



**IMPLEMENTASI METODE *WEIGHTED MOVING AVERAGE* DALAM  
PREDIKSI JUMLAH STOK DARAH PADA PALANG MERAH INDONESIA  
JEMBER**

**SKRIPSI**

Oleh

**Rizky Fauziah**

**NIM 112410101005**

**PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER**

**2016**



**IMPLEMENTASI METODE *WEIGHTED MOVING AVERAGE* DALAM  
PREDIKSI JUMLAH STOK DARAH PADA PALANG MERAH INDONESIA  
JEMBER**

**SKRIPSI**

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

Oleh

**Rizky Fauziah**

**NIM 112410101005**

**PROGRAM STUDI SISTEM INFORMASI  
UNIVERSITAS JEMBER**

**2016**

## PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Allah SWT, yang telah memberikan kelancaran dan kemudahan dalam menyelesaikan skripsi ini;
2. Ibunda Isnaini serta Ayahanda Choirul Anwar atas segala kesabaran, keikhlasan, limpahan kasih sayang, doa, dan motivasi serta dukungan yang luar biasa selama hidup saya;
3. Kakakku tercinta Arif Kurniawan dan adikku Alfian Alfarisy;
4. Saudara-saudaraku beserta seluruh keluarga besar;
5. Sahabatku bersama doa dan bantuannya;
6. Guru-guruku sejak taman kanak-kanak hingga perguruan tinggi;
7. Almater Program Studi Sistem Informasi Universitas Jember.



**MOTO**

إِنَّمَا أَمْرُهُ إِذَا أَرَادَ شَيْئًا أَنْ يَقُولَ لَهُ كُنْ فَيَكُونُ

“Sesungguhnya keadaan kekuasaannya apabila Ia menghendaki sesuatu hanyalah berkata kepadanya: ‘Jadilah!’, maka terjadilah ia.”

(QS. Yaasiin: 82)

*“Man sara ‘ala ad-darbi washala”*

“Barang siapa berjalan pada jalannya, dia akan sampai”

**PERNYATAAN**

Saya yang bertanda tangan di bawah ini:

Nama : Rizky Fauziah

NIM : 112410101005

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Implementasi Metode *Weighted Moving Average* Dalam Prediksi Jumlah Stok Darah Pada Palang Merah Indonesia 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, 22 Juni 2016

Yang menyatakan,

Rizky Fauziah

**SKRIPSI**

**IMPLEMENTASI METODE *WEIGHTED MOVING AVERAGE* DALAM  
PREDIKSI JUMLAH STOK DARAH PADA PALANG MERAH INDONESIA  
JEMBER**

oleh :

**Rizky Fauziah**

**NIM 112410101005**

Pembimbing :

Dosen Pembimbing Utama

: Drs. Antonius C.P.,M.App.,Sc.,Ph.D.

NIP.196909281993021001

Dosen Pembimbing Pendamping

: Yanuar Nurdiansyah ST.,M.Cs.

NIP 198201012010121004

**PENGESAHAN PEMBIMBING**

Skripsi berjudul Implementasi Metode *Weighted Moving Average* Dalam Prediksi Jumlah Stok Darah Pada Palang Merah Indonesia Jember, telah diuji dan disahkan pada :

hari, tanggal : Rabu, 22 Juni 2016

tempat : Program Studi Sistem Informasi Universitas Jember.

Disetujui oleh :

Pembimbing I,

Pembimbing II,

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

NIP.196909281993021001

Yanuar Nurdiansyah ST.,M.Cs.

NIP 198201012010121004



**PENGESAHAN PENGUJI**

Skripsi berjudul Implementasi Metode *Weighted Moving Average* Dalam Prediksi Jumlah Stok Darah Pada Palang Merah Indonesia Jember, telah diuji dan disahkan pada :

hari, tanggal : Rabu, 22 Juni 2016

tempat : Program Studi Sistem Informasi Universitas Jember.

Tim Penguji :

Penguji I,

Penguji II,

Dr. Saiful Bukhori, ST., M.Kom

NIP.196811131994121001

M. Arief Hidayat, S.Kom., M.Kom

NIP 198101232010121003

Mengesahkan

Ketua Program Studi,

Prof. Drs. Slamini, M.Comp.Sc.,Ph.D

NIP 196704201992011001

## Implementasi Metode *Weighted Moving Average* Dalam Prediksi Jumlah Stok Darah Pada Palang Merah Indonesia Jember

**Rizky Fauziah**

*Jurusan Sistem Informasi, Program Studi Sistem Informasi, Universitas Jember*

### **ABSTRAK**

Palang Merah Indonesia (PMI) adalah sebuah organisasi perhimpunan nasional di Indonesia yang bergerak dalam bidang sosial kemanusiaan dan kesehatan. Sebagai organisasi yang besar, PMI mempunyai banyak divisi atau unit kerja yang masing-masing membutuhkan barang dalam pekerjaannya. Bagian logistik Unit Transfusi Darah (UTD) merupakan gudang tempat penyimpanan barang-barang yang dibutuhkan oleh masing-masing unit kerja. Namun proses pengolahan data barang baik barang masuk, barang keluar, pesanan barang serta pembuatan laporan belum berjalan dengan baik karena belum terkomputerisasinya sistem persediaan barang yang ada pada UTD PMI ini. Pentingnya ketersediaan akan darah di PMI mengharuskan PMI untuk selalu menjaga jumlah dari darah yang tersedia untuk memenuhi kebutuhan akan transfusi darah.

Metode penelitian yang digunakan yaitu metode *weighted moving average* dipilih karena metode ini cocok dalam proses prediksi jumlah stok darah, diperlukan pemahaman dalam alur perhitungan metode *weighted moving average* secara manual hal ini dikarenakan pada perhitungan metode tersebut memiliki ciri khas dengan menghitung 6 periode. Sehingga tabel perhitungan hasil prediksi dapat terisi pada baris ketujuh dan seterusnya.

**Kata kunci:** PMI, Darah, *Weighted Moving Average*

## RINGKASAN

Ketersediaan dari darah yang tersedia untuk memenuhi kebutuhan akan transfusi darah. Namun sulitnya ketersediaan darah pada PMI disebabkan karena masih banyak masyarakat yang tidak memahami pentingnya arti donor darah, hal ini dilihat dari banyaknya masyarakat yang kesulitan mencari darah. Maka dari itu masyarakat yang mencari pasokan darah di PMI ini sangat kesulitan dan membutuhkan waktu yang cukup lama. Pernyataan ini didapat dari masyarakat yang kesulitan mencari darah. Berdasarkan permasalahan diatas, maka dibutuhkan pemanfaatan teknologi informasi untuk mengatasi permasalahan yang ada pada PMI Jember.

Pada penelitian ini dibangun Sistem Informasi Prediksi menggunakan beberapa variable seperti jumlah stok dan jumlah pengeluaran darah setiap periode. Metode yang digunakan dalam sistem ini adalah *weighted moving average*. Metode ini digunakan untuk menghitung rata-rata yang menghasilkan nilai prediksi. Sistem ini menghasilkan output berupa prediksi jumlah stok darah perkantong dan juga nilai tingkat keakuratan metode.

Penelitian ini dilaksanakan dalam tiga tahap yaitu tahap pengumpulan data, tahap analisis dan tahap pengembangan sistem. Tahap pengumpulan data dilakukan melalui studi literatur dan wawancara kepada narasumber yaitu PMI Jember. Tahap analisis dilakukan dengan menentukan pola data dan jumlah pengeluaran sample data darah keluar. Tahap pengembangan sistem dilakukan dengan mengadopsi model *waterfall* yang dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, pembuatan desain sistem, penulisan kode program dan pengujian sistem. Sistem ini dibangun menggunakan bahasa pemrograman *Page Hypertext Pre-Processor* (PHP) dan *framework Code Igniter* (CI). Berdasarkan hasil pengujian terhadap sistem, implementasi metode *weighted moving average* pada sistem ini dapat memberikan

prediksi jumlah stok darah masing-masing perkantong darah beserta nilai tingkat keakuratan hasil prediksi metode.



## PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Implementasi Metode *Weighted Moving Average* Dalam Prediksi Jumlah Stok Darah Pada Palang Merah Indonesia 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. Ibunda Isnaini serta Ayahanda Choirul Anwar serta seluruh keluarga yang telah mencurahkan perhatian, dukungan, motivasi, dan limpahan doa demi terselesaikannya skripsi ini;
2. Prof. Drs. Slamun, M.Comp.Sc., Ph.D., selaku Ketua Program Studi Sistem Informasi Universitas Jember;
3. Drs. Antonius Cahya P., M.App., Sc., Ph.D. selaku Dosen Pembimbing Utama dan Yanuar Nurdiansyah ST., M.Cs. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
4. Prof. Drs. Slamun, M.Comp.Sc., Ph.D., selaku Dosen Pembimbing Akademik yang telah membimbing selama penulis menjadi mahasiswa;
5. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember;
6. Kakakku Arif Kurniawan dan adikku Alfan Alfarisy yang telah menjadi motivasi terbesarku;
7. Saudara-saudaraku berserta seluruh keluarga besar;
8. Sahabat hatiku Andean Pradana yang selalu memberi semangat dan motivasi;
9. Sahabat seperjuangan yang selalu menemani dan memberikan semangat Fepri Kharismasari dan Diannisa Wildan Aviva;

10. Keluarga besar NEFOTION semua mahasiswa Program Studi Sistem Informasi angkatan 2011 yang telah menjadi keluarga kecil bagi penulis selama menempuh pendidikan S1;
11. PMI Jember yang telah meluangkan waktu dan memberikan data terkait penulisan skripsi ini;
12. Semua pihak yang tidak dapat disebutkan satu-persatu.

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, 22 Juni 2016

Penulis

## DAFTAR ISI

BAB 1. PENDAHULUAN .....	<b>Error! Bookmark not defined.</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan.....	3
1.4 Manfaat.....	3
1.5 Sistematika Penulisan.....	3
BAB 2. TINJAUAN PUSTAKA .....	5
2.1 Kajian Penelitian Terdahulu.....	5
2.2 Metode <i>Weighted Moving Average</i> .....	7
2.2.1 Relevansi Metode <i>Weighted Moving Average</i> Terhadap Peramalan Jumlah Stok Darah Pada PMI.....	8
2.2.2 Metode Tingkat Pengukuran Tingkat Error Suatu Hasil Peramalan.....	8
BAB 3. METODOLOGI PENELITIAN.....	10
3.1 Waktu dan Tempat Penelitian .....	10
3.2 Tahapan Penelitian .....	10
3.4.1 Tahap Pengumpulan Data .....	11
3.4.2 Identifikasi Masalah .....	12
3.4.3 Perancangan Sistem.....	12
3.3 Coding (Pengkodean).....	14
3.4 Implementasi Penerapan Metode <i>Weighted Moving Average</i> pada sistem.....	15
3.5 Pengujian.....	16
BAB 4. DESAIN DAN PENGEMBANGAN SISTEM .....	18
4.1 Deskripsi Umum Sistem.....	18
4.1.1 Statement of Purpose.....	18
4.1.2 Karakteristik Sistem .....	18
4.2 Analisis Kebutuhan .....	19

4.2.1	Kebutuhan Fungsional.....	19
4.2.2	Kebutuhan Non-Fungsional .....	20
4.3	Desain Sistem.....	20
4.3.1	Business Process .....	20
4.3.2	<i>Usecase Diagram</i> .....	21
4.3.3	Skenario Sistem.....	24
4.3.4	Activity Diagram.....	31
4.3.5	Sequence Diagram.....	37
4.3.6	<i>Class Diagram</i> .....	42
4.3.7	Entity Relation Diagram (ERD) .....	43
4.3	Desain Sistem.....	44
4.4	Implementasi Perancangan.....	44
4.5	Pengujian.....	44
4.5.1	<i>Black box</i> .....	45
4.5.2	<i>White box</i> .....	48
BAB 5.	HASIL DAN PEMBAHASAN.....	52
5.1	Sistem Peramalan Jumlah Stok Darah Pada PMI Dengan Menggunakan Metode <i>Weighted Moving Average</i> .....	52
5.2	Tahap Implementasi Sistem Peramalan Jumlah Stok Darah Pada PMI Dengan Menggunakan Metode <i>Weighted Moving Average</i> .....	52
5.3	Hasil Implementasi <i>Coding</i> Pada Sistem Informasi Peramalan Jumlah Stok Darah Pada PMI .....	53
5.3.1	Tampilan <i>login screen</i> .....	53
5.3.2	Tampilan halaman awal admin .....	54
5.3.3	Tampilan halaman data darah (admin).....	54
5.3.4	Tampilan form tambah data darah .....	55
5.3.5	Tampilan form <i>edit</i> data darah .....	56
5.3.6	Tampilan peringatan hapus data darah.....	56



5.3.7	Tampilan halaman darah masuk (admin) .....	57
5.3.8	Tampilan form tambah data darah masuk .....	58
5.3.9	Tampilan <i>edit</i> data darah masuk.....	58
5.3.10	Tampilan peringatan hapus data darah masuk .....	59
5.3.11	Tampilan halaman data darah keluar (admin).....	60
5.3.12	Tampilan form tambah data darah keluar .....	61
5.3.13	Tampilan <i>edit</i> data darah keluar .....	61
5.3.14	Tampilan peringatan hapus data darah masuk .....	62
5.3.15	Tampilan Halaman Stok Darah .....	63
5.3.16	Tampilan Peramalan Darah Keluar .....	63
5.3.17	Tampilan Peramalan Darah Masuk .....	64
5.3.18	Tampilan Perhitungan <i>Manual Weighted Moving Average</i> .....	65
5.3.19	Logout .....	66
5.4	Pengujian Implementasi Metode <i>Weighted Moving Average</i> Pada Jumlah Stok Darah .....	66
5.5	Pembahasan.....	80
5.5.1	Pembahasan Hasil Implementasi Metode <i>Weighted Moving Average</i> Pada Sistem Informasi Prediksi Jumlah Stok Darah Pada PMI .....	81
5.5.2	Pembahasan Hasil Pengujian Sistem.....	83
BAB 6.	PENUTUP.....	85
6.1	Kesimpulan.....	85
6.2	Saran.....	86
LAMPIRAN A.	PERANCANGAN SISTEM.....	88
A1.	Perancangan Skenario .....	88
A2.	Perancangan <i>Activity Diagram</i> .....	103
A3.	Perancangan <i>Sequence Diagram</i> .....	110
A4.	Implementasi Program .....	118
A5.	Pengujian White Box .....	119
A6.	Kode Program .....	122

## DAFTAR GAMBAR

Gambar 3.1 Diagram Alir Penelitian .....	10
Gambar 3.2 Tahapan Metode Waterfall.....	13
Gambar 3.3 Weighted Moving Average yang implemented .....	15
Gambar 4.1 Bussiness Process .....	21
Gambar 4.2 Usecase Diagram.....	22
Gambar 4.3 <i>Activity Diagram</i> Input Data Darah Keluar.....	32
Gambar 4.4 <i>Activity Diagram</i> Edit Data Darah Keluar .....	33
Gambar 4.5 <i>Activity Diagram</i> Hapus Data Darah Keluar.....	34
Gambar 4.6 <i>Activity Diagram</i> Peramalan Darah Keluar.....	34
Gambar 4.7 <i>Activity Diagram</i> Peramalan Darah Masuk.....	35
Gambar 4.8 <i>Sequence Diagram</i> Input Data Darah Masuk.....	38
Gambar 4.9 <i>Sequence Diagram</i> Edit Data Darah Masuk .....	39
Gambar 4.10 <i>Sequence Diagram</i> Hapus Data Darah Masuk.....	40
Gambar 4.11 <i>Sequence Diagram</i> Peramalan Darah Keluar.....	40
Gambar 4.12 <i>Class Diagram</i> Sistem Peramalan Jumlah Stok Darah .....	43
Gambar 4.13 <i>Entity Relation Diagram</i> Sistem Peramalan Stok Darah .....	44
Gambar 4.14 Listing Program function FormUpdateDarah().....	49
Gambar 4.15 Listing Program function DarahKeluarInsertDb() .....	50
Gambar 5.1 Kode Program Menghitung <i>Weighted Moving Average</i> .....	53
Gambar 5.2 Tampilan login screen .....	54
Gambar 5.3 Halaman awal admin.....	54
Gambar 5.4 Halaman data darah.....	55
Gambar 5.5 Form tambah data darah.....	55
Gambar 5.6 Form edit data darah.....	56
Gambar 5.7 Peringatan hapus data darah.....	56
Gambar 5.8 Tampilan halaman data darah masuk .....	57
Gambar 5.9 Form tambah darah masuk .....	58

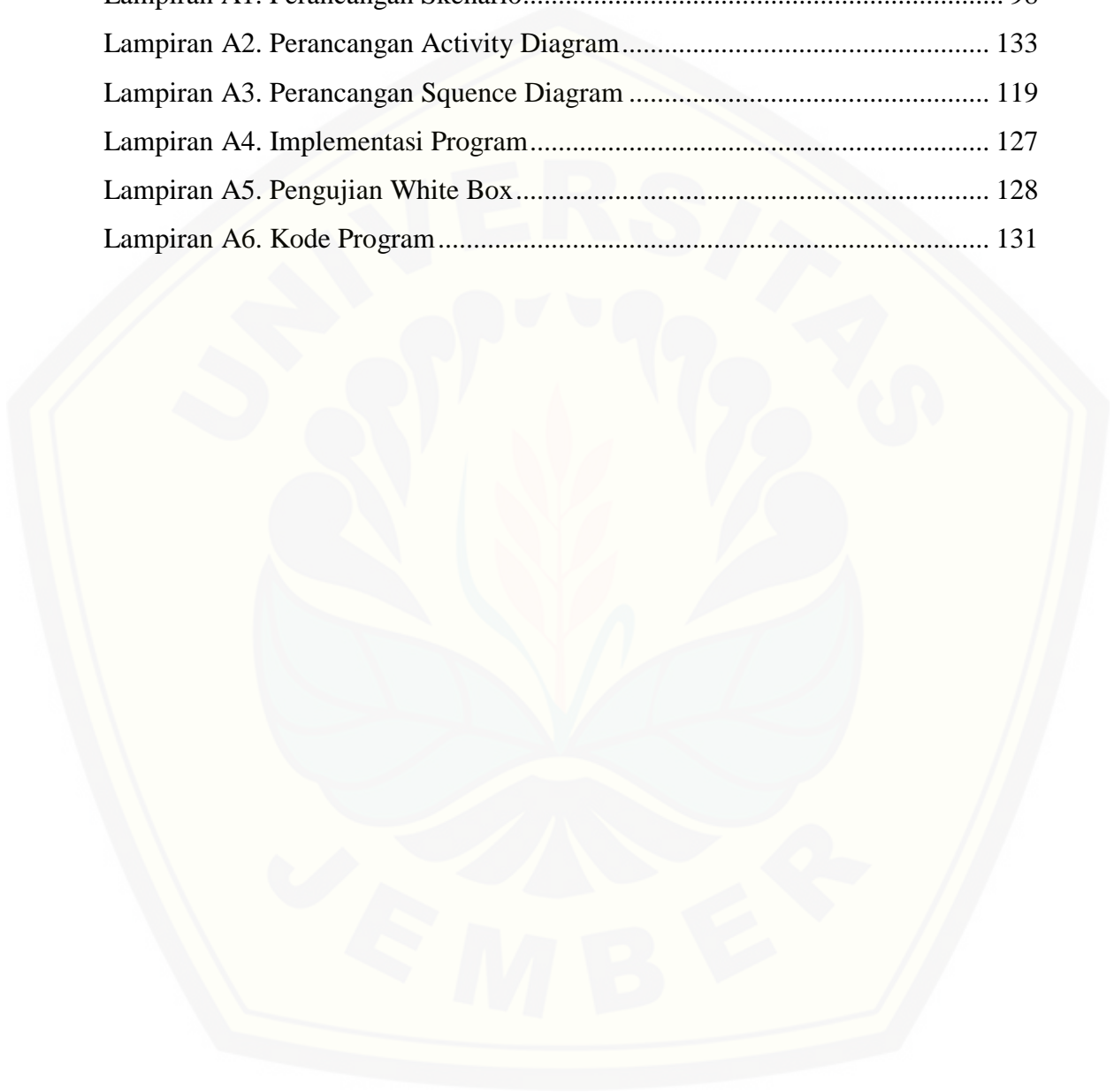
Gambar 5.10 <i>Form edit data darah masuk</i> .....	59
Gambar 5.11 Peringatan hapus data darah masuk .....	59
Gambar 5.12 halaman data darah keluar.....	60
Gambar 5.13 Form tambah darah keluar.....	61
Gambar 5.14 Form edit data darah keluar.....	62
Gambar 5.15 Peringatan hapus data darah masuk .....	62
Gambar 5.16 Halaman awal Stok Darah.....	63
Gambar 5.17 Halaman Peramalan WMA Pada Data Darah Keluar .....	64
Gambar 5.18 Halaman Peramalan WMA Pada Data Darah Masuk .....	65
Gambar 5.19 Halaman Perhitungan Manual Weighted Moving Average .....	66

**DAFTAR TABEL**

Tabel 4.1 Definisi Aktor .....	22
Tabel 4.2 Definisi Usecase.....	23
Tabel 4.3 Skenario Input data darah masuk .....	24
Tabel 4.4 Skenario Edit data darah masuk.....	26
Tabel 4.5 Skenario Hapus data darah masuk .....	27
Tabel 4.6 Pengujian <i>Black box</i> .....	45
Tabel 5.1 Peramalan Perhitungan WMA Darah Masuk.....	67
Tabel 5.2 perhitungan MAD .....	74
Tabel 5.3 Perbandingan Nilai Aktual dan Prediksi .....	82

**DAFTAR LAMPIRAN**

Lampiran A1. Perancangan Skenario.....	98
Lampiran A2. Perancangan Activity Diagram.....	133
Lampiran A3. Perancangan Squence Diagram .....	119
Lampiran A4. Implementasi Program.....	127
Lampiran A5. Pengujian White Box.....	128
Lampiran A6. Kode Program.....	131



## **BAB 1. PENDAHULUAN**

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

### **1.1 Latar Belakang**

Darah menjadi kebutuhan yang setiap hari harus dipenuhi oleh beberapa rumah sakit di setiap daerah. Kebutuhan darah yang cukup besar, terkadang tidak bisa dipenuhi oleh Palang Merah Indonesia (PMI) pada waktu tertentu. PMI akan melakukan berbagai upaya untuk menjaga stok darah, setiap harinya PMI membutuhkan 100-130 kantong darah untuk memenuhi kebutuhan pasien.

Untuk mengantisipasi kebutuhan darah, Departemen Kesehatan membentuk PMI yang bertugas menyediakan darah yang layak untuk digunakan. Darah yang layak diberikan kepada pasien adalah darah yang telah lulus uji saring dan cocok dengan darah pasien. Dalam setiap pelayanan di PMI mulai dari pengambilan darah donor sampai dengan pemberian darah kepada pasien dilakukan pencatatan.

Pemenuhan ketersediaan darah merupakan suatu hal yang sangat penting. Stok darah yang terlalu menumpuk akan mengakibatkan kerugian yang tidak sedikit, karena darah memiliki masa kadaluarsa. Sebaliknya, jika stok darah terlalu sedikit akan menyebabkan tidak terpenuhinya kebutuhan darah untuk masyarakat dan akan membuat kepercayaan masyarakat terhadap pelayanan PMI menjadi menurun. PMI masih sering mengalami kekurangan persediaan darah sehingga pada beberapa kasus menyebabkan banyak nyawa melayang karena pada saat membutuhkan darah tidak ada persediaan yang cukup. Bahkan pada bulan-bulan tertentu, PMI sering mengalami kurangnya stok darah dimana pada bulan tersebut kecelakaan jalan raya sering terjadi, misalnya pada saat bulan Ramadhan. Selain bulan tersebut, bencana alam yang datang secara tiba-tiba selalu membuat PMI membutuhkan donasi yang lebih. Berbagai upaya telah dilakukan PMI dalam memenuhi stok darah salah satunya dengan cara melakukan program donor darah diberbagai macam tempat, seperti

program donor darah di sekolah, kantor pemerintahan, dan universitas yang melibatkan warga sekitar area tersebut tentunya dengan melakukan uji tes kesehatan terlebih dulu kepada pendonor.

Prediksi terhadap persediaan darah sangat diperlukan untuk mengatasi permasalahan terkait stok darah dan banyaknya permintaan darah. Peramalan (*forecasting*) adalah seni dan ilmu untuk memperkirakan kejadian di masa depan. Hal ini dapat dilakukan dengan melibatkan pengambilan data masa lalu dan menempatkannya ke masa yang akan datang dengan suatu bentuk matematis (Heizer dan Render, 2005). Dengan adanya peramalan tersebut PMI dapat menyediakan stok yang cukup untuk mengantisipasi banyaknya permintaan darah di masa yang akan datang. Peramalan diharapkan dapat meminimalisir adanya kelebihan dan kekurangan stok darah. Penerapan peramalan sebagai salah satu upaya mengontrol terhadap persediaan darah dinilai penting. Hal itu diperlukan untuk meminimalkan jumlah pemesanan dan penyimpanan darah yang harus ditanggung PMI. Untuk itu, diperlukan suatu metode peramalan yang tepat agar dapat menghasilkan sebuah perkiraan yang akurat terhadap persediaan darah. Pada penelitian ini peramalan kebutuhan darah dilakukan dengan menggunakan metode *Weighted Moving Average* karena pola data stok dan permintaan darah yang dimiliki oleh PMI mengikuti pola data *trend*. Peramalan jumlah order oleh PMI Jember menggunakan data masa lalu selama 3 tahun, yaitu periode 2013. Penggunaan metode WMA pada peramalan jumlah penerimaan darah dan jumlah pengeluaran darah diharapkan mampu menjadi solusi atas permasalahan persediaan darah dengan meramalkan jumlah darah PMI pada bulan selanjutnya.

## 1.2 Rumusan Masalah

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

1. Bagaimana cara mengetahui peramalan jumlah stok darah di PMI Jember menggunakan metode *Weighted Moving Average*?

2. Bagaimana merancang dan membuat sistem peramalan stok darah PMI menggunakan WMA dalam menentukan stok darah perbulan ?

### 1.3 Tujuan

Tujuan dari penelitian ini adalah :

1. Untuk mengetahui cara peramalan jumlah stock darah PMI menggunakan metode *Weighted Moving Average*.
2. Untuk mengetahui tingkat keakuratan penggunaan metode *Weighted Moving Average* dalam meramalkan jumlah stock darah PMI.

### 1.4 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Diharapkan dapat membantu PMI pencatatan darah masuk dan darah keluar.
2. Dengan menerapkan system informasi peramalan maka dapat meminimalisir terjadinya kerugian PMI akibat pengadaan darah berlebih dan kurangnya persediaan darah.
3. Diharapkan dapat menjadi dasar bagi peneliti pada penelitian selanjutnya untuk pengembangan system informasi peramalan yang lebih baik.

### 1.5 Sistematika Penulisan

Sistematika penulisan dalam penyusunan tugas akhir ini adalah sebagai berikut:

1. Pendahuluan  
Bab ini terdiri atas latar belakang, rumusan masalah, tujuan, manfaat dan sistematika penulisan.
2. Tinjauan Pustaka  
Bab ini berisi tentang kajian materi dan informasi apa saja yang digunakan dalam penelitian ini. Dimulai dari kajian pustaka mengenai penjelasan dari aplikasi sistem prediksi jumlah stok darah.
3. Metodologi Penelitian



Bab ini menguraikan tentang metodologi penelitian apa yang dilakukan selama penelitian. Dimulai dari tahap pencarian permasalahan hingga pengujian aplikasi prediksi jumlah stok darah yang akan dibuat.

#### 4. Desain dan Pengembangan Aplikasi

Bab ini akan menguraikan tentang desain dan pengembangan aplikasi pembuatan aplikasi secara keseluruhan. Proses perancangan aplikasi dimulai dari analisis kebutuhan fungsional dan non-fungsional aplikasi, dilanjutkan dengan pembuatan *usecase diagram*, skenario *usecase*, *sequence diagram*, *activity diagram*, *class diagram* dan *entity relationship diagram* (ERD).

#### 5. Hasil dan Pembahasan

Bab ini menjelaskan hasil dan pembahasan dari penelitian yang telah dilakukan. Dengan menampilkan hasil penelitian dan hasil percobaan pengimplementasian aplikasi.

#### 6. Penutup

Bab ini berisi kesimpulan dari penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.

## BAB 2. TINJAUAN PUSTAKA

Tinjauan pustaka akan menjelaskan teori-teori serta pustaka yang dipakai pada saat penelitian, dan akan dijelaskan mengenai metode yang digunakan untuk penelitian. Berikut ini merupakan teori-teori yang akan digunakan dan dibahas dalam penelitian.

### 2.1 Kajian Penelitian Terdahulu

Bagian ini menjelaskan penelitian-penelitian sebelumnya yang memiliki relevansi dengan sistem informasi peramalan yang menggunakan metode *weighted moving average*. Penelitian yang berjudul *Prediksi Daya Listrik Geothermal Power Plant Berdasarkan Metode Weighted Moving Average di PT. GEO DIPA ENERGI Unit Dieng* dilakukan oleh Katherin Indrawati Junaldi yang merupakan mahasiswa Institut Teknologi Sepuluh November. Penelitian tersebut menjelaskan tentang metode *Weighted Moving Average* (WMA) berdasarkan nilai *Root Mean Square Error* (RMSE) untuk memprediksi laju aliran massa yang keluar dari separator menuju demister. Metode ini dianggap paling cocok untuk memprediksi data *time series* karena hasil prediksi dapat mengikuti *trend* dengan baik. Sehingga diharapkan dengan memprediksi laju aliran massa pada demister maka dapat diketahui nilai daya listrik yang dihasilkan dan juga efisiensi *plant*.

Penelitian yang kedua adalah Sistem Peramalan Persediaan Barang Dengan *Weighted Moving Average* Di Toko The Kids 24. Toko The Kids merupakan sebuah toko yang menyediakan keperluan anak dengan tingkat penjualan yang semakin meningkat dari tahun ke tahun namun dalam pencatatan persediaan barang masih dilakukan secara manual yang kemudian pada akhir bulan hasil pencatatan tersebut akan dilaporkan kepada pemilik toko guna melakukan pengadaan barang yang diperlukan untuk stok toko. Dalam hal pengadaan barang, pemilik toko masih menggunakan perkiraan dalam melakukan pemesanan barang kepada supplier yang

dapat berdampak pada ketidaksesuaian kebutuhan barang toko. Dalam menghadapi persaingan bisnis sejenis yang semakin ketat, pemilik toko harus memiliki strategi yang dapat mempermudah pengadaan barang dengan melakukan perhitungan perkiraan jumlah barang yang harus dibeli dengan menggunakan metode perhitungan tertentu yang dapat menghasilkan jumlah barang yang harus dibeli dengan tingkat akurasi yang tinggi. Metode yang digunakan dalam perhitungan prediksi jumlah barang yang harus dibeli oleh pemilik toko adalah metode *weighted moving average* dengan pembobotan tiga dan lima periode yang kemudian dibandingkan dengan mencari nilai error menggunakan metos MAD, MSE, dan MAPE. Hasil menunjukkan bahwa WMA dengan pembobotan tiga periode lebih tinggi nilai akurasinya dibandingkan dengan pembobotan WMA lima periode.

Penelitian yang ketiga berjudul Sistem Penjualan Dan Analisa Peramalan Untuk Penjualan Produk Elektronik Dengan Metode *Weighted Moving Average* dilakukan oleh Mirsa Hadi yang merupakan mahasiswa Program Studi Teknik Informatika di UPN Veteran Yogyakarta. Dipenelitian ini adalah bagaimana menerapkan metode *weighted moving average (WMA)* dalam memecahkan masalah peramalan penjualan produk elektronika pada periode berikutnya dengan studi kasus produk *merk Sharp*. WMA merupakan model rata-rata bergerak terbobot yang lebih responsif terhadap perubahan, karena data dari periode yang baru biasanya diberi bobot lebih besar. Konsep dari teori tentang metode *moving average* diatas sudah dapat diterapkan dalam berbagai kebutuhan salah satunya dalam penjualan produk, dalam industri elektronika perlu adanya studi khusus dari perusahaan tentang selera dan daya beli konsumen terhadap produk yang dibuat demi kebutuhan pasar. Hal tersebut juga yang harus dilakukan oleh perusahaan *merk sharp* dimana perlu bagaimana meramalkan penjualan produk di masa mendatang berdasarkan data yang telah direkam sebelumnya. Sedangkan untuk menghitung ketepatan peramalan dlam pembahasan ini menggunakan metode Nilai Rata-rata Kesalahan (*Mean Absolute Deviation*) yang sering disebut MAD.

Pola data stok dan permintaan darah di PMI Jember, yaitu pola data *trend*. Data *trend* yaitu apabila data dalam jangka panjang mempunyai kecenderungan, baik yang arahnya meningkat dari waktu ke waktu maupun menurun. Metode WMA digunakan sebagai prediksi daya listrik dapat diadopsi untuk meprediksi kebutuhan darah di PMI Jember karena dengan memprediksi jumlah stok darah yang masih tersedia di PMI Jember akan sangat membantu untuk mangantisipasi kekurangan stok darah yang ada dan tingkat kadaluarsa stok darah yang lama tak terpakai. Maka dari itu untuk dapat memprediksi jumlah kebutuhan dan ketersediaan stok darah yang ada dapat digunakan metode WMA dan dari penelitian kedua juga dapat diadopsi untuk menghitung nilai rata-rata kesalahan (*Mean Absolute Deviation*) yang disering disebut MAD sehingga hasil dari perhitungan metode WMA lebih tepat dan akurat.

## 2.2 Metode *Weighted Moving Average*

Peramalan (*forecasting*) merupakan salah satu unsur yang sangat penting dalam pengambilan suatu keputusan. Efektif atau tidaknya suatu keputusan pada umumnya tergantung pada beberapa factor yang tidak dapat dilihat pada saat keputusan tersebut diambil (Soejoeti, 1987).

Salah satu cara untuk meramalkan data time series yang memiliki trend linear adalah dengan menggunakan *weighted moving average*. Metode ini secara tidak langsung dinamakan set pertama dihitung *moving averagenya* dan set kedua dihitung sebagai *moving average* dari set pertama. Pertama kali dilakukan *moving average* kemudian baru dilakukan lagi *moving average* untuk data yang tadi yang sudah di *moving average* pertama kali (Harjo, 2014). Berikut ini adalah rumus yang dipakai pada peramalan ini yaitu:

$$WMA = \frac{\sum(\text{bobot untuk periode } n)(\text{permintaan untuk periode } n)}{\sum \text{bobot}} \quad (1)$$

Keterangan:

n = jumlah periode dalam rata-rata bergerak tertimbang.

### 2.2.1 Relevansi Metode Weighted Moving Average Terhadap Peramalan Jumlah Stok Darah Pada PMI

Berdasarkan permasalahan yang dialami oleh PMI mengenai prediksi jumlah darah masuk dan darah keluar untuk menanggulangi adanya kelebihan dan kekurangan stok darah, maka peneliti menerapkan peramalan dengan metode *weighted moving average*.

Penerapan metode *weighted moving average* dipilih berdasarkan pola data yang dimiliki oleh PMI, yaitu pola data trend. Penggunaan metode ini sangat tepat karena mengolah data yang bersifat non stasioner. Perhitungan yang ganda dapat menambah nilai keakuratan hasil peramalan dari metode tersebut. Metode WMA tepat untuk digunakan pada objek yang memiliki data 3 tahun. Sehingga penggunaan metode ini menjadi sebuah solusi dalam pemecahan permasalahan penentuan jumlah stok darah untuk mengontrol jumlah stok darah yang dimiliki di masa yang akan datang.

### 2.2.2 Metode Tingkat Pengukuran Tingkat Error Suatu Hasil Peramalan

Dalam sistem peramalan ada beberapa perhitungan yang biasa digunakan untuk menghitung kesalahan peramalan (*forecast error*) total. Perhitungan ini dapat digunakan untuk membandingkan model peramalan yang berbeda, juga untuk mengawasi peramalan, untuk memastikan peramalan berjalan baik tiga dari perhitungan yang paling terkenal adalah Deviasi Rata-rata Absolut (*mean absolute deviation-MAD*), (Heizer dan Render, 2001).

1. Deviasi Rata-rata Absolut (*Mean Absolute Deviation = MAD*). MAD merupakan ukuran pertama kesalahan peramalan keseluruhan untuk sebuah model. Nilai ini dihitung dengan mengambil jumlah nilai *absolute* dari tiap kesalahan peramalan dibagi dengan jumlah periode data (*n*). Rumus menghitung MAD adalah sebagai berikut :

$$MAD = \frac{\sum_{t=1}^n |X_t - F_t|}{n} \quad (2)$$

Keterangan :

$X_t$  = Data actual pada periode t

$F_t$  = Nilai peramalan pada periode t

N = Jumlah Data

2. Kesalahan Rata-rata Kuadrat (*Mean Square Error = MSE*) MSE merupakan cara kedua untuk mengukur kesalahan prediksi keseluruhan. MSE merupakan rata-rata selisih kuadrat antara nilai yang diramalkan dan yang diamati. Kekurangan penggunaan MSE adalah bahwa metode ini cenderung menonjolkan deviasi yang besar karena adanya pengkuadratan. Rumus [3], MSE dapat dirumuskan sebagai berikut :

$$MSE = \frac{\sum_{t=1}^n |n_t - 1 (X_t - F_t)^2}{n} ; \text{ untuk deviasi rata-rata kuadrat (3)}$$

Keterangan :

$X_t$  = Data actual pada periode t

$F_t$  = Nilai prediksi pada periode t

N = Jumlah Data

3. *Mean Absolute Percentage Error (MAPE)* merupakan pengukuran kesalahan yang menghitung ukuran presentase penyimpangan antara data actual dengan data prediksi. Rumus[4], MAPE dapat dirumuskan sebagai berikut :

$$MAPE = (100/n) \sum_{t=1}^n \frac{|n_t - 1 X_t - F_t|}{n} ; \text{ untuk deviasi presentasi absolut (4)}$$

Keterangan :

$X_t$  = Data actual pada periode t

$F_t$  = Nilai prediksi pada periode t

N = Jumlah Data

### BAB 3. METODOLOGI PENELITIAN

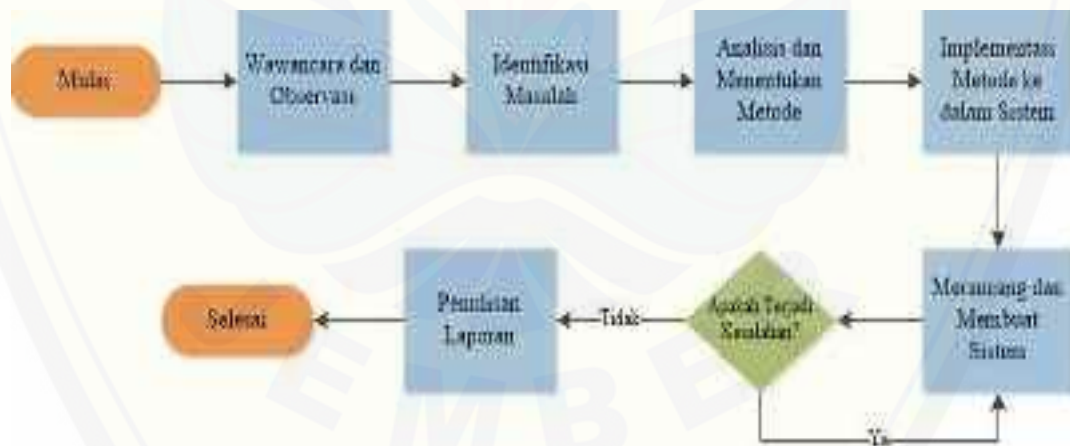
Metodologi penelitian ini dapat memberikan informasi mengenai langkah-langkah yang dilakukan untuk membangun sistem ini. Pada bagian ini disajikan tentang waktu dan tempat penelitian serta penjelasan mengenai tahap penelitian.

#### 3.1 Waktu dan Tempat Penelitian

Waktu dan tempat penelitian yang dilakukan dalam penelitian ini yaitu selama kurang lebih 5 bulan, dimulai pada bulan Juli sampai dengan bulan November 2015 dan bertempat di Palang Merah Indonesia Jember.

#### 3.2 Tahapan Penelitian

Dalam tahap penelitian dilakukan dengan tahap diantaranya pengumpulan data, dengan tahap analisis dan tahap perancangan system. Tahap ini dilakukan dengan diagram alir pada gambar 3.1.



Gambar 0.1 Diagram Alir Penelitian

Pada tahapan awal untuk memulai penelitian, peneliti melakukan wawancara kepada pihak-pihak terkait dan sekaligus untuk mendapatkan data-data yang diperlukan untuk membuat sebuah aplikasi. Tahapan kedua setelah melakukan wawancara yaitu dengan melakukan pengidentifikasian masalah. Setelah

mengidentifikasi masalah pada data yang diperoleh, tahapan selanjutnya yaitu menentukan metode dan menganalisis data yang telah didapatkan untuk merancang dan membuat sebuah aplikasi. *Testing* atau pengujian dilakukan setelah perancangan dan pembuatan aplikasi, jika aplikasi belum sesuai dengan rencana yang diharapkan maka kembali ke tahapan mengumpulkan data sedangkan jika hasil pengujian telah sesuai dengan apa yang direncanakan maka akan berlanjut ke tahap penulisan laporan dan penelitian telah selesai dan untuk lebih jelasnya akan dijelaskan pada sub-bab berikut ini.

### **3.4.1 Tahap Pengumpulan Data**

Tahap pengumpulan data dalam penelitian ini sebagai berikut :

#### **3.4.1.1 Wawancara**

Penelitian ini menggunakan wawancara tidak berstruktur. Wawancara tidak berstruktur adalah wawancara yang hanya memuat garis besar persoalan yang ada relevansinya dengan dengan permasalahan pokok dalam penelitian ini. Pengertian wawancara itu sendiri adalah percakapan dengan maksud tertentu. Percakapan itu dilakukan oleh dua pihak, yaitu pewawancara (*Interviewer*) yang mengajukan pertanyaan dan terwawancara (*Interviewee*) yang memberikan jawaban atas pertanyaan (Moleong, 2011). Dalam penelitian ini, wawancara dilakukan kepada Direktur PMI yang bertanggung jawab atas semua proses persediaan seperti menempatkan pesanan dengan pelayanan darah, pemesanan unit darah untuk pasien, tes darah untuk kompatibilitas dengan pasien (pencocokan), dan memastikan bahwa kualitas standar kantong darah terpenuhi dan dapat memenuhi permintaan sesuai kebutuhan pasien karena dengan melakukan wawancara ini peneliti dapat menerima secara langsung informasi detail mengenai jumlah kebutuhan stok dan permintaan darah yang akan di teliti.



### 3.4.1.2 Studi Literature

Studi *literature* merupakan tahapan pengumpulan data dan informasi yang diperlukan untuk proses perancangan sistem. Data dan informasi dapat diperoleh dari lokasi penelitian yaitu PMI Jember. Selain itu studi *literature* juga dapat diperoleh melalui jurnal ilmiah, *paper*, serta berbagai buku referensi yang berkaitan dengan penelitian.

### 3.4.1.2 Dokumentasi

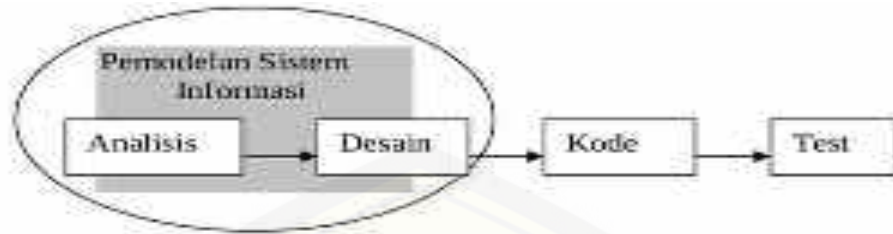
Dokumentasi dilakukan dengan cara pengumpulan data melalui arsip-arsip, termasuk juga buku mengenai pencatatan tentang jumlah kebutuhan stok darah yang diperoleh dari arsip kegiatan donor darah.

### 3.4.2 Identifikasi Masalah

Permasalahan yang akan diteliti pada penelitian kali ini adalah bagaimana meramalkan jumlah stok darah yang ada di PMI Jember dalam jangka waktu tiga tahun. Pengolahan data yang digunakan untuk dapat menentukan peramalan tersebut dengan data yang ada pada tahun sebelumnya yaitu tahun 2012 dan akan diramalkan (*forecasting*) dengan menggunakan metode *Weighted Moving Average* (WMA).

### 3.4.3 Perancangan Sistem

Perancangan dilakukan untuk memberikan gambaran secara umum mengenai sistem yang akan dibangun. Pembuatan perancangan perangkat lunak ini mengadopsi metode *waterfall*. Metode *waterfall* merupakan metode yang sistematis dan sekuensial yang mulai pada tingkat dan kemajuan sistem sampai pada analisis, desain, kode, dan test. (Roger S. Pressman, 2002). Tahapan *Waterfall* digambarkan pada gambar 3.2.



Gambar 0.2 Tahapan Metode Waterfall

(Sumber : Roger S. Pressman , 2002)

Tahap pertama pada proses perancangan perangkat lunak ini adalah sebagai berikut:

a. Analisis Kebutuhan

Pada tahap ini, peneliti mencari permasalahan yang ada untuk dapat dianalisis kebutuhan yang diperlukan, sebagai solusi dari permasalahan yang muncul. Data-data yang telah didapat kemudian dikelompokkan menjadi kebutuhan fungsional dan non-fungsional. Setelah itu, menentukan fungsi dan fasilitas apa saja yang akan dibuat dalam aplikasi.

b. Desain Sistem

Jika proses analisis kebutuhan telah diketahui maka proses selanjutnya adalah pada tahapan desain aplikasi. Proses pendesainan aplikasi yang akan dibangun yaitu dengan menggunakan *Unified Modeling Language (UML)*. Penggunaan UML karena sudah menggunakan konsep *Object Oriented Design* yang tentunya akan sangat memudahkan developer untuk membangun sebuah aplikasi. Dalam UML ada beberapa diagram yang akan dibuat antara lain:

1. *Business Process*

*Business process* merupakan diagram yang menggambarkan proses dari sebuah sistem yang meliputi *input*, *output*, dan *goal* yang merupakan tujuan dari sebuah sistem yang dibangun.

2. *Usecase Diagram*

*UseCase Diagram* menggambarkan fungsionalitas dari sebuah sistem dan interaksi antara *user* dengan sistem untuk melakukan suatu pekerjaan tertentu. *Use Case Diagram* juga menggambarkan hak akses dari *user*.

3. *Scenario*

*Scenario* digunakan untuk menjelaskan fitur yang ada pada *usecase diagram*.

4. *Activity Diagram*

*Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

5. *Sequence Diagram*

*Sequence diagram* (diagram urutan) adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek didalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, display, dan sebagainya berupa pesan atau message.

6. *Class Diagram*

*Class Diagram* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

7. *Entity Relationship Diagram (ERD)*

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.

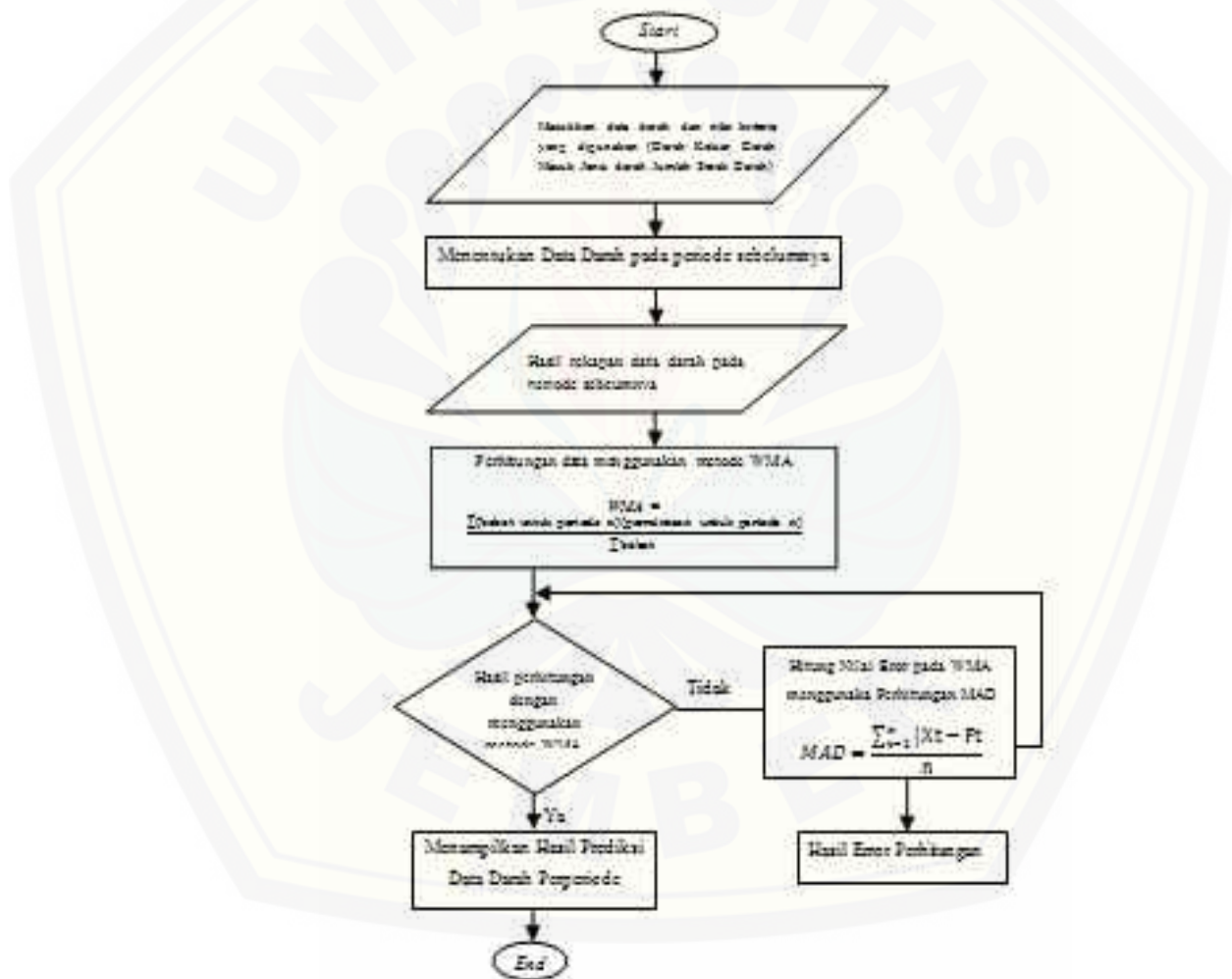
### 3.3 Coding (Pengkodean)

Setelah proses desain system dikerjakan, proses selanjutnya adalah *coding* atau penulisan kode program. Bahasa pemrograman yang dipakai adalah php (*Hypertext Preprocessor*) dengan *Code Igniter Framework*, sedangkan *tool* yang digunakan

adalah Netbeans. Database yang digunakan yaitu MySQL dengan *tool* yang digunakan yaitu XAMPP.

### 3.4 Implementasi Penerapan Metode *Weighted Moving Average* pada sistem

Sistem dibangun menggunakan *framework codeigniter*, *coding* metode diletakkan pada *controller*. *Weighted Moving Average* diterapkan pada sistem sesuai dengan perhitungan metode *Weighted Moving Average*.



Gambar 3.4 Weighted Moving Average yang implemented

Pada gambar 3.4 menjelaskan tentang proses alur kerja sistem permalan jumlah stok darah pada PMI dengan menginputkan data kriteria yaitu golongan darah, darah masuk, darah keluar, dan jumlah stok darah. Tahap selanjutnya menginputkan nilai bobot dari setiap kriteria yang ada sehingga nilai yang diinputkan dihitung menggunakan metode WMA dan memperoleh hasil akhir untuk prediksi dibulan berikutnya. Data-data yang diinputkan diperoleh dari hasil pengumpulan data serta wawancara pada pihak-pihak terkait.

### 3.5 Pengujian

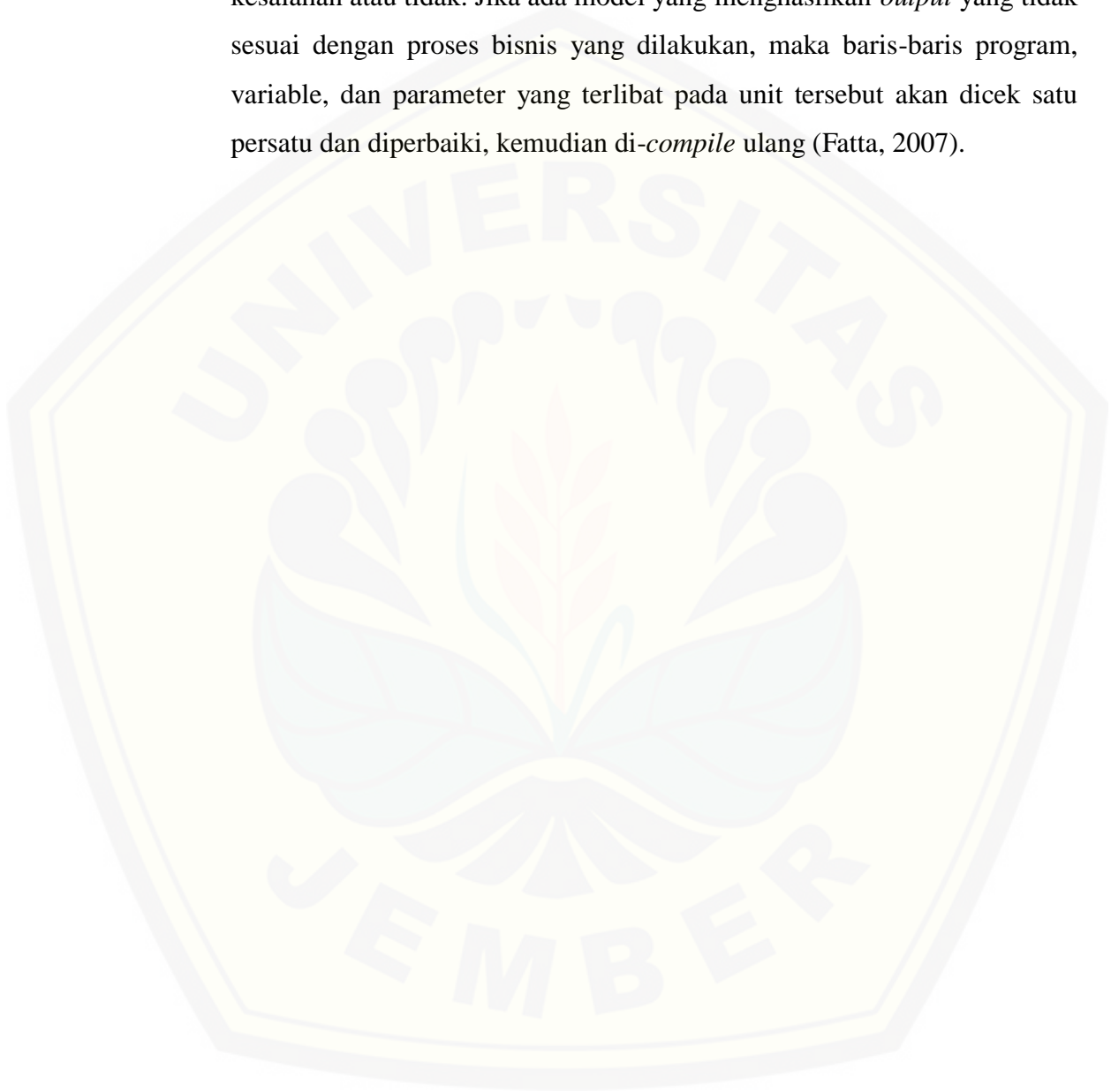
Pengujian ini dilakukan uji coba sistem yang telah dibuat dengan pengujian *white box* dan *black box*. Pengujian *white box* adalah cara pengujian dengan meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak sedangkan *black box* merupakan cara pengujian dengan melakukan *running* program dengan menguji coba berbagai kemungkinan kesalahan yang ada.

1. *Black box Testing*

*Black Box Testing* adalah metode pengujian perangkat lunak yang memeriksa fungsionalitas dari aplikasi yang berkaitan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi atau struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Metode ini memfokuskan pada keperluan fungsionalitas dari software (Wildan Agissa, 2013). Pada pengujian *black box* ini, aplikasi yang dibangun pada penelitian ini akan diuji dengan mengujikan langsung *running* aplikasi dan melakukan kegiatan pengujian dengan menganalisis proses *input* dan *output* yang dihasilkan aplikasi. Dalam metode *black box* juga dilakukan pengujian dengan cara memasukkan data normal dan data salah, dari *penginputan* ini nantinya akan dilakukan analisis terhadap reaksi yang muncul pada aplikasi.

2. *White box Testing*

*White box testing* adalah cara pengujian dengan melihat ke dalam modul untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Jika ada model yang menghasilkan *output* yang tidak sesuai dengan proses bisnis yang dilakukan, maka baris-baris program, variable, dan parameter yang terlibat pada unit tersebut akan dicek satu persatu dan diperbaiki, kemudian di-*compile* ulang (Fatta, 2007).



## BAB 4. DESAIN DAN PENGEMBANGAN SISTEM

Bab ini menjelaskan proses pengembangan sistem , mengimplementasikan metode *Weighted Moving Average (WMA)* untuk sistem memprediksi jumlah stok darah pada PMI. Proses pengembangan sistem ini menggunakan model pengembangan sistem Waterfall sesuai dengan yang dijelaskan pada bab 3.

### 4.1 Deskripsi Umum Sistem

Deskripsi umum dari sistem dalam penelitian ini akan dijelaskan lebih detail pada SOP (*Statement Of Perpose*) sistem dan karakteristik sistem.

#### 4.1.1 Statement of Purpose

Sistem Peramalan Jumlah Stok Darah PMI Menggunakan Metode *Weighted Moving Average* merupakan sistem yang digunakan untuk membantu instansi terkait yang masih kesulitan dalam menentukan stok darah yang mengalami kekurangan atau kelebihan darah selama satu bulan, sehingga memudahkan instansi terkait dalam menentukan stock darah.

#### 4.1.2 Karakteristik Sistem

Karakteristik sistem ini menjelaskan fungsi – fungsi yang dimiliki oleh sebuah sistem pakar, meliputi :

a. Fungsi *Autentifikasi*

Fungsi autentifikasi ini merupakan proses *login* dan *logout* dari sistem. Fungsi *login* dimaksudkan agar keamanan data yang dimiliki dapat terjamin kerahasiaannya melalui proses tersebut. Proses *login* ini hanya bisa diakses oleh user yang terkait dengan memiliki sebuah username dan password.

Sedangkan fungsi *logout* dimaksudkan agar setelah user selesai mengolah data maka diwajibkan untuk *logout* dari akunnya agar tidak diretas oleh user

lainnya. Fungsi ini pada dasarnya sama yaitu untuk melindungi dan membatasi hak akses user terhadap data yang dimiliki.

b. Fungsi *Input, Update, Delete*

Fungsi *input, update* dan *delete* data merupakan fungsi dasar dari sebuah sistem. Sistem digunakan untuk memudahkan user dalam mengolah data. Data yang dimaksud adalah data yang nantinya akan diolah, dan menghasilkan *output* yang diinginkan. Data tersebut harus diinputkan terlebih dahulu agar bisa diolah.

Data yang sudah melalui prose *input* sebaiknya dapat di perbaiki lagi atau dapat melakukan proses *update*. Begitu juga dengan fungsi *delete*, agar data bisa dihapus untuk meminimalisir data yang ganda, atau kesalahan data.

## 4.2 Analisis Kebutuhan

Proses analisis kebutuhan sistem pada penelitian ini dengan cara permasalahan yang ada, kemudian dipahami agar dapat dijadikan bahan untuk membangun sistem penunjang penentuan lahan tanam. Analisis kebutuhan yang akan dilakukan meliputi analisis kebutuhan fungsional dan analisis kebutuhan non fungsional.

### 4.2.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan hal pokok yang harus dapat dilakukan oleh sistem dalam menerima masukan untuk diproses sehingga menghasilkan luaran yang sesuai dengan kebutuhan inti sistem. Kebutuhan fungsional dari sistem penunjang penentuan lahan tanam antara lain :

1. Sistem dapat memanajemen user
2. Sistem dapat *reset* stok darah perbulan
3. Sistem dapat mengelola data darah masuk
4. Sistem dapat mengelola data darah keluar
5. Sistem dapat menampilkan rekap data stok darah



6. Sistem dapat menampilkan hasil perhitungan darah dari perhitungan *Weighted Moving Average*

#### 4.2.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional dari sistem ini dibuat guna mendukung aktivitas sistem dengan kebutuhan-kebutuhan fungsional yang telah dibuat sebelumnya. Kebutuhan non-fungsional dari sistem ini yaitu :

1. Sistem berbasis *website*
2. Sistem memiliki tampilan dan bahasa yang mudah dipahami semua user, khususnya bahasa yang mudah dipahami pihak terkait.
3. Sistem memiliki batasan hak akses pengguna dengan menggunakan *username* dan *password*

#### 4.3 Desain Sistem

Desain sistem merupakan tahapan untuk memodelkan sistem informasi penunjang keputusan yang akan dibuat. Dokumen desain sistem yang akan dibuat meliputi *Business process*, *Use case diagram*, skenario, *Activity diagram*, *Sequence diagram*, *Class diagram* dan *Entity Relationship Diagram* (ERD).

##### 4.3.1 Business Process

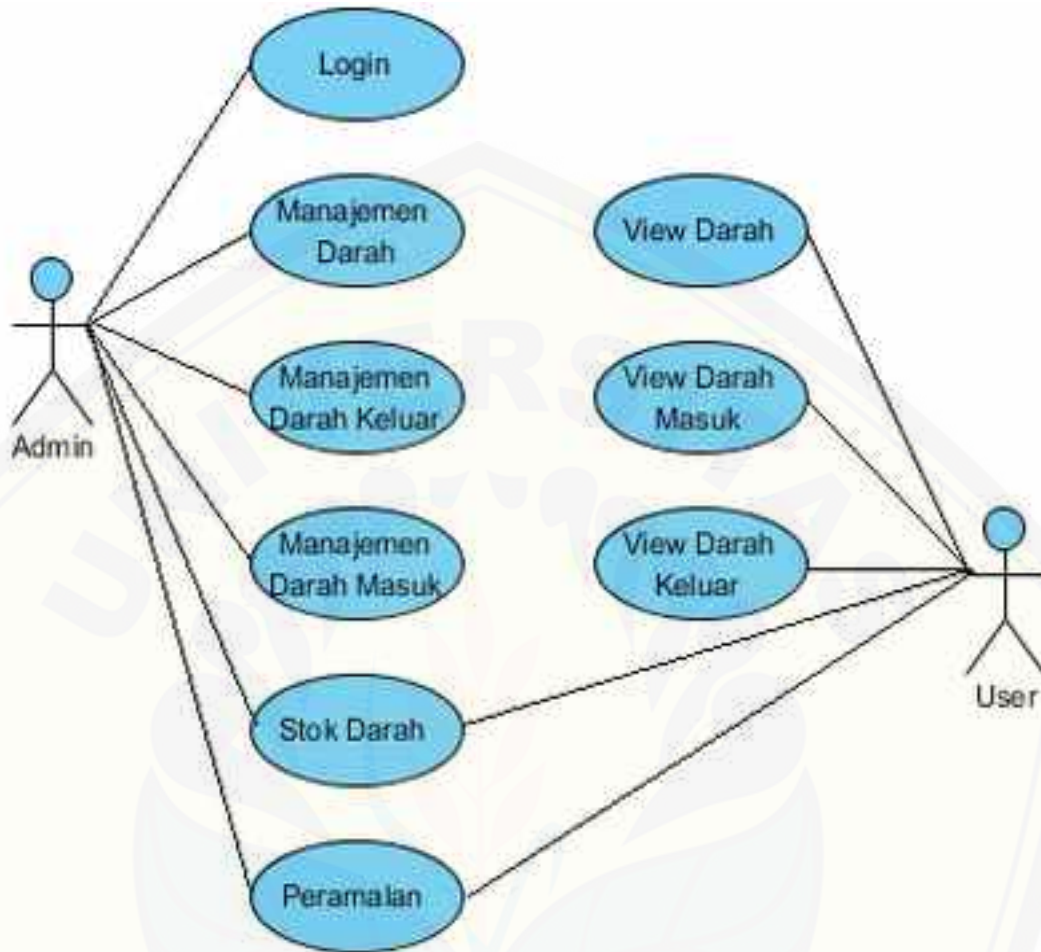
*Business Process* digunakan untuk menggambarkan masukkan data yang dibutuhkan sistem (*input*) , keluaran dari sistem (*output*) , media yang digunakan oleh sistem (*uses*) , serta tujuan dari pembuatan sistem (*goal*). *Business Process* sistem peramalan jumlah stok darah pmi menggunakan metode *weighted moving average* menjelaskan sekumpulan proses yang dilakukan untuk mengetahui jumlah stok darah perbulan. Pada gambar 4.1 dijelaskan *input*, *output*, *uses* dan *goal* dari sistem yang akan dibangun.



Gambar 4.1 Bussiness Process

#### 4.3.2 Usecase Diagram

*Usecase diagram* digunakan untuk menggambarkan fitur dan aktor yang terdapat dalam sistem yang dikembangkan. *Usecase diagram* mendeskripsikan interaksi antara suatu aktor dengan sistem yang akan dibuat. *Usecase diagram* bertujuan untuk mengetahui fungsi apa saja yang dapat diakses oleh suatu aktor. *Usecase diagram* sistem dapat dilihat pada Gambar 4.2.



Gambar 4.2 Usecase Diagram

*Usecase* sistem sistem peramalan jumlah stok darah pmi menggunakan metode *weighted moving average* terdapat penjelasan berupa definisi aktor yang menjelaskan aktor siapa saja yang terdapat pada sistem tersebut dapat dilihat pada tabel 4.1, selain itu terdapat definisi *usecase* yang menggambarkan fungsionalitas dari setiap *usecase*. Definisi *usecase* dari sistem peramalan jumlah stok darah pmi menggunakan metode *weighted moving average* dapat dilihat pada tabel 4.1

Tabel 4.1 Definisi Aktor

No.	Aktor	Definisi Tugas
1	Admin	Admin merupakan aktor yang memiliki hak

		akses tertinggi dalam sistem. Aktor dapat melihat dan memanajemen beberapa aktivitas user dalam sistem seperti misalnya melakukan manajemen data darah, manajemen darah masuk, manajemen darah keluar, view stok darah, view stok darah, view peramalan dan dapat melakukan proses <i>login</i> .
2	Aktor	Aktor merupakan user yang memiliki hak akses untuk melihat data darah masuk, melihat data darah keluar, melihat data darah, melihat rekap stok darah perbulan, dan melihat hasil peramalan stok darah.

Tabel 4.2 Definisi Usecase

No.	Aktor	Definisi Tugas
1	Login	Merupakan <i>usecase</i> proses autentifikasi untuk masuk kedalam sistem.
2	Manajemen Darah	Merupakan <i>usecase</i> untuk untuk melihat stok darah perbulan yang akan di <i>reset</i> pada awal bulan selanjutnya (data berlaku hanya satu bulan).
3	Manajemen Darah Keluar	Merupakan proses manajemen ( <i>input</i> , edit, dan delete) data darah keluar dan view darah keluar.
4	Manajemen Darah Masuk	Merupakan proses manajemen ( <i>input</i> , edit, dan delete) data darah masuk dan view darah masuk.

5	View Peramalan	Merupakan <i>usecase</i> untuk melihat hasil data darah yang dihitung dalam sistem
6	View Darah	Merupakan <i>usecase</i> untuk melihat data darah di dalam sistem.
7	View Darah Keluar	Merupakan <i>usecase</i> untuk melihat data darah keluar di dalam sistem.
8	View Darah Masuk	Merupakan <i>usecase</i> untuk melihat data darah masuk di dalam sistem.
9	View Stok Darah	Merupakan <i>usecase</i> untuk melihat jumlah stok darah perbulan di dalam sistem.

### 4.3.3 Skenario Sistem

Skenario sistem merupakan gambaran lebih rinci dari *usecase*, skenario berfungsi menggambarkan alur dari sebuah sistem beserta alternatif yang dilakukan oleh aktor. Skenario dari sistem peramalan jumlah stok darah pmi menggunakan metode *weighted moving average* dapat diketahui pada tabel – tabel dibawah ini.

#### 4.3.3.1 Skenario Manajemen Data Darah Masuk - Input Data Darah Masuk

Skenario ini menjelaskan alur proses dari manajemen data darah masuk yaitu tambah data darah. Skenario utama merupakan alur utama dari proses tambah data darah. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses tambah data hama dan penyakit. Kondisi setelah skenario ini dijalankan adalah admin berhasil menambahkan data darah baru. Skenario tambah data darah dapat dilihat pada tabel 4.3 berikut.

Tabel 4.3 Skenario Input data darah masuk

<b>Nama</b>	<b>Input data darah masuk</b>
<b>Aktor</b>	<b>Admin</b>

<b>Pre-Condition</b>	- Admin duduk di depan computer - Menghidupkan computer
<b>Post-Condition</b>	- Data darah masuk telah ditambah
<b>Skenario Utama : Input Data Darah Masuk</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1. Pilih menu manajemen darah masuk	
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>
3. Memilih tombol "Tambah Data"	
	4. Menampilkan form tambah data darah
5. Mengisi Form tambah data baru dengan data darah dan mengklik tombol "Simpan"	
	6. Menyimpan data, menampilkan kembali informasi data darah yang baru saja ditambahkan
<b>Alternative Flow : Terdapat Data Kurang Lengkap</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
5b. Mengisi Form tambah data baru dengan data darah dan Mengklik tombol "Simpan"	
	6b. Menampilkan <i>alert</i> "data tidak boleh kosong" dan menampilkan kembali halaman form tambah data darah

--	--

#### 4.3.3.2 Skenario Manajemen Darah Masuk – Edit Data Darah

Skenario ini menjelaskan alur proses dari manajemen data darah mask yaitu edit data darah masuk. Skenario utama merupakan alur utama dari proses edit data darah masuk. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses edit data darah. Kondisi setelah skenario ini dijalankan adalah admin berhasil merubah data darah. Skenario edit data darah dapat dilihat pada tabel 4.4 berikut.

Tabel 4.4 Skenario Edit data darah masuk

<b>Nama</b>	<b>Edit data darah masuk</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	-	<b>Admin duduk di depan computer</b>
	-	<b>Menghidupkan computer</b>
<b>Post-Condition</b>	-	<b>Data darah masuk telah ter update</b>
<b>Skenario Utama : Edit Data Darah Masuk</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
1. Memilih menu “Darah Masuk”		
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>	
3. Memilih data yang akan diedit lalu mengklik tombol “Edit”		
	4. Menampilkan form edit data darah masuk	
5. Mengedit data darah masuk dan memilih tombol “Simpan”		
	6. Menyimpan data, menampilkan	

	kembali informasi data darah masuk yang sudah di edit
<b>Skenario Alternatif : Data kurang lengkap</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
5b. Mengklik tombol “Simpan”	
	6b. Menampilkan <i>alert</i> “data tidak boleh kosong” dan menampilkan kembali halaman form edit data darah

#### 4.3.3.3 Skenario Manajemen Darah Masuk – Edit Data Darah

Skenario ini menjelaskan alur proses dari manajemen data darah masuk yaitu hapus data darah. Skenario utama merupakan alur utama dari proses hapus data hama darah masuk. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses hapus data darah masuk. Kondisi setelah skenario ini dijalankan adalah admin berhasil menghapus data darah masuk. Skenario hapus data darah dapat dilihat pada tabel 4.5 berikut.

Tabel 4.5 Skenario Hapus data darah masuk

<b>Nama</b>	<b>Hapus data darah masuk</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	- Admin duduk di depan komputer - Menghidupkan computer
<b>Post-Condition</b>	- Data darah telah ter hapus
<b>Skenario Utama : Hapus Data Darah Masuk</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1. Memilih menu “Darah Masuk”	
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di



	<i>inputkan</i>
3. Memilih data darah masuk yang akan dihapus lalu klik tombol “delete”	
	4. Menampilkan dialog box konfirmasi yang berisi tombol ‘ok’ dan ‘cancel’
5. Mengklik tombol “delete” pada dialog box	
	6. Menghapus data dan menampilkan kembali data darah
<b>Skenario Alternatif : Pembatalan Hapus Data</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
5a. Mengklik tombol “cancel” pada dialog box	
	6a. Menampilkan kembali list data darah

#### 4.3.3.4 Skenario Peramalan

Skenario ini menjelaskan alur proses dari peramalan yang dilakukan oleh user. Skenario utama merupakan alur utama dari proses peramalan. Kondisi setelah skenario ini dijalankan adalah admin berhasil melakukan proses peramalan dan berhasil mendapatkan informasi hasil peramalan data darah keluar dan darah masuk. Skenario peramalan dapat dilihat pada tabel 4.6 berikut.

Tabel 4.6 Skenario Peramalan

<b>Nama</b>	<b>Peramalan</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	- <b>User duduk di depan computer</b>

	- <b>Menghidupkan computer</b>
<b>Post-Condition</b>	- <b>User mendapatkan hasil data darah dari peramalan</b>
<b>Skenario Utama : Peramalan</b>	
<b>User</b>	<b>Reaksi Sistem</b>
1. Memilih menu Peramalan	
	2. Menampilkan <i>dropdown</i> darah masuk dan darah keluar
3. Memilih <i>dropdown</i> darah masuk	
	4. Sistem menampilkan hasil peramalan darah masuk yang dilakukan oleh user
5. Memilih menu "Home"	
	6. Menampilkan halaman awal system
<b>Alternative Flow : Pembatalan Peramalan</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
6a. Memilih tombol Home	
	7a. Menampilkan kembali halaman awal system

#### 4.3.3.5 Skenario Edit dan Hapus Data Darah – Edit data darah

Skenario ini menjelaskan alur proses dari edit dan hapus data darah yaitu edit data darah. Skenario utama merupakan alur utama dari proses tambah data darah. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses tambah data darah. Kondisi setelah skenario ini dijalankan adalah admin berhasil menambahkan data darah. Skenario tambah data darah dapat dilihat pada lampiran A1.

#### 4.3.3.6 Skenario Edit dan Hapus Data Darah – Hapus data darah

Skenario ini menjelaskan alur proses dari edit dan hapus data darah yaitu hapus data darah. Skenario utama merupakan alur utama dari proses hapus data darah. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses hapus data darah. Kondisi setelah skenario ini dijalankan adalah admin berhasil menghapus data darah. Skenario hapus data darah dapat dilihat pada lampiran A1.

#### 4.3.3.7 Skenario Manajemen Data Darah Keluar - Input Data Darah Keluar

Skenario ini menjelaskan alur proses dari manajemen data darah keluar yaitu tambah data darah keluar. Skenario utama merupakan alur utama dari proses tambah data darah keluar. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses tambah data darah keluar. Kondisi setelah skenario ini dijalankan adalah admin berhasil menambahkan data darah keuar baru. Skenario tambah data darah keluar dapat dilihat pada lampiran A1.

#### 4.3.3.8 Skenario Manajemen Data Darah Keluar - Edit Data Darah Keluar

Skenario ini menjelaskan alur proses dari manajemen data darah keluar yaitu edit data darah keluar. Skenario utama merupakan alur utama dari proses edit data darah keluar. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses edit data darah keluar. Kondisi setelah skenario ini dijalankan adalah pakar berhasil merubah data darah keluar. Skenario edit data darah keluar dapat dilihat pada lampiran A1.

#### 4.3.3.9 Skenario Manajemen Data Darah Keluar – Hapus Data Darah Keluar

Skenario Skenario ini menjelaskan alur proses dari manajemen data darah keluar yaitu hapus data data darah keluar. Skenario utama merupakan alur utama dari proses hapus data data darah keluar. Sedangkan skenario alternatif merupakan bagian yang menangani *exception* atau alur alternatif dari proses hapus data data darah

keluar. Kondisi setelah skenario ini dijalankan adalah pakar berhasil menghapus data data darah keluar. Skenario hapus data data darah keluar dapat dilihat pada lampiran A1.

#### 4.3.3.10 Skenario *View* Darah

Skenario ini menjelaskan alur untuk melihat laporan data darah perbulan atau jumlah stok darah yang keluar dan darah masuk yang telah diinputkan. Skenario *view* data darah dapat dilihat pada lampiran A1.

#### 4.3.3.11 Skenario *View* Darah Masuk

Skenario ini menjelaskan alur untuk melihat laporan data darah masuk perbulan yang telah diinputkan dengan memilih laporan berdasarkan bulan dan tahun. Skenario *view* data darah masuk dapat dilihat pada lampiran A1.

#### 4.3.3.12 Skenario *View* Darah Keluar

Skenario ini menjelaskan alur untuk melihat laporan data darah keluarperbulan yang telah diinputkan dengan memilih laporan berdasarkan bulan dan tahun. Skenario *view* data darah keluar dapat dilihat pada lampiran A1.

#### 4.3.3.13 Skenario *View Stok* Darah

Skenario ini menjelaskan alur untuk melihat laporan data darah masuk dan darah keluar selama sebulan atau jumlah stok darah yang telah diinputkan dengan memilih laporan berdasarkan bulan dan tahun. Skenario *view* stok darah dapat dilihat pada lampiran A1.

### 4.3.4 Activity Diagram

*Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang akan dibangun, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* sistem peramalan jumlah

stok darah pmi menggunakan metode *weighted moving average* dapat diketahui pada tabel berikut.

#### 4.3.4.1 Activity Diagram Manajemen Data Darah Keluar - Input Data Darah Keluar

*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah keluar yaitu tambah data darah. Admin dapat menambahkan data darah baru kedalam sistem. Data yang sudah tersimpan akan ditampilkan dalam tabel data darah keluar. *Activity diagram* tambah data darah keluar dapat dilihat pada gambar 4.3 berikut.



Gambar 4.3 Activity Diagram Input Data Darah Keluar

#### 4.3.4.2 Activity Diagram Darah Keluar – Edit Data Darah

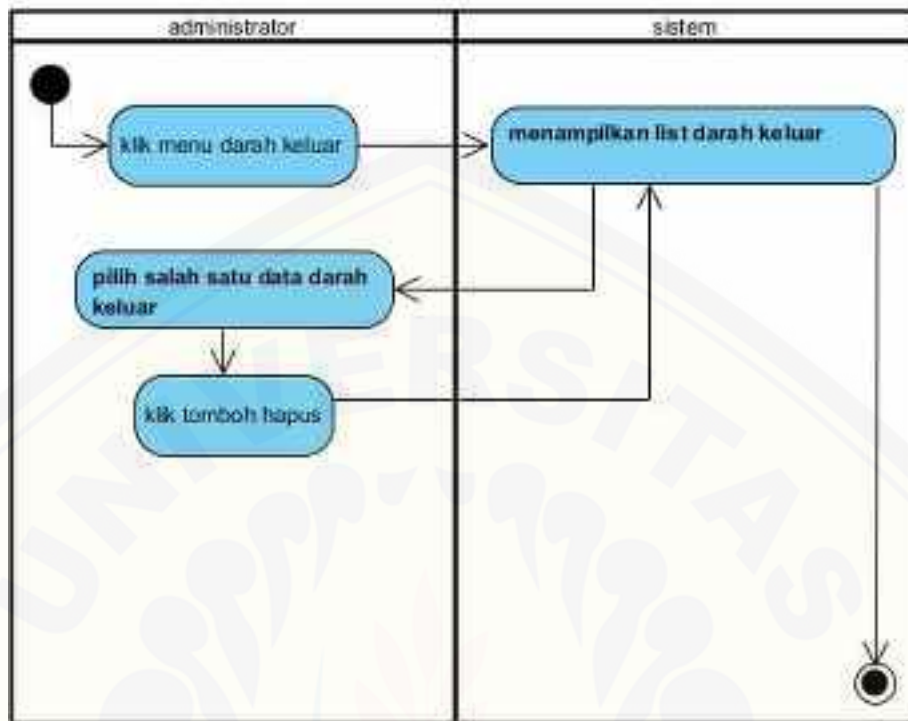
*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah keluar yaitu edit data darah keluar. Admin dapat mengubah data data darah keluar yang ada dengan data baru kedalam sistem. Data yang sudah tersimpan akan ditampilkan dalam tabel data darah keluar. *Activity diagram* edit data darah keluar dapat dilihat pada gambar 4.4 berikut.



Gambar 4.4 *Activity Diagram* Edit Data Darah Keluar

#### 4.3.4.3 *Activity Diagram* Darah Keluar – Hapus Data Darah

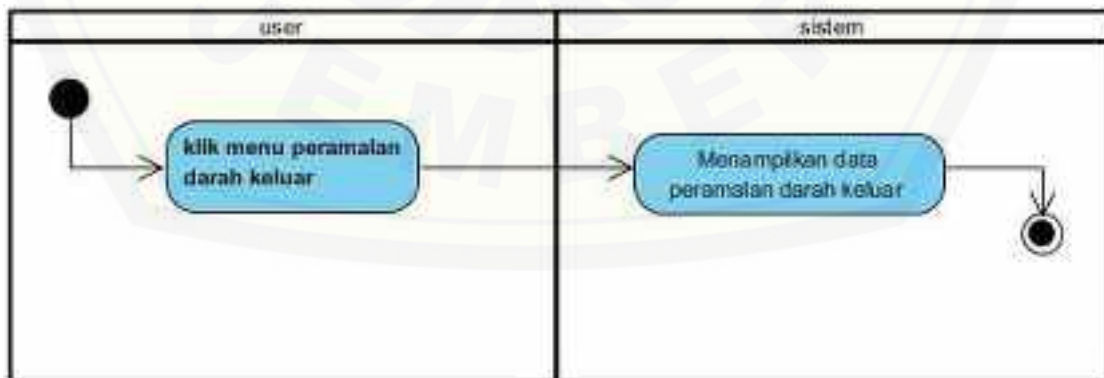
*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah keluar yaitu hapus data darah keluar. Admin dapat menghapus data darah keluar yang ada dalam sistem. Setelah data dihapus akan menampilkan kembali tabel data darah keluar. *Activity diagram* hapus data darah keluar dapat dilihat pada gambar 4.5 berikut.



Gambar 4.5 Activity Diagram Hapus Data Darah Keluar

#### 4.3.4.4 Activity Diagram Peramalan Darah Keluar

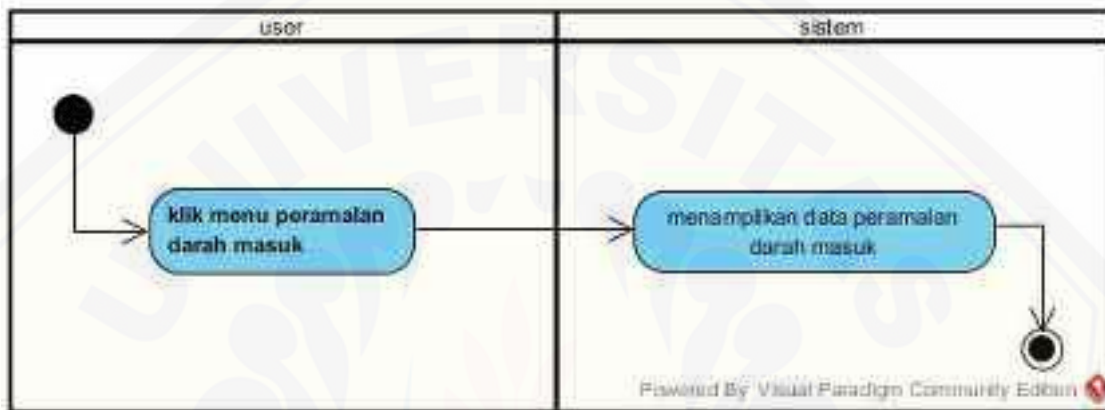
Activity diagram ini menjelaskan alur proses dari peramalan yang dilakukan oleh admin. Admin berhasil melakukan proses peramalan dan berhasil mendapatkan informasi hasil peramalan data darah keluar. Activity diagram peramalan dapat dilihat pada gambar 4.6 berikut.



Gambar 4.6 Activity Diagram Peramalan Darah Keluar

#### 4.3.4.5 Activity Diagram Peramalan Darah Masuk

*Activity diagram* ini menjelaskan alur proses dari peramalan yang dilakukan oleh admin. Admin berhasil melakukan proses peramalan dan berhasil mendapatkan informasi hasil peramalan data darah masuk. *Activity diagram* peramalan dapat dilihat pada gambar 4.7.



Gambar 4.7 Activity Diagram Peramalan Darah Masuk

#### 4.3.4.6 Activity Diagram Edit dan Hapus Data Darah – Edit data darah

*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah yaitu *edit* data darah. Admin dapat mengubah data darah baru kedalam sistem. Data yang sudah dirubah akan ditampilkan dalam tabel data darah. *Activity diagram edit* data darah dapat dilihat pada lampiran A2.

#### 4.3.4.7 Activity Diagram Edit dan Hapus Data Darah – Hapus data darah

*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah yaitu hapus data darah. Admin dapat menghapus data darah didalam sistem. Setelah berhasil menghapus data darah, sistem akan menampilkan kembali tabel data darah. *Activity diagram* hapus data darah dapat dilihat pada lampiran A2.



#### 4.3.4.8 *Activity Diagram* Manajemen Data Darah Masuk – Tambah data darah masuk

*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah masuk yaitu tambah data darah. Admin dapat menambahkan data darah baru kedalam sistem. Data yang sudah tersimpan akan ditampilkan dalam tabel data darah masuk. *Activity diagram* tambah data darah masuk dapat dilihat pada lampiran A2.

#### 4.3.4.9 *Activity Diagram* Manajemen Data Darah Masuk – Edit data darah masuk

*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah masuk yaitu edit data darah masuk. Admin dapat mengubah data data darah masuk yang ada dengan data baru kedalam sistem. Data yang sudah tersimpan akan ditampilkan dalam tabel data darah masuk. *Activity diagram* edit data darah masuk dapat dilihat pada lampiran A2.

#### 4.3.4.10 *Activity Diagram* Manajemen Data Darah Masuk – Edit data darah masuk

*Activity diagram* ini menggambarkan alur kegiatan manajemen data darah masuk yaitu hapus data darah masuk. Admin dapat menghapus data darah masuk yang ada dalam sistem. Setelah data dihapus akan menampilkan kembali tabel data darah masuk data darah masuk. *Activity diagram* hapus data darah masuk dapat dilihat pada lampiran A2.

#### 4.3.4.11 *Activity diagram View* Darah

*Activity diagram view* darah merupakan penggambaran alur kerja sistem pada fitur Darah dimana berfungsi untuk melihat data darah pergolongan selama sebulan. *Activity diagram view* darah dapat dilihat pada lampiran A.2.

#### 4.3.4.12 *Activity diagram View* Darah Masuk

*Activity diagram view* darah masuk merupakan penggambaran alur kerja sistem pada fitur Darah Masuk. Darah Masuk dimana berfungsi untuk melihat data darah

dengan mencari berdasarkan bulan dan tahun. *Activity diagram view* darah masuk dapat dilihat pada lampiran A.2.

#### 4.3.4.13 *Activity diagram View* Darah Keluar

*Activity diagram view* darah keluar merupakan penggambaran alur kerja sistem pada fitur Darah Keluar. dimana berfungsi untuk melihat data darah dengan mencari berdasarkan bulan dan tahun. *Activity diagram view* darah keluar dapat dilihat pada lampiran A.2.

#### 4.3.4.14 *Activity diagram View* Stok Darah

*Activity diagram view* stok darah merupakan penggambaran alur kerja sistem pada fitur Stok Darah dimana berfungsi untuk melihat laporan data darah masuk dan darah keluar selama sebulan atau jumlah stok darah yang telah diinputkan dengan memilih laporan berdasarkan bulan dan tahun. *Activity diagram* laporan transaksi dapat dilihat pada lampiran A.2.

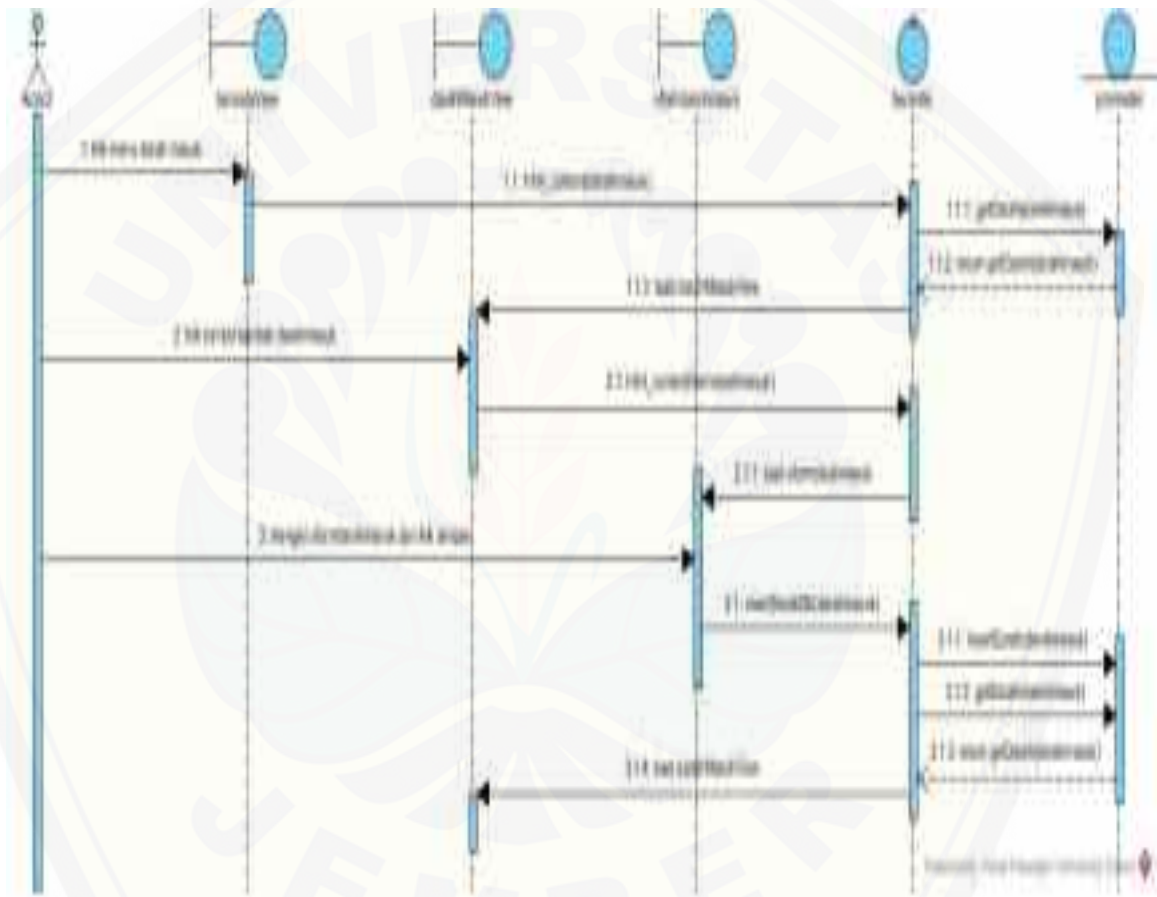
### 4.3.5 **Sequence Diagram**

*Sequence Diagram* pada sistem peramalan jumlah stok darah PMI ini digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian atau even untuk menghasilkan output tertentu. *Sequence Diagram* diawali dari apa yang me-trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan.

Sistem peramalan jumlah stok darah PMI ini menggunakan arsitektur pemrograman *Model-View-Controller (MVC)*. MVC yakni untuk memisahkan pemroses data yang disebut sebagai *Model* dan penyaji antarmuka (UI) atau HTML template dalam hal ini disebut *View*. Sedangkan *Controller* sebagai pengikat *Model View* untuk menangani permintaan pengguna sehingga data ditampilkan dengan benar.

#### 4.3.5.1 *Sequence Diagram* Manajemen Data Darah Masuk - Input Data Darah Masuk

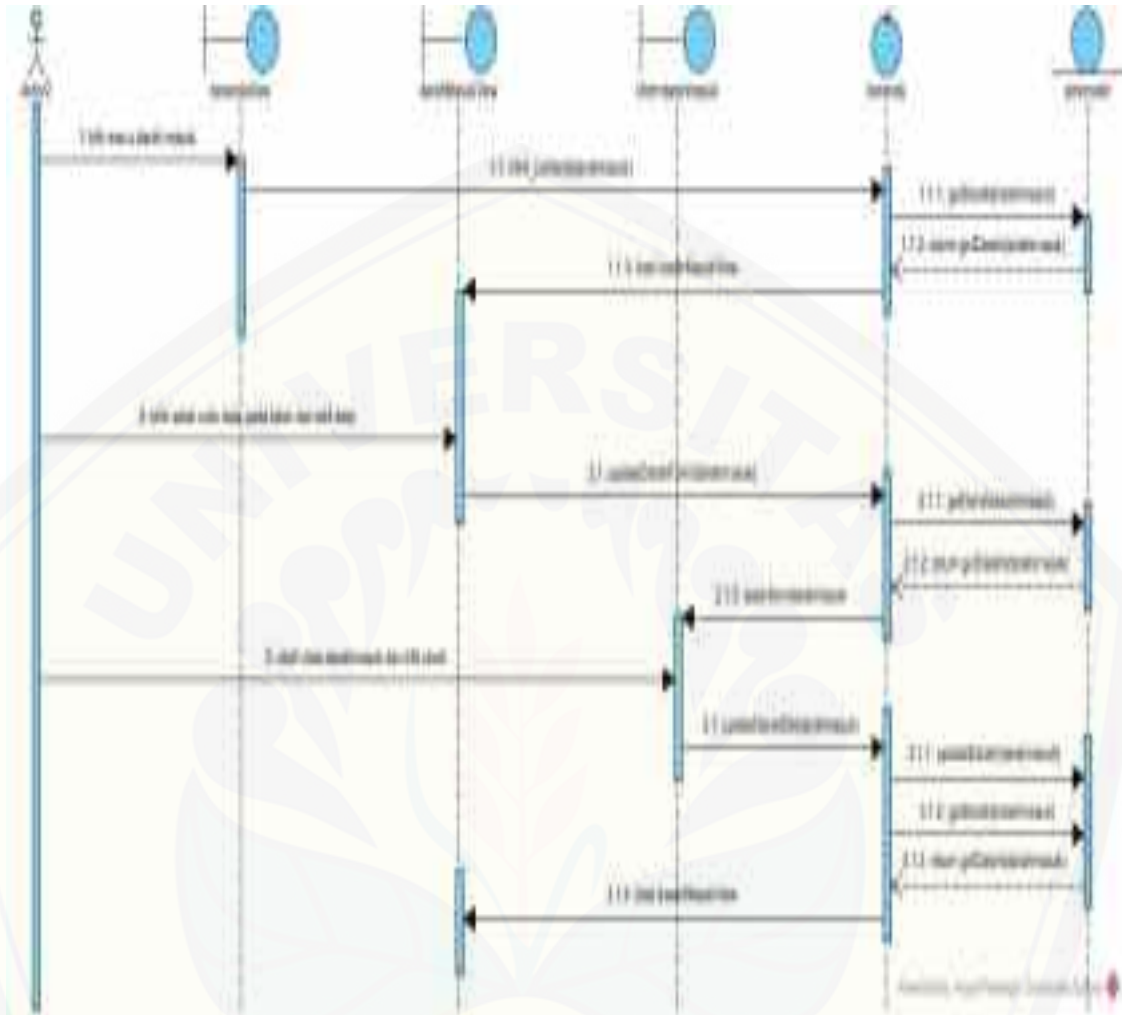
*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk manajemen data darah masuk yaitu tambah data darah masuk dengan aktor admin. *Sequence diagram* tambah data darah masuk dapat dilihat pada gambar 4.8 berikut.



Gambar 4.8 *Sequence Diagram* Input Data Darah Masuk

#### 4.3.5.2 *Sequence Diagram* Manajemen Data Darah Masuk - Edit Data Darah Masuk

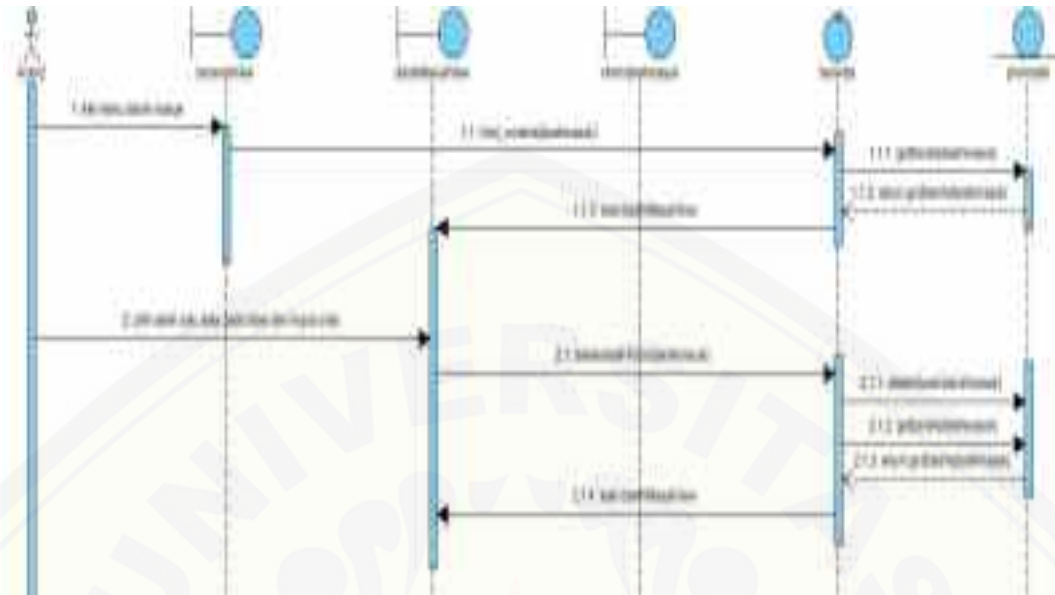
*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk manajemen data darah masuk yaitu *edit* darah masuk dengan aktor admin. *Sequence diagram* tambah data data darah masuk dapat dilihat pada gambar 4.9 berikut.



Gambar 4.9 *Sequence Diagram* Edit Data Darah Masuk

#### 4.3.5.3 *Sequence Diagram* Manajemen Data Darah Masuk - Hapus Data Darah Masuk

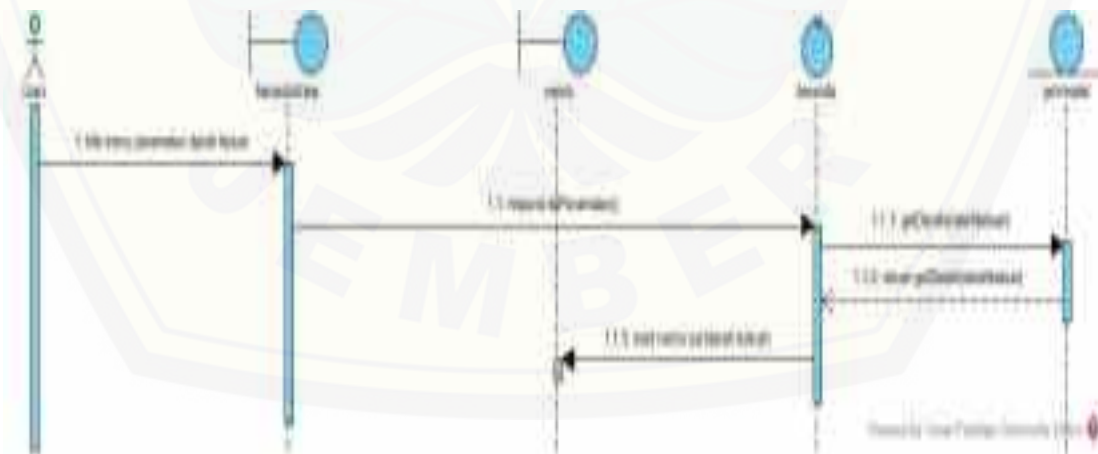
*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk manajemen data darah masuk yaitu hapus data darah masuk dengan aktor admin. *Sequence diagram* tambah data darah masuk dapat dilihat pada gambar 4.10 berikut



Gambar 4.10 *Sequence Diagram* Hapus Data Darah Masuk

#### 4.3.5.4 *Sequence Diagram* Peramalan Darah Keluar

*Activity diagram* ini menjelaskan alur proses dari peramalan yang dilakukan oleh admin. Admin berhasil melakukan proses peramalan dan berhasil mendapatkan informasi hasil peramalan data darah masuk. *Sequence diagram* peramalan dapat dilihat pada gambar 4.11 berikut



Gambar 4.11 *Sequence Diagram* Peramalan Darah Keluar

#### 4.3.5.5 *Sequence Diagram* Edit dan Hapus Data Darah – Edit data darah

*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk manajemen data darah yaitu *edit* data darah dengan aktor admin. *Sequence diagram* tambah data darah dapat dilihat pada lampiran A3.

#### 4.3.5.6 *Sequence Diagram* Edit dan Hapus Data Darah – Hapus data darah

*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk manajemen data darah yaitu hapus data darah dengan aktor admin. *Sequence diagram* tambah data darah dapat dilihat pada lampiran A3.

#### 4.3.5.7 *Sequence Diagram* Manajemen Data Darah Keluar - Input Data Darah Keluar

*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk tambah data keluar yaitu tambah data keluar dengan aktor admin. *Sequence diagram* tambah data data keluar dapat dilihat pada lampiran A3.

#### 4.3.5.8 *Sequence Diagram* Manajemen Data Darah Keluar – Edit Data Darah Keluar

*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk mengubah data darah keluar yaitu edit darah keluar dengan aktor admin. *Sequence diagram* edit data keluar dapat dilihat pada lampiran A3.

#### 4.3.5.9 *Sequence Diagram* Manajemen Data Darah Keluar – Hapus Data Darah Keluar

*Sequence diagram* ini merupakan gambaran alur kegiatan interaksi pada masing – masing class untuk manajemen darah keluar yaitu hapus darah keluar dengan aktor admin. *Sequence diagram* darah keluar dapat dilihat pada lampiran A3.

#### 4.3.5.10 *Sequence Diagram View Darah*

*Sequence diagram* lihat *View Darah* merupakan penggambaran skenario dan alur interaksi antar objek pada fitur Darah. Terdapat dua *view*, satu *controller*, dan satu model. *Sequence diagram View Darah* dapat dilihat pada lampiran A3.

#### 4.3.5.11 *Sequence Diagram View Darah Masuk*

*Sequence diagram* lihat *View Darah Masuk* merupakan penggambaran skenario dan alur interaksi antar objek pada fitur Darah Masuk. Terdapat dua *view*, satu *controller*, dan satu model. *Sequence diagram View Darah* dapat dilihat pada lampiran A3.

#### 4.3.5.12 *Sequence Diagram View Darah Keluar*

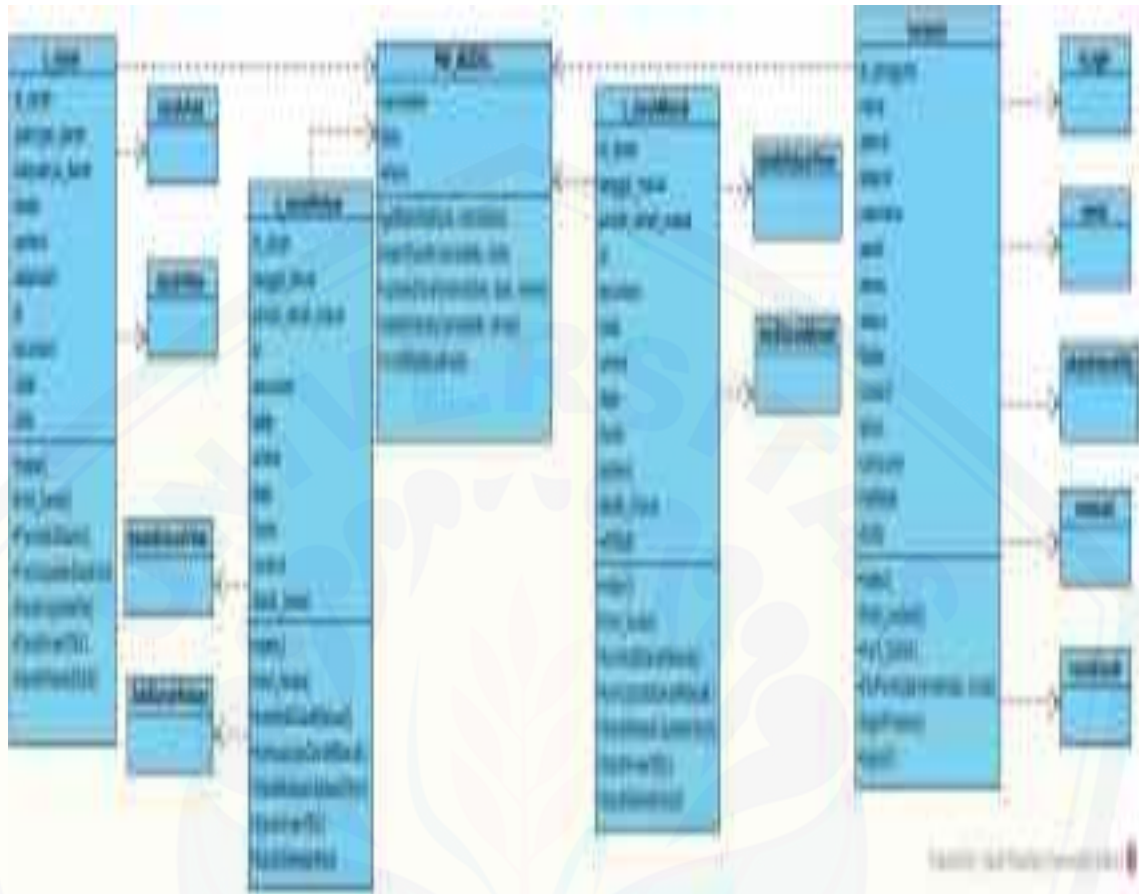
*Sequence diagram* lihat *View Darah Keluar* merupakan penggambaran skenario dan alur interaksi antar objek pada fitur Darah Keluar. Terdapat dua *view*, satu *controller*, dan satu model. *Sequence diagram View Darah Keluar* dapat dilihat pada lampiran A3.

#### 4.3.5.13 *Sequence Diagram View Stok Darah*

*Sequence diagram* lihat *View Stok Darah* merupakan penggambaran skenario dan alur interaksi antar objek pada fitur Stok Darah. Terdapat dua *view*, satu *controller*, dan satu model. *Sequence diagram View Stok Darah* dapat dilihat pada lampiran A3.

### 4.3.6 *Class Diagram*

*Class Diagram* menjelaskan semua *function* yang dipakai dalam sistem. *Class diagram* juga menjelaskan bagaimana relasi antar *class* dalam suatu sistem yang terdiri dari model, view, dan controller yang masing-masing berisi method dan data yang berbeda namun memiliki hubungan dengan yang lainnya. Diagram ini memengaruhi bagaimana implementasi desain dalam *coding* program. Seperti dijelaskan pada gambar 4.12.



**Gambar 4.12** Class Diagram Sistem Peramalan Jumlah Stok Darah

#### 4.3.7 Entity Relation Diagram (ERD)

*Entity Relation Diagram* (ERD) merupakan diagram yang menjelaskan relasi antar tabel. Pada tahap ini, maka akan dibuat sebuah desain berisi basis data-data yang akan digunakan dalam pembangunan sistem. *Entity Relation Diagram* (ERD) yang digunakan dalam sistem peramalan stok darah ini dijelaskan pada gambar 4.13.





Gambar 4.13 *Entity Relation Diagram* Sistem Peramalan Stok Darah

#### 4.3 Desain Sistem

Desain sistem yang akan dibangun berdasarkan masing-masing *usecase* dan perancangan yang lain seperti *Class* dan *Entity Relation Diagram*.

#### 4.4 Implementasi Perancangan

Setelah tahap desain perancangan selesai, tahap selanjutnya dalam penelitian ini yaitu tahap pengimplementasian desain perancangan ke dalam bahasa pemrograman. Bahasa pemrograman yang dipakai yaitu bahasa pemrograman *php*, *html* dan *css* dengan *framework codeigniter* dan *database* yang digunakan adalah *DBMS MySql*.

#### 4.5 Pengujian

Pengujian ini dilakukan guna mengevaluasi sistem yang telah dibuat. Proses pengujian dengan pengujian *black box*. Pengujian *black box* digunakan untuk mengetahui fungsi-fungsi dari *input* dan *output* sistem, mengetahui keakurasian sistem ketika sistem berhadapan langsung dengan user yang mayoritas adalah petani dan pakar pertanian. Pada tabel 4.4 dapat dijelaskan bahwa proses ini dilakukan untuk mengetahui menu yang telah berhasil dijalankan dan di implementasikan.

#### 4.5.1 Black box

Dalam pengujian *black box* ini semua menu berhasil dijalankan pada sistem informasi peramalan jumlah stok darah menggunakan metode WMA. Pengujian sistem penentuan lahan menggunakan menggunakan metode WMA dengan metode *black box* dapat dilihat pada tabel 4.7.

Tabel 4.6 Pengujian *Black box*

No	Menu	Fungsi	Kasus	Hasil	Ket
1	View Data Darah	Melihat informasi kaitan dengan banyaknya data darah pergolongan setiap bulan	Ketika user memilih menu Data Darah	Menampilkan semua daftar informasi yang telah <i>diinputkan</i> admin	OK
2	View Darah Keluar	Melihat kriteria darah pergolongan dalam melakukan perhitungan	User memilih menu Darah Keluar	Menampilkan halaman yang berisi data darah keluar	OK
			User klik tombol detail Data Darah	Menampilkan informasi lebih lengkap sesuai dengan baris yang dipilih	OK
3	View Darah Masuk	Melihat kriteria darah	User memilih menu Darah	Menampilkan halaman yang	OK

		pergolongan dalam melakukan perhitungan	Masuk	berisi data darah masuk	
			User klik tombol detail Data Darah	Menampilkan informasi lebih lengkap sesuai dengan baris yang dipilih	OK
4	Function Cari	Mencari data darah perbulan sesuai golongan	User memilih menu cari dan memilih <i>dropdown</i>	Menampilkan halaman cari yang berisi hasil dan proses perhitungan metode	OK
5	Manajemen admin	Memajemen data darah	Admin memilih menu data darahh	Menampilkan semua data darah yang terdapat dalam system	OK
			Admin klik tombol tambah user	Menampilkan form tambah data darah	OK
			Admin klik tombol detail	Menampilkan data darah baris terpilih	OK
			Admin klik tombol hapus	Menghapus data darah yang terdapat dari system	OK
6	Fitur Darah	Memajemen	Admin klik	Menampilkan form	OK

		data darah yang ditampilkan pada halaman user	tombol tambah tambah data darah	tambah data darah/info	
			Admin klik tombol edit	Menampilkan halaman edit data darah	OK
			Admin klik tombol hapus	Menghapus informasi baris terpilih	OK
7	Fitur Darah Masuk	Memanajemen data darah masuk yang menjadi alternatif dalam perhitungan	Admin memilih menu detail darah masuk	Menampilkan semua data darah masuk	OK
			Admin klik tombol tambah data darah	Menampilkan form <i>inputan</i> tambah darah	OK
			Admin klik tombol edit	Menampilkan halaman edit data darah	OK
			Admin klik tombol hapus	Menghapus lahan yang sebelumnya sudah pernah <i>diinputkan</i> oleh admin	OK
8	Fitur Darah	Memanajemen	Admin	Menampilkan	OK

	Keluar	data darah keluar yang menjadi alternatif dalam perhitungan	memilih menu detail darah keluar	semua data darah keluar	
			Admin klik tombol edit	Menampilkan form edit data darah	OK
9	View Stock Darah	Memajemen stok darah yang ditampilkan pada halaman user	Admin klik fitur stok data darah	Menampilkan semua stok data darah	OK
10	Fitur Log Out	Keluar dari system	User/admin klik tombol keluar	Menampilkan halaman login dengan fungsi destroy/unset username dan password	OK

#### 4.5.2 White box

Pengujian *White Box* yang dilakukan pada penelitian ini dimulai dari pembuatan diagram alir dari listing program yang diujikan. Dalam pengujian whitebox terdapat beberapa tahapan yaitu cyclomatic complexity, listing program, penentuan jalur independen, dan test case. Pada tahap pengujian ini, fungsi yang diuji adalah sebagai berikut.

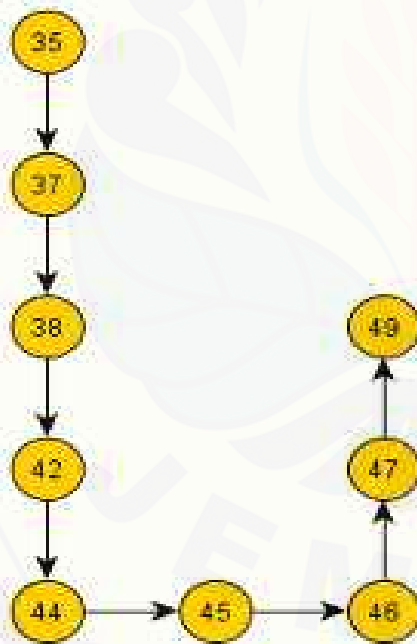
1. Pengujian Listing Program function FormUpdateDarah()

```

34
35
36 public function FormUpdateDarah($id)
37 {
38     $qr = $this->getDarah($id);
39     $data = array();
40     $id_darah => $qr[0]['id_darah'];
41     $golongan_darah => $qr[0]['golongan_darah'];
42     $banyaknya_darah => $qr[0]['banyaknya_darah'];
43 }
44
45 $response = array();
46 $header => $this->getHeader();
47 $content => $this->loadView('dashDarah', $data, TRUE);
48
49 $this->loadView('dashboard', $response);
50

```

Gambar 4.14 Listing Program function FormUpdateDarah()



a. Kompleksitas Siklometrik

Perhitungan kompleksitas siklometrik untuk grafik alir  $G$  menggunakan rumus  $V(G) = E - N + 2$ .  $E$  adalah jumlah edge dan  $N$  adalah jumlah node pada grafik alir.

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

$$= 8 - 9 + 2$$

$$= 1$$

b. Basis Set

Jalur 1 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9

c. Test Case

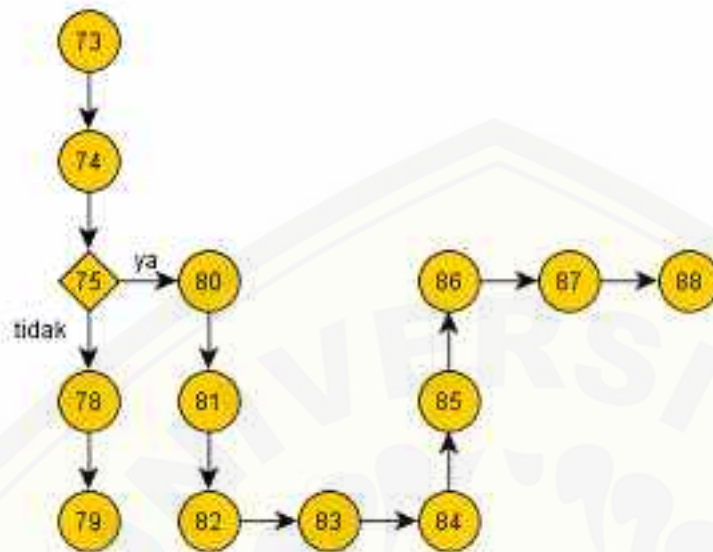
<b>Test Case</b>	Menambah form pada update darah
<b>Target yang diharapkan</b>	Mendapatkan form pada update darah
<b>Hasil Pengujian</b>	Benar
<b>Path / Jalur</b>	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9

2. Pengujian Listing Program function DarahKeluarInsertDb()

```

87 public function DarahKeluarInsertDb()
88 {
89     $iddarahkeluar = $_POST['iddarahkeluar'];
90     $tanggal_keluar = $_POST['tanggal_keluar'];
91     $jumlah_darah = $_POST['jumlah_darah'];
92     $golongan_darah = $_POST['golongan_darah'];
93     //echo "iddarahkeluar: ".$iddarahkeluar." / tanggal_keluar: ".$tanggal_keluar." / golongan_darah: ".
94     $data_insert = array(
95         'id_darah' => $iddarahkeluar,
96         'tanggal_keluar' => $tanggal_keluar,
97         'jumlah_darah_keluar' => $jumlah_darah
98     );
99     $where = array('id_darah' => $iddarahkeluar);
100     $qr = $this->model->updateDarah('darah_keluar', $data_insert, $where);
101     redirect('/c_darahkeluar');
102 }
    
```

Gambar 4.15 Listing Program function DarahKeluarInsertDb()



a. Kompleksitas Siklometik

Perhitungan kompleksitas siklometik untuk grafik alir G menggunakan rumus

$V(G) = E - N + 2$ . E adalah jumlah edge dan N adalah jumlah node pada grafik alir.

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 13 - 14 + 2 \\
 &= 1
 \end{aligned}$$

b. Basis Set

Jalur 1 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12

Jalur 2 : 1 - 2 - 3 - 13 - 14

c. Test Case

<b>Test Case</b>	Menambah form pada data Darah Keluar
<b>Target yang diharapkan</b>	Mendapatkan form pada data Darah Keluar
<b>Hasil Pengujian</b>	Benar
<b>Path / Jalur</b>	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 13 - 14



## BAB 6. PENUTUP

Bab ini merupakan bagian akhir dalam penulisan skripsi, yang berisi tentang kesimpulan dan saran. Kesimpulan yang ditulis merupakan hasil dari penelitian yang telah dilakukan dan saran lanjutan untuk dilakukan pada penelitian selanjutnya.

### 6.1 Kesimpulan

Kesimpulan pada penelitian yang telah dilakukan adalah sebagai berikut :

1. Implementasi metode WMA pada sistem peramalan stok darah memiliki langkah secara berturut-turut antara lain: penentuan data kriteria darah dan penentuan data bobot yang bertujuan untuk menghitung kriteria darah dan bobot dari data darah, sehingga menghasilkan nilai untuk dikalikan sesuai dengan metode perhitungan. Data 6 bulan sebelumnya harus disimpan untuk peramalan bulanan, setelah melakukan analisis diketahui bahwa periode masa lalu 6 bulan paling optimal karena menghasilkan nilai data darah masuk dan keluar serta *Mean Absolute Deviation* yang paling kecil.
2. Dalam merancang dan membangun sistem informasi prediksi jumlah stok darah pada PMI, diperlukan pemahaman dalam alur perhitungan metode *weighted moving average* secara manual. Sistem informasi prediksi jumlah stok darah telah dibuat dengan hak akses, yaitu admin. Admin dapat mengakses fitur prediksi yang menghasilkan nilai prediksi secara bulanan sehingga membantu admin dalam menentukan jumlah stok pada masing-masing pergolongan darah dan jumlah kantong darah yang telah *expired* dapat diminimalisir jumlahnya sehingga PMI dapat memperkirakan permintaan darah di bulan yang akan datang dan berapa jumlah kantong darah yang akan keluar. Dengan demikian secara otomatis juga dapat meminimalisir jumlah kantong darah yang terbuang. Selain itu juga meminimalisir waktu dan biaya yang dikeluarkan selama proses perjalanan pengambilan darah ke PMI.

3. Admin memiliki kewenangan untuk manajemen data darah, manajemen data darah masuk, manajemen data darah keluar, menetapkan data kriteria dan bobot. User juga memiliki hak akses untuk melihat rekap data darah, rekap data darah masuk, rekap data darah keluar, stok darah, dan peramalan darah.
4. Perhitungan prediksi menggunakan metode *weighted moving average* dapat memberikan hasil error yang cukup kecil yaitu antara 10 sampai 20, sedangkan jika hasil error lebih dari rata-rata dikarenakan terjadinya kenaikan jumlah permintaan darah.

## 6.2 Saran

Adapun saran yang ditunjukkan untuk memberikan masukan yang lebih baik yaitu:

1. Sistem prediksi jumlah stok darah PMI dapat dilengkapi dengan fitur detail darah masuk sehingga pengguna juga dapat mengetahui tanggal masuk dan tanggal kadaluarsa tiap bulan.
2. Memungkinkan pengguna dapat menyimpan hasil prediksi pada masing-masing periode.
3. Dalam metode peramalan untuk menentukan jumlah stok darah PMI untuk di masa yang akan datang dapat digunakan metode-metode yang lain. Hal ini bertujuan mengetahui tingkat keakuratan peramalan dan tingkat kesalahan dari masing-masing sistem yang menggunakan metode berbeda unruk mendapatkan hasil yang paling optimal.

**DAFTAR PUSTAKA**

- Soejoeti, Z. 1987. *Analisis Runtun waktu*. Jakarta : Karunia Jakarta
- Sugiyono. 2010. *Metode Penelitian Kuantitatif Kualitatif & RND*. Bandung : Alfabeta
- Harjo. 2014. *Peramalan – Forecasting*. Diakses pada tanggal 17 Maret 2015  
<http://harjo820.blogspot.com/2014/12/peramalan-forecasting.html>
- Agissa, W. 2013. *White Box and Black Box Testing*. Diakses pada tanggal 26 Maret 2015. <http://bangwildan.web.id>
- Pressman, R. S. 2002. *Rekayasa Perangkat Lunak Pendekatan Praktisi Buku1*. Yogyakarta : Andi
- Moleong, Lexy J. (2011). *Metodologi penelitian Kualitatif*. Bandung: PT Remaja Rosdakarya
- Aminudin, M. 2011, *Peramalan Cuaca Kota Surabaya Tahun 2011 menggunakan Metode Moving Average dan Klasifikasi Naive Bayes*, PENS ITS, Indonesia
- Junaldi, K. I. 2012. *Prediksi Daya Listrik Geothermal Power Plant Berdasarkan Metode Weighted Moving Averagedi* PT. GEO DIPA ENERGI Unit Dieng, PENS, ITS, Indonesia
- Jay, H., dan Barry, R. 2005. *Prinsip-Prinsip Management Operasi*. Jakarta : Salemba 4
- Koorag, J. F. 2010. *Berbagi Nyawa Hidup Bahagia dengan Berdonor Darah*. Yogyakarta : Pustaka Marwa
- Husni. (2007). *Pemrograman Database Berbasis Web*. Yogyakarta : Graha Ilmu
- Ladjamudin. Al Bahra Bin. (2005). *Analisis dan Desain Sistem Informasi*. Yogyakarta : Graha Ilmu.
- Minartiningtyas, Arie. (2013). *Artikel Teknik Informatika dan Sistem Informasi*. Diakses 14 April 2015, dari <http://informatika.web.id/>.
- Pressman, R. S. 2002. *Rekayasa Perangkat Lunak Pendekatan Praktisi*. Yogyakarta: Andi.

## LAMPIRAN A. PERANCANGAN SISTEM

## A1. Perancangan Skenario

## 1. Skenario Login

<b>Nama</b>	<b>Login</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>	
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Admin berhasil <i>login</i> system</li> </ul>	
<b>Skenario Utama : Login</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
1. Membuka/menjalankan halaman system		
	2. Menampilkan halaman <i>login</i>	
3. Memasukan <i>username</i> dan <i>password</i>		
4. Mengklik tombol <i>login</i>		
	5. Mengecek data <i>login</i> ke <i>database</i>	
	6. Data <i>login</i> yang diinputkan benar	
<b>Skenario Alternatif : Login</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
5a. Mengklik tombol " <i>login</i> "		
	6a. Data <i>login</i> yang diinputkan salah	
	6b. Menampilkan halaman <i>login</i> kembali	

## 2. Skenario Edit Data Darah

<b>Nama</b>	<b>Edit data darah</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	-	<b>Admin duduk di depan computer</b>
	-	<b>Menghidupkan computer</b>
<b>Post-Condition</b>	-	<b>Data darah telah ter update</b>
<b>Skenario Utama : Edit Data Darah</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
1. Memilih menu "Darah"		
	7. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>	
8. Memilih data yang akan diedit lalu mengklik tombol "Edit"		
	9. Menampilkan form edit data darah	
10. Mengedit data darah dan memilih tombol "Simpan"		
	11. Menyimpan data, menampilkan kembali informasi data darah masuk yang sudah di edit	
<b>Skenario Alternatif : Data kurang lengkap</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
5b. Mengklik tombol "Simpan"		
	6b. Menampilkan <i>alert</i> "data tidak boleh kosong" dan menampilkan kembali halaman form edit data darah	

## 2a. Skenario Hapus Data Darah

<b>Nama</b>	<b>Hapus data darah</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	-	<b>Admin duduk di depan komputer</b>
	-	<b>Menghidupkan komputer</b>
<b>Post-Condition</b>	-	<b>Data darah telah ter hapus</b>
<b>Skenario Utama : Hapus Data Darah</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
7. Memilih menu "Darah"		
	8. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>	
9. Memilih data darah masuk yang akan dihapus lalu klik tombol "delete"		
	10. Menampilkan dialog box konfirmasi yang berisi tombol 'ok' dan 'cancel'	
11. Mengklik tombol "delete" pada dialog box		
	12. Menghapus data dan menampilkan kembali data darah	
<b>Skenario Alternatif : Pembatalan Hapus Data</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
5a. Mengklik tombol "cancel" pada dialog box		
	6a. Menampilkan kembali data darah	

## 3. Skenario Peramalan Darah Masuk

<b>Nama</b>	<b>Peramalan Darah Masuk</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- <b>User duduk di depan computer</b></li> <li>- <b>Menghidupkan computer</b></li> </ul>	
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- <b>User mendapatkan hasil data darah dari peramalan</b></li> </ul>	
<b>Skenario Utama : Peramalan Darah Masuk</b>		
<b>User</b>	<b>Reaksi Sistem</b>	
7. Memilih menu Peramalan		
	8. Menampilkan <i>dropdown</i> darah masuk dan darah keluar	
9. Memilih <i>dropdown</i> darah masuk		
	10. Sistem menampilkan hasil peramalan darah masuk yang dilakukan oleh user	
11. Memilih menu "Home"		
	12. Menampilkan halaman awal system	
<b>Alternative Flow : Pembatalan Peramalan</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
6a. Memilih tombol Home		
	7a. Menampilkan kembali halaman awal system	

## 3a. Skenario Skenario Peramalan Darah Keluar

<b>Nama</b>	<b>Peramalan Darah Keluar</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- User duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	- User mendapatkan hasil data darah dari peramalan
<b>Skenario Utama : Peramalan Darah Keluar</b>	
<b>User</b>	<b>Reaksi Sistem</b>
4. Memilih menu Peramalan	
	5. Menampilkan <i>dropdown</i> darah masuk dan darah keluar
3. Memilih <i>dropdown</i> darah keluar	
	4. Sistem menampilkan hasil peramalan darah masuk yang dilakukan oleh user
5. Memilih menu "Home"	
	6. Menampilkan halaman awal sistem
<b>Alternative Flow : Pembatalan Peramalan</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
6a. Memilih tombol Home	
	7a. Menampilkan kembali halaman awal system

## 4. Skenario Manajemen Darah Masuk

<b>Nama</b>	<b>Input data darah masuk</b>
<b>Aktor</b>	<b>Admin</b>



<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	- Data darah masuk telah ditambah
<b>Skenario Utama : Input Data Darah Masuk</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1. Pilih menu manajemen darah masuk	
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>
3. Memilih tombol "Tambah Data"	
	4. Menampilkan form tambah data darah
5. Mengisi Form tambah data baru dengan data darah dan mengklik tombol "Simpan"	
	6. Menyimpan data, menampilkan kembali informasi data darah yang baru saja ditambahkan
<b>Alternative Flow : Terdapat Data Kurang Lengkap</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
5b. Mengisi Form tambah data baru dengan data darah dan Mengklik tombol "Simpan"	
	6b. Menampilkan <i>alert</i> "data tidak boleh kosong" dan menampilkan kembali halaman form tambah data darah

--	--

## 4a. Skenario Edit Darah Masuk

<b>Nama</b>	<b>Edit data darah masuk</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Data darah masuk telah ter update</li> </ul>

**Skenario Utama : Edit Data Darah Masuk**

<b>Admin</b>	<b>Reaksi Sistem</b>
1. Memilih menu "Darah Masuk"	
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>
3. Memilih data yang akan diedit lalu mengklik tombol "Edit"	
	4. Menampilkan form edit data darah masuk
5. Mengedit data darah masuk dan memilih tombol "Simpan"	
	6. Menyimpan data, menampilkan kembali informasi data darah masuk yang sudah di edit

**Skenario Alternatif : Data kurang lengkap**

<b>Admin</b>	<b>Reaksi Sistem</b>
5b. Mengklik tombol "Simpan"	
	6b. Menampilkan <i>alert</i> "data tidak boleh"

	kosong” dan menampilkan kembali halaman form edit data darah
--	--

4b. Skenario Hapus Darah Masuk

<b>Nama</b>	<b>Hapus data darah masuk</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>	
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Data darah telah ter hapus</li> </ul>	
<b>Skenario Utama : Hapus Data Darah Masuk</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
1. Memilih menu “Darah Masuk”		
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>	
3. Memilih data darah masuk yang akan dihapus lalu klik tombol “delete”		
	4. Menampilkan dialog box konfirmasi yang berisi tombol ‘ok’ dan ‘cancel’	
5. Mengklik tombol “delete” pada dialog box		
	6. Menghapus data dan menampilkan kembali data darah	
<b>Skenario Alternatif : Pembatalan Hapus Data</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
5a. Mengklik tombol “cancel” pada		

dialog box	
	6a. Menampilkan kembali list data darah

#### 5. Skenario Manajemen Darah Keluar

<b>Nama</b>	<b>Input data darah keluar</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	- Data darah keluar telah ditambah
<b>Skenario Utama : Input Data Darah Keluar</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1. Pilih menu manajemen darah keluar	
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>
3. Memilih tombol "Tambah Data"	
	4. Menampilkan form tambah data darah
5. Mengisi Form tambah data baru dengan data darah dan mengklik tombol "Simpan"	
	6. Menyimpan data, menampilkan kembali informasi data darah yang baru saja ditambahkan

<b>Alternative Flow : Terdapat Data Kurang Lengkap</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
5b. Mengisi Form tambah data baru dengan data darah dan Mengklik tombol "Simpan"	
	6b. Menampilkan <i>alert</i> "data tidak boleh kosong" dan menampilkan kembali halaman form tambah data darah

5a. Skenario Edit Data Darah Keluar

<b>Nama</b>	<b>Edit data darah keluar</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Data darah keluar telah ter update</li> </ul>

**Skenario Utama : Edit Data Darah Keluar**

<b>Admin</b>	<b>Reaksi Sistem</b>
1. Memilih menu "Darah Keluar"	
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>
1. Memilih data yang akan diedit lalu mengklik tombol "Edit"	
	2. Menampilkan form edit data darah masuk
3. Mengedit data darah keluar dan	

memilih tombol “Simpan”	
	4. Menyimpan data, menampilkan kembali informasi data darah keluar yang sudah di edit
<b>Skenario Alternatif : Data kurang lengkap</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
5b. Mengklik tombol “Simpan”	
	6b. Menampilkan <i>alert</i> “data tidak boleh kosong” dan menampilkan kembali halaman form edit data darah

## 5b. Skenario Hapus Data Darah Keluar

<b>Nama</b>	<b>Hapus data darah keluar</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Data darah telah ter hapus</li> </ul>
<b>Skenario Utama : Hapus Data Darah Keluar</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1. Memilih menu “Darah Keluar”	
	2. Menampilkan halaman dengan tabel yang berisi data darah yang telah di <i>inputkan</i>
3. Memilih data darah keluar yang akan dihapus lalu klik tombol “delete”	
	4. Menampilkan dialog box konfirmasi yang berisi tombol ‘ok’ dan ‘cancel’

5. Mengklik tombol “delete” pada dialog box	
	6. Menghapus data dan menampilkan kembali data darah
<b>Skenario Alternatif : Pembatalan Hapus Data</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
5a. Mengklik tombol “cancel” pada dialog box	
	6a. Menampilkan kembali list data darah

## 5c. Skenario Detail Darah Keluar

<b>Nama</b>	<b>Detail Darah Keluar</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Admin melihat informasi terkait darah keluar</li> </ul>
<b>Skenario Utama : Detail Darah Keluar</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1 Memilih menu “Darah Keluar”	
	2 Menampilkan tabel yang berisi informasi-informasi tentang darah keluar
3 Klik tombol detail darah keluar	
	4 Menampilkan rincian informasi sesuai dengan golongan darah, ID darah dan banyaknya darah

## 6. View Darah

<b>Nama</b>	<b>View Darah</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>	
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Admin melihat informasi terkait darah</li> </ul>	
<b>Skenario Utama : View Darah</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	
1. Memilih menu "Darah"		
	2. Menampilkan halaman dengan tabel yang berisi data dari informasi-informasi yang telah di <i>inputkan</i>	
	3. Menampilkan previous dan next	
4. Klik tombol next		
	5. Menampilkan data dan isi informasi yang telah <i>inputkan</i> dihalam berikutnya	

## 7. View Darah Keluar

<b>Nama</b>	<b>View Darah Keluar</b>	
<b>Aktor</b>	<b>Admin</b>	
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>	
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Admin melihat informasi terkait darah</li> </ul>	
<b>Skenario Utama : View Darah Keluar</b>		
<b>Admin</b>	<b>Reaksi Sistem</b>	



1. Memilih menu “Darah Keluar”	
	2. Menampilkan halaman dengan tabel yang berisi data dari informasi-informasi yang telah di <i>inputkan</i>
	3. Menampilkan previous dan next
4. Klik tombol next	
	5. Menampilkan data dan isi informasi yang telah <i>inputkan</i> dihalam berikutnya

8. View Darah Masuk

<b>Nama</b>	<b>View Darah Masuk</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Admin melihat informasi terkait darah</li> </ul>
<b>Skenario Utama : View Darah Masuk</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1. Memilih menu “Darah Masuk”	
	2. Menampilkan halaman dengan tabel yang berisi data dari informasi-informasi yang telah di <i>inputkan</i>
	3. Menampilkan previous dan next
4. Klik tombol next	
	5. Menampilkan data dan isi informasi yang telah <i>inputkan</i> dihalam berikutnya

## 9. View Stok Darah

<b>Nama</b>	<b>View Stok Darah</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Menghidupkan computer</li> </ul>
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Admin melihat informasi terkait stok darah</li> </ul>
<b>Skenario Utama : View Stok Darah</b>	
<b>Admin</b>	<b>Reaksi Sistem</b>
1. Memilih menu "Stok Darah"	
	2. Menampilkan halaman dengan tabel yang berisi data dari informasi-informasi yang telah di <i>inputkan</i>
	3. Menampilkan previous dan next
4. Klik tombol next	
	5. Menampilkan data dan isi informasi yang telah <i>inputkan</i> dihalam berikutnya

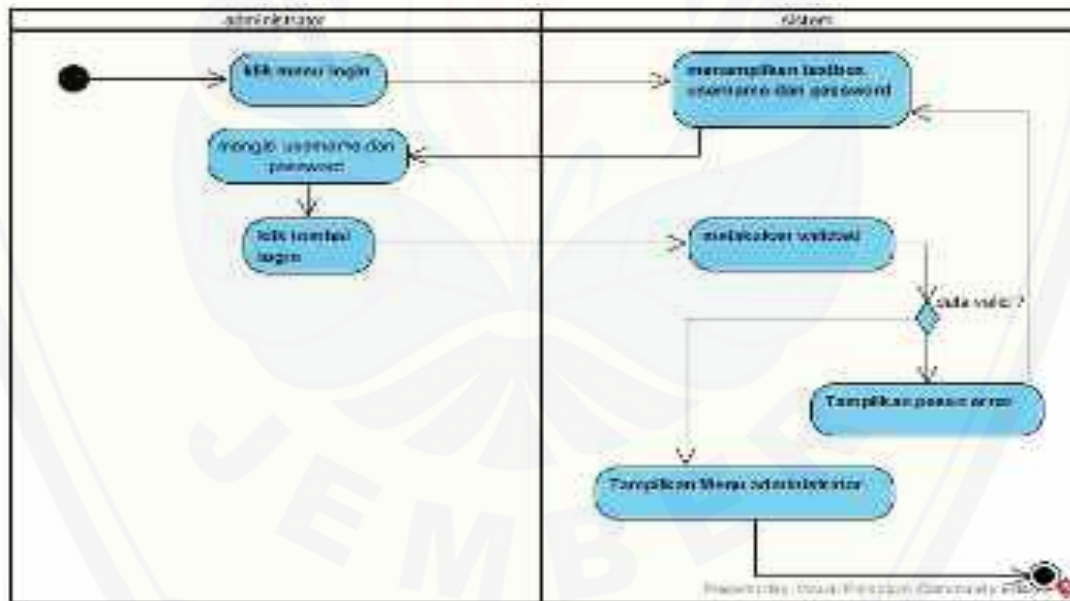
## 10. Logout

<b>Nama</b>	<b>Logout</b>
<b>Aktor</b>	<b>Admin</b>
<b>Pre-Condition</b>	<ul style="list-style-type: none"> <li>- Admin duduk di depan computer</li> <li>- Aktor masuk kedalam sistem dengan username dan password</li> </ul>
<b>Post-Condition</b>	<ul style="list-style-type: none"> <li>- Aktor keluar sistem dengan function destroy username dan password</li> </ul>
<b>Skenario Utama : View Darah Keluar</b>	

Admin	Reaksi Sistem
1. Aktor berada di dalam sistem dengan username dan password login	
	2. Menampilkan halaman system
3. Klik icon user yang berada pada pojok kanan atas halaman dan pilih logout	
	4. Keluar halaman system

## A2. Perancangan Activity Diagram

### 1. Login



2. Manajemen Darah





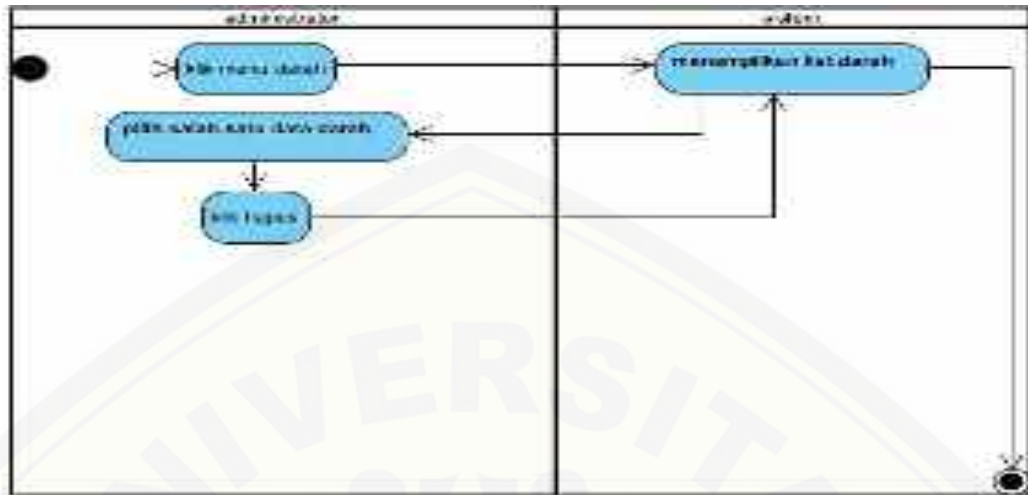
3. Manajemen Darah – Input Data Darah



4. Manajemen Darah – Edit Data Darah



5. Manajemen Darah – Hapus Data Darah



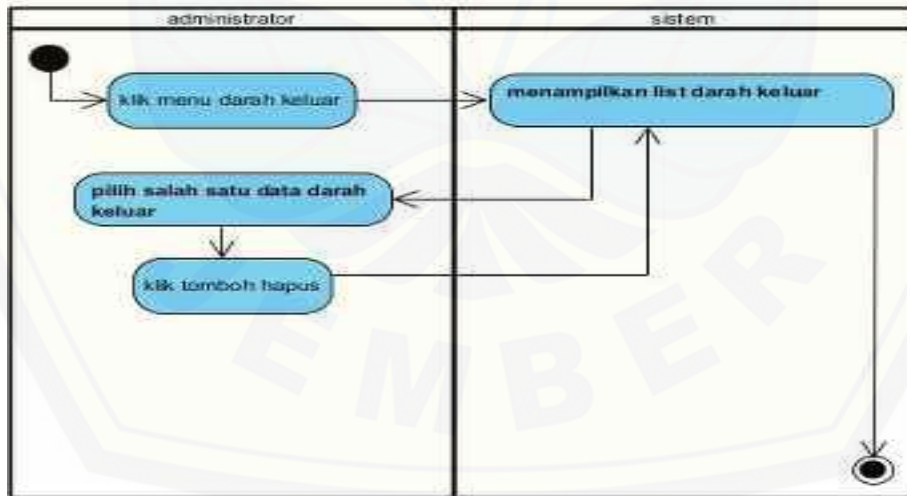
6. Manajemen Darah Masuk – Input Data Darah



7. Manajemen Darah Masuk – Edit Data Darah



8. Manajemen Darah Masuk– Hapus Data Darah

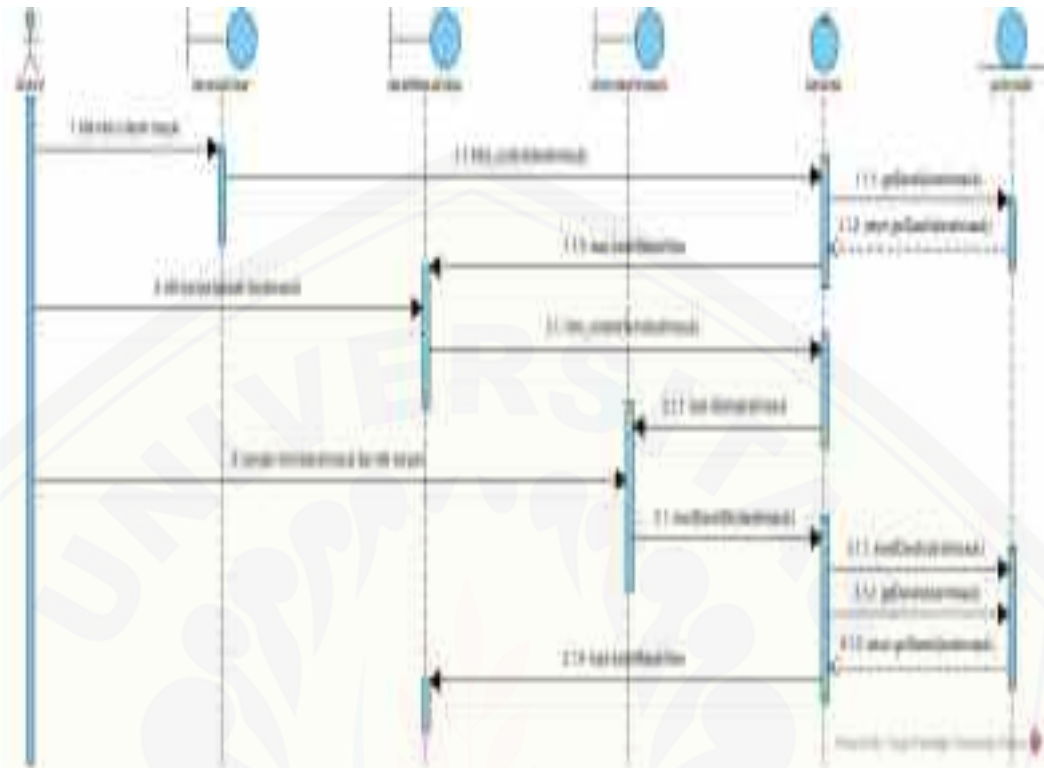


9. Manajemen Darah Masuk

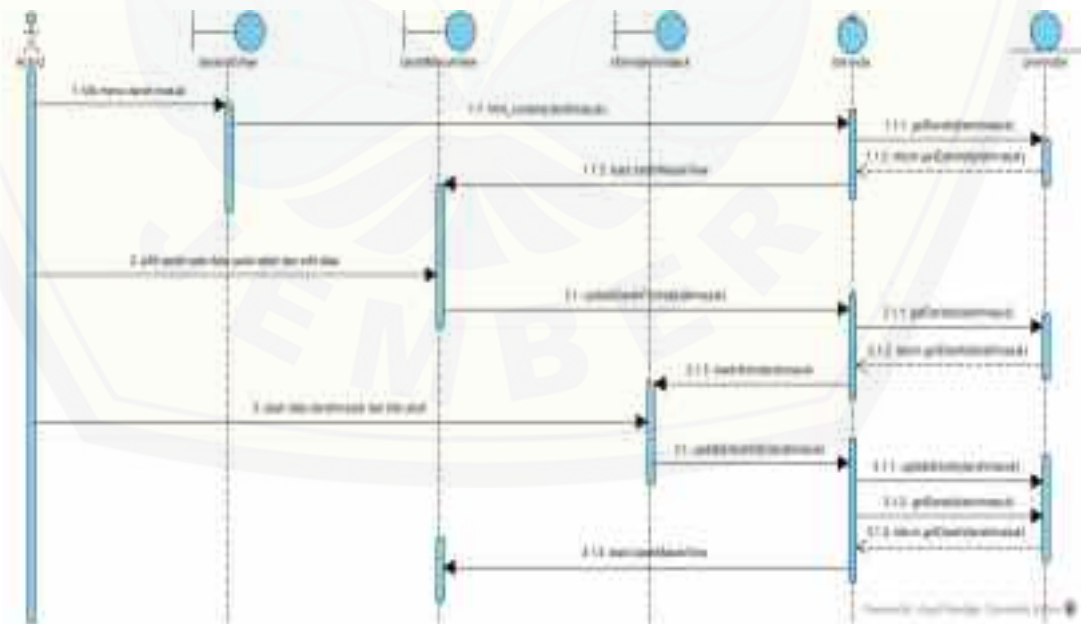




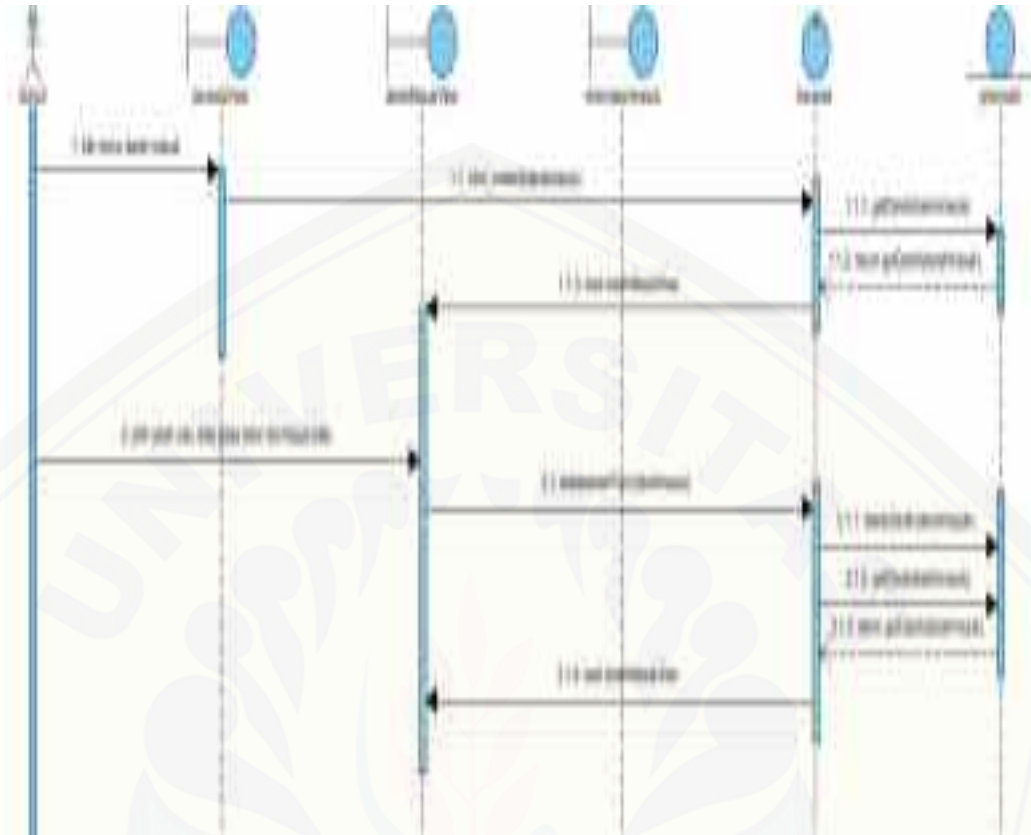




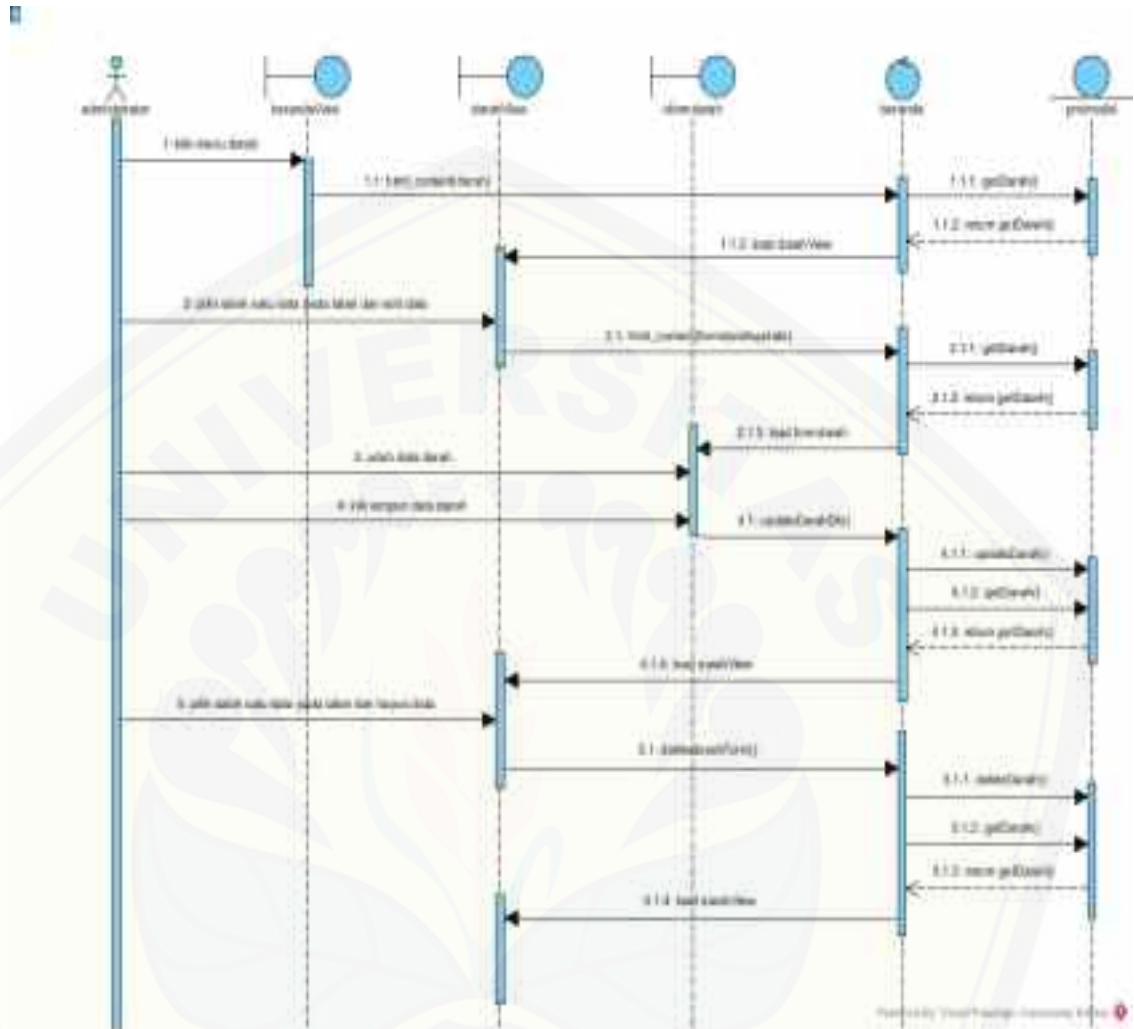
3. Manajemen Darah Masuk – Edit Data Darah



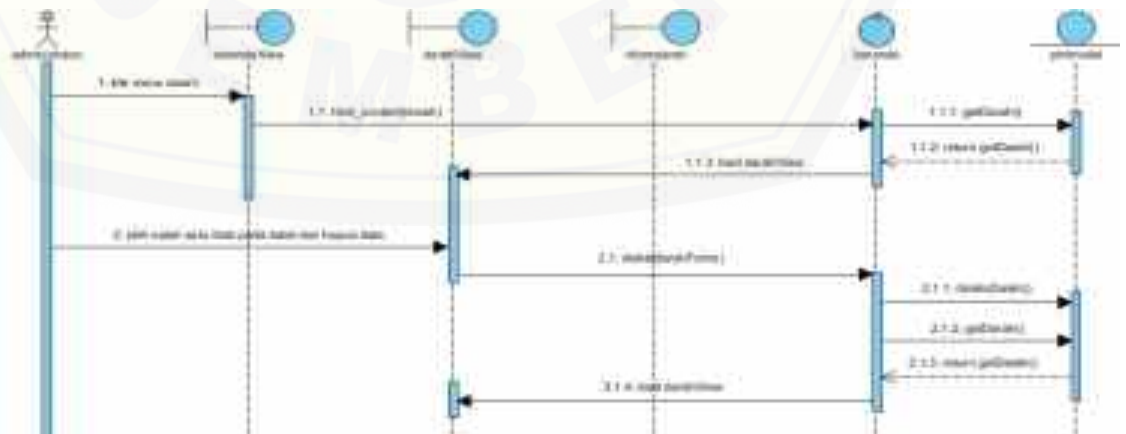
4. Manajemen Darah Masuk – Hapus Data Darah



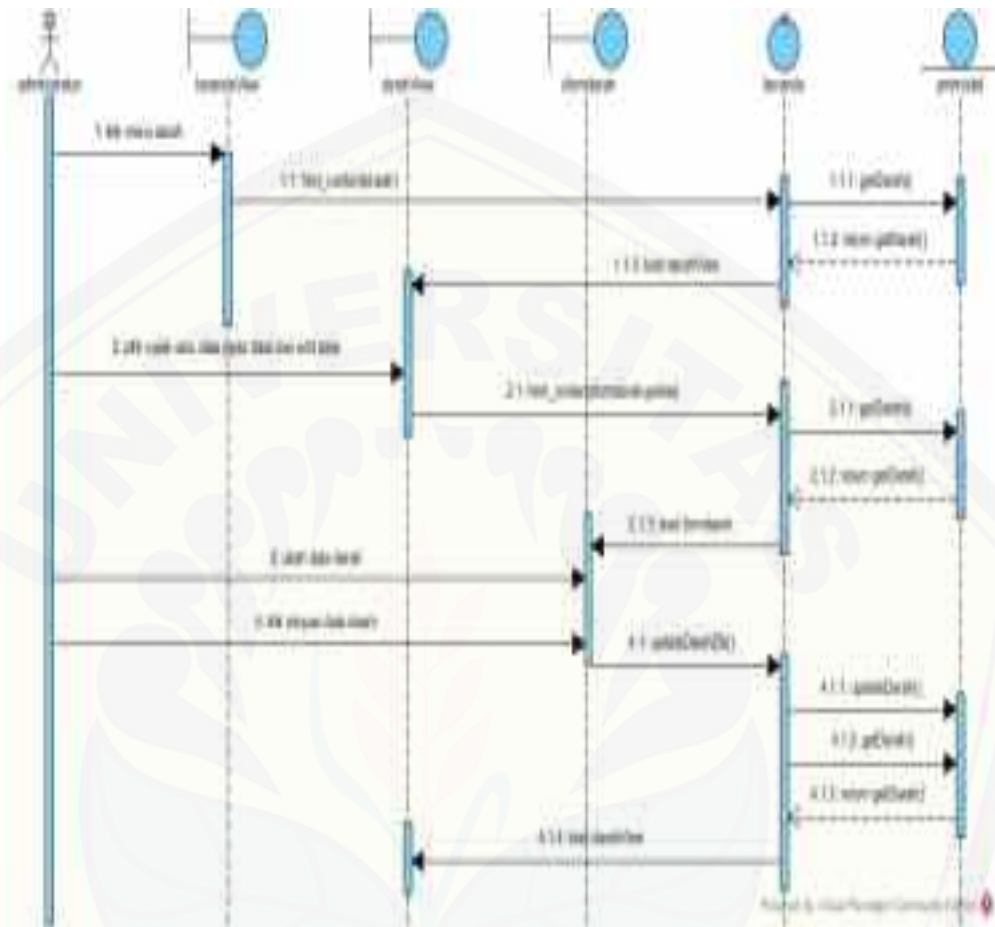
5. Manajemen Darah



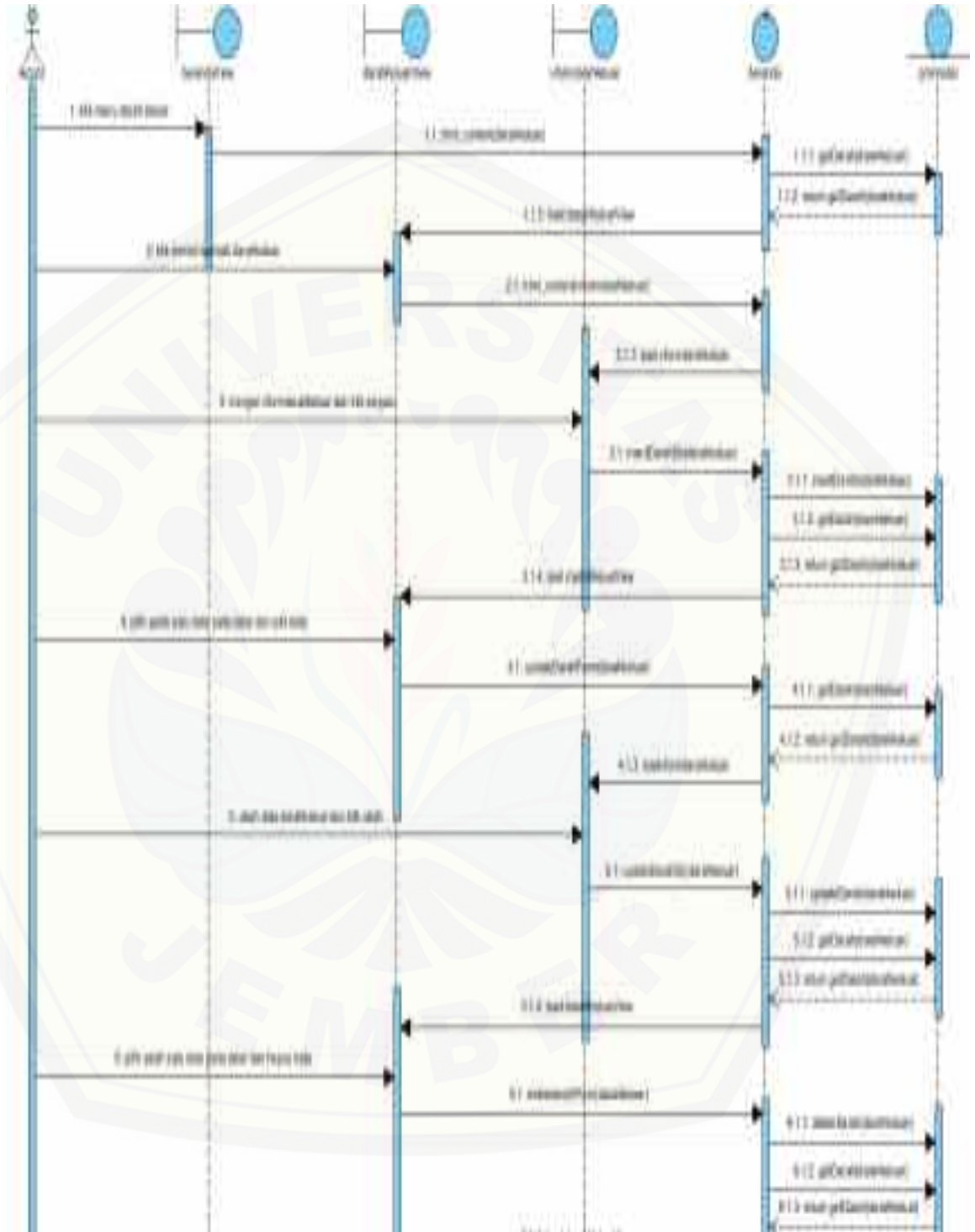
6. Manajemen Darah – Hapus Data Darah



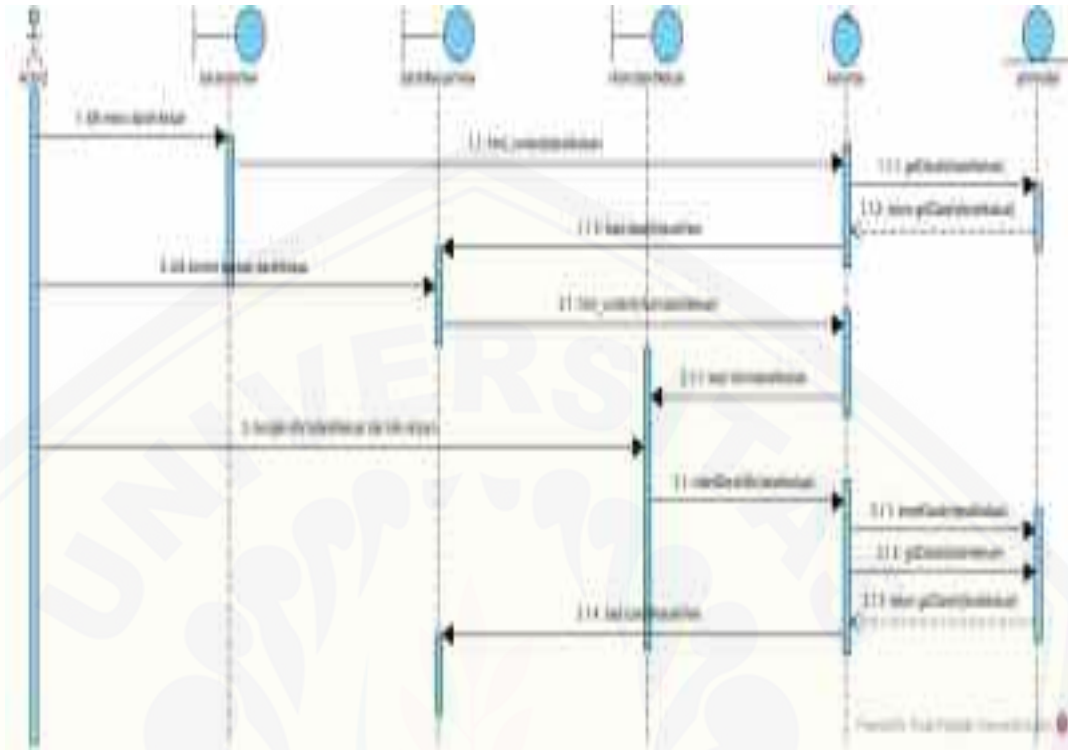
7. Manajemen Darah – Edit Data Darah



8. Manajemen Darah Keluar



9. Manajemen Darah Keluar – Input Data Darah

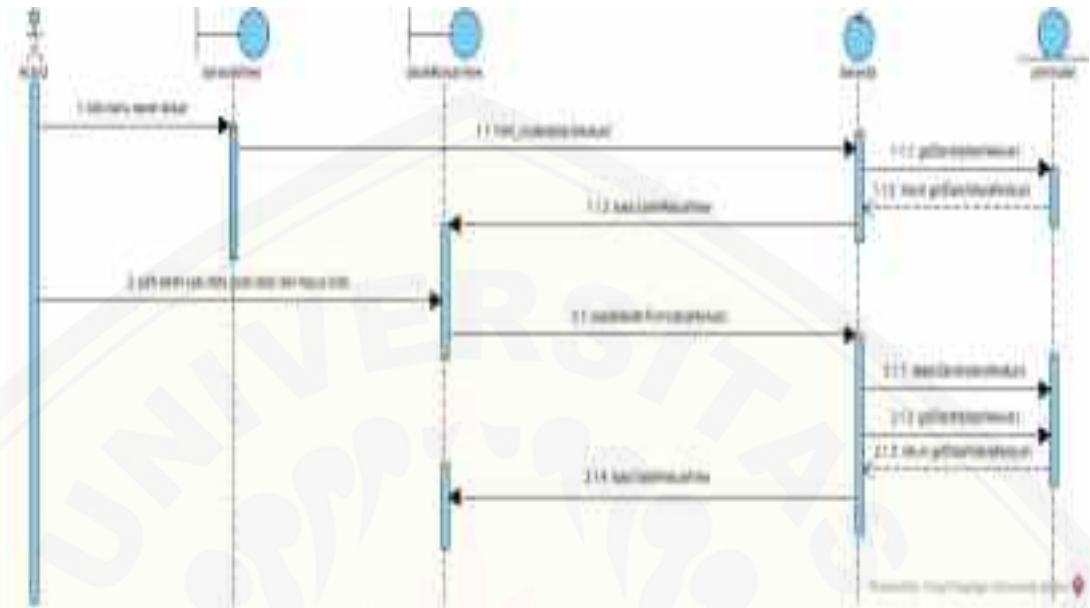


10. Manajemen Darah Keluar – Edit Data Darah





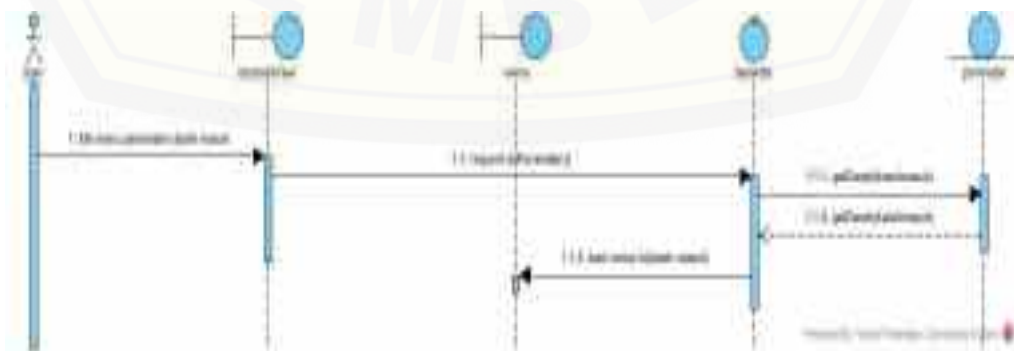
11. Manajemen Darah Keluar – Hapus Data Darah



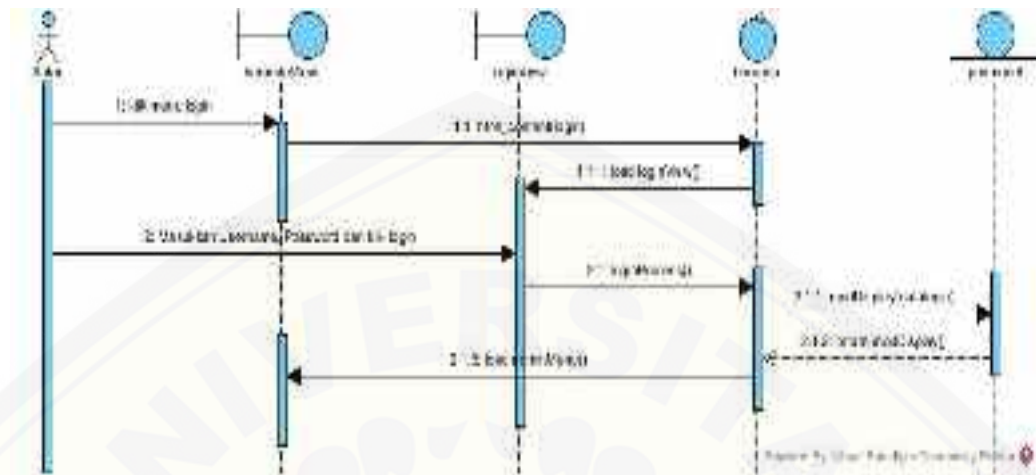
12. Peramalan Darah Keluar



13. Peramalan Darah Masuk



14. Login



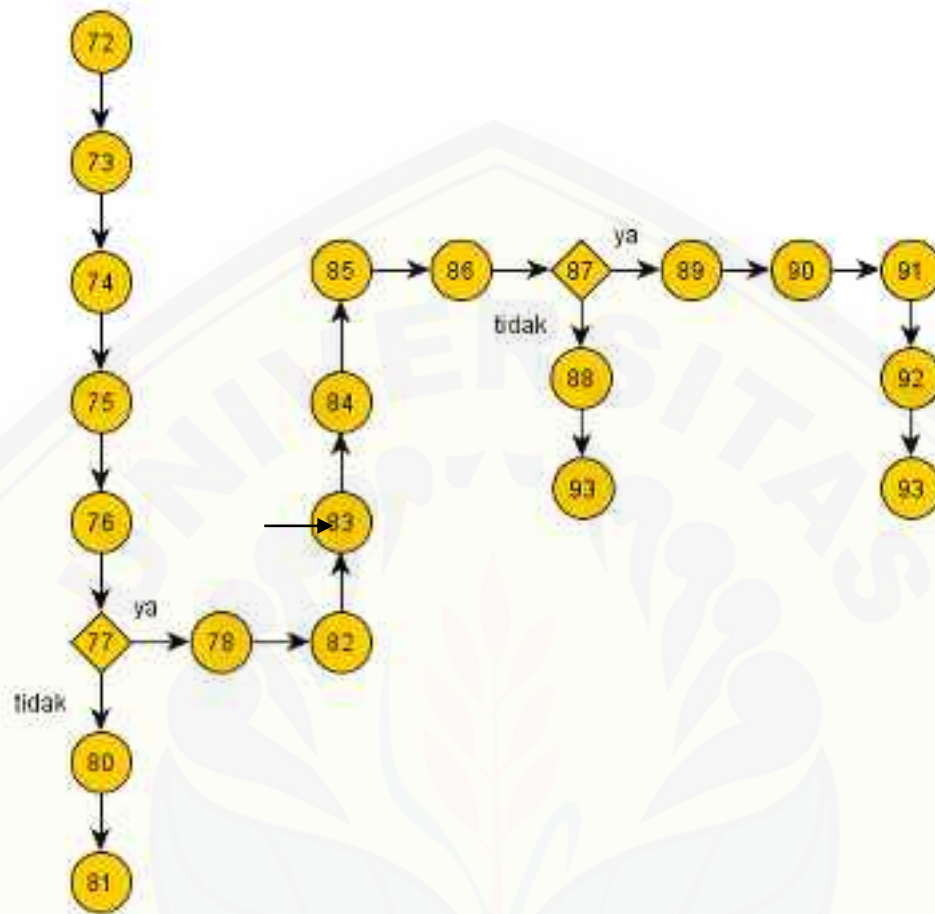
A4. Implementasi Program

1. Hasil peramalan darah masuk

MEMBER	DEWAN	AKTUAL				PERAMALAN				MAAD			
		1	2	3	4	1	2	3	4	1	2	3	4
1	JAWABAN	225	240	250	229	2	3	2	3	225	240	227	225
2	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225
3	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225
4	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225
5	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225
6	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225
7	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225
8	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225
9	DEWAN	225	240	250	229	2	3	2	3	225	240	227	225

2. Detail darah masuk





c. Cyclomatic Complexity

$CC = \text{EDGE} - \text{NODE} + 2$ $CC = 20 - 21 + 2$ $CC = 1$
--

d. Jalur Independen

Jalur 1 = 1 - 2 - 3 - 4 - 7 - 8 - 9 - 10 - 11 - 12 - 13 - 14 - 15

Jalur 2 = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 13 - 16 - 17 - 18 - 19 - 20

Jalur 3 = 1 - 2 - 3 - 4 - 5 - 6 - 21 - 22

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Insert data pada function data darah masuk	1 – 2 – 3 – 4 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15	1 – 2 – 3 – 4 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15	[ ] berhasil [ ] gagal

#### 4. Delete Darah Keluar

##### a. Listing Program

```

public function DeleteDarahKeluar($id) {
    $koneksi = mysqli_connect("localhost", "root", "", "db_sdm");
    $sql = "DELETE FROM darah_keluar WHERE id_darah_keluar = $id";
    $query = mysqli_query($koneksi, $sql);
}

```

##### b. Diagram Alir



##### c. Cyclomatic Complexity

$$CC = \text{EDGE} - \text{NODE} + 2$$

$$CC = 4 - 5 + 2$$

$$CC = 1$$

d. Jalur Independen

$$\text{Jalur} = 1 - 2 - 3 - 4$$

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	keterangan
1	Delete data pada function data DarahKeluarDb	1 - 2 - 3 - 4	1 - 2 - 3 - 4	[ ] berhasil [ ] gagal

#### A6. Kode Program

Kode program peramalan stok darah dengan menggunakan perhitungan metode *Weighted Moving Average*

```
<?php
    $bobot=6;
    $ab=$this->db->query("select * from darah")-
>result_array();
    if($mode=="in"){
        $allData=$this->db->query("SELECT
YEAR(tanggal_masuk) AS Tahun FROM vmonthyearin GROUP BY
YEAR(tanggal_masuk)")->result_array();
        $dtA= $this->db->query("SELECT
COUNT(*) AS jumlah FROM vmonthyearin")->result_array();
```

```
    }
    elseif($mode=="out"){
        $allData=$this->db->query("SELECT
YEAR(tanggal_keluar) AS Tahun FROM vmonthyearout GROUP BY
YEAR(tanggal_keluar)")->result_array();

        $dtA= $this->db->query("SELECT
COUNT(*) AS jumlah FROM vmonthyearout")->result_array();
    }
    $jmlgolA=$dtA[0]['jumlah'];
    $Tahunn[]=null;
    foreach($allData as $ad){
        $nt=$ad['Tahun'];
        array_push($Tahunn,$nt);
    }

    $cekMonth=$jmlgolA% 12; $hitung =count($allData);

    if($cekMonth==0){
        $newYear=$Tahunn[$hitung]+1;
        array_push($Tahunn,$newYear);
        $hitung=count($Tahunn);
        foreach($Tahunn as $T){
            // echo $T;
        }
        $hitung=$hitung-1;

        for($i=1;$i<=$hitung;$i++){
            // echo $Tahunn[$i];
        }
    }
}
```

```
    }  
  }  
  
?>  
  
<div class="panel ">  
  <div class="panel-heading">  
    <h4><?php  
      if($mode=="in"){  
        echo      "<strong>PERAMALAN  
WEIGHTED MOVING AVERAGE DARAH MASUK</strong>";  
      }  
      elseif($mode=="out"){  
        echo      "<strong>PERAMALAN  
WEIGHTED MOVING AVERAGE DARAH KELUAR</strong>";  
      }  
    ?>  
  
    </h4>  
  </div>  
</div>  
  
<div class="table-responsive">  
  <?php $nomor=null; $A[]=null; $AB[]=null; $B[]=null;  
$O[]=null;  
      if($mode=="out"){  
        for($TT=1;$TT<=$hitung;$TT++){  
          $tahun = $Tahunn[$TT];  
          for($i=1;$i<=12;$i++){ $nomor++;
```



```
// echo "$nomor||$tahun||</br>";
// foreach($ab as $value){
//
//     $siddarah=$value['id_darah'];
//
//     $jmlBlood= $this->db-
// >query("SELECT SUM(jumlah_darah_keluar) AS jumlah_darah_keluar FROM
// vdarahkeluar WHERE YEAR(tanggal_keluar)='$tahun' AND
// MONTH(tanggal_keluar)='$i' AND id_darah='$siddarah'")->result_array();
//
//     $jml darah=$jmlBlood[0]['jumlah_darah_keluar'];
//     if($siddarah==1){
//         array_push($A,$jml darah);
//     }elseif($siddarah==2){
//         array_push($B,$jml darah);
//     }elseif($siddarah==4){
//         array_push($AB,$jml darah);
//     }elseif($siddarah==3){
//         array_push($O,$jml darah);
//     }
// }
// echo
// "$siddarah||$jml darah||";
//
// }
// echo "</br>";
// }
```

```
    }
  }
  elseif($mode=="in"){
    for($TT=1;$TT<=$hitung;$TT++){
      $tahun = $Tahunn[$TT];
      for($i=1;$i<=12;$i++){ $nomor++;
//      echo "$nomor||$tahun|</br>";
      foreach($ab as $value){

        $iddarah=$value['id_darah'];

        $jmlBlood= $this->db-
>query("SELECT SUM(jumlah_darah_masuk) AS jumlah_darah_masuk FROM
vdarahmasuk WHERE YEAR(tanggal_masuk)='$tahun' AND
MONTH(tanggal_masuk)='$i' AND id_darah='$iddarah')->result_array());

        $jml darah=$jmlBlood[0]['jumlah_darah_masuk'];

        if($iddarah==1){

array_push($A,$jml darah);

        }elseif($iddarah==2){

array_push($B,$jml darah);

        }elseif($iddarah==4){

array_push($AB,$jml darah);

        }elseif($iddarah==3){

array_push($O,$jml darah);
```

```

}
//
echo
"$iddarah||$jml darah||";
}
//
echo "</br>";
}
}
}
}
$FA[]=null;$FB[]=null;$FAB[]=null;$FO[]=null;
$dtForecasting=$jml golA+1;
for($i=1;$i<=$dtForecasting;$i++){
    if($i > $bobot){
        $tFA=null;$tFB=null;$tFAB=null;$tFO=null;
        $batas=$i-$bobot;$control=$bobot;
        for($j=$i-1;$j>=$batas;$j--){
            $mulA=$A[$j]*$control;$mulB=$B[$j]*$control;$mulAB=$AB[$j]*$control
;$mulO=$O[$j]*$control;
            $tFA=$tFA+$mulA;
            $tFB=$tFB+$mulB;
            $tFAB=$tFAB+$mulAB;
            $tFO=$tFO+$mulO;
            //echo
"$tFA=$tFA+$mulA;$A[$j]*$control||$A[$j]||$i</br>";
            $control=$control-1;

```

```

        }
        array_push($FA,$tFA);
        array_push($FB,$tFB);
        array_push($FAB,$tFAB);
        array_push($FO,$tFO);
    }else{
        array_push($FA,"0");
        array_push($FB,"0");
        array_push($FAB,"0");
        array_push($FO,"0");
    }
}
?>

<table class="table table-striped table-bordered table-hover" id="dataTables-
example">
    <thead>
        <tr class="danger">
            <th rowspan="2"><div align="center" style="margin-bottom:
15px;">NOMOR</div></th>
            <th rowspan="2"><div align="center" style="margin-
bottom: 15px;">BULAN</div></th>
            <th colspan="4"><div align="center" >AKTUAL</div></th>
            <th colspan="4"><div align="center"
>PERAMALAN</div></th>
            <th colspan="4"><div align="center" >MAD</div></th>
        </tr>

```

```
<tr class="danger">
    <?php $stdcount=null;
        foreach($sab as $svalue){ $stdcount++;
            echo
" <th>$value[golongan_darah]</th>";
        }
        $stdcount=null;
        foreach($sab as $svalue){ $stdcount++;
            echo
" <th>$value[golongan_darah]</th>";
        }
        $stdcount=null;
        foreach($sab as $svalue){ $stdcount++;
            echo
" <th>$value[golongan_darah]</th>";
        }
    ?>
</tr>
</thead>
<tbody>
    <?php
        $nomor=0;$tbobot=null;
        for($i=0;$bobot>$i;$bobot--){
            $tbobot=$tbobot+$bobot;
        }
        //echo $tbobot;
        if($mode=="out"){
$TAA=null;$TAB=null;$TAO=null;          $TAAB=null;$TPA=null;$TPB=null;
```

```
$TPO=null;$TPAB=null;

for($TT=1;$TT<=$hitung;$TT++){
    $tahun = $Tahunn[$TT];
    for($i=1;$i<=12;$i++){
        $total=null; $nomor++;

        $nameMonth=getBulan($i);

        echo "<tr>
            <td>$nomor</td>
            <td>$nameMonth
$tahun</td>";$aktual[]=null;
        foreach($ab as $value){
            $iddarah=$value['id_darah'];
            $jmlBlood= $this->db->query("SELECT SUM(jumlah_darah_keluar) AS jumlah_darah_keluar FROM
vdarahkeluar WHERE YEAR(tanggal_keluar)='$tahun' AND
MONTH(tanggal_keluar)='$i' AND id_darah='$iddarah')->result_array());

            $jmldarah=$jmlBlood[0]['jumlah_darah_keluar'];

            $total=$total+$jmldarah;

            array_push($aktual,$jmldarah);

            echo
            "<td>$jmldarah</td>";
        }
    }
}
if($nomor >
```



```
for($c=1;$c<=4;$c++){  
  
    if($c==1){  
  
        $MRA=0;  
  
        echo "<td>$MRA</td>";  
  
    }elseif($c==2){  
  
        $MRB=0;  
  
        echo "<td>$MRB</td>";  
  
    }elseif($c==3){  
  
        $MRO=0;  
  
        echo "<td>$MRO</td>";  
  
    }elseif($c==4){  
  
        $MRAB=0;  
  
        echo "<td>$MRAB</td>";  
  
    }  
}
```



```
    }  
    elseif($nomor >  
$bobot) {  
  
    for($c=1;$c<=4;$c++){  
  
    if($c==1){ $TAA=$TAA+$aktual[$c]; $TPA=$TPA+$RA;  
  
    $MRA=$aktual[$c]-$RA;  
  
    echo "<td>$MRA</td>";  
  
    }elseif($c==2){ $TAB=$TAB+$aktual[$c]; $TPB=$TPB+$RB;  
  
    $MRB=$aktual[$c]-$RB;  
  
    echo "<td>$MRB</td>";  
  
    }elseif($c==3){ $TAO=$TAO+$aktual[$c]; $TPO=$TPO+$RO;  
  
    $MRO=$aktual[$c]-$RO;  
  
    echo "<td>$MRO</td>";  
  
    }elseif($c==4){ $TAAB=$TAAB+$aktual[$c]; $TPAB=$TPAB+$RAB;  
  
    $MRAB=$aktual[$c]-$RAB;
```

```
echo "<td>$MRAB</td>";  
  
    }  
    }  
    }  
else{  
  
for($c=1;$c<=4;$c++){  
  
if($c==1){  
  
$MRA=0;  
  
echo "<td>$MRA</td>";  
  
}elseif($c==2){  
  
$MRB=0;  
  
echo "<td>$MRB</td>";  
  
}elseif($c==3){  
  
$MRO=0;  
  
echo "<td>$MRO</td>";  
  
}
```

```
}elseif($c==4){

$MRAB=0;

echo "<td>$MRAB</td>";

}

}

}

unset($aktual);

echo "</tr>";

}

}

elseif($mode=="in"){

$TAA=null;$TAB=null;$TAO=null;          $TAAB=null;$TPA=null;$TPB=null;

$TPO=null;$TPAB=null;

for($TT=1;$TT<=$hitung;$TT++){

    $tahun = $Tahun[$TT];

    for($i=1;$i<=12;$i++){

        $total=null; $nomor++;

        $nameMonth=getBulan($i);

        echo "<tr>

            <td>$nomor</td>

            <td>$nameMonth

            $tahun</td>";$aktual[]=null;

        foreach($ab as $value){
```



```
    }  
  
    echo  
    " $RA</td>";      echo     " $RB</td>";      echo     " $RAB</td>";      echo     " $RO</td>";      if($nomor > $jmlgolA){     for($c=1;$c<=4;$c++){     if($c==1){     $MRA=0;     echo "<td>$MRA</td>";     }elseif($c==2){     $MRB=0;     echo "<td>$MRB</td>"; | | | |
```

```
    }elseif($c==3){

        $MRO=0;

        echo "<td>$MRO</td>";

    }elseif($c==4){

        $MRAB=0;

        echo "<td>$MRAB</td>";

    }

}

elseif($nomor >

$bobot) {

    for($c=1;$c<=4;$c++){

        if($c==1){ $TAA=$TAA+$aktual[$c]; $TPA=$TPA+$RA;

        $MRA=$aktual[$c]-$RA;

        echo "<td>$MRA</td>";

    }elseif($c==2){$TAB=$TAB+$aktual[$c]; $TPB=$TPB+$RB;
```

```
$MRB=$aktual[$c]-$RB;

echo "<td>$MRB</td>";

}elseif($c==3){$TAO=$TAO+$aktual[$c]; $TPO=$TPO+$RO;

$MRO=$aktual[$c]-$RO;

echo "<td>$MRO</td>";

}elseif($c==4){$TAAB=$TAAB+$aktual[$c]; $TPAB=$TPAB+$RAB;

$MRAB=$aktual[$c]-$RAB;

echo "<td>$MRAB</td>";

}

}

}

else{

for($c=1;$c<=4;$c++){

if($c==1){

$MRA=0;
```

```
echo "<td>$MRA</td>";

}elseif($c==2){

$MRB=0;

echo "<td>$MRB</td>";

}elseif($c==3){

$MRO=0;

echo "<td>$MRO</td>";

}elseif($c==4){

$MRAB=0;

echo "<td>$MRAB</td>";

}

}

}

unset($aktual);

echo "</tr>";

}

}

}
```



```

?>

</tbody>
</table>

<?php
    $RTA=$TAA-$TPA;      $RTB=$TAB-$TPB;      $RTO=$TAO-
$TPO;      $RTAB=$TAAB-$TPAB;
    echo "MAD GOLONGAN DARAH A      : $TAA - $TPA      =
$RTA<br>";
    echo "MAD GOLONGAN DARAH B      : $TAB      - $TPB
= $RTB<br>";
    echo "MAD GOLONGAN DARAH O      : $TAO      - $TPO      =
$RTO<br>";
    echo "MAD GOLONGAN DARAH AB      : $TAAB - $TPAB =
$RTAB<br>";

?>

<hr />

</div>

<?PHP
function getBulan($data){
    if($data==1){
        $nmBulan="JANUARI";
    }elseif($data==2){
        $nmBulan="FEBRUARI";
    }elseif($data==3){
        $nmBulan="MARET";
    }
}

```

```
}elseif($data==4){
    $nmBulan="APRIL";
}elseif($data==5){
    $nmBulan="MEI";
}elseif($data==6){
    $nmBulan="JUNI";
}elseif($data==7){
    $nmBulan="JULI";
}elseif($data==8){
    $nmBulan="AGUSTUS";
}elseif($data==9){
    $nmBulan="SEPTEMBER ";
}elseif($data==10){
    $nmBulan="OKTOBER";
}elseif($data==11){
    $nmBulan="NOVEMBER";
}elseif($data==12){
    $nmBulan="DESEMBER";
}

return $nmBulan;
}
?>
```