
- Peraturan Menteri Kesehatan Republik Indonesia Nomor 9. (2014, Januari 19). *Tentang Klinik*. Jakarta: Berita Negara Republik Indonesia Tahun 2014 Nomor 232.
- Pressman, R. S. (2002). *Rekayasa Perangkat Lunak Pendekatan Praktisi* (Buku). Yogyakarta: ANDI.
- Quick, J. D., Dias, V., & Rankin, J. (2012). *Inventory Management in Managing Drug Supply. Third Edititon, Managing Access to Medicines and Health Technologies*. Arlington: Management Sciences for Health.
- Reddy, V. V. (2008). *Hospital Material Management In A.V. Srinivasan (Ed) (Managing a Modern Hospital (2nd Ed) ed.)*. New Delhi: Sage Publication.

Singh, V., Singh, H., & Singh, S. (2015). Drug Inventory Management of A Pharmacy Store by Combined AbcVed Analysis. *International Journal on Mechanical Engineering and Robotics (IJMER)*, 3(5), 19-22.

Undang-Undang Republik Indonesia Nomor 36 Tahun 2009. (2009, Oktober 13). *Tentang Kesehatan*. Jakarta: Lembaran Negara Republik Indonesia Tahun 2009 Nomor 144.

Wandalkar, P., Pandit, P. T., & Zite, A. R. (2013). ABC and VED Analysis of The Drug Store of A Tertiary Care Teaching Hospital. *Indian Journal of Basic and Applied Medical Research*, 3(1), 126-131.



LAMPIRAN

Lampiran A. Skenario

A.1. Skenario Login

No Use Case	USC 01
Nama Use Case	Login
Aktor	Apoteker, Asisten Apoteker
Prekondisi	Aktor sudah membuka halaman login
Prakondisi	Aktor sudah melakukan login

**SKENARIO NORMAL**  
**“Login”**

Aksi Aktor	Reaksi Sistem
1. Membuka Halaman Login	2. Menampilkan Halaman Login
3. Memasukkan Username dan Password	
4. Tekan tombol Login	5. Menampilkan halaman Beranda yang menampilkan <ul style="list-style-type: none"> <li>a. Resep yang diterima hari ini</li> <li>b. Resep yang diterima kemarin</li> <li>c. Resep yang diterima minggu ini</li> <li>d. Resep yang diterima bulan ini</li> <li>e. Resep yang diterima tahun ini</li> <li>f. Resep yang diterima rata-rata harian bulan lalu</li> <li>g. Peringatan obat kadaluarsa</li> </ul>

**SKENARIO ALTERNATIF**  
**“Username dan Password tidak diisi”**

Aksi Aktor	Reaksi Sistem
3. Tekan tombol Login	4. Menampilkan notifikasi “Please fill out this field”

**SKENARIO ALTERNATIF**  
**“Username dan Password salah”**

3. Memasukkan Username dan Password	
-------------------------------------	--



4. Tekan tombol Login	5. Menampilkan halaman login dengan tambahan pesan "Username atau Password yang anda masukkan salah"
-----------------------	--

A.2. Skenario Melihat Obat

No Use Case	USC 02
Nama Use Case	Lihat data obat
Aktor	Apoteker, Asisten Apoteker
Prekondisi	Aktor sudah melakukan login
Prakondisi	Aktor sudah melihat data obat

**SKENARIO NORMAL**  
**"Melihat data obat"**

Aksi Aktor	Reaksi Sistem
1. Pilih menu "Master Obat"	2. Menampilkan Halaman Master Obat yang berisi tabel data obat yang berisi : a. No b. Kode Obat c. Nama d. Kategori e. Golongan f. Kekuatan g. Aksi yang berisi Tombol Detail Serta tombol tambah, edit dan hapus untuk Apoteker

**SKENARIO Alternatif**  
**"Melihat detail data obat"**

3. Tekan tombol Detail pada baris obat yang dipilih	4. Menampilkan Halaman Detail Obat yang berisi : a. Kode Obat b. Nama Obat c. Kategori d. Golongan e. Produsen f. Kekuatan g. Satuan h. Kemasan i. Bentuk
---	--

- j. Rute
  - k. Harga Beli
  - l. Harga Jual
  - m. Kategori VEN
- Serta tombol edit untuk Apoteker

A.3. Skenario Menambah Obat

No Use Case	USC 03
Nama Use Case	Tambah data obat
Aktor	Apoteker
Prekondisi	Aktor sudah membuka Halaman Master Obat
Prakondisi	Aktor sudah menambah data obat

**SKENARIO NORMAL**  
**“Menambah data obat”**

Aksi Aktor	Reaksi Sistem
1. Tekan tombol Tambah	2. Menampilkan Halaman Form Obat Kosong yang berisi: <ul style="list-style-type: none"> <li>a. Kode Obat</li> <li>b. Nama Obat</li> <li>c. Kategori</li> <li>d. Golongan</li> <li>e. Produsen</li> <li>f. Kekuatan</li> <li>g. Satuan</li> <li>h. Kemasan</li> <li>i. Bentuk</li> <li>j. Rute</li> <li>k. Harga Beli</li> <li>l. Harga Jual</li> <li>m. Kategori VEN</li> </ul> Serta tombol Simpan, Reset dan Batal
3. Memasukkan data obat baru	
4. Tekan tombol Simpan	5. Menampilkan Halaman Detail Obat yang berisi data yang baru ditambahkan

**SKENARIO ALTERNATIF**  
**“Ada field penting yang tidak diisi”**

Aksi Aktor	Reaksi Sistem
3. Tekan tombol Simpan	

4. Menampilkan notifikasi “Please fill out this field”

**SKENARIO ALTERNATIF  
“Batal menambahkan data”**

3. Tekan tombol Batal

4. Menampilkan Halaman Master Obat yang berisi tabel data obat

A.4. Skenario Mengubah Obat

No Use Case	USC 04
Nama Use Case	Ubah data obat
Aktor	Apoteker
Prekondisi	Aktor sudah membuka Halaman Master Obat
Prakondisi	Aktor sudah mengubah data obat

**SKENARIO NORMAL  
“Mengubah data obat”**

Aksi Aktor	Reaksi Sistem
1. Tekan tombol Ubah pada baris obat yang dipilih	2. Menampilkan Halaman Form Obat berisi data yang ada dari obat yang dipilih yang berisi : a. Kode Obat b. Nama Obat c. Kategori d. Golongan e. Produsen f. Kekuatan g. Satuan h. Kemasan i. Bentuk j. Rute k. Harga Beli l. Harga Jual m. Kategori VEN Serta tombol Simpan, Reset dan Batal
3. Mengubah data obat	
4. Tekan tombol Simpan	5. Menampilkan Halaman Detail Obat yang berisi data yang baru diubah

<b>SKENARIO ALTERNATIF</b> <b>“Ada field penting yang tidak diisi”</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
3. Mengubah data obat	
4. Tekan tombol Simpan	
	5. Menampilkan notifikasi “Please fill out this field”

<b>SKENARIO ALTERNATIF</b> <b>“Batal mengubah data”</b>	
3. Tekan tombol Batal	
	5. Menampilkan Halaman Detail Obat yang berisi data lama

A.5. Skenario Menghapus Obat

No Use Case	USC 05
Nama Use Case	Hapus data obat
Aktor	Apoteker
Prekondisi	Aktor sudah membuka Halaman Master Obat
Prakondisi	Aktor sudah mengubah data obat

<b>SKENARIO NORMAL</b> <b>“Menghapus data obat”</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Tekan tombol Hapus pada baris obat yang dipilih	
	2. Menampilkan pesan “Apakah anda yakin untuk menghapus data obat ini?”
3. Tekan tombol Ya	
	4. Menampilkan Halaman Master Obat yang berisi tabel data obat

<b>SKENARIO ALTERNATIF</b> <b>“Batal menghapus data obat”</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
3. Tekan tombol Tidak	
	4. Menampilkan Halaman Master Obat yang berisi tabel data obat

A.6. Skenario Melihat Data Stok

No Use Case	USC 06
Nama Use Case	Lihat data stok obat
Aktor	Apoteker, Asisten Apoteker

Prekondisi	Aktor sudah melakukan login
Prakondisi	Aktor sudah melihat data stok obat
<b>SKENARIO NORMAL</b>	
<b>“Melihat data stok”</b>	

Aksi Aktor	Reaksi Sistem
1. Pilih menu “Stok Gudang”	2. Menampilkan Halaman Stok Obat yang berisi tabel data stok obat yang berisi : a. No b. Kode Obat c. Nama d. Produsen e. Kekuatan f. Jumlah g. Aksi yang berisi Tombol Detail Serta tombol tambah untuk Apoteker

<b>SKENARIO Alternatif</b>	
<b>“Melihat detail data stok obat tertentu”</b>	

3. Tekan tombol Detail pada baris obat yang dipilih	4. Menampilkan Halaman Detail Stok Obat yang berisi : a. Kode Obat b. Nama Obat c. Kategori d. Golongan e. Produsen f. Kekuatan
---	---

A.7. Skenario Menambah Data Stok

No Use Case	USC 07
Nama Use Case	Tambah data stok obat
Aktor	Apoteker
Prekondisi	Aktor sudah membuka Halaman Stok Gudang
Prakondisi	Aktor sudah menambah data stok

<b>SKENARIO NORMAL</b>	
<b>“Menambah data stok”</b>	

Aksi Aktor	Reaksi Sistem
1. Tekan tombol Tambah	

	2. Menampilkan Halaman Form Stok Baru yang berisi : a. Nama Obat b. Kadaluarsa c. Jumlah d. Aksi yang berisi Tombol Detail Serta tombol Simpan, Reset dan Batal
3. Memasukkan data Stok baru	
4. Tekan tombol Simpan	
	5. Menampilkan Halaman Stok Obat yang berisi tabel data stok obat

**SKENARIO ALTERNATIF  
"Ada field penting yang tidak diisi"**

Aksi Aktor	Reaksi Sistem
3. Tekan tombol Simpan	
	4. Menampilkan notifikasi "Please fill out this field"

**SKENARIO ALTERNATIF  
"Data sudah ada"**

Aksi Aktor	Reaksi Sistem
3. Tekan tombol Simpan	
	4. Menampilkan notifikasi "Stok obat ini sudah ada pada database"

**SKENARIO ALTERNATIF  
"Batal menambahkan data"**

3. Tekan tombol Batal	
	4. Menampilkan Halaman Stok Obat yang berisi tabel data stok obat

A.8. Skenario Melihat Data Transaksi

No Use Case	USC 08
Nama Use Case	Lihat data transaksi
Aktor	Apoteker, Asisten Apoteker
Prekondisi	Aktor sudah melakukan login
Prakondisi	Aktor sudah melihat data transaksi

**SKENARIO NORMAL  
"Melihat data transaksi"**

Aksi Aktor	Reaksi Sistem
1. Pilih menu "Transaksi"	
	2. Menampilkan Submenu
3. Pilih Daftar Transaksi	

4. Menampilkan Halaman Transaksi yang berisi tabel data transaksi
  - a. No
  - b. Tanggal Transaksi
  - c. Nilai Uang
  - d. Jenis
  - e. Oleh
  - f. Aksi yang berisi tombol Detail

Serta tombol tambah, edit dan hapus untuk Asisten Apoteker

**SKENARIO Alternatif  
“Melihat detail data transaksi tertentu”**

5. Tekan tombol Detail pada baris transaksi yang dipilih
6. Menampilkan Halaman Detail Transaksi yang berisi :
  - a. Tanggal Transaksi
  - b. Jenis Transaksi
  - c. Total Nilai
  - d. Nama Obat
  - e. Tanggal Kadaluarsa
  - f. Jumlah
  - g. Harga per item

A.9. Skenario Menambah Data Transaksi

No Use Case	USC 09
Nama Use Case	Tambah data transaksi
Aktor	Asisten Apoteker
Prekondisi	Aktor sudah membuka Halaman Transaksi
Prakondisi	Aktor sudah menambah data transaksi

**SKENARIO NORMAL  
“Menambah data transaksi”**

Aksi Aktor	Reaksi Sistem
1. Tekan tombol Tambah	2. Menampilkan pilihan
3. Pilih jenis transaksi	4. Menampilkan Halaman Form Transaksi yang berisi : <ol style="list-style-type: none"> <li>a. Tanggal Transaksi</li> <li>b. Jenis Transaksi</li> <li>c. Total Nilai</li> </ol>

	d. Nama Obat
	e. Tanggal Kadaluarsa
	f. Jumlah
	g. Harga per item
	h. Pilihan obat
	i. Tombol tambah obat
	Serta tombol Simpan, Reset dan Batal
5.	Memasukkan data transaksi baru
6.	Tekan tombol Simpan
	7. Menampilkan Halaman Detail Transaksi yang berisi data yang baru ditambahkan

**SKENARIO ALTERNATIF  
"Ada field penting yang tidak diisi"**

Aksi Aktor	Reaksi Sistem
5. Tekan tombol Simpan	6. Menampilkan notifikasi "Please fill out this field"

**SKENARIO ALTERNATIF  
"Batal menambahkan data"**

5. Tekan tombol Batal	6. Menampilkan Halaman Transaksi yang berisi tabel data transaksi
-----------------------	---

A.10. Skenario Mengubah Data Transaksi

No Use Case	USC 10
Nama Use Case	Ubah data transaksi
Aktor	Asisten Apoteker
Prekondisi	Aktor sudah membuka Halaman Transaksi
Prakondisi	Aktor sudah mengubah data transaksi

**SKENARIO NORMAL  
"Mengubah data transaksi"**

Aksi Aktor	Reaksi Sistem
1. Tekan tombol Ubah pada baris transaksi yang dipilih	2. Menampilkan Halaman Form Transaksi berisi data yang ada dari transaksi yang dipilih yang berisi : Transaksi yang berisi : a. Tanggal Transaksi b. Jenis Transaksi c. Total Nilai



	d. Nama Obat
	e. Tanggal Kadaluarsa
	f. Jumlah
	g. Harga per item
	h. Pilihan obat
	i. Tombol tambah obat
	Serta tombol Simpan, Reset dan Batal
3.	Mengganti data transaksi
4.	Tekan tombol Simpan
	5. Menampilkan Halaman Detail Transaksi yang berisi data yang baru diubah

**SKENARIO ALTERNATIF  
"Ada field penting yang tidak diisi"**

Aksi Aktor	Reaksi Sistem
3. Mengubah data transaksi	
4. Tekan tombol Simpan	
	5. Menampilkan notifikasi "Please fill out this field"

**SKENARIO ALTERNATIF  
"Batal mengubah data"**

3. Tekan tombol Batal	
	4. Menampilkan Halaman Detail Obat yang berisi data lama

A.11. Skenario Menghapus Data Transaksi

No Use Case	USC 11
Nama Use Case	Hapus data transaksi
Aktor	Asisten Apoteker
Prekondisi	Aktor sudah membuka Halaman Transaksi
Prakondisi	Aktor sudah mengubah data transaksi

**SKENARIO NORMAL  
"Menghapus data transaksi"**

Aksi Aktor	Reaksi Sistem
1. Tekan tombol Hapus pada baris transaksi yang dipilih	
	2. Menampilkan pesan "Apakah anda yang untuk menghapus data transaksi ini?"
3. Tekan tombol Ya	
	4. Menampilkan Halaman Transaksi yang berisi tabel data transaksi

<b>SKENARIO ALTERNATIF</b> <b>“Batal menghapus data transaksi”</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
3. Tekan tombol Tidak	4. Menampilkan Halaman Transaksi yang berisi tabel data transaksi

A.12. Skenario Melihat Data Pembuangan Obat

No Use Case	USC 12
Nama Use Case	Lihat data pembuangan obat
Aktor	Apoteker, Asisten Apoteker
Prekondisi	Aktor sudah melakukan login
Prakondisi	Aktor sudah melihat data pembuangan obat

<b>SKENARIO NORMAL</b> <b>“Melihat data pembuangan obat”</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Pilih menu “Pembuangan”	2. Menampilkan Halaman Pembuangan Obat yang berisi tabel data obat kadaluarsa yang berisi : a. No b. Kode Obat c. Nama d. Tanggal Kadaluarsa e. Jumlah f. Aksi yang berisi Tombol Buang dan tabel data pembuangan obat yang berisi : a. No b. Kode Obat c. Nama d. Kadaluarsa e. Jumlah f. Keterangan g. Tanggal Pembuangan

A.13. Skenario Menambah Data Pembuangan Obat

No Use Case	USC 13
Nama Use Case	Tambah data pembuangan obat
Aktor	Apoteker, Asisten Apoteker

Prekondisi	Aktor sudah membuka Halaman Pembuangan
Prakondisi	Aktor sudah menambah data pembuangan obat

**SKENARIO NORMAL**  
**“Menambah data pembuangan obat”**

Aksi Aktor	Reaksi Sistem
1. Tekan tombol Buang pada baris obat kadaluarsa yang dipilih	2. Menampilkan pesan “Apakah anda yakin untuk membuang obat ini?” beserta form pengisian data pembuangan yang berisi : a. Kode Obat b. Nama c. Kadaluarsa d. Jumlah e. Keterangan f. Tanggal Pembuangan
3. Memasukkan data pembuangan obat	
4. Tekan tombol Hapus	4. Menampilkan Halaman Pembuangan

**SKENARIO ALTERNATIF**  
**“Ada field penting yang tidak diisi”**

Aksi Aktor	Reaksi Sistem
3. Memasukkan data pembuangan obat	
4. Tekan tombol Simpan	5. Menampilkan notifikasi “Please fill out this field”

**SKENARIO ALTERNATIF**  
**“Batal menambah data pembuangan obat”**

Aksi Aktor	Reaksi Sistem
3. Tekan tombol Tutup	
	4. Menampilkan Halaman Pembuangan

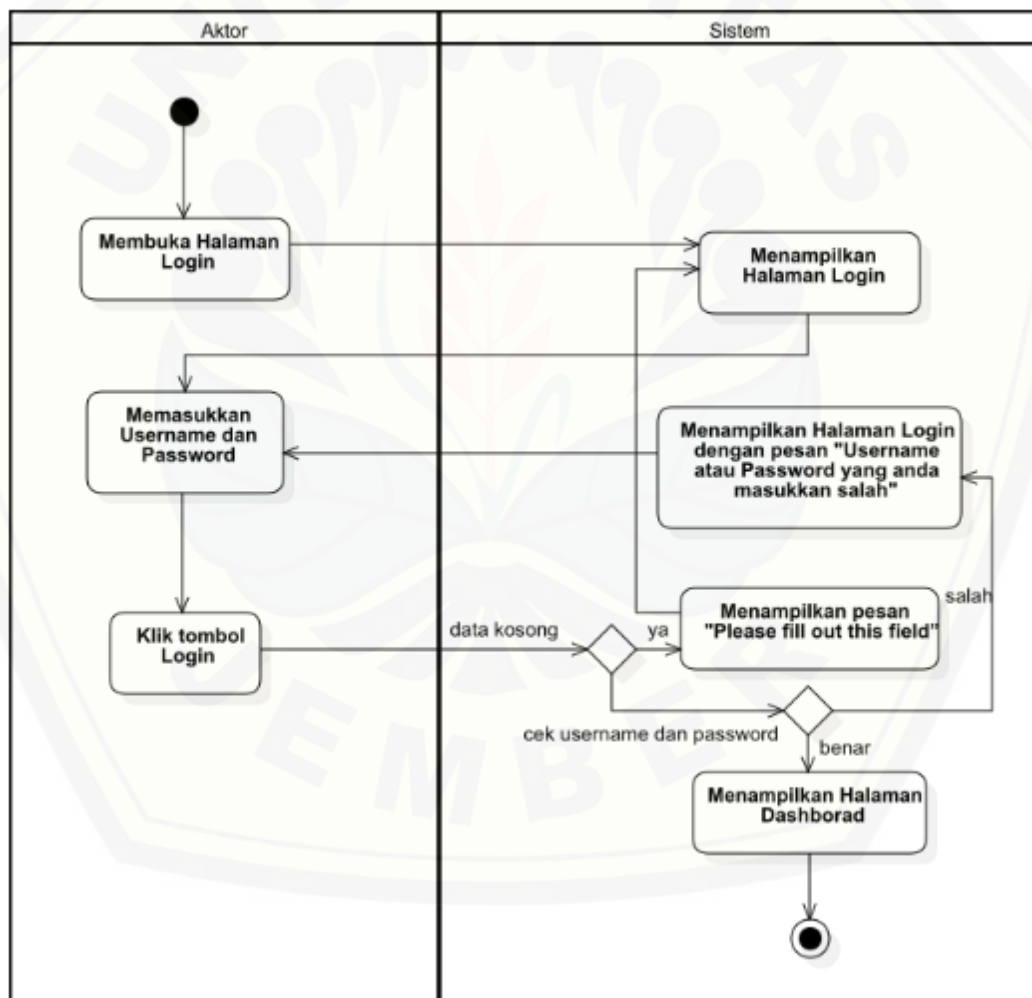
A.14. Skenario Logout

No Use Case	USC 15
Nama Use Case	Logout
Aktor	Apoteker, Asisten Apoteker

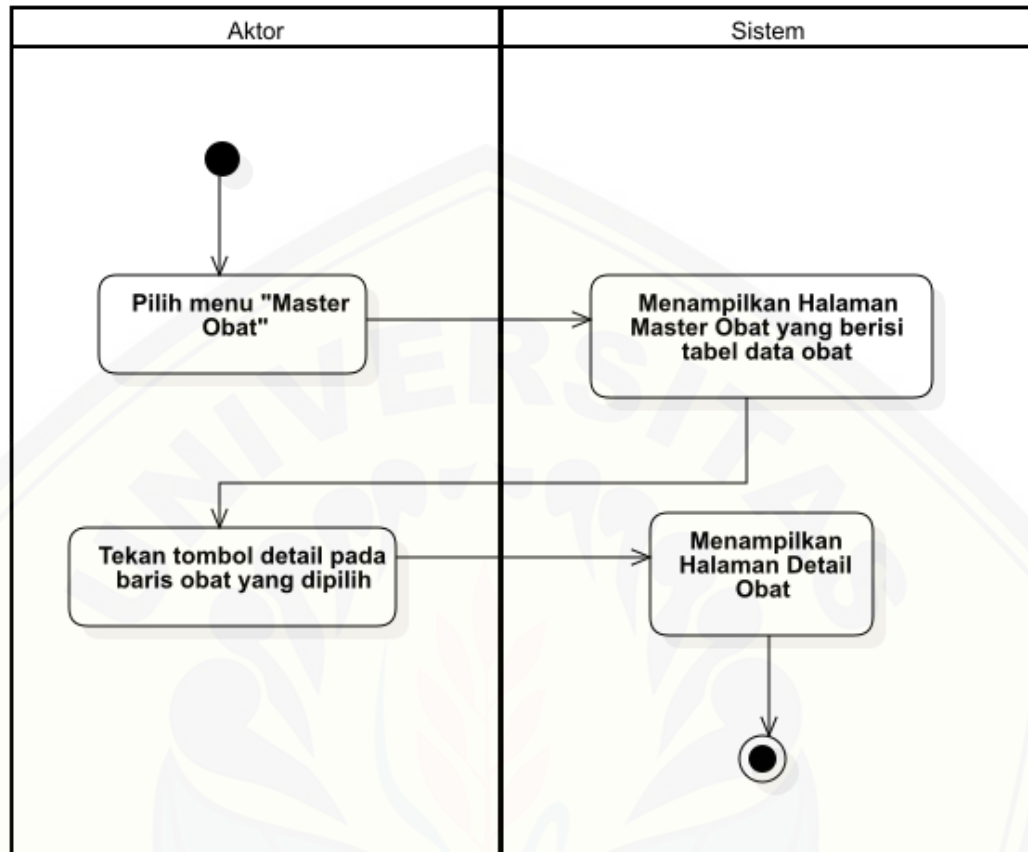
Prekondisi	Aktor sudah melakukan login pada sistem
Prakondisi	Aktor sudah melakukan logout
<b>SKENARIO NORMAL</b>	
<b>“Logout”</b>	
<b>Aksi Aktor</b>	<b>Reaksi Sistem</b>
1. Klik tombol Logout	2. Menampilkan Halaman Login

**Lampiran B. Activity Diagram**

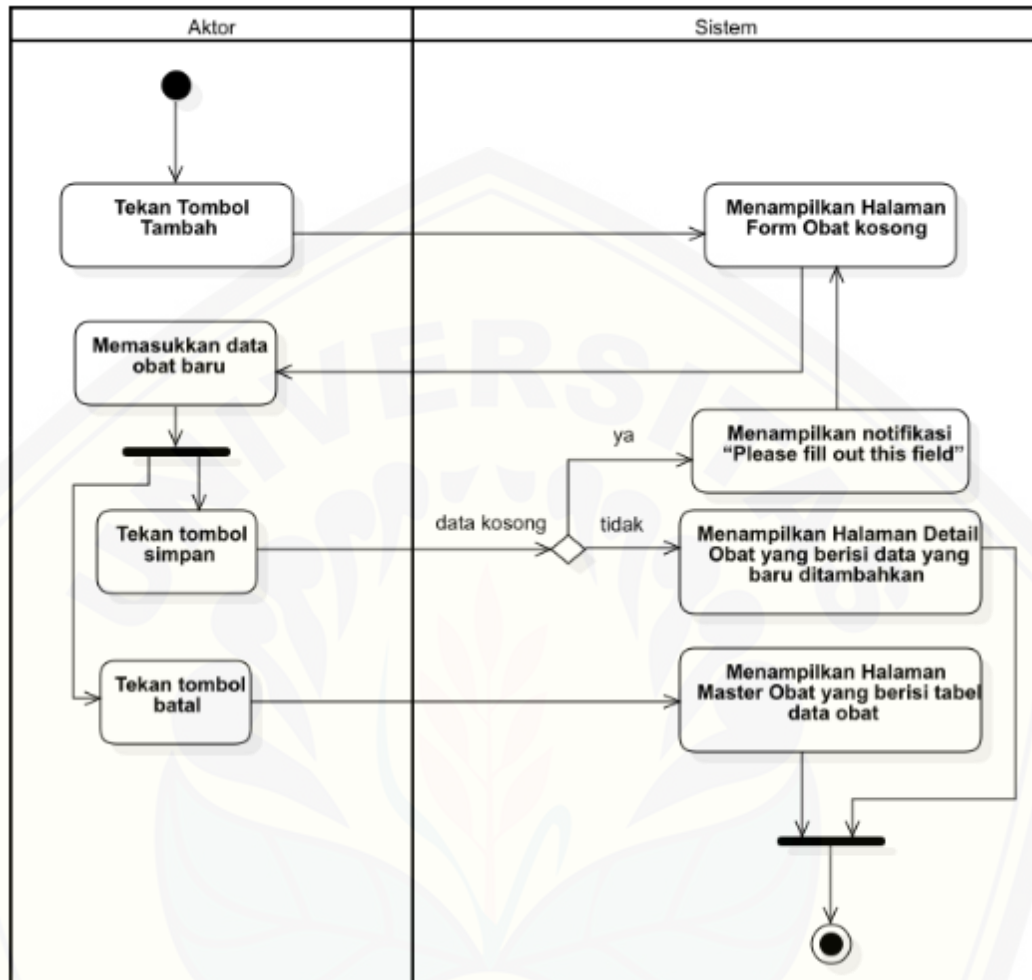
B.1. Activity Diagram Login



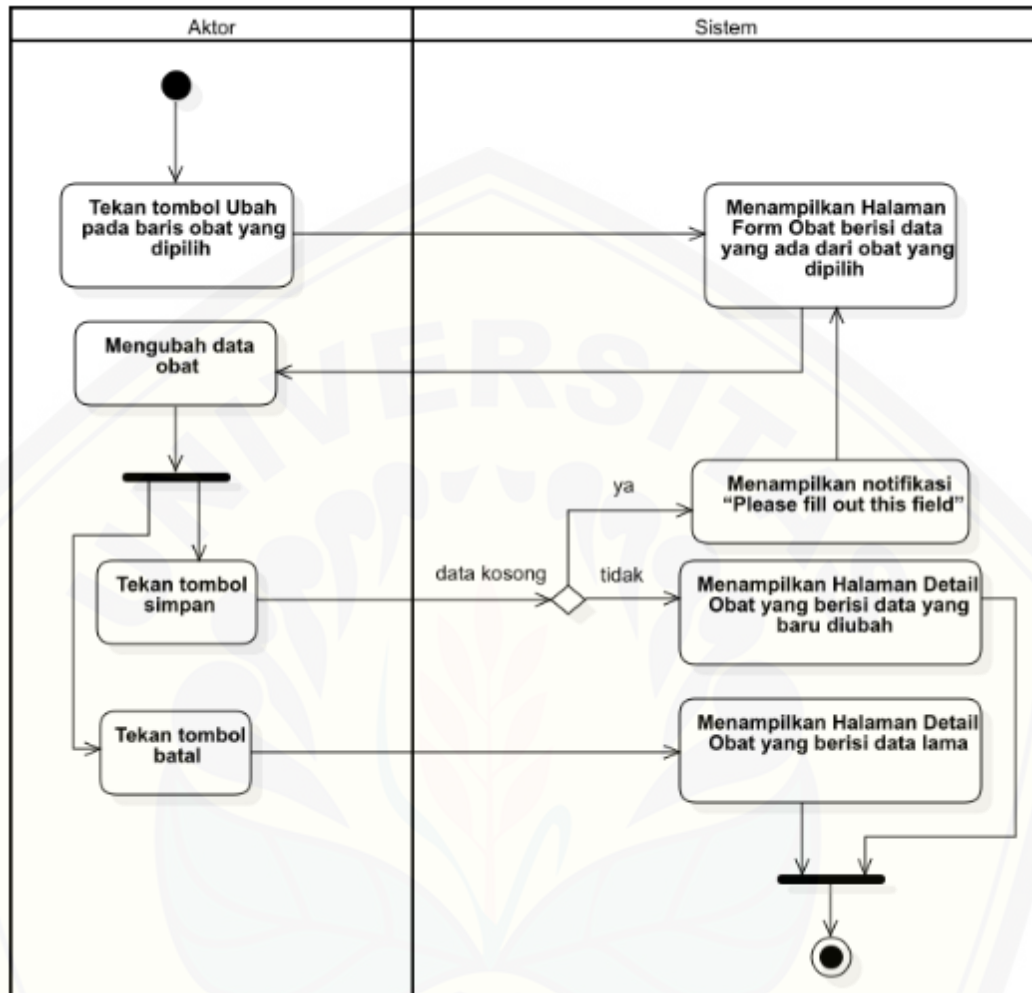
B.2. Activity Diagram Melihat Obat



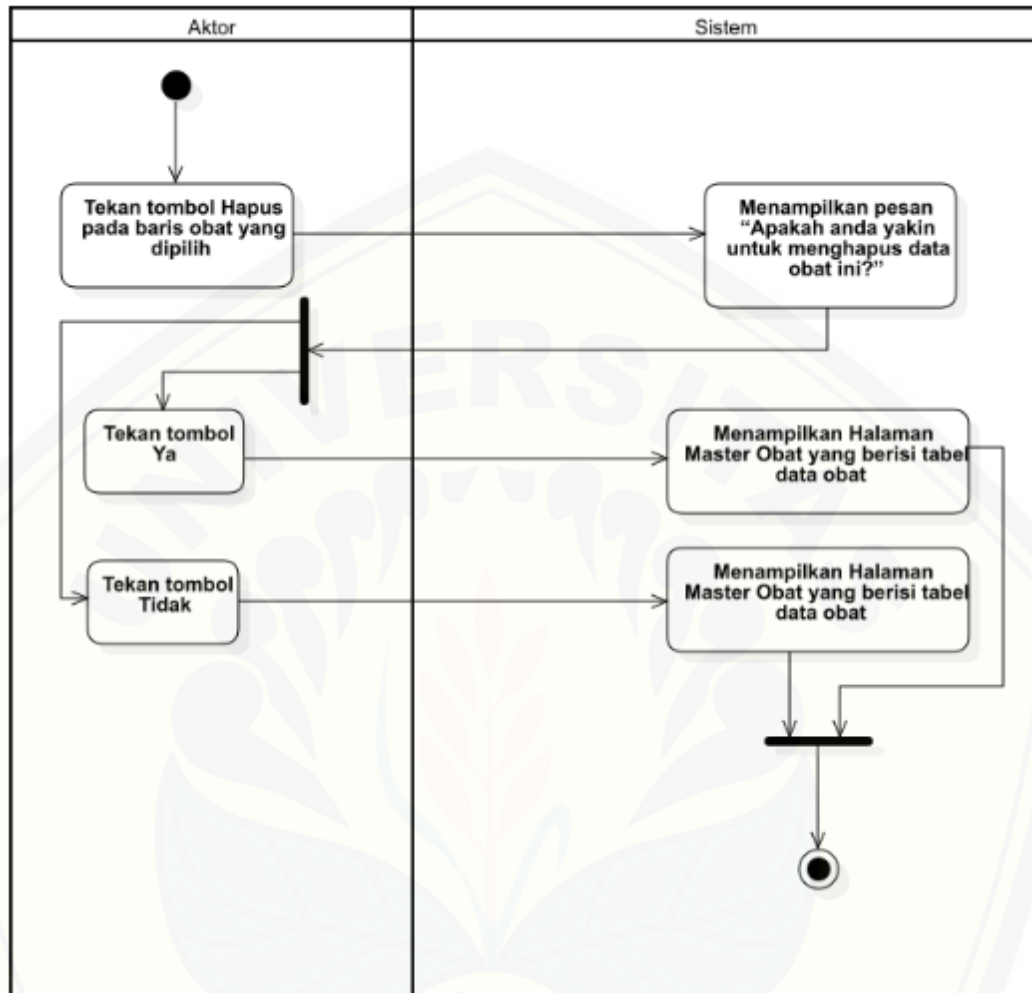
B.3. Activity Diagram Menambah Obat



B.4. Activity Diagram Mengubah Obat

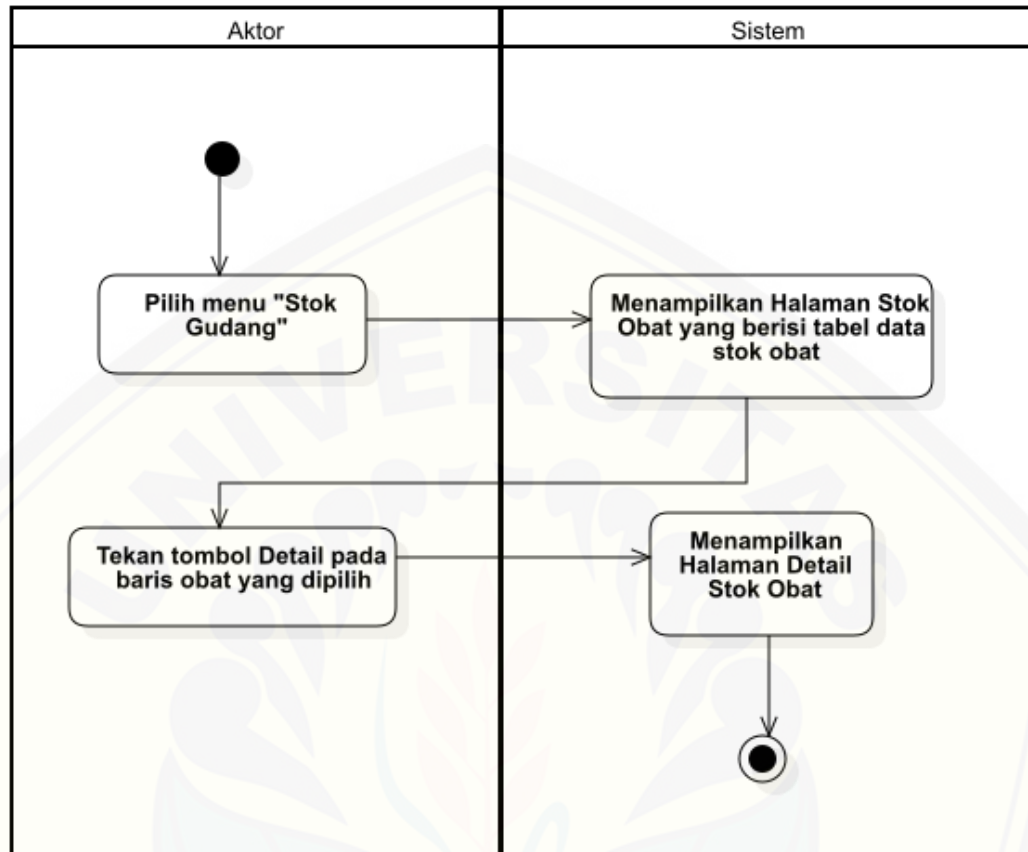


B.5. Activity Diagram Menghapus Obat

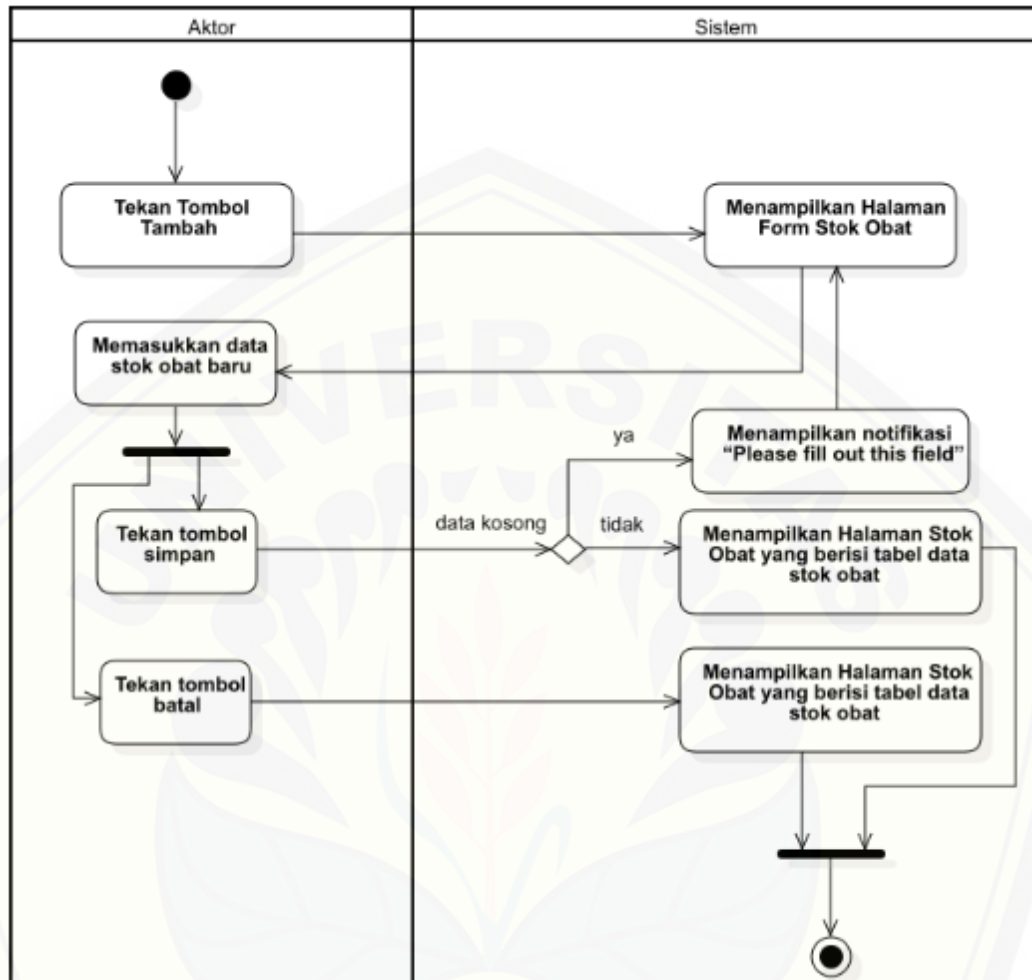




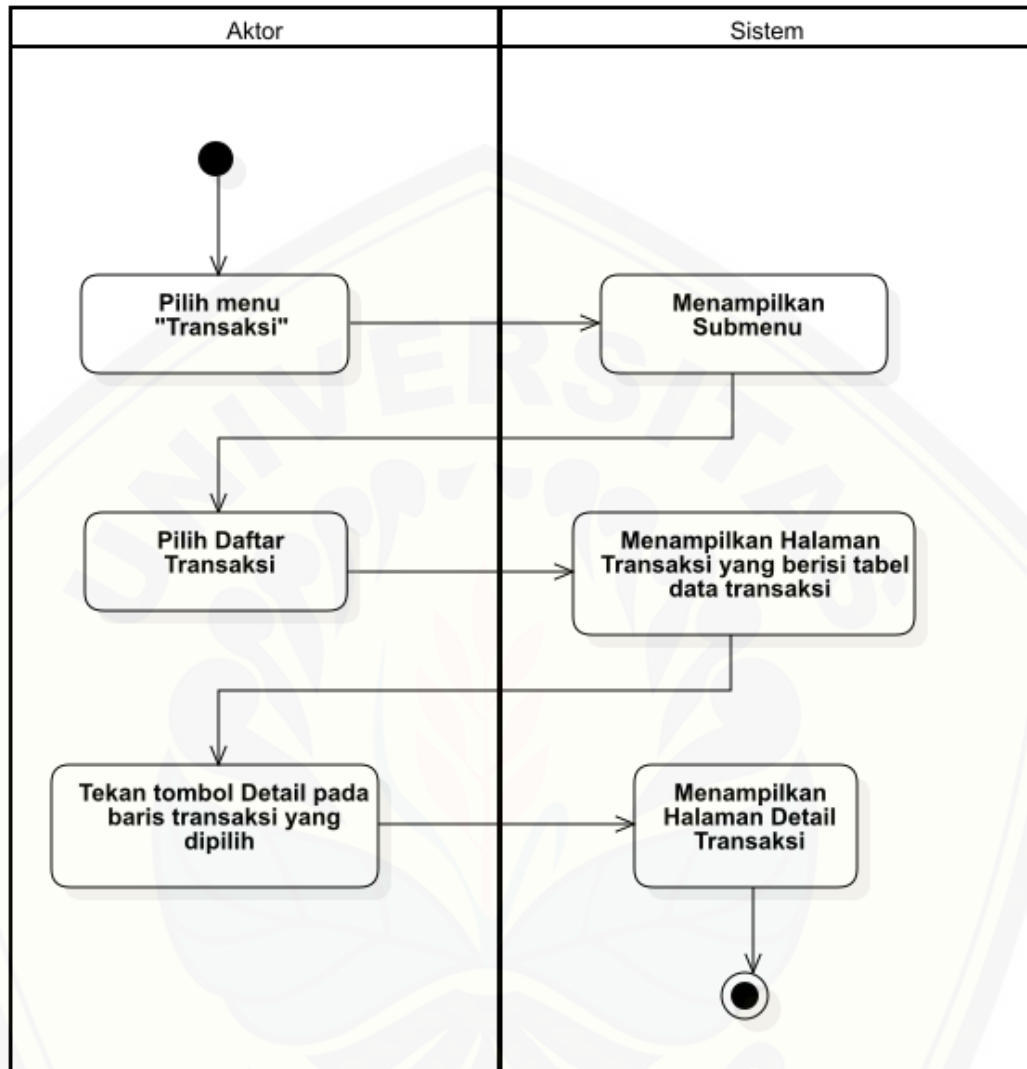
B.6. Activity Diagram Melihat Data Stok



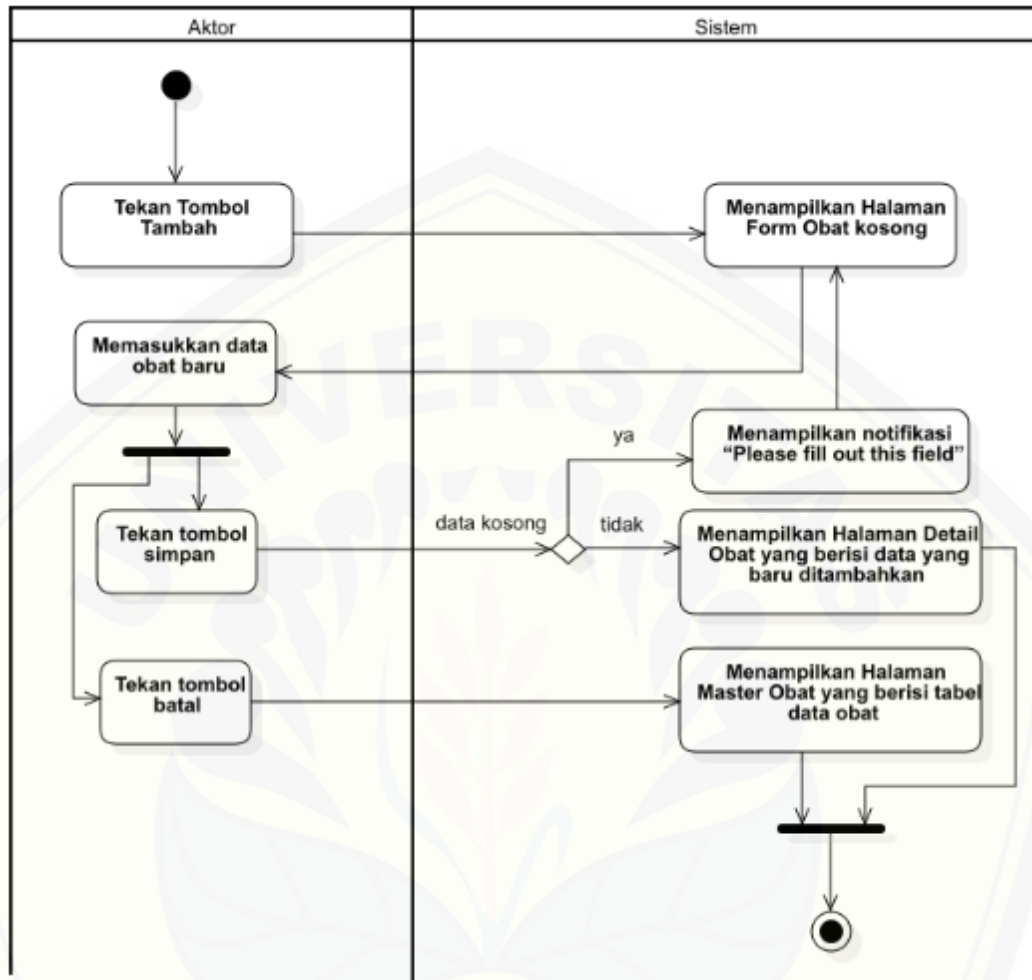
B.7. Activity Diagram Menambah Data Stok



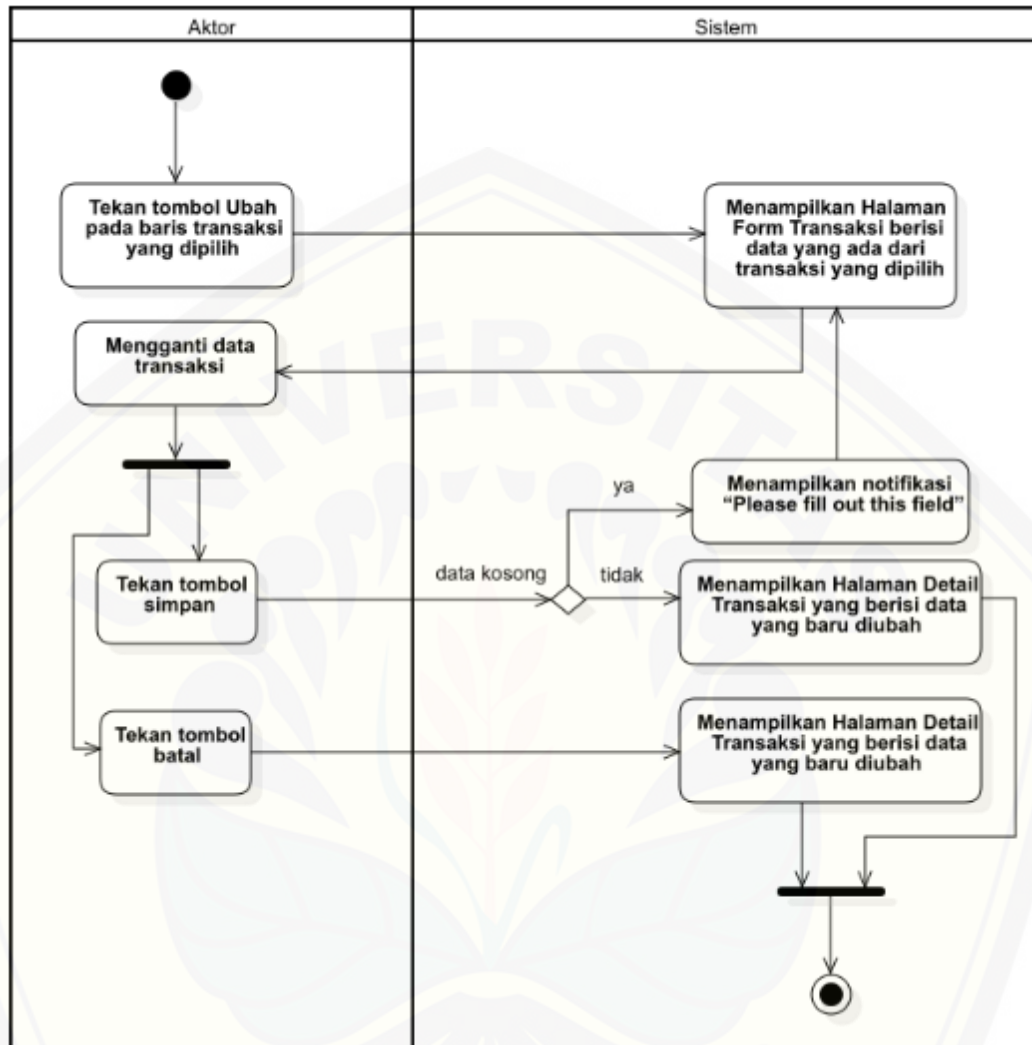
B.8. Activity Diagram Melihat Data Transaksi



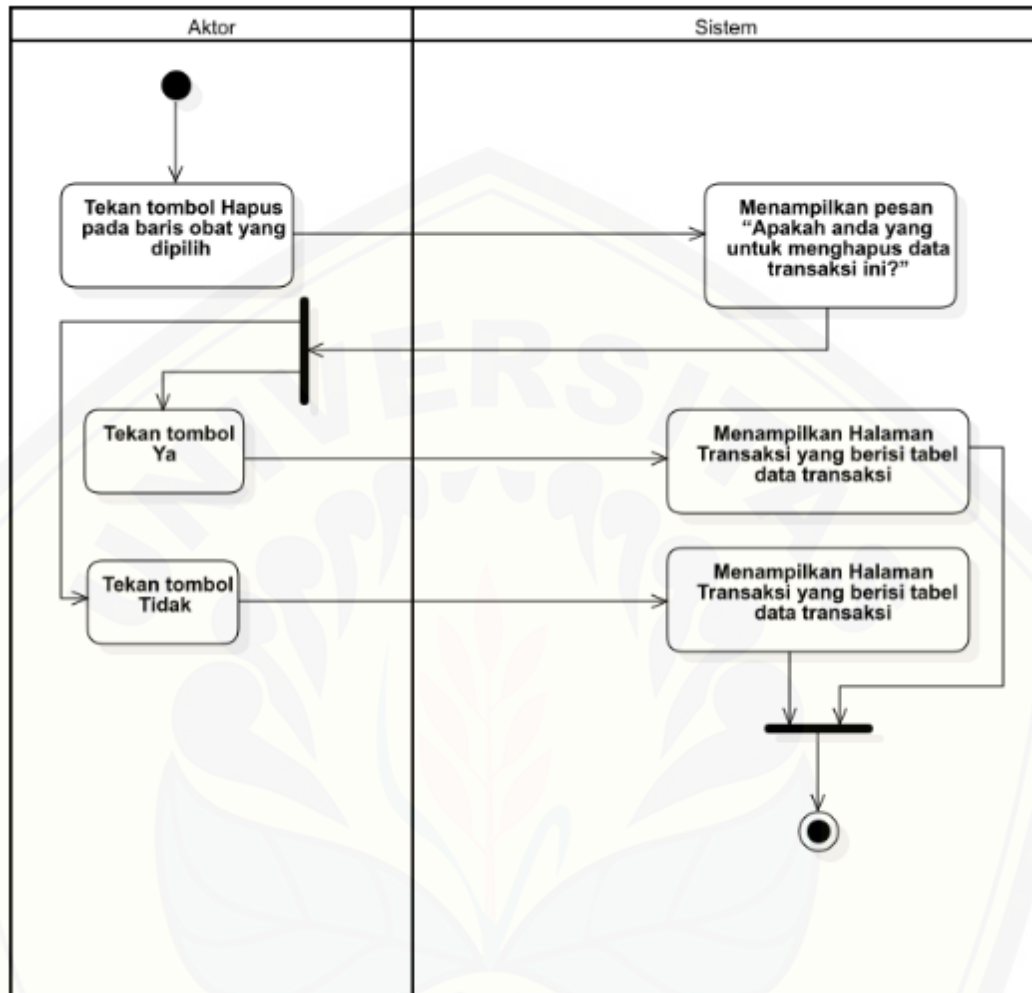
B.9. Activity Diagram Menambah Data Transaksi



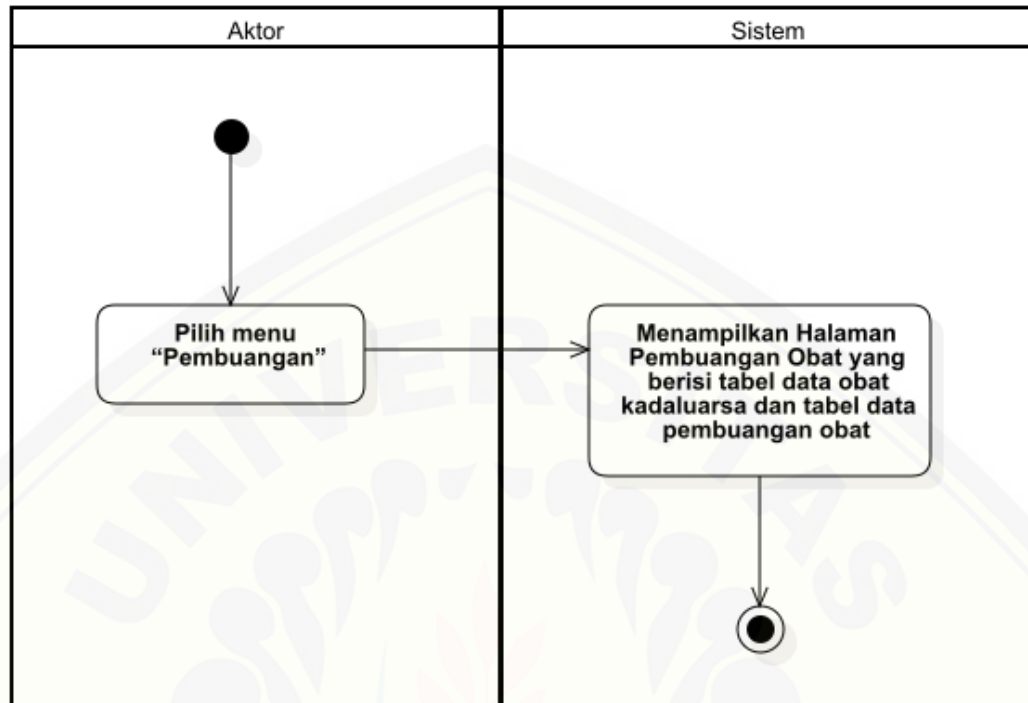
B.10. Activity Diagram Mengubah Data Transaksi



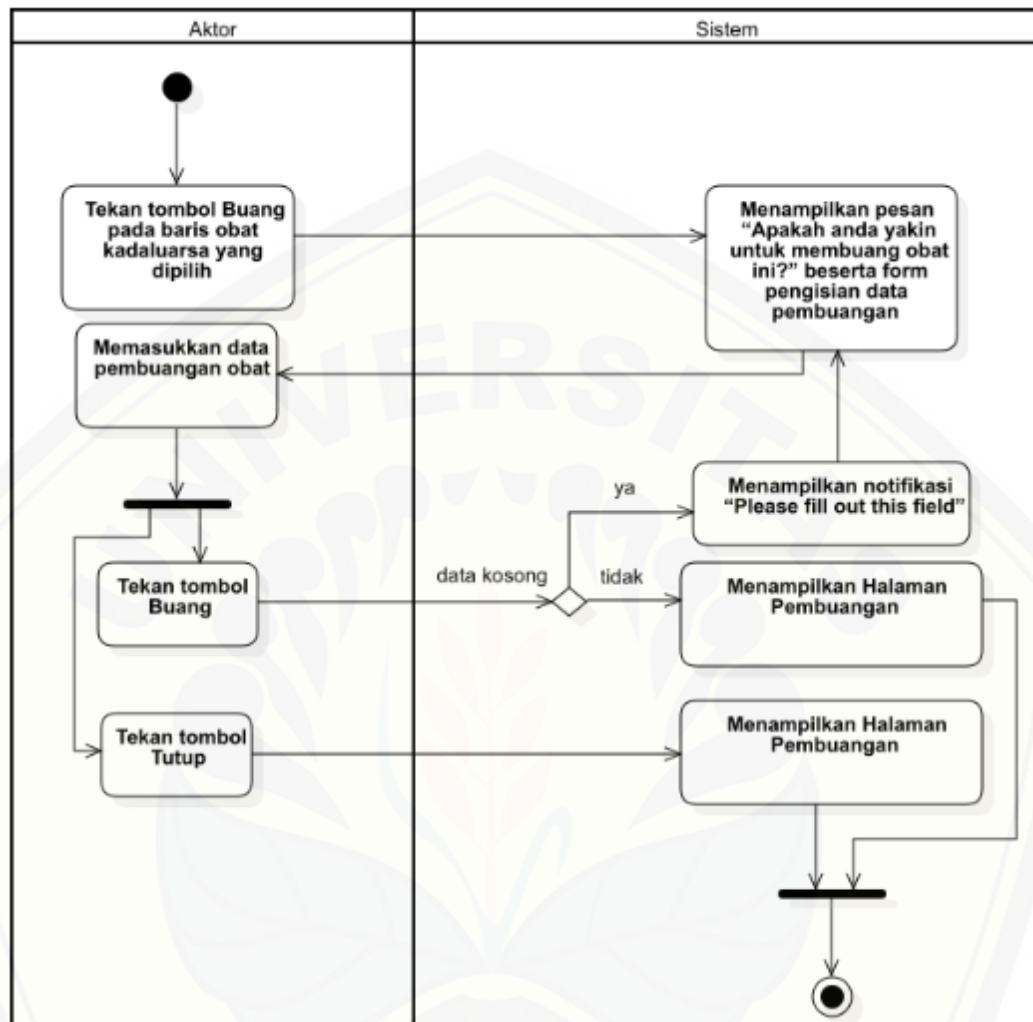
B.11. Activity Diagram Menghapus Data Transaksi



B.12. Activity Diagram Melihat Data Pembuangan Obat

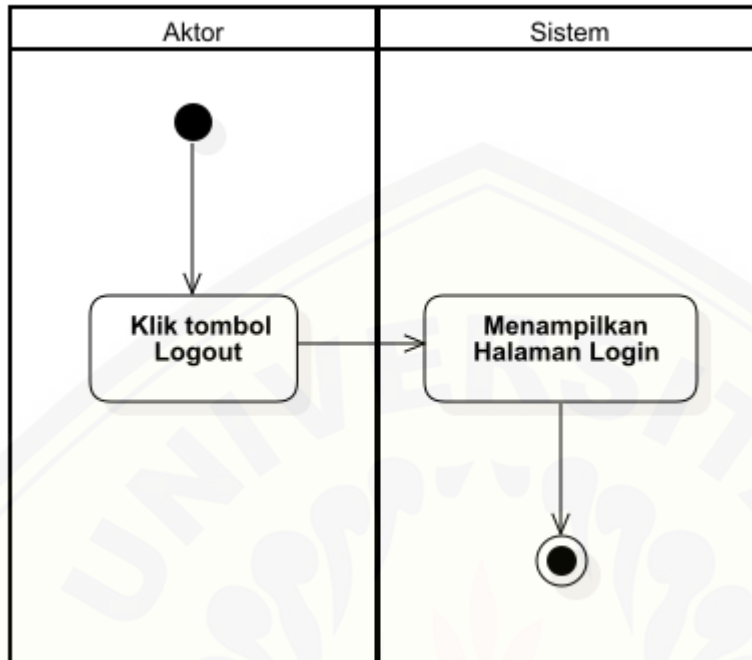


B.13. Activity Diagram Menambah Data Pembuangan Obat



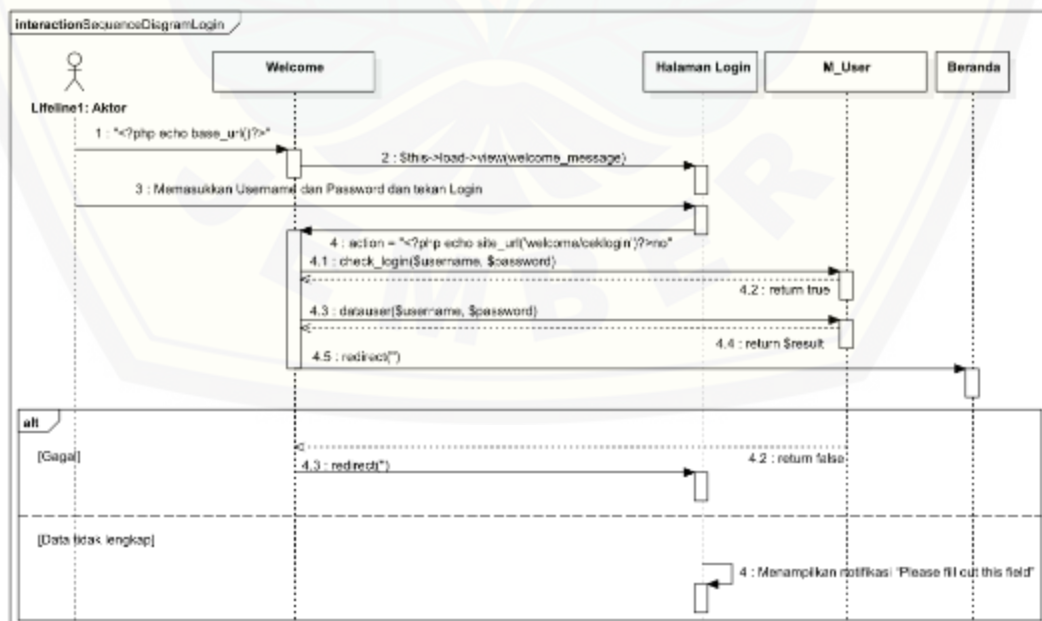


B.14. Activity Diagram Logout

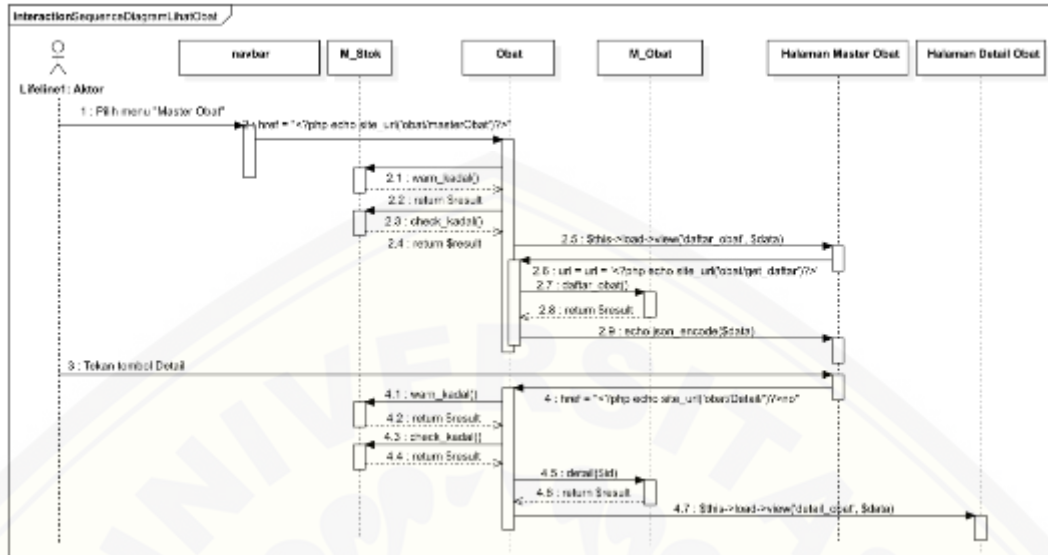


Lampiran C. Sequence Diagram

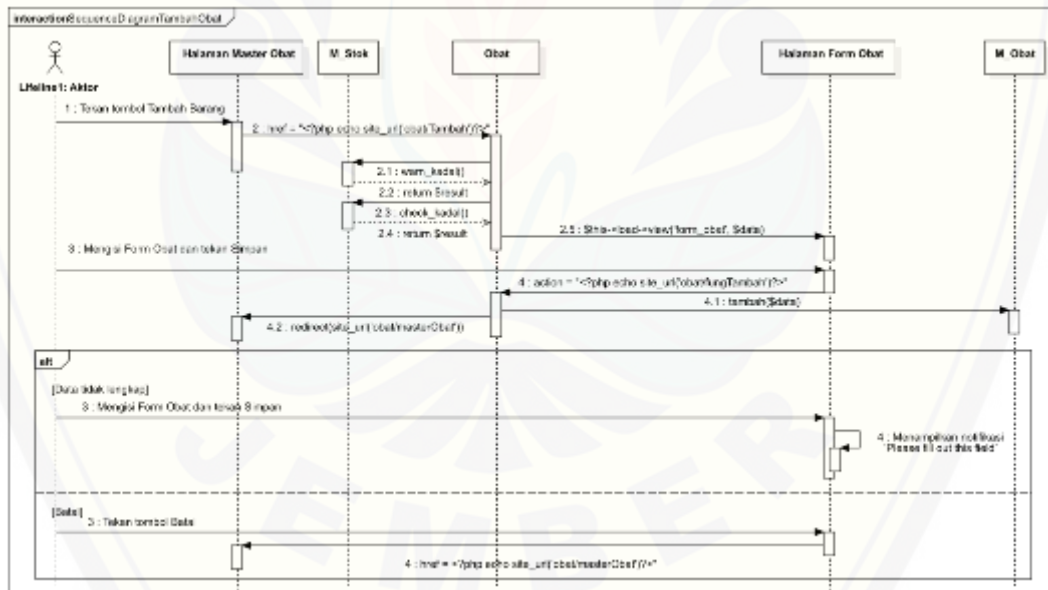
C.1. Sequence Diagram Login



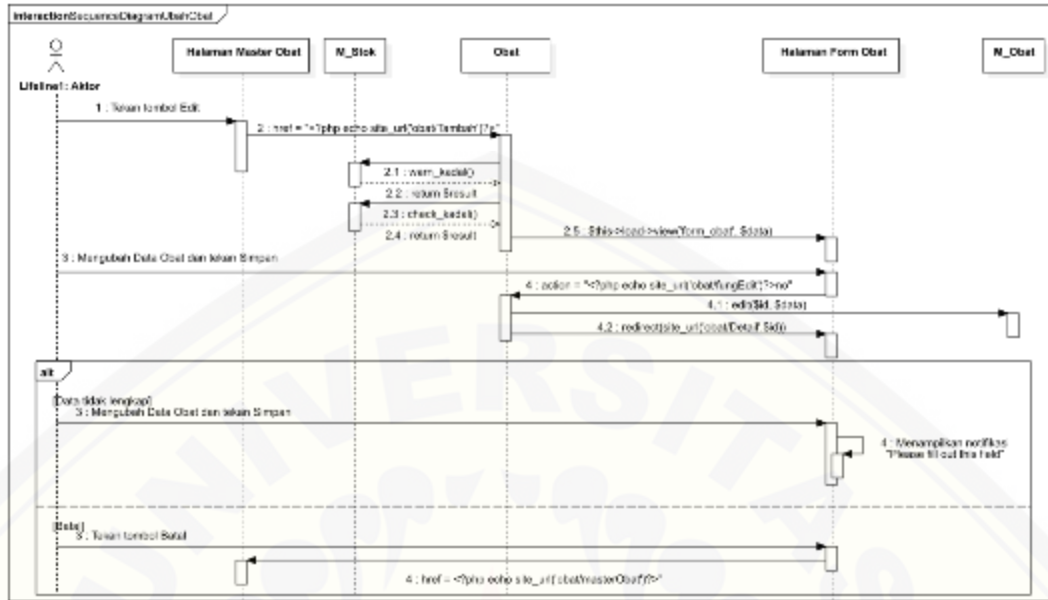
C.2. Sequence Diagram Melihat Obat



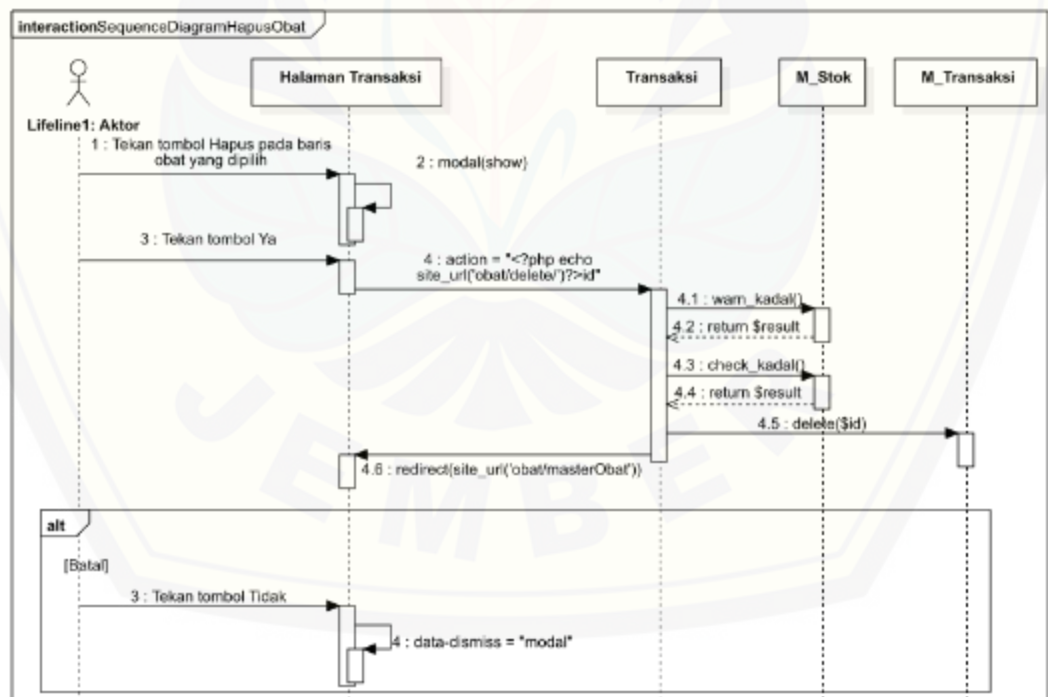
C.3. Sequence Diagram Menambah Obat



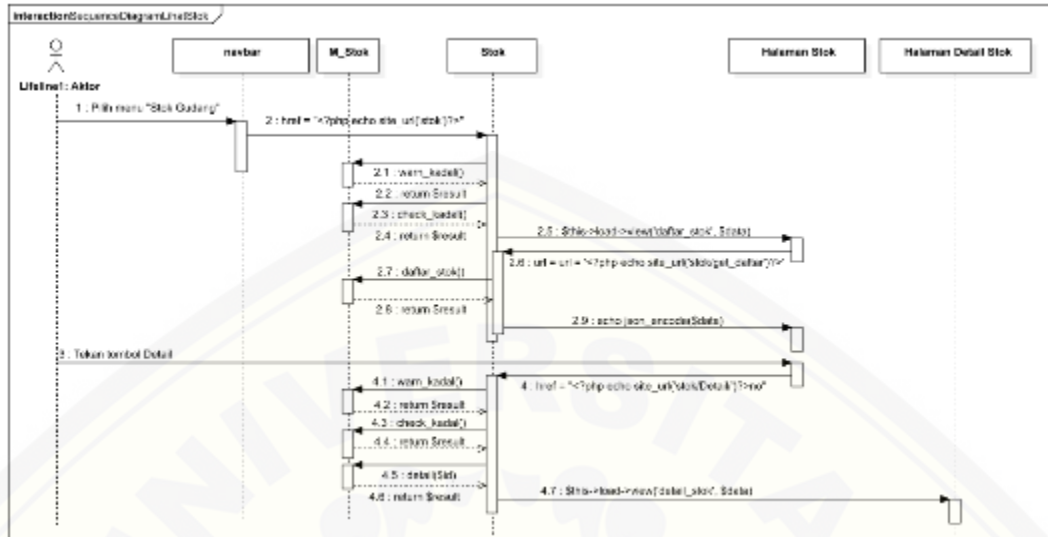
C.4. Sequence Diagram Mengubah Obat



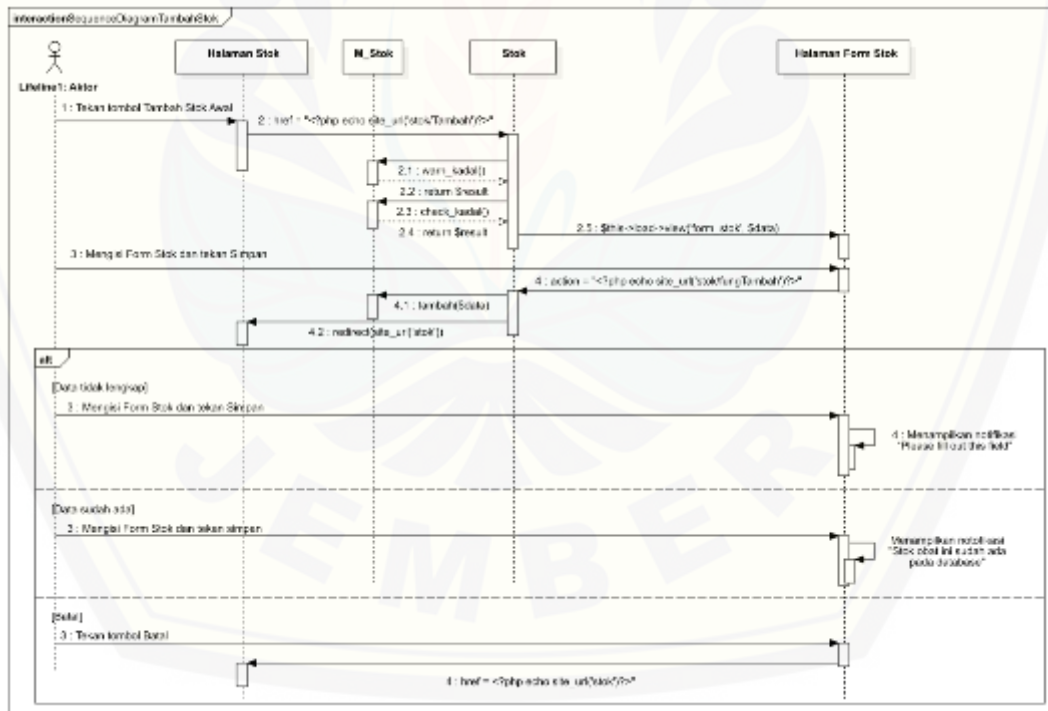
C.5. Sequence Diagram Menghapus Obat



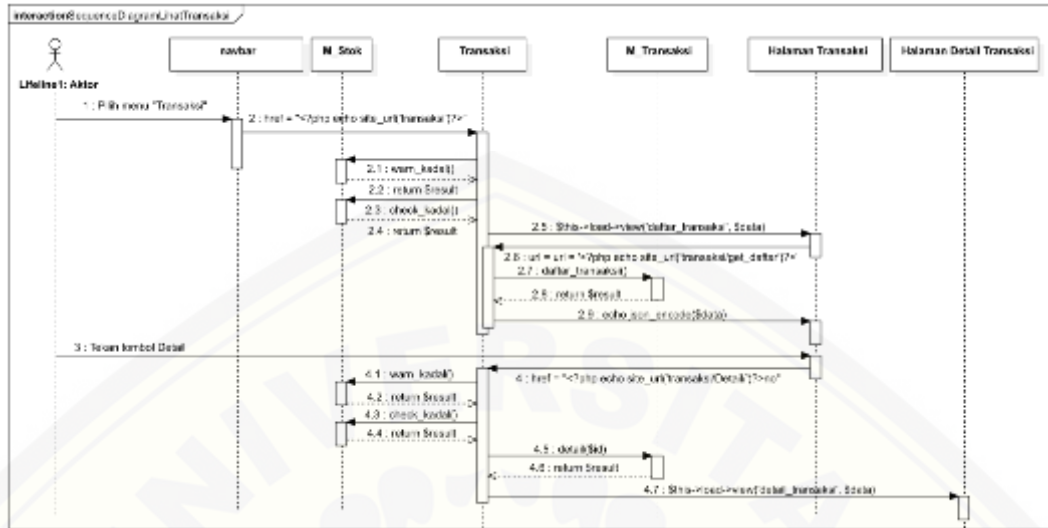
C.6. Sequence Diagram Melihat Data Stok



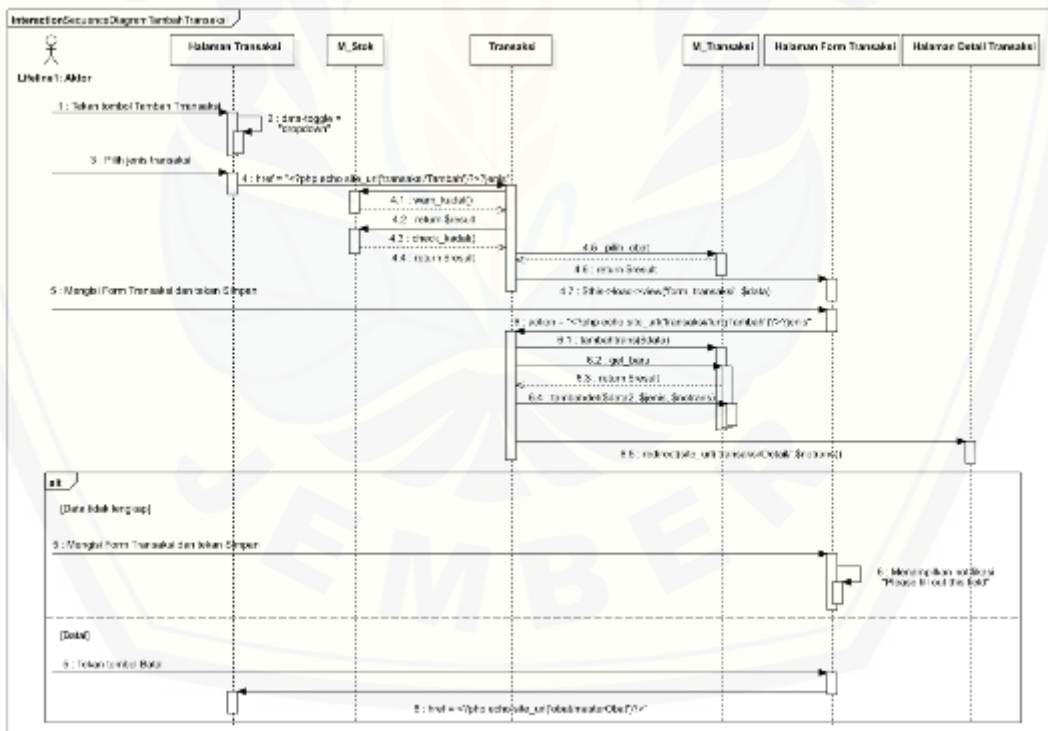
C.7. Sequence Diagram Menambah Data Stok



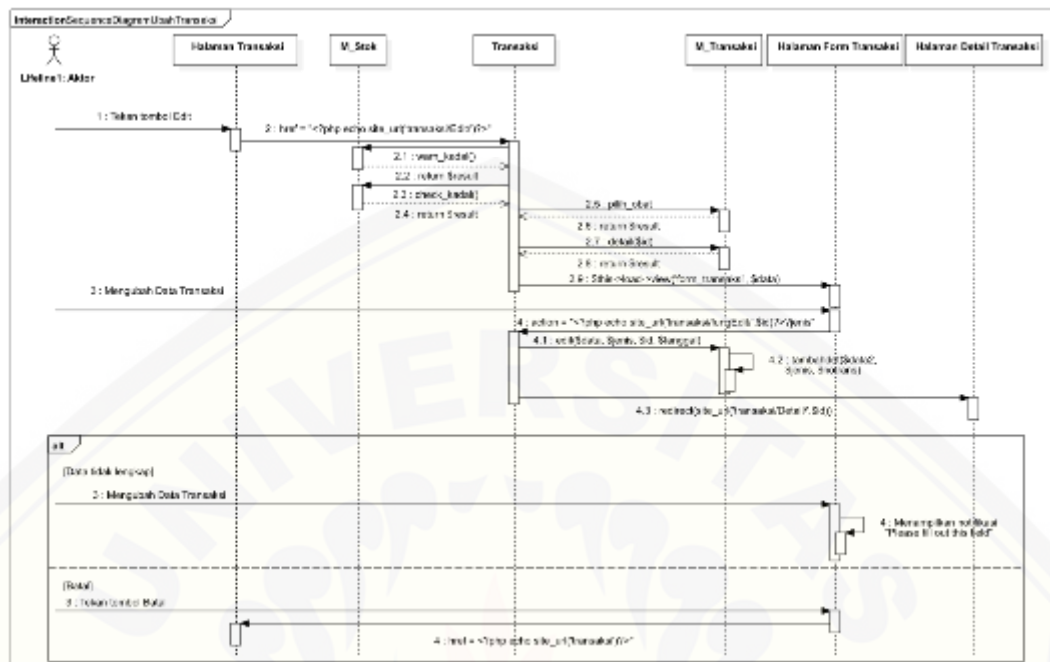
C.8. Sequence Diagram Melihat Data Transaksi



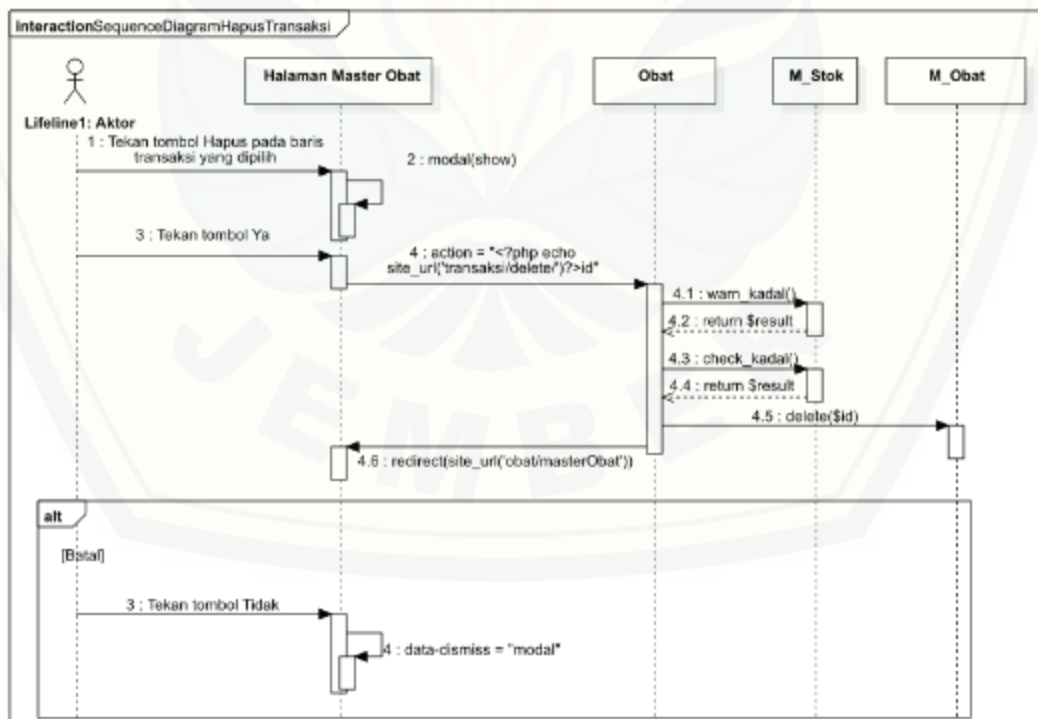
C.9. Sequence Diagram Menambah Data Transaksi



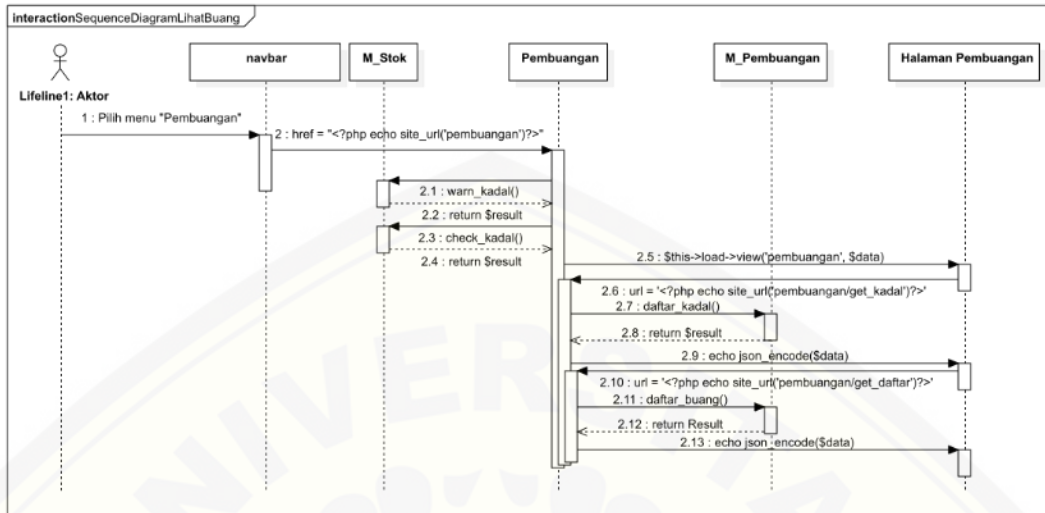
C.10. Sequence Diagram Mengubah Data Transaksi



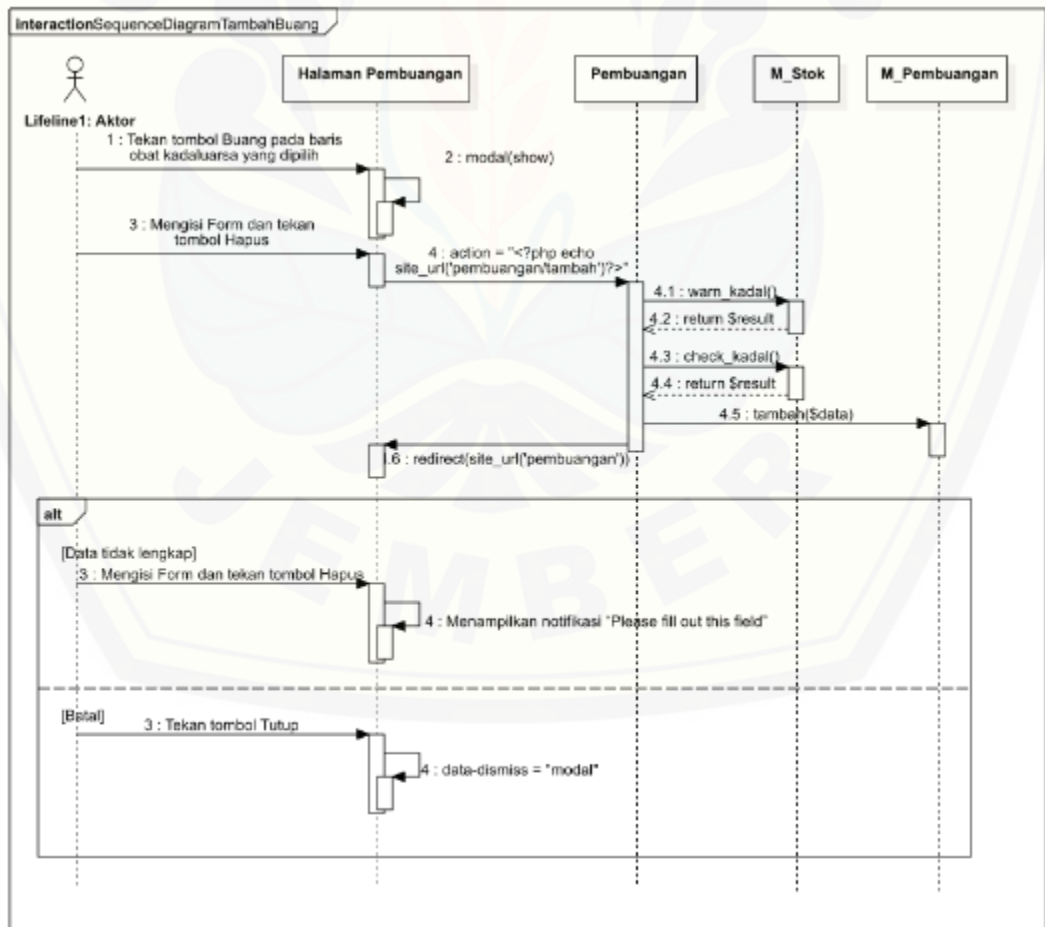
C.11. Sequence Diagram Menghapus Data Transaksi



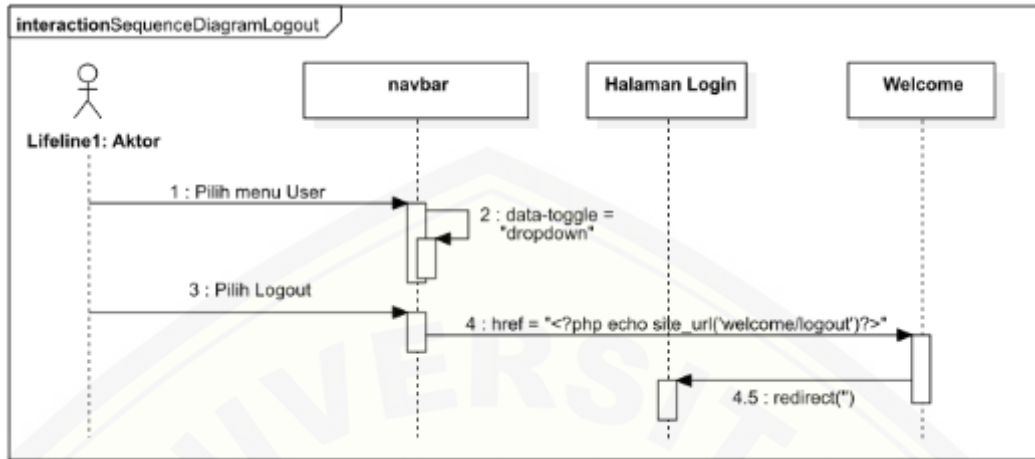
C.12. Sequence Diagram Melihat Data Pembuangan Obat



C.13. Sequence Diagram Menambah Data Pembuangan Obat



C.14. Sequence Diagram Logout



Lampiran D. Kode Program

D.1. Kode Program Pengguna

Kode Program pada Class Controller Welcome.php

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Welcome extends CI_Controller {
5
6     /**
7      * Index Page for this controller.
8      *
9      * Maps to the following URL
10     * http://example.com/index.php/welcome
11     * - or -
12     * http://example.com/index.php/welcome/index
13     * - or -
14     * Since this controller is set as the default controller in
15     * config/routes.php, it's displayed at http://example.com/
16     *
17     * So any other public methods not prefixed with an underscore will
18     * map to /index.php/welcome/<method_name>
19     * @see https://codeigniter.com/user_guide/general/urls.html
20     */
21     public function __construct(){
22         parent::__construct();
23         $this->load->model('m_user', 'muser');

```



---

```

24     }
25     public function index(){
26         if(!$this->session->userdata('logged')){
27             $this->load->view('welcome_message');
28             unset($_SESSION['passed']);
29         } else {
30             redirect(site_url('obat'));
31         }
32     }
33     public function ceklogin(){
34         $username = $this->input->post('username');
35         $password = $this->input->post('password');
36         $cek_login = $this->muser->check_login($username, $password);
37         if($cek_login === TRUE){
38             $datauser = $this->muser->datauser($username, $password);
39             $param_sess = array(
40                 'username' => $username,
41                 'id' => $datauser->row()->id,
42                 'akses' => $datauser->row()->akses,
43                 'jabatan' => $datauser->row()->jabatan,
44                 'logged' => TRUE,
45             );
46             $this->session->set_userdata($param_sess);
47             redirect("");
48         }else{
49             $this->session->set_userdata('passed', 'NO');
50             redirect("");
51         }
52     }
53     public function logout(){
54         $this->session->sess_destroy();
55         redirect("");
56     }
57 }
58

```

---

Kode Program pada Class Model M\_User.php

---

```

1 <?php
2
3 if(!defined('BASEPATH')) exit ('No direct script access allowed');
4
5 class M_User extends CI_Model{
6
7     public function datauser($username, $password){
8

```

---

---

```

    $query = "SELECT id, username, akses, jabatan FROM user WHERE
9  username = ".$username." AND password = ".$password."";
10  $result = $this->db->query($query);
11  return $result;
12  }
13  public function check_login($username, $password){
    $query = "SELECT * FROM user WHERE username =
14  ".$username." AND password = ".$password."";
15  $result = $this->db->query($query);
16  if($result->num_rows() > 0){
17  return true;
18  }else{
19  return false;
20  }
21  }
22  }

```

---

## D.2. Kode Program Data Obat

### Kode Program pada Class Controller Obat.php

---

```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class Obat extends CI_Controller {
5
6  /**
7   * Index Page for this controller.
8   *
9   * Maps to the following URL
10  *   http://example.com/index.php/welcome
11  * - or -
12  *   http://example.com/index.php/welcome/index
13  * - or -
14  * Since this controller is set as the default controller in
15  * config/routes.php, it's displayed at http://example.com/
16  *
17  * So any other public methods not prefixed with an underscore will
18  * map to /index.php/welcome/<method_name>
19  * @see https://codeigniter.com/user_guide/general/urls.html
20  */
21  private $warn;
22  private $cek;
23  public function __construct(){
24  parent::__construct();
25  if(!$this->session->userdata('logged')) {

```

---

---

```
26     redirect(site_url());
27     }
28     $this->load->model('M_Obat', 'mobat');
29     $this->load->model('M_Stok', 'mstok');
30     $this->warn=$this->mstok->warn_kadal()->result_array();
31     $this->cek=$this->mstok->check_kadal()->result_array();
32     }
33     public function index(){
34     $data['cek']=$this->cek;
35     $data['warn']=$this->warn;
36     $data['resep']=$this->mobat->resep();
37     $this->load->view('beranda', $data);
38     }
39     public function masterObat(){
40     $data['cek']=$this->cek;
41     $data['warn']=$this->warn;
42     $this->load->view('daftar_obat', $data);
43     }
44     public function get_daftar(){
45     $data=$this->mobat->daftar_obat()->result();
46     echo json_encode($data);
47     }
48     public function detail($id){
49     $data['cek']=$this->cek;
50     $data['warn']=$this->warn;
51     $detail=$this->mobat->detail($id);
52     $data['obat']=$detail->result_array();
53     $data['judul']=$data['obat'][0]['namaobat'];
54     $this->load->view('detail_obat', $data);
55     }
56     public function Tambah(){
57     $data['cek']=$this->cek;
58     $data['warn']=$this->warn;
59     $data['obat'][0]=array(
60     'no'=>",
61     'kodeobat'=>",
62     'namaobat'=>",
63     'kategori'=>",
64     'golongan'=>",
65     'produsen'=>",
66     'kekuatan'=>",
67     'satuan'=>",
68     'kemasan'=>",
69     'bentuk'=>",
70     'rute'=>",
71     'VEN'=>",
```

---

---

```

72     'hargajual'=>",
73     'hargabeli'=>",
74 );
75 $data['judul']='Baru';
76     $this->load->view('form_obat', $data);
77 }
78 public function fungTambah(){
79 $data=array(
80     'kodeobat'=>$this->input->post('kodeObat'),
81     'namaobat'=>$this->input->post('namaObat'),
82     'kategori'=>$this->input->post('kategori'),
83     'golongan'=>$this->input->post('golongan'),
84     'produsen'=>$this->input->post('produsen'),
85     'kekuatan'=>$this->input->post('kekuatan'),
86     'satuan'=>$this->input->post('satuan'),
87     'kemasan'=>$this->input->post('kemasan'),
88     'bentuk'=>$this->input->post('bentuk'),
89     'rute'=>$this->input->post('rute'),
90     'VEN'=>$this->input->post('VEN'),
91     'hargajual'=>$this->input->post('hjual'),
92     'hargabeli'=>$this->input->post('hbeli'),
93 );
94 $this->mobat->tambah($data);
95     redirect(site_url('obat/masterObat'));
96 }
97 public function Edit($id){
98 $data['cek']=$this->cek;
99 $data['warn']=$this->warn;
100 $data['obat']=$this->mobat->detail($id)->result_array();
101 $data['judul']=$data['obat'][0]['namaobat'];
102     $this->load->view('form_obat', $data);
103 }
104 public function fungEdit($id){
105 $data=array(
106     'kodeobat'=>$this->input->post('kodeObat'),
107     'namaobat'=>$this->input->post('namaObat'),
108     'kategori'=>$this->input->post('kategori'),
109     'golongan'=>$this->input->post('golongan'),
110     'produsen'=>$this->input->post('produsen'),
111     'kekuatan'=>$this->input->post('kekuatan'),
112     'satuan'=>$this->input->post('satuan'),
113     'kemasan'=>$this->input->post('kemasan'),
114     'bentuk'=>$this->input->post('bentuk'),
115     'rute'=>$this->input->post('rute'),
116     'VEN'=>$this->input->post('VEN'),
117     'hargajual'=>$this->input->post('hjual'),

```

---

---

```

118     'hargabeli'=>$this->input->post('hbeli'),
119     );
120     $this->mobat->edit($id, $data);
121     redirect(site_url('obat/Detail/'.$id));
122     }
123     public function delete(){
124         $id=$this->input->post('kode');
125         $this->mobat->delete($id);
126         redirect(site_url('obat/masterObat'));
127     }
128 }
129

```

---

Kode Program pada Class Model M\_Obat.php

---

```

1 <?php
2
3 if(!defined('BASEPATH')) exit ('No direct script access allowed');
4
5 class M_Obat extends CI_Model{
6
7     public function resepe(){
8         $result = $this->db->query("SELECT COUNT(notrans) as jumlah
9 FROM transaksi WHERE jenis='jual' AND DATEDIFF(tanggaltrans,
10 NOW())=0");
11         $data['sekarang']= $result->row()->jumlah;
12         $result = $this->db->query("SELECT COUNT(notrans) as jumlah
13 FROM transaksi WHERE jenis='jual' AND DATEDIFF(tanggaltrans,
14 NOW())=1");
15         $data['kemarin']= $result->row()->jumlah;
16         $result = $this->db->query("SELECT COUNT(notrans) as jumlah
17 FROM transaksi WHERE jenis='jual' AND YEAR(tanggaltrans) =
18 YEAR(NOW()) AND WEEK(tanggaltrans) = WEEK(NOW())");
19         $data['minggu']= $result->row()->jumlah;
20         $result = $this->db->query("SELECT COUNT(notrans) as jumlah
21 FROM transaksi WHERE jenis='jual' AND YEAR(tanggaltrans) =
22 YEAR(NOW()) AND MONTH(tanggaltrans) = MONTH(NOW())");
23         $data['bulan']= $result->row()->jumlah;
24         $result = $this->db->query("SELECT COUNT(notrans) as jumlah
25 FROM transaksi WHERE jenis='jual' AND YEAR(tanggaltrans) =
26 YEAR(NOW())");
27         $data['tahun']= $result->row()->jumlah;
28         $result = $this->db->query("SELECT COUNT(notrans) as jumlah,
29 COUNT(DISTINCT extract(day FROM tanggaltrans)) as hari FROM
30 transaksi WHERE jenis='jual' AND

```

---

---

```

PERIOD_DIFF(EXTRACT(YEAR_MONTH FROM NOW()),
EXTRACT(YEAR_MONTH FROM tanggaltrans)) = 1");
19     $data['rata']= round($result->row()->jumlah/$result->row()->hari);
20     return $data;
21 }
22 public function daftar_obat(){
23     $query = "SELECT * FROM obat";
24     $result = $this->db->query($query);
25     return $result;
26 }
27 public function detail($id){
28     $query = "SELECT * FROM obat WHERE no = '". $id. "'";
29     $result = $this->db->query($query);
30     return $result;
31 }
32 public function tambah($data){
33     $this->db->insert('obat',$data);
34 }
35 public function edit($id, $data){
36     $this->db->where('no', $id);
37     $this->db->update('obat',$data);
38 }
39 public function delete($id){
40     $query = "SELECT namaobat FROM obat WHERE no = '". $id. "'";
41     $result = $this->db->query($query)->row();
42     $hapus = '(DELETE)'.$result->namaobat;
43     $this->db->update('obat', array('namaobat' => $hapus), array('no' =>
44     $id));
45 }
46

```

---

### D.3. Kode Data Stok Obat

#### Kode Program pada Class Controller Stok.php

---

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Stok extends CI_Controller {
5
6     /**
7      * Index Page for this controller.
8      *
9      * Maps to the following URL
10     * http://example.com/index.php/welcome

```

---

---

```

11     * - or -
12     *   http://example.com/index.php/welcome/index
13     * - or -
14     * Since this controller is set as the default controller in
15     * config/routes.php, it's displayed at http://example.com/
16     *
17     * So any other public methods not prefixed with an underscore will
18     * map to /index.php/welcome/<method_name>
19     * @see https://codeigniter.com/user_guide/general/urls.html
20     */
21     private $warn;
22     private $cek;
23     public function __construct(){
24         parent::__construct();
25         if(!$this->session->userdata('logged')) {
26             redirect(site_url());
27         }
28         $this->load->model('M_Stok', 'mstok');
29         $this->warn=$this->mstok->warn_kadal()->result_array();
30         $this->cek=$this->mstok->check_kadal()->result_array();
31     }
32     public function index(){
33         $data['cek']=$this->cek;
34         $data['warn']=$this->warn;
35         $data['obat']=$this->mstok->daftar_stok();
36         $this->load->view('daftar_stok', $data);
37     }
38     public function get_daftar(){
39         $data=$this->mstok->daftar_stok()->result();
40         echo json_encode($data);
41     }
42     public function detail($id){
43         $data['cek']=$this->cek;
44         $data['warn']=$this->warn;
45         $detail=$this->mstok->detail($id);
46         $data['obat']=$detail->result_array();
47         $data['judul']=$data['obat'][0]['namaobat'];
48         $this->load->view('detail_stok', $data);
49     }
50     public function Tambah(){
51         $data['obat']=$this->mstok->pilih_obat();
52         $data['cek']=$this->cek;
53         $data['warn']=$this->warn;
54         $this->load->view('form_stok', $data);
55     }
56     public function cek_stok($id, $tanggal){

```

---

---

```

57     $data=$this->mstok->stok($id, $tanggal)->result();
58         echo json_encode($data);
59     }
60     public function fungTambah(){
61     $data=array(
62         'no'=>$this->input->post('obat'),
63         'kadaluarsa'=>$this->input->post('tanggal'),
64         'stok'=>$this->input->post('jumlah'),
65     );
66     $this->mstok->tambah($data);
67         redirect(site_url('stok'));
68     }
69 }
70 
```

---

Kode Program pada Class Model M\_Stok.php

---

```

1 <?php
2
3 if(!defined('BASEPATH')) exit ('No direct script access allowed');
4
5 class M_Stok extends CI_Model{
6
7     public function daftar_stok(){
8         $query = "SELECT *, SUM(stok) as totalstok FROM obat JOIN stok
9 ON obat.no = stok.no WHERE stok.stok > 0 GROUP BY obat.no";
10        $result = $this->db->query($query);
11        return $result;
12    }
13    public function detail($id){
14        $query = "SELECT *, DATEDIFF(kadaluarsa, NOW()) as selisih
15 FROM obat JOIN stok ON obat.no = stok.no WHERE stok.stok > 0 AND
16 obat.no = ".$id."";
17        $result = $this->db->query($query);
18        if ($result->num_rows() > 0) {
19            return $result;
20        }else {
21            $result = $this->db->query("SELECT * FROM obat WHERE no =
22 ".$id."");
23            return $result;
24        }
25    }
26    public function pilih_obat(){
27        $query = "SELECT * FROM obat WHERE namaobat NOT LIKE
28 '%(DELETE)%'";
29        $result = $this->db->query($query);
30    }
31 }
32 
```

---



---

```

25     return $result;
26     }
27     public function tambah($data){
28         $this->db->insert('stok',$data);
29     }
30     public function stok($id, $tanggal){
31         $query = "SELECT * FROM stok WHERE kadaluarsa
= '$tanggal.' AND no = '$id.'";
32         $result = $this->db->query($query);
33         return $result;
34     }
35     public function check_kadal(){
36         $query = "SELECT no, SUM(stok) as total FROM stok WHERE
kadaluarsa <= NOW() and stok > 0 GROUP BY no";
37         $result = $this->db->query($query);
38         return $result;
39     }
40     public function warn_kadal(){
41         $query = "SELECT no, SUM(stok) as total FROM stok WHERE
DATEDIFF(kadaluarsa, NOW()) <= 7 AND DATEDIFF(kadaluarsa,
NOW()) > 0 AND stok > 0 GROUP BY no";
42         $result = $this->db->query($query);
43         return $result;
44     }
45     public function delete($id){
46         $this->db->where('no', $id);
47         $this->db->delete('obat',$data);
48     }
49 }
50

```

---

#### D.4. Kode Program Transaksi

##### Kode Program pada Class Controller Transaksi.php

---

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Transaksi extends CI_Controller {
5
6     /**
7      * Index Page for this controller.
8      *
9      * Maps to the following URL
10     * http://example.com/index.php/welcome
11     * - or -

```

---

---

```

12     *   http://example.com/index.php/welcome/index
13     * - or -
14     * Since this controller is set as the default controller in
15     * config/routes.php, it's displayed at http://example.com/
16     *
17     * So any other public methods not prefixed with an underscore will
18     * map to /index.php/welcome/<method_name>
19     * @see https://codeigniter.com/user_guide/general/urls.html
20     */
21     private $warn;
22     private $cek;
23     public function __construct(){
24         parent::__construct();
25         if(!$this->session->userdata('logged')) {
26             redirect(site_url());
27         }
28         $this->load->model('M_Transaksi', 'mtrans');
29         $this->load->model('M_Stok', 'mstok');
30         $this->warn=$this->mstok->warn_kadal()->result_array();
31         $this->cek=$this->mstok->check_kadal()->result_array();
32     }
33     public function index(){
34         $data['cek']=$this->cek;
35         $data['warn']=$this->warn;
36         $this->load->view('daftar_transaksi', $data);
37     }
38     public function get_daftar(){
39         $data=$this->mtrans->daftar_transaksi()->result();
40         echo json_encode($data);
41     }
42     public function detail($id){
43         $data['cek']=$this->cek;
44         $data['warn']=$this->warn;
45         $data['detail']=$this->mtrans->detail($id)->result_array();
46         $data['totalnilai']=0;
47         foreach ($data['detail'] as $v) {
48             $data['totalnilai']+=( $v['jumlah']*$v['harga'] );
49         }
50         $this->load->view('detail_transaksi', $data);
51     }
52     public function Tambah(){
53         $data['cek']=$this->cek;
54         $data['warn']=$this->warn;
55         $data['obat']=$this->mtrans->pilih_obat();
56         $data['tanggal']=date('Y-m-d');
57         $data['detail'][0]=array( 'no' => ",

```

---

---

```

58         'kodeobat' => ",
59         'namaobat' => ",
60         'VEN' => ",
61         'notrans' => ",
62         'kadaluarsa' => ",
63         'jumlah' => ",
64         'harga' => ",
65         'tanggaltrans' => ",
66         'jenis' => $_GET['jenis'],
67         'id' => ",
68         'username' => ");
69     $data['totalnilai']=0;
70     $this->load->view('form_transaksi', $data);
71 }
72 public function fungTambah(){
73     $id=$_SESSION['id'];
74     $jenis=$_GET['jenis'];
75     $tanggal=$this->input->post('tgl');
76     $data1=array('id' => $id, 'tanggaltrans' => $tanggal, 'jenis' => $jenis);
77     $this->mtrans->tambahtrans($data1);
78     $notrans=$this->mtrans->get_baru()->row()->baru;
79     $data2['no']=$this->input->post('no');
80     $data2['harga']=$this->input->post('harga');
81     $data2['jumlah']=$this->input->post('jumlah');
82     if ($jenis!='jual') {
83         $data2['kadaluarsa']=$this->input->post('kadal');
84     }
85     $this->mtrans->tambahdet($data2, $jenis, $notrans, $tanggal);
86     redirect(site_url('transaksi/Detail/'.$notrans));
87 }
88 public function Edit($id){
89     $data['cek']=$this->cek;
90     $data['warn']=$this->warn;
91     $data['obat']=$this->mtrans->pilih_obat();
92     $data['detail']=$this->mtrans->detail($id)->result_array();
93     $data['totalnilai']=0;
94     foreach ($data['detail'] as $v) {
95         $data['totalnilai']+=( $v['jumlah']*$v['harga']);
96     }
97     $data['tanggal']=$data['detail'][0]['tanggaltrans'];
98     $this->load->view('form_transaksi', $data);
99 }
100 public function fungEdit($id){
101     $jenis=$_GET['jenis'];
102     $tanggal=$this->input->post('tgl');
103     $data2['no']=$this->input->post('no');

```

---

---

```

104     $data2['harga']=$this->input->post('harga');
105     $data2['jumlah']=$this->input->post('jumlah');
106     if ($jenis!='jual') {
107         $data2['kadal']=$this->input->post('kadal');
108     }
109     $this->mtrans->edit($data, $jenis, $id, $tanggal);
110     redirect('transaksi/Detail/'.$id);
111 }
112 public function delete(){
113     $id=$this->input->post('kode');
114     $this->mtrans->delete($id);
115     redirect(site_url('transaksi'));
116 }
117 }
118 
```

---

Kode Program pada Class Model M\_Transaksi.php

---

```

1 <?php
2
3 if(!defined('BASEPATH')) exit ('No direct script access allowed');
4
5 class M_Transaksi extends CI_Model{
6
7     public function daftar_transaksi(){
8         $query = "SELECT transaksi.notrans, tanggaltrans, jenis, jabatan,
9             username, SUM(detailtransaksi.harga*detailtransaksi.jumlah) as jumlah,
10            user.id FROM detailtransaksi JOIN transaksi ON detailtransaksi.notrans =
11            transaksi.notrans JOIN user on transaksi.id = user.id GROUP BY
12            detailtransaksi.notrans ORDER BY tanggaltrans DESC";
13
14         $result = $this->db->query($query);
15         return $result;
16     }
17     public function detail($id){
18         $jenis=$this->db->query("SELECT jenis FROM transaksi WHERE
19             notrans='".$id."'")->row()->jenis;
20         if($jenis=='jual'){
21             $query = "SELECT obat.no, kodeobat, namaobat, VEN,
22                 detailtransaksi.notrans, SUM(jumlah) as jumlah, kadaluarsa, harga,
23                 tanggaltrans, jenis, user.id, username FROM obat JOIN detailtransaksi on
24                 obat.no = detailtransaksi.no JOIN transaksi on detailtransaksi.notrans =
25                 transaksi.notrans JOIN user on transaksi.id = user.id WHERE
26                 detailtransaksi.notrans = '".$id.'" GROUP BY obat.no ORDER BY obat.no,
27                 kadaluarsa ASC";
28         } elseif ($jenis=='beli') {

```

---

---

```

17     $query = "SELECT obat.no, kodeobat, namaobat, VEN,
detailtransaksi.notrans, kadaluarsa, jumlah, harga, tanggaltrans, jenis,
user.id, username FROM obat JOIN detailtransaksi on obat.no =
detailtransaksi.no JOIN transaksi on detailtransaksi.notrans =
transaksi.notrans JOIN user on transaksi.id = user.id WHERE
detailtransaksi.notrans = ".$id." ORDER BY obat.no, kadaluarsa ASC";
18     }
19     $result = $this->db->query($query);
20     return $result;
21     }
22     public function get_baru(){
23         $query = "SELECT MAX(notrans) as baru FROM transaksi";
24         $result = $this->db->query($query);
25         return $result;
26     }
27     public function pilih_obat(){
28         $query = "SELECT obat.no, kodeobat, namaobat, kekuatan,
hargabeli, hargajual, sum(if(kadaluarsa > now(), stok, 0)) as jumlah FROM
obat LEFT OUTER JOIN stok on obat.no=stok.no WHERE namaobat
NOT LIKE '%(DELETE)%' GROUP BY obat.no";
29         $result = $this->db->query($query);
30         return $result;
31     }
32     public function tambahtrans($data){
33         $this->db->insert('transaksi',$data);
34     }
35     public function tambahdet($data, $jenis, $notrans, $tanggal){
36         if ($jenis=='beli') {
37             for ($i=0; $i < count($data['no']); $i++) {
38                 $tambah = array('no' => $data['no'][$i], 'notrans' => $notrans,
'kadaluarsa' => $data['kadaluarsa'][$i], 'jumlah' => $data['jumlah'][$i],
'harga' => $data['harga'][$i], );
39                 $this->db->insert('detailtransaksi', $tambah);
40                 $this->db->update('obat', array('hargabeli' => $data['harga'][$i]),
array('no' => $data['no'][$i]));
41                 $result=$this->db->query("SELECT * FROM stok WHERE no =
".$data['no'][$i]." AND kadaluarsa = ".$data['kadaluarsa'][$i].""");
42                 if ($result->num_rows() > 0) {
43                     $jumlah = $result->row()->stok + $data['jumlah'][$i];
44                     $this->db->update('stok', array('stok' => $jumlah), array('no' =>
$data['no'][$i], 'kadaluarsa' => $data['kadaluarsa'][$i]));
45                 } else {
46                     $tambah = array('no' => $data['no'][$i], 'kadaluarsa' =>
$data['kadaluarsa'][$i], 'stok' => $data['jumlah'][$i], );
47                     $this->db->insert('stok',$tambah);
48                 }

```

---

---

```

49     }
50     } else if ($jenis=='jual') {
51         for ($i=0; $i < count($data['no']); $i++) {
52             $result=$this->db->query("SELECT * FROM stok WHERE no =
53             ".$data['no'][$i]."" AND stok > 0 AND kadaluarsa > ".$tanggal."" ORDER
54             BY kadaluarsa ASC");
55             foreach ($result->result_array() as $v) {
56                 if ($v['stok']>=$data['jumlah'][$i]) {
57                     $this->db->insert('detailtransaksi', array('no' => $data['no'][$i],
58                     'notrans' => $notrans, 'kadaluarsa' => $v['kadaluarsa'], 'jumlah' =>
59                     $data['jumlah'][$i], 'harga' => $data['harga'][$i], ));
60                     $this->db->update('obat', array('hargajual' => $data['harga'][$i]),
61                     array('no' => $data['no'][$i]));
62                     $sisas=$v['stok']-$data['jumlah'][$i];
63                     $this->db->update('stok', array('stok' => $sisas), array('no' =>
64                     $data['no'][$i], 'kadaluarsa' => $v['kadaluarsa'][$i]));
65                     break;
66                 } else {
67                     $this->db->insert('detailtransaksi', array('no' => $data['no'][$i],
68                     'notrans' => $notrans, 'kadaluarsa' => $v['kadaluarsa'], 'jumlah' =>
69                     $v['stok'], 'harga' => $data['harga'][$i], ));
70                     $data['jumlah'][$i]-=$v['stok'];
71                     $this->db->update('stok', array('stok' => 0), array('no' =>
72                     $data['no'][$i], 'kadaluarsa' => $v['kadaluarsa'][$i]));
73                 }
74             }
75         }
76     }
77     }
78     }
79     public function edit($data, $jenis, $notrans, $tanggal){
80         $this->db->update('transaksi', array('tanggaltrans' => $tanggal),
81         array('notrans' => $notrans));
82         if ($jenis=='jual'){
83             for ($i=0; $i < count($data['no']); $i++) {
84                 $query = "SELECT * FROM detailtransaksi WHERE notrans =
85                 ".$notrans."" AND no = ".$data['no'][$i]."";
86                 $result = $this->db->query($query);
87                 if ($result->num_rows() > 0) {
88                     $jumlah = $result->row()->stok + $data['jumlah'][$i];
89                     $this->db->update('stok', array('stok' => $jumlah), array('no' =>
90                     $data['no'][$i], 'kadaluarsa' => $data['kadaluarsa'][$i]));
91                 } else {
92                     $tambah = array('no' => $data['no'][$i], 'kadaluarsa' =>
93                     $data['kadaluarsa'][$i], 'stok' => $data['jumlah'][$i], );
94                     $this->db->insert('stok',$tambah);
95                 }
96             }
97         }

```

---

---

```

82     }
83     } else if ($jenis=='beli'){
84         for ($i=0; $i < count($data['no']); $i++) {
85             $query = "SELECT * FROM detailtransaksi WHERE notrans =
86             '$notrans.'" AND no = '$data['no'][$i].'";
87             $result = $this->db->query($query);
88             if ($result->num_rows() > 0) {
89                 $jumlah = $result->row()->stok - $data['jumlah'][$i];
90                 $this->db->update('stok', array('stok' => $jumlah), array('no' =>
91                 $data['no'][$i], 'kadaluarsa' => $data['kadaluarsa'][$i]));
92             } else {
93                 $tambah = array('no' => $data['no'][$i], 'kadaluarsa' =>
94                 $data['kadaluarsa'][$i], 'stok' => $data['jumlah'][$i], );
95                 $this->db->insert('stok',$tambah);
96             }
97         }
98     }
99     public function delete($id){
100         $this->db->delete('detailtransaksi', array('notrans'=>$notrans));
101         $this->tambahdet($data, $jenis, $notrans, $tanggal);
102     }
103 }

```

---

#### D.5. Kode Program Pembuangan

##### Kode Program pada Class Controller Pembuangan.php

---

```

1 <?php
2 defined('BASEPATH') OR exit('No direct script access allowed');
3
4 class Pembuangan extends CI_Controller {
5
6     /**
7      * Index Page for this controller.
8      *
9      * Maps to the following URL
10     * http://example.com/index.php/welcome
11     * - or -
12     * http://example.com/index.php/welcome/index
13     * - or -
14     * Since this controller is set as the default controller in
15     * config/routes.php, it's displayed at http://example.com/

```

---

---

```
16  *
17  * So any other public methods not prefixed with an underscore will
18  * map to /index.php/welcome/<method_name>
19  * @see https://codeigniter.com/user_guide/general/urls.html
20  */
21  private $warn;
22  private $cek;
23  public function __construct(){
24      parent::__construct();
25      if(!$this->session->userdata('logged')) {
26          redirect(site_url());
27      }
28      $this->load->model('M_Pembuangan', 'mbuang');
29      $this->load->model('M_Stok', 'mstok');
30      $this->warn=$this->mstok->warn_kadal()->result_array();
31      $this->cek=$this->mstok->check_kadal()->result_array();
32  }
33  public function index(){
34      $data['cek']=$this->cek;
35      $data['warn']=$this->warn;
36      $this->load->view('pembuangan', $data);
37  }
38  public function get_daftar(){
39      $data=$this->mbuang->daftar_buang()->result();
40      echo json_encode($data);
41  }
42  public function get_kadal(){
43      $data=$this->mbuang->daftar_kadal()->result();
44      echo json_encode($data);
45  }
46  public function tambah(){
47      $data=array(
48          'no'=>$this->input->post('no'),
49          'jumlah'=>$this->input->post('jum'),
50          'ket'=>$this->input->post('ket'),
51          'kadaluarsa'=>$this->input->post('tgl'),
52          'tanggal'=>$this->input->post('buang'),
53      );
54      $this->mbuang->tambah($data);
55      redirect(site_url('pembuangan'));
56  }
57 }
58
```

---



Kode Program pada Class Model M\_Pembuangan.php

```

1 <?php
2
3 if(!defined('BASEPATH')) exit ('No direct script access allowed');
4
5 class M_Pembuangan extends CI_Model{
6
7     public function daftar_buang(){
8         $query = "SELECT * FROM obat JOIN pembuangan ON obat.no =
9         pembuangan.no";
10        $result = $this->db->query($query);
11        return $result;
12    }
13    public function daftar_kadal(){
14        $query = "SELECT * FROM obat JOIN stok ON obat.no = stok.no
15        WHERE DATEDIFF(kadaluarsa, NOW()) <= 0 AND stok > 0";
16        $result = $this->db->query($query);
17        return $result;
18    }
19    public function tambah($data){
20        $this->db->insert('pembuangan',$data);
21        $result = $this->db->query("SELECT stok FROM stok WHERE
22        kadaluarsa = ".$data['kadaluarsa']."' AND no = ".$data['no']."");
23        if ($result->result()->stok > $data['jumlah']) {
24            $this->db->update('stok', array('stok' => $result->result()->stok -
25            $data['jumlah']), array('no' => $data['no'][$i], 'kadaluarsa' =>
26            $v['kadaluarsa'][$i]));
27        }
28        else {
29            $this->db->delete('stok', array('no'=>$data['no'],
30            'kadaluarsa'=>$data['kadaluarsa']));
31        }
32    }
33 }
34 }

```

**Lampiran E. Pengujian Black Box**

Tabel Pengujian Black Box

Hasil Pengujian Black Box

No. Menu	Aksi	Hasil	Ket
----------	------	-------	-----

2. Login	Mengisi username dan password yang benar pada form login kemudian menekan tombol login	Login berhasil dan menampilkan Halaman Beranda	√
	Menekan tombol login tanpa mengisi username dan password	Menampilkan notifikasi "Please fill out this field"	√
	Mengisi username dan menekan tombol login	Menampilkan notifikasi "Please fill out this field"	√
	Mengisi password dan menekan tombol login	Menampilkan notifikasi "Please fill out this field"	√
	Mengisi username dan password yang salah dan menekan tombol login	Menampilkan Halaman Login dengan pesan "Username atau Password yang anda masukkan salah"	√
3. Melihat Data Obat	Pilih menu Master Obat	Menampilkan halaman Master Obat yang berisi tabel daftar Obat	√
	Tekan tombol detail pada baris obat yang dipilih	Menampilkan halaman detail data obat yang dipilih	√
4. Menambah Data Obat	Tekan tombol Tambah	Menampilkan form obat kosong	√
	Mengisi form dan tekan tombol simpan	Menampilkan halaman detail data obat yang baru dimasukkan	√
	Mengisi form tapi kode obat tidak diisi dan tekan tombol simpan	Menampilkan notifikasi "please fill out this field"	√
	Mengisi form tapi nama obat tidak diisi dan tekan tombol simpan	Menampilkan notifikasi "please fill out this field"	√

		Tekan tombol Batal	Menampilkan halaman Master Obat yang berisi tabel daftar Obat	√
5. Mengubah Data Obat		Tekan tombol Edit pada baris obat yang dipilih	Menampilkan form obat yang berisi data obat yang dipilih	√
		Mengubah data obat dan tekan tombol simpan	Menampilkan halaman detail data obat yang telah diubah	√
		Mengosongkan kode obat dan tekan tombol simpan	Menampilkan notifikasi "please fill out this field"	√
		Mengosongkan nama obat dan tekan tombol simpan	Menampilkan notifikasi "please fill out this field"	√
		Tekan tombol Batal	Menampilkan halaman Master Obat yang berisi tabel daftar Obat	√
6. Menghapus Data Obat		Tekan tombol hapus pada baris obat yang dipilih	Menampilkan modal	√
		Tekan tombol Hapus	Menampilkan halaman Master Obat yang berisi tabel daftar Obat	√
		Tekan tombol Batal	Menampilkan halaman Master Obat yang berisi tabel daftar Obat	√
7. Melihat Data Stok Obat		Pilih menu Stok Gudang	Menampilkan Halaman Stok Obat yang berisi tabel data stok obat	√
		Tekan tombol Detail pada baris obat yang dipilih	Menampilkan Halaman Detail Stok Obat	√
8.		Mengisi form dan tekan tombol simpan	Menampilkan halaman data stok	√

Menambah Data Stok Obat	Tekan tombol Simpan tanpa memilih obat	Menampilkan notifikasi "please fill out this field"	√
	Tekan tombol simpan tanpa mengisi tanggal kadaluarsa	Menampilkan notifikasi "please fill out this field"	√
	Tekan tombol Simpan tanpa mengisi jumlah	Menampilkan notifikasi "please fill out this field"	√
	Mengisi form dengan data yang sudah ada di database	Menampilkan notifikasi "Stok obat ini sudah ada pada database"	√
9. Melihat Data Transaksi	Pilih menu Daftar Transaksi	Menampilkan Halaman Transaksi yang berisi tabel data transaksi	√
	Tekan tombol detail pada baris transaksi yang dipilih	Menampilkan halaman detail data transaksi yang dipilih	√
10. Menambah Data Transaksi	Pilih menu Masukkan transaksi jual baru	Menampilkan halaman form transaksi jual kosong	√
	Pilih menu Masukkan transaksi beli baru	Menampilkan halaman form transaksi jual kosong	√
	Mengisi form dan tekan tombol Simpan	Menampilkan Halaman Detail Transaksi yang baru dimasukkan	
	Tekan tombol simpan tetapi ada <i>field</i> yang kosong	Menampilkan notifikasi "Please fill out this field"	√
	Tekan tombol batal	Menampilkan Halaman Transaksi yang berisi tabel data transaksi	√
11. Mengubah Data Transaksi	Tekan tombol Edit pada transaksi yang dipilih	Menampilkan halaman form transaksi yang berisi data transaksi yang dipilih	√
	Tekan tombol simpan tetapi ada <i>field</i> yang kosong	Menampilkan notifikasi "Please fill out this field"	√

		Tekan tombol batal	Menampilkan Halaman Transaksi yang berisi tabel data transaksi	√
12. Menghapus Data Transaksi		Tekan tombol hapus	Menampilkan modal	√
		Tekan tombol hapus	Hapus data dan menampilkan halaman daftar transaksi	√
		Tekan tombol batal	Tutup modal	√
13. Melihat Data Pembuangan Obat		Pilih menu Pembuangan	Menampilkan halaman Pembuangan	√
14. Menambah Data Pembuangan Obat		Tekan tombol Buang	Menampilkan modal form pembuangan	√
		Mengisi form dan tekan tombol Buang	Menampilkan halaman Pembuangan	√
		Tekan tombol buang tanpa mengisi form	Menampilkan notifikasi "Please fill out this field"	√
		Tekan tombol tutup	Menutup modal	√
15. Logout		Menekan tombol Logout	Keluar dari sistem dan menampilkan Halaman Login	√