



**SISTEM PERENCANAAN DAN PERAMALAN DISTRIBUSI PRODUK
BERDASARKAN JUMLAH PERMINTAAN MENGGUNAKAN METODE
WEIGHT MOVING AVERAGE (WMA)
(STUDI KASUS : PUSAT OLEH-OLEH PURNAMAJATI)**

SKRIPSI

Oleh

ANINDYA PALMITRAAZZAH

NIM 132410101080

PROGRAM STUDI SISTEM INFORMASI

UNIVERSITAS JEMBER

2017



**SISTEM PERENCANAAN DAN PERAMALAN DISTRIBUSI PRODUK
BERDASARKAN JUMLAH PERMINTAAN MENGGUNAKAN METODE
WEIGHT MOVING AVERAGE (WMA)
(STUDI KASUS : PUSAT OLEH-OLEH PURNAMA JATI)**

SKRIPSI

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

Oleh

ANINDYA PALMITRAAZZAH

NIM 132410101080

PROGRAM STUDI SISTEM INFORMASI

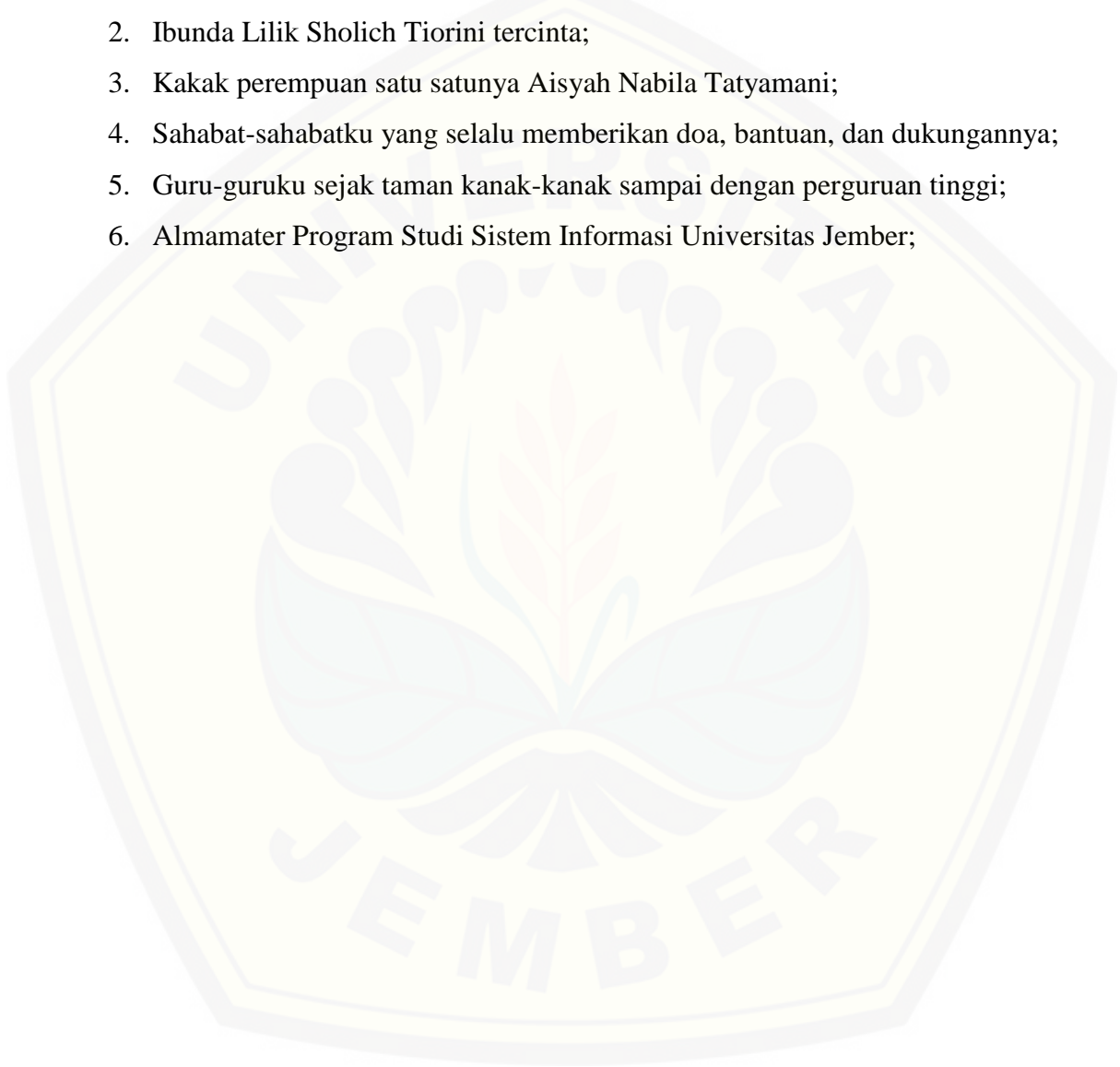
UNIVERSITAS JEMBER

2017

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT, Tuhan Yang Maha Pengasih yang senantiasa memberikan kemudahan dan kelancaran dalam menyelesaikan tugas akhir ini;
2. Ibunda Lilik Sholich Tiorini tercinta;
3. Kakak perempuan satu satunya Aisyah Nabila Tatyamani;
4. Sahabat-sahabatku yang selalu memberikan doa, bantuan, dan dukungannya;
5. Guru-guruku sejak taman kanak-kanak sampai dengan perguruan tinggi;
6. Almamater Program Studi Sistem Informasi Universitas Jember;



MOTO

“Jadilah seperti bola bekel, yang akan tetap melambung jika dijatuhkan ke bawah,
Semakin keras menjatuhkannya, semakin keatas dia akan melambung”
(Anindya Palmitraazzah)



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

nama : Anindya Palmitraazzah

NIM : 132410101080

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Sistem Perencanaan Dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan Menggunakan Metode *Weight Moving Average* (Wma) (Studi Kasus : Pusat Oleh-Oleh Purnamajati)”, 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 mendapatkan sanksi akademik jika di kemudian hari pernyataan ini tidak benar.

Jember, Mei 2017

Yang menyatakan,

Anindya Palmitraazzah

NIM 132410101080

SKRIPSI

SISTEM PERENCANAAN DAN PERAMALAN DISTRIBUSI PRODUK
BERDASARKAN JUMLAH PERMINTAAN MENGGUNAKAN METODE
WEIGHT MOVING AVERAGE (WMA)
(STUDI KASUS : PUSAT OLEH-OLEH PURNAMAJATI)

Oleh

Anindya Palmitraazzah

NIM 132410101080

Pembimbing :

Dosen Pembimbing Utama : Prof. Drs. Slamir, M.Comp.Sc.,Ph.D

Dosen Pembimbing Pendamping : Oktalia Juwita, S.kom., M.MT

PENGESAHAN PEMBIMBING

Skripsi berjudul “Sistem Perencanaan Dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus : Pusat Oleh-Oleh Purnamajati)”, telah diuji dan disahkan pada:

hari, tanggal : Jumat, 19 Mei 2017

Tempat : Program Studi Sistem Informasi Universitas Jember

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Prof. Drs. Slamir, M.Comp.Sc.,Ph.D
NIP. 196704201992011001

Oktalia Juwita, S.kom., M.MT
NIP. 198110202014042001

PENGESAHAN PENGUJI

Skripsi berjudul “Sistem Perencanaan Dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus : Pusat Oleh-Oleh Purnamajati)”, telah diuji dan disahkan pada:

hari, tanggal : Jumat, 19 Mei 2017

Tempat : Program Studi Sistem Informasi Universitas Jember

Tim Penguji,

Penguji I,

Penguji II,

Dr. Saiful Bukhori, S.T., M.Kom.
NIP 196811131994121001

M. Arief Hidayat, S.Kom., M.Kom.
NIP 1981101232010121003

Mengesahkan

Ketua Program Studi,

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

NIP. 196704201992011001

RINGKASAN

Sistem Perencanaan dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus: Pusat Oleh-Oleh Purnama Jati)

Pusat oleh-oleh purnama jati merupakan salah satu *home industry* di Jember yang bergerak dalam bidang makanan. Salah satu produk unggulan yang diproduksi oleh purnama jati adalah proll tape. Tingginya permintaan konsumen terhadap produk proll tape menyebabkan pihak purnama jati memperluas jaringan pemasaran produknya dengan mendistribusikan ke sejumlah *outlet* yang telah bekerja sama, akan tetapi dalam penelitian ini penulis membatasi penelitian hanya fokus kepada 6 *outlet* saja. Permasalahan yang sering terjadi pada pusat oleh-oleh purnama jati adalah kurangnya analisa dalam memperkirakan permintaan konsumen pada masing masing *outlet* sehingga proses pendistribusian yang dilakukan oleh pihak purnama jati disama ratakan untuk masing masing *outlet* tanpa memperhitungkan *outlet* mana yang memiliki jumlah konsumen lebih banyak diantara yang lain dan *stock* yang masih tersedia pada masing masing *outlet* tersebut. Hal ini menyebabkan terjadinya proses pengiriman yang berulang setiap harinya untuk *outlet-outlet* tertentu yang memang memiliki jumlah konsumen lebih tinggi diantara yang lain. Untuk membantu meramalkan jumlah produk yang akan didistribusikan pada masing masing *outlet* maka digunakan metode *Weight Moving Average* (WMA) yang diimplementasikan pada sistem peramalan tersebut. Sedangkan untuk mengetahui kondisi *stock* produk pada setiap *outlet* maka diperlukan sistem yang mampu menghubungkan pihak marketing dari purnama jati dengan pihak pengelola pada setiap *outlet*. Sehingga sangat diperlukan pengembangan aplikasi dari “Sistem Perencanaan dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus: Pusat Oleh-Oleh Purnama Jati)”.

PRAKATA

Puji syukur kehadiran Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Sistem Peramalan Permintaan Bahan Baku Pakan Ternak untuk Menunjang *Supply Chain Management* Menggunakan Metode *Winter Exponential Smoothing* (Studi Kasus : UD. Setia Budi Kabupaten 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. Allah S.W.T. dan Rosullullah S.A.W dan Para Sahabat.
2. Prof. Drs. Slamini, M.Comp.Sc., Ph.D., selaku Ketua Program Studi Sistem Informasi Universitas Jember.
3. Prof. Drs. Slamini, M.Comp.Sc., Ph.D., selaku Dosen Pembimbing Utama dan Oktalia Juwita, S.kom., M.MT selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi;
4. Yanuar Nurdiansyah ST,.M.Cs. sebagai dosen pembimbing akademik, yang telah mendampingi penulis selama menempuh pendidikan S1.
5. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember;
6. Ibunda Lilik Sholich Tiorini yang selalu mendoakan dan mendukung;
7. Kakak Perempuan Aisyah Nabila Tatyamani;
8. Kakek Soemardi dan Kakek Soetamin beserta keluarga besar;
9. Sahabat seperjuangan yang selalu menemani dan memberikan semangat serta doa Khoirunnisa' Afandi, Khoirun Nisaa H, Putri Damayanti, Wenny Hardiyanti P, Safitri Febryanti A, Sugiarti, Helma Daniar.

10. Teman-teman seperjuangan yang menemani dan membantu dalam terselesaikannya skripsi ini M. Bustommy M, Angga Septiawan;
11. Teman-teman seperjuanganku INTENTION angkatan 2013 dan semua mahasiswa Program Studi Sistem Informasi yang telah menjadi keluarga kecil bagi penulis selama menempuh pendidikan S1;
12. Keluarga besar kost az-zahraa Jalan Nias 1 Nomer 3 yang telah menjadi rumah singgah dan keluarga penulis selama di Jember;
13. Keluarga besar Ibu Dina sebagai pemilik dari Pusat Oleh-oleh Purnama Jati yang selalu memberi doa dan dukungan sepenuhnya bagi penulis selama penelitian berlangsung;
14. 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, Mei 2017

Penulis

DAFTAR ISI

SKRIPSI.....	ii
PERSEMBAHAN.....	iii
MOTO.....	iv
PERNYATAAN.....	v
PENGESAHAN PEMBIMBING.....	vii
PENGESAHAN PENGUJI.....	viii
RINGKASAN.....	ix
PRAKATA.....	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xviii
DAFTAR GAMBAR.....	xxii
BAB. 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	3
1.3 Tujuan.....	3
1.4 Batasan Masalah.....	3
1.5 Sistematika Penulisan.....	4
BAB. 2 TINJAUAN PUSTAKA.....	5
2.1 Penelitian Terdahulu.....	5
2.2 Proll Tape.....	6
2.3 Peramalan Permintaan (<i>Forecasting Demand</i>).....	6
2.4 Distribusi Produk.....	7
2.5 <i>Weight Moving Average (WMA)</i>	7
2.6 Nilai Ketepatan Peramalan.....	8
2.7 Perhitungan Metode <i>Weight Moving Average (WMA)</i>	8
2.8 Contoh Penentuan Kesalahan Peramalan.....	9
BAB. 3 METODOLOGI PENELITIAN.....	11
3.1 Tempat dan Waktu Penelitian.....	11
3.2 Tahap Penelitian.....	11
3.3 Tahap Pengumpulan Data.....	12
3.3.1 Studi Literatur.....	12

3.3.2	Tahap Wawancara.....	13
3.3.3	Tahap Dokumentasi	13
3.4	Tahap Perancangan Sistem	13
3.4.1	Analisis Kebutuhan	14
3.4.2	Desain Sistem.....	15
3.4.3	Implementasi	15
3.4.4	Tahap Pengujian.....	15
3.5	Perawatan (<i>Maintenance</i>).....	16
3.6	Gambaran Sistem	16
BAB. 4	ANALISIS DAN PERANCANGAN SISTEM	19
4.1	Analisis Kebutuhan	19
4.1.1	Kebutuhan Fungsional	19
4.1.2	Kebutuhan Non-Fungsional	20
4.2	Desain Sistem.....	20
4.2.1	<i>Business Process</i>	20
4.2.2	<i>Use Case Diagram</i>	21
4.2.3	<i>Use Case Skenario</i>	26
4.2.4	<i>Sequence Diagram</i>	30
4.2.5	<i>Activity Diagram</i>	32
4.2.6	<i>Class Diagram</i>	34
4.2.7	<i>Entity Relationship Diagram (ERD)</i>	35
4.3	Penulisan Kode Program.....	36
4.3.1	Kode Program <i>Login</i>	36
4.3.2	Kode Program Pengelolaan Data Pengguna	36
4.3.3	Kode Program Melihat Data Pengguna.....	36
4.3.4	Kode Program Pengelolaan Data <i>Outlet</i>	37
4.3.5	Kode Program Melihat Data <i>Outlet</i>	37
4.3.6	Kode Program Pengelolaan Data Hasil Peramalan	37
4.3.7	Kode Program Melihat Data Hasil Peramalan	38
4.3.8	Kode Program Pengelolaan Data Pendistribusian Produk	39
4.3.9	Kode Program Melihat Detail Hasil Penjualan dan Sisa Produk.....	39

4.3.10	Kode Program Melihat Data Hasil Penjualan dan Pendistribusian Produk.	39
4.3.11	Kode Program Input Transaksi Penjualan.....	39
4.3.12	Kode Program Melihat Rekap Data Hasil Transaksi Penjualan.	39
4.4	Pengujian Sistem.....	39
4.4.1	Pengujian White Box	39
4.4.2	Pengujian Black Box.....	41
BAB. 5	HASIL DAN PEMBAHASAN.....	43
5.1	Sistem peramalan pendistribusian produk menggunakan metode <i>Weight Moving Average (WMA)</i>	43
5.1.1	Data Produk.....	44
5.1.2	Data Outlet	44
5.2	Hasil Implementasi Coding pada Sistem Peramalan Distribusi Produk	45
5.2.1	Tampilan Login Screen	45
5.2.2	Tampilan Halaman Awal (<i>Dashboard</i>) Admin.....	45
5.2.3	Tampilan Halaman Data Pengguna (Admin).....	46
5.2.4	Tampilan Edit Data Pengguna (Admin).....	47
5.2.5	Tampilan Halaman Awal (<i>Dashboard</i>) Marketing	48
5.2.6	Tampilan Halaman Data Peramalan (Marketing)	49
5.2.7	Tampilan Halaman Data Pengiriman (Marketing).....	50
5.2.8	Tampilan Halaman Input Data Pengiriman (Marketing)	52
5.2.9	Tampilan Halaman Detail Data Pengiriman (Marketing).....	52
5.2.10	Tampilan Halaman Data <i>Outlet</i> (Marketing)	53
5.2.11	Tampilan Halaman Edit Data <i>Outlet</i> (Marketing).	54
5.2.12	Tampilan Halaman Utama (<i>Dashboard</i>) Eksekutif.	55
5.2.13	Tampilan Halaman Data Pengguna (Eksekutif).....	56
5.2.14	Tampilan Halaman Data Peramalan (Eksekutif).....	57
5.2.15	Tampilan Halaman Data Pengiriman (Eksekutif).....	57
5.2.16	Tampilan Halaman Data <i>Outlet</i> (Eksekutif).	58
5.2.17	Tampilan Halaman Utama (<i>Dashboard</i>) Admin <i>Outlet</i>	59
5.2.18	Tampilan Halaman Data Penjualan (Admin <i>Outlet</i>).	59
5.2.19	Tampilan Halaman Detail Data Penjualan (Admin <i>Outlet</i>).....	60

5.2.20	Tampilan Halaman Transaksi Penjualan (Admin <i>Outlet</i>).....	61
5.2.21	Tampilan Halaman Detail Transaksi Penjualan (Admin <i>Outlet</i>).....	62
5.3	Pengujian Implementasi Metode <i>Weight Moving Average</i> Pada Peramalan Distribusi Produk.....	63
5.3.1	Hasil Peramalan Produk Proll Tape pada <i>Outlet A</i>	64
5.3.1.1	Proll Tape Besar	64
5.3.1.2	Proll Tape Kecil.....	65
5.3.2	Hasil Peramalan Produk Proll Tape pada <i>Outlet B</i>	66
5.3.2.1	Proll Tape Besar	66
5.3.2.2	Proll Tape Kecil.....	67
5.3.3	Hasil Peramalan Produk Proll Tape pada <i>Outlet C</i>	68
5.3.3.1	Proll Tape Besar	68
5.3.3.1	Proll Tape Kecil.....	69
5.3.4	Hasil Peramalan Produk Proll Tape pada <i>Outlet D</i>	70
5.3.4.1	Proll Tape Besar	70
5.3.4.2	Proll Tape Kecil.....	71
5.3.5	Hasil Peramalan Produk Proll Tape pada <i>Outlet E</i>	72
5.3.5.1	Proll Tape Besar	72
5.3.5.2	Proll Tape Kecil.....	73
5.3.6	Hasil Peramalan Produk Proll Tape pada <i>Outlet F</i>	74
5.3.6.1	Proll Tape Besar	74
5.3.6.2	Proll Tape Kecil.....	75
5.3.7	Perhitungan metode <i>Weight Moving Average</i> (WMA).....	76
5.4	Pembahasan.....	80
5.4.1	Pembahasan Hasil Implentasi Metode <i>Weight Moving Average</i> pada Sistem Peramalan Distribusi Produk pada Pusat Oleh-oleh Purnama Jati	81
BAB. 6	PENUTUP.....	85
6.1	Kesimpulan	85
6.2	Saran.....	85
	DAFTAR PUSTAKA	87
	LAMPIRAN.....	88
A.	<i>Use Case</i> Skenario	88

A.1	Skenario Use Case Login	88
A.2	Skenario <i>Use Case</i> Pengelolaan Data Pengguna	95
A.3	Skenario <i>Use Case</i> Melihat Data Pengguna.....	99
A.4	Skenario <i>Use Case</i> Pengelolaan Data <i>Outlet</i>	100
A.5	Skenario <i>Use Case</i> Melihat Data <i>Outlet</i>	103
A.6	Skenario <i>Use Case</i> Melihat Data Hasil Peramalan	104
A.7	Skenario <i>Use Case</i> Pengelolaan Data Pendistribusian Produk	105
A.8	Skenario <i>Use Case</i> Melihat Detail Hasil Penjualan dan Sisa Produk	108
A.9	Skenario <i>Use Case</i> Melihat Data Hasil Penjualan dan Pendistribusian Produk	110
A.10	Skenario Input Transaksi Penjualan.....	111
A.11	Skenario Melihat Rekap Hasil Transaksi Penjualan	113
B.	<i>Sequence Diagram</i>	116
B.1	<i>Sequence Diagram</i> Login	116
B.2	<i>Sequence Diagram</i> Pengelolaan Data Pengguna	118
B.3	<i>Sequence Diagram</i> Melihat Data Pengguna.....	119
B.4	<i>Sequence Diagram</i> Pengelolaan Data <i>Outlet</i>	119
B.5	<i>Sequence Diagram</i> Melihat Data <i>Outlet</i>	120
B.6	<i>Sequence Diagram</i> Melihat Data Hasil Peramalan	120
B.7	<i>Sequence Diagram</i> Pengelolaan Data Pendistribusian	121
B.8	<i>Sequence Diagram</i> Melihat Detail Hasil Penjualan dan Sisa Produk	122
B.9	<i>Sequence Diagram</i> Melihat Data Hasil Penjualan dan Pendistribusian Produk	123
B.10	<i>Sequence Diagram</i> Input Transaksi Penjualan.....	124
B.11	<i>Sequence Diagram</i> Melihat Rekap Hasil Transaksi Penjualan	125
C.	<i>Activity Diagram</i>	126
C.1	<i>Activity Diagram</i> Login.....	126
C.2	<i>Activity Diagram</i> Pengelolaan Data Pengguna	131
C.3	<i>Activity Diagram</i> Melihat Data Pengguna	132
C.4	<i>Activity Diagram</i> Pengelolaan Data <i>Outlet</i>	132
C.5	<i>Activity Diagram</i> Melihat Data <i>Outlet</i>	133
C.6	<i>Activity Diagram</i> Melihat Data Hasil Peramalan.....	134
C.7	<i>Activity Diagram</i> Pengelolaan Data Pendistribusian Produk.....	135

C.8	<i>Activity Diagram</i> Melihat Detail Hasil Penjualan dan Sisa Produk..	136
C.9	<i>Activity Diagram</i> Melihat Data Hasil Penjualan dan Pendistribusian Produk	137
C.10	<i>Activity Diagram</i> Input Transaksi Penjualan	138
C.11	<i>Activity Diagram</i> Melihat Rekap Hasil Transaksi Penjualan.....	139
D.	Kode Program	140
D.1	Kode Program <i>Login</i>	140
D.2	Kode Program Pengelolaan Data Pengguna	141
D.3	Kode Program Pengelolaan Data <i>Outlet</i>	146
D.4	Kode Program Pengelolaan Data Peramalan	148
D.5	Kode Program Pengelolaan Data Pendistribusian Produk	150
D.6	Kode Program Pengelolaan Data Pendistribusian Produk.....	153
D.7	Kode Program Pengelolaan Data Penjualan Produk.....	156
E.	Pengujian <i>White Box</i>	163
E.1	Kelas Controller	163
E.2	Kelas Model	171
F.	Pengujian Black Box.....	172
F.1	Fitur <i>Login</i>	172
F.2	Fitur Pengelolaan Data Pengguna	172
F.3	Fitur Melihat Data Pengguna	173
F.4	Fitur Pengelolaan Data <i>Outlet</i>	173
F.5	Fitur Melihat Data <i>Outlet</i>	174
F.6	Fitur Melihat Data Hasil Peramalan.....	174
F.7	Fitur Pengelolaan Data Pendistribusian Produk.....	174
F.8	Fitur Pengelolaan Data Penjualan Produk	175
F.9	Fitur Melihat Data Hasil Penjualan dan Pendistribusian Produk.....	176
F.10	Fitur Input Transaksi Penjualan	176
F.11	Fitur Melihat Rekap Hasil Transaksi Penjualan.....	177

DAFTAR TABEL

Tabel 2.1 Hasil Peramalan Metode Weight Moving Average dengan 3 dan 5 bobot	9
Tabel 2.2 Hasil Perhitungan MAD, MSE, MAPE dengan 3 bobot	9
Tabel 2.3 Hasil Perhitungan MAD, MSE, MAPE dengan 5 bobot	10
Tabel 4.1 Definisi Aktor	23
Tabel 4.2 Definisi Use Case.....	24
Tabel 4.3 Skenario Use Case Pengelolaan Data Peramalan Distribusi Produk	27
Tabel 4.4 Kode Program Function tambah	37
Tabel 4.5 Kode Program Function tambah	38
Tabel 4.6 Test Case Function Tambah.....	40
Tabel 4.7 Pengujian Black Box Pengelolaan Data Hasil Peramalan	42
Tabel 5.1 Data Produk.....	44
Tabel 5.2 Data Outlet	44
Tabel 5.3 Hasil Peramalan Perhitungan Metode WMA pada Outlet A	64
Tabel 5.4 Hasil Peramalan Perhitungan Metode WMA pada Outlet A	65
Tabel 5.5 Hasil Peramalan Perhitungan Metode WMA pada Outlet B	66
Tabel 5.6 Hasil Peramalan Perhitungan Metode WMA pada Outlet B	67
Tabel 5.7 Hasil Peramalan Perhitungan Metode WMA pada Outlet C	68
Tabel 5.8 Hasil Peramalan Perhitungan Metode WMA pada Outlet C	69
Tabel 5.9 Hasil Peramalan Perhitungan Metode WMA pada Outlet D	70
Tabel 5.10 Hasil Peramalan Perhitungan Metode WMA pada Outlet D	71
Tabel 5.11 Hasil Peramalan Perhitungan Metode WMA pada Outlet E.....	72
Tabel 5.12 Hasil Peramalan Perhitungan Metode WMA pada Outlet E.....	73
Tabel 5.13 Hasil Peramalan Perhitungan Metode WMA pada Outlet F.....	74
Tabel 5.14 Hasil Peramalan Perhitungan Metode WMA pada Outlet F.....	75
Tabel 5.15 Contoh Perhitungan Metode Weight Moving Average	76
Tabel 5.16 Contoh Tabel Analisa Perbandingan Metode WMA	82
Tabel A. 1 Skenario Use Case Login Admin.....	88
Tabel A. 2 Skenario Use Case Login Marketing	89
Tabel A. 3 Skenario Use Case Login Eksekutif.....	90
Tabel A. 4 Skenario Use Case Login Admin Outlet A	90
Tabel A. 5 Skenario Use Case Login Admin Outlet B	91
Tabel A. 6 Skenario Use Case Login Admin Outlet C	92
Tabel A. 7 Skenario Use Case Login Admin Outlet D.....	93
Tabel A. 8 Skenario Use Case Login Admin Outlet E	94
Tabel A. 9 Skenario Use Case Login Admin Outlet F.....	95
Tabel A. 10 Skenario Use Case Pengelolaan Data Pengguna.....	96
Tabel A. 11 Skenario Use Case Melihat Data Pengguna.....	99
Tabel A. 12 Skenario Use Case Pengelolaan Data Outlet	100

Tabel A. 13 Skenario Use Case Melihat Data Outlet.....	103
Tabel A. 14 Skenario Use Case Melihat Data Hasil Peramalan	104
Tabel A. 15 Skenario Use Case Pengelolaan Data Pendistribusian Produk	105
Tabel A. 16 Skenario Use Case Melihat Detail Hasil Penjualan dan Sisa Produk	108
Tabel A. 17 Skenario Use Case Melihat Data Hasil Penjualan dan Pendistribusian Produk	110
Tabel A. 18 Skenario Use Case Pengelolaan Data Transaksi Penjualan	111
Tabel A. 19 Skenario Use Case Melihat Rekap Hasil Transaksi Penjualan	113
Tabel D. 1 Kode Program Function Login.....	140
Tabel D. 2 Kode Program Function Construct.....	140
Tabel D. 3 Kode Program Function Index	141
Tabel D. 4 Kode Program Function Login.....	141
Tabel D. 5 Kode Program Function Logout.....	141
Tabel D. 6 Kode Program Function Register	142
Tabel D. 7 Kode Program Function TampilData	142
Tabel D. 8 Kode Program Function getStatus	142
Tabel D. 9 Kode Program Function getJabatan	142
Tabel D. 10 Kode Program Function getOutlet	143
Tabel D. 11 Kode Program Function edit	143
Tabel D. 12 Kode Program Function Update.....	143
Tabel D. 13 Kode Program Function delete.....	144
Tabel D. 14 Kode Program Function index	144
Tabel D. 15 Kode Program Function submit	144
Tabel D. 16 Kode Program Function edit	145
Tabel D. 17 Kode Program Function update	145
Tabel D. 18 Kode Program Function delete.....	146
Tabel D. 19 Kode Program Function Register.....	146
Tabel D. 20 Kode Program Function getData.....	146
Tabel D. 21 Kode Program Function edit	146
Tabel D. 22 Kode Program Function update	147
Tabel D. 23 Kode Program Function delete.....	147
Tabel D. 24 Kode Program Function index	147
Tabel D. 25 Kode Program Function submit	148
Tabel D. 26 Kode Program Function edit	148
Tabel D. 27 Kode Program Function update	148
Tabel D. 28 Kode Program Function delete.....	148
Tabel D. 29 Kode Program Function viewData.....	149
Tabel D. 30 Kode Program Function viewOutlet.....	149
Tabel D. 31 Kode Program Function viewProduk.....	149
Tabel D. 32 Kode Program Function index	149

Tabel D. 33 Kode Program Function filter	150
Tabel D. 34 Kode Program Function tambah	150
Tabel D. 35 Kode Program Function viewData.....	150
Tabel D. 36 Kode Program Function viewOutlet	151
Tabel D. 37 Kode Program Function viewProduk.....	151
Tabel D. 38 Kode Program Function viewStatus.....	151
Tabel D. 39 Kode Program Function edit	151
Tabel D. 40 Kode Program Function update	151
Tabel D. 41 Kode Program Function index	152
Tabel D. 42 Kode Program Function filter	152
Tabel D. 43 Kode Program Function edit	152
Tabel D. 44 Kode Program Function update	153
Tabel D. 45 Kode Program function viewData.....	153
Tabel D. 46 Kode Program Function ViewOutlet	153
Tabel D. 47 Kode Program Function ViewProduk.....	154
Tabel D. 48 Kode Program Function ViewStatus.....	154
Tabel D. 49 Kode Program Function Edit	154
Tabel D. 50 Kode Program Function Update.....	154
Tabel D. 51 Kode Program Function detail	155
Tabel D. 52 Kode Program Function Index	155
Tabel D. 53 Kode Program Function Filter	155
Tabel D. 54 Kode Program Function Edit	156
Tabel D. 55 Kode Program Function Update.....	156
Tabel D. 56 Kode Program Function Detail	156
Tabel D. 57 Kode Program function viewData.....	156
Tabel D. 58 Kode Program Function viewOutlet	157
Tabel D. 59 Kode Program Function viewProduk.....	157
Tabel D. 60 Kode Program Function edit	157
Tabel D. 61 Kode Program Function Transaksi.....	157
Tabel D. 62 Kode Program Function Sisa	157
Tabel D. 63 Kode Program Function Beli	158
Tabel D. 64 Kode Program Function LogicTrans.....	158
Tabel D. 65 Kode Program Function detailTransaksi.....	159
Tabel D. 66 Kode Program Function ProdukDetail.....	160
Tabel D. 67 Kode Program Function totalPembelian	160
Tabel D. 68 Kode Program Function detail	160
Tabel D. 69 Kode Program Function index	161
Tabel D. 70 Kode Program Function Transaksi.....	161
Tabel D. 71 Kode Program Function Beli	161
Tabel D. 72 Kode Program Function detailTransaksi.....	162
Tabel D. 73 Kode Program Function filter	162
Tabel D. 74 Kode Program Function edit	162
Tabel D. 75 Kode Program Function update	162

Tabel D. 76 Kode Program Function Detail	162
Tabel F. 1 Pengujian Black Box Fitur Login	172
Tabel F. 2 Pengujian Black Box Fitur Pengelolaan Data Pengguna	172
Tabel F. 3 Pengujian Black Box Fitur Melihat Data Pengguna	173
Tabel F. 4 Pengujian Black Box Fitur Pengelolaan Data Outlet	173
Tabel F. 5 Pengujian Black Box Melihat Data Outlet	174
Tabel F. 6 Pengujian Black Box Melihat Data Hasil Peramalan	174
Tabel F. 7 Pengujian Black Box Pengelolaan Data Pendistribusian Produk	174
Tabel F. 8 Pengujian Black Box Pengelolaan Data Penjualan Produk	175
Tabel F. 9 Pengujian Black Box Melihat Data Hasil Penjualan dan Pendistribusian Produk	176
Tabel F. 10 Fitur Input Transaksi Penjualan	176
Tabel F. 11 Melihat Rekap Hasil Transaksi Penjualan	177
Tabel E. 1 Test case Function Submit (Outlet)	163
Tabel E. 2 Test Case Function Index (Pengiriman)	164
Tabel E. 3 Test Case Function Update (Pengiriman)	165
Tabel E. 4 Test Case Function Submit (Pengguna)	166
Tabel E. 5 Test Case Function Edit (Pengguna)	168
Tabel E. 6 Test Case Function Update (Pengguna)	169
Tabel E. 7 Test Case Function Index (Detail)	170
Tabel E. 8 Test Case Function Update (Detail)	170
Tabel E. 9 Test Case Function Beli (Detail)	171

DAFTAR GAMBAR

Gambar 3.1 Diagram Penelitian.....	12
Gambar 3.2 Tahapan Metode Agile.....	14
Gambar 3.3 Flowchart Sistem.....	17
Gambar 4.1 Business Process.....	21
Gambar 4.2 Use Case Diagram.....	22
Gambar 4.3 Sequence Pengelolaan Data Hasil Peramalan.....	31
Gambar 4.4 Activity Diagram Pengelolaan Data Hasil Peramalan.....	33
Gambar 4.5 Class Diagram.....	35
Gambar 4.6 Entity Relationship Diagram.....	36
Gambar 4.7 Function tambah.....	40
Gambar 5.1 Halaman Login.....	45
Gambar 5.2 Halaman Awal (Dashboard) Admin.....	46
Gambar 5.3 Halaman Data Pengguna (Admin).....	47
Gambar 5.4 Halaman Edit Data Pengguna (Admin).....	48
Gambar 5.5 Halaman Awal (Dashboard) Marketing.....	49
Gambar 5.6 Halaman Data Peramalan (Marketing).....	50
Gambar 5.7 Halaman Data Pengiriman (Marketing).....	51
Gambar 5.8 Halaman Input Data Pengiriman (Marketing).....	52
Gambar 5.9 Halaman Detail Data Pengiriman (Marketing).....	53
Gambar 5.10 Halaman Data Outlet (Marketing).....	54
Gambar 5.11 Halaman Edit Data Outlet (Marketing).....	55
Gambar 5.12 Halaman Utama (Dashboard) Eksekutif.....	56
Gambar 5.13 Halaman Data Pengguna (Eksekutif).....	56
Gambar 5.14 Halaman Data Peramalan (Eksekutif).....	57
Gambar 5.15 Halaman Data Pengiriman (Eksekutif).....	58
Gambar 5.16 Halaman Data Outlet (Eksekutif).....	58
Gambar 5.17 Halaman Utama (Dashboard) Admin Outlet.....	59
Gambar 5.18 Halaman Data Penjualan (Admin Outlet).....	60
Gambar 5.19 Halaman Input Data Penjualan (Admin Outlet).....	61
Gambar 5.20 Halaman Transaksi Penjualan (Admin Outlet).....	62
Gambar 5.21 Halaman Detail Transaksi Penjualan (Admin Outlet).....	63
Gambar 5.22 Output Hasil Peramalan Oleh Sistem.....	84
Gambar B. 1 Sequence Diagram Login Admin.....	116
Gambar B. 2 Sequence Diagram Login Admin.....	116
Gambar B. 3 Sequence Diagram Login Eksekutif.....	116
Gambar B. 4 Sequence Diagram Admin Outlet A.....	117
Gambar B. 5 Sequence Diagram Admin Outlet B.....	117
Gambar B. 6 Sequence Diagram Admin Outlet C.....	117
Gambar B. 7 Sequence Diagram Admin Outlet D.....	118
Gambar B. 8 Sequence Diagram Admin Outlet E.....	118

Gambar B. 9 Sequence Diagram Admin Outlet F.....	118
Gambar B. 10 Sequence Diagram Pengelolaan Data Pengguna	119
Gambar B. 11 Sequence Diagram Melihat Data Pengguna	119
Gambar B. 12 Sequence Diagram Pengelolaan Data Outlet.....	120
Gambar B. 13 Sequence Diagram Melihat Data Outlet.....	120
Gambar B. 14 Sequence Diagram Melihat Data Hasil Peramalan.....	121
Gambar B. 15 Sequence Diagram Pengelolaan Data Pendistribusian Produk....	122
Gambar B. 16 Sequence Diagram Melihat Detail Hasil Penjualan dan Sisa Produk	123
Gambar B. 17 Sequence Diagram Melihat Data Hasil Penjualan dan Pendistribusian Produk	124
Gambar B. 18 Sequence Diagram Input Transaksi Penjualan	124
Gambar B. 19 Sequence Diagram Melihat Rekap Hasil Transaksi Penjualan ...	125
Gambar C. 1 Activity Diagram Login Admin	126
Gambar C. 2 Activity Diagram Login Marketing.....	127
Gambar C. 3 Activity Diagram Login Eksekutif	127
Gambar C. 4 Activity Diagram Login Admin Outlet A.....	128
Gambar C. 5 Activity Diagram Login Admin Outlet B.....	128
Gambar C. 6 Activity Diagram Login Admin Outlet C.....	129
Gambar C. 7 Activity Diagram Login Admin Outlet C.....	129
Gambar C. 8 Activity Diagram Login Admin Outlet E.....	130
Gambar C. 9 Activity Diagram Login Admin Outlet F	130
Gambar C. 10 Activity Diagram Pengelolaan Data Pengguna	131
Gambar C. 11 Activity Diagram Melihat Data Pengguna	132
Gambar C. 12 Activity Diagram Pengelolaan Data Outlet	133
Gambar C. 13 Activity Diagram Melihat Data Outlet	134
Gambar C. 14 Activity Diagram Melihat Data Hasil Peramalan.....	135
Gambar C. 15 Activity Diagram Pengelolaan Data Pendistribusian Produk.....	136
Gambar C. 16 Activity Diagram Melihat Detail Hasil Penjualan dan Sisa Produk	137
Gambar C. 17 Activity Diagram Melihat Data Hasil Penjualan dan Pendistribusian Produk	138
Gambar C. 18 Activity Diagram Input Transaksi Penjualan	139
Gambar C. 19 Activity Diagram Melihat Rekap Hasil Transaksi Penjualan.....	139
Gambar E. 1 Function Submit (outlet).....	163
Gambar E. 2 Function Index (pengiriman)	164
Gambar E. 3 Function update (pengiriman).....	165
Gambar E. 4 Function Submit (pengguna)	166
Gambar E. 5 Function Edit (pengguna)	168
Gambar E. 6 Function update (pengguna)	168

Gambar E. 7 Function Index (detail).....	169
Gambar E. 8 Function Update (detail).....	170
Gambar E. 9 Function Beli (Detail).....	171



BAB. 1 PENDAHULUAN

Bab ini merupakan tahap awal dari penulisan tugas akhir ini. Bab ini menjelaskan mengenai latar belakang penulisan, perumusan masalah, tujuan, batasan masalah, dan sistematika penulisan.

1.1 Latar Belakang

Perkembangan *home industry* saat ini telah mengalami kemajuan yang sangat pesat. Hal ini ditunjukkan dengan munculnya banyak persaingan bisnis yang bergerak pada bidang yang sama. Melihat kondisi tersebut, maka perlu adanya strategi untuk menunjang kemajuan bisnis yang dijalankan. Dalam mempertahankan eksistensinya, pemilik *home industry* berusaha untuk selalu meningkatkan kualitas produk dan kualitas pelayanan yang meliputi pendistribusian produk ke beberapa *reseller* hingga mengendalikan persediaan produk. Upaya tersebut harus dilakukan agar kepuasan pelanggan selalu terjaga, sehingga *home industry* tersebut dapat mempertahankan keuntungan yang diperoleh.

Pusat Oleh-oleh Purnama Jati merupakan salah satu *home industry* di Kabupaten Jember yang bergerak dibidang makanan. Produk terkenal yang dihasilkan dari Pusat Oleh-oleh Purnama Jati yaitu Proll Tape. Permintaan konsumen terhadap produk tersebut selalu mengalami peningkatan disetiap harinya. Untuk memenuhi permintaan konsumen, Pusat Oleh-oleh Purnama Jati telah mendistribusikan produk produknya kepada 6 *outlet* pusat oleh-oleh yang tersebar di Kabupaten Jember. Jumlah produk yang didistribusikan kepada 6 *outlet* setiap harinya disama ratakan tanpa memperhitungkan *outlet* mana yang memiliki jumlah konsumen lebih banyak diantara yang lain dan *stock* yang masih tersedia di masing masing *outlet* tersebut. Hal ini tentu menimbulkan permasalahan pada proses distribusi di Pusat Oleh-oleh Purnama Jati. Permasalahan tersebut yaitu tidak sesuai jumlah produk yang didistribusikan dengan kebutuhan dari masing-masing *outlet*, sehingga mengakibatkan proses pengiriman produk menjadi kurang efektif. Hal ini dapat dilihat dari proses pengiriman yang dilakukan lebih dari satu

kali dalam satu hari. Sehingga, Pusat Oleh-oleh Purnama Jati harus mengeluarkan biaya lebih untuk pengiriman produk secara berulang hampir disetiap harinya. Untuk mengatasi permasalahan tersebut, perlu dibangun suatu sistem peramalan yang berguna dalam manajemen sistem distribusi produk.

Berdasarkan pemaparan diatas maka penulis merasa perlu untuk membangun suatu sistem perencanaan dan peramalan distribusi produk berdasarkan jumlah permintaan produk pada masing-masing *outlet*. Dalam hal ini penulis memilih menggunakan metode *Weight Moving Average* (WMA) untuk pengelolaan data peramalan pada sistem yang akan diterapkan pada Pusat Oleh-oleh Purnama Jati di Kabupaten Jember, dengan harapan sistem yang dibangun nantinya dapat meramalkan banyaknya produk yang harus didistribusikan agar semua permintaan konsumen dapat terpenuhi serta proses pendistribusian menjadi lebih sesuai dengan kebutuhan di masing-masing *outlet*.

Metode *Weight Moving Average* (WMA) dipilih karena pada dasarnya metode ini sesuai apabila diterapkan untuk mengolah data yang bersifat *time-series* atau data yang berubah dari waktu ke waktu. Sehingga metode *Weight Moving Average* (WMA) ini dirasa sesuai apabila diterapkan dalam pengolahan data permintaan konsumen pada Pusat Oleh-oleh Purnama Jati karena pola data yang dimiliki selalu berubah disetiap waktunya. Metode *Weight Moving Average* (WMA) juga cocok digunakan untuk mengolah data yang perubahannya tidak terlalu cepat, hal ini sesuai dengan pola data permintaan pada Pusat Oleh-oleh Purnama Jati. Selain itu karena data permintaan pada Pusat Oleh-oleh Purnama Jati berupa data harian, maka metode *Weight Moving Average* (WMA) ini dipilih karena bisa digunakan untuk peramalan dalam jangka pendek Sehingga hasil pengolahan data diharapkan mampu untuk meramalkan jumlah produk yang harus didistribusikan dengan periode harian. Dengan begitu proses pemenuhan kebutuhan konsumen menjadi lebih optimal serta tidak ada lagi proses pengiriman yang terjadi secara berulang karena kurangnya ketelitian dalam memprediksi permintaan konsumen.

1.2 Perumusan Masalah

Berdasarkan uraian diatas, permasalahan yang muncul adalah:

1. Bagaimana penerapan metode peramalan data terhadap tingkat kesesuaian jumlah barang yang didistribusikan oleh Pusat Oleh-oleh Purnama Jati menggunakan metode *Weight Moving Average* (WMA)?
2. Bagaimana merancang dan membangun sistem untuk peramalan distribusi produk berdasarkan jumlah permintaan dengan menggunakan metode *Weight Moving Average* (WMA)?

1.3 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

1. Mengetahui penerapan metode peramalan data terhadap tingkat kesesuaian jumlah barang yang didistribusikan oleh Pusat Oleh-oleh Purnama Jati menggunakan metode *Weight Moving Average* (WMA).
2. Merancang dan membangun sistem untuk peramalan distribusi produk berdasarkan jumlah permintaan dengan menggunakan metode *Weight Moving Average* (WMA).

1.4 Batasan Masalah

1. Sistem ini menggunakan metode untuk pengolahan data yang bersifat *time-series* yaitu metode *Weight Moving Average* (WMA).
2. Objek penelitian fokus pada salah satu produk dari Pusat Oleh-oleh Purnama Jati yaitu Proll Tape.
3. Penelitian ini hanya difokuskan pada proses peramalan jumlah produk yang akan di distribusikan dari kantor pusat kepada 6 *outlet* yang telah bekerja sama dengan Pusat Oleh-oleh Purnama Jati.
4. Tidak meliputi arus keuangan dan pembayaran

1.5 Sistematika Penulisan

Sistematika dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Pendahuluan

Bab ini menjelaskan mengenai latar belakang, perumusan masalah, tujuan, batasan masalah, dan sistematika penulisan.

2. Tinjauan Pustaka

Bab ini berisi tentang penelitian terdahulu, teori serta materi materi yang digunakan untuk menunjang penelitian ini.

3. Metodologi Penelitian

Bab ini memaparkan tentang metode yang digunakan dalam penelitian ini yang meliputi tahap pengumpulan data dan analisis data, model perancangan sistem, dan gambaran umum sistem yang akan dibangun.

4. Analisis dan Perancangan Sistem

Bab ini menguraikan tentang analisis dan perancangan sistem yang dibangun.

5. Hasil dan Pembahasan

Bab ini menjelaskan mengenai hasil dan pembahasan dari penelitian yang telah dilakukan.

6. Penutup

Bab ini terdiri dari kesimpulan dari penelitian yang telah dilakukan serta saran yang diberikan untuk dijadikan acuan pada penelitian selanjutnya.

BAB. 2 TINJAUAN PUSTAKA

Bab ini berisi pembahasan mengenai tinjauan pustaka, definisi, serta penjelasan mengenai teori-teori yang berhubungan dengan permasalahan yang akan dibahas dalam penelitian.

2.1 Penelitian Terdahulu

Metode *Weight Moving Average* (WMA) telah diterapkan pada Sistem Peramalan Persediaan Barang di Toko The Kids 24 dengan menggunakan metode *Weight Moving Average* (WMA). Penelitian yang dilakukan oleh Shinta Siti Sundari, Susanto, dan Wivia Revianti (2015), menjelaskan bahwa sistem pencatatan hasil penjualan pada toko The Kids 24 masih menggunakan sistem yang tradisional yaitu dengan melakukan pencatatan pada buku kas setiap harinya oleh bagian keuangan lalu akan dilaporkan diakhir bulan kepada pemilik toko. Akhirnya untuk menentukan berapa jumlah produk yang harus disediakan untuk memenuhi kebutuhan konsumen pada bulan selanjutnya, pemilik hanya menggunakan perkiraan, sehingga tidak sesuai dengan jumlah kebutuhan konsumen.

Pada penelitian ini, penulis menggunakan metode *Weight Moving Average* (WMA) dalam meramalkan jumlah produk yang harus disediakan, sehingga untuk memenuhi kebutuhan konsumen dan menghindari kerugian maka perlu dibangun aplikasi yang mampu membantu pemilik toko untuk menentukan berapa jumlah yang harus disediakan untuk masing-masing produk. Berdasarkan hasil penelitian yang telah dilakukan oleh Shinta Siti Sundari, Susanto, dan Wivia Revianti dikemukakan bahwa peramalan jumlah produk yang harus disediakan dengan menggunakan metode *Weight Moving Average* (WMA) dengan pengambilan data dalam waktu 3 bulan kebelakang menghasilkan hasil yang lebih baik. Hal ini dapat dilihat dari keuntungan lebih yang didapat oleh toko setelah pemilik beralih menggunakan sistem ini.

2.2 Proll Tape

Proll Tape merupakan salah satu makanan khas Kabupaten Jember yang memiliki tekstur semi basah serta memiliki rasa yang manis. Proll dibuat dengan menggunakan bahan dasar tape singkong, dimana singkong itu sendiri merupakan tanaman yang memiliki kandungan gizi lengkap. Menurut Muhammad Asnawi et al (2013) rasa yang ada pada tape disebabkan karena adanya aktifitas dari mikroorganisme yang terkandung dalam ragi. Dalam proses pembuatan proll tape, tape singkong yang digunakan yaitu tape singkong dengan kematangan medium atau dalam arti proses fermentasi yang terjadi masih berlangsung sekitar 50%. Hal ini dikarenakan, apabila kondisi tape yang digunakan terlalu matang maka rasa tape sudah berubah menjadi lebih masam, sehingga akan merubah cita rasa proll tape yang dihasilkan.

2.3 Peramalan Permintaan (*Forecasting Demand*)

Menurut Tri Bowo Atmojo, et al (2013) Peramalan permintaan (*Forecasting demand*) merupakan salah satu cara yang biasa digunakan untuk mengoptimalkan kinerja dari *Supply Chain Management*. Ada banyak keuntungan yang diperoleh dengan adanya kegiatan peramalan diantaranya perusahaan mampu menghemat biaya penyimpanan barang, perusahaan mampu mengendalikan persediaan barang, serta perencanaan pengiriman barang menjadi lebih optimal. Proses peramalan yang dilakukan pada umumnya berdasarkan data yang terdapat pada masa lampau yang dianalisis dengan menggunakan metode atau cara cara tertentu. Baik tidaknya suatu peramalan yang dibuat selain ditentukan oleh metode yang digunakan, juga ditentukan oleh baik tidaknya informasi yang digunakan. Apabila informasi yang digunakan tidak dapat meyakinkan maka hasil peramalan juga akan sulit untuk dipercaya ketepatannya (Miftahatul Hakimah et al, 2015). Peramalan dilakukan dengan tujuan agar peramalan yang kita buat mampu meminimumkan pengaruh ketidakpastian terhadap perusahaan atau dengan kata lain dapat meminimumkan kesalahan peramalan (*forecast error*) yang biasanya diukur dengan menghitung nilai nilai *mean absolute deviation* (MAD), *mean squared error* (MSE) dan *mean absolute percentage error* (MAPE).

2.4 Distribusi Produk

Distribusi produk merupakan usaha yang dilakukan produsen untuk memasarkan produk produknya dengan tujuan untuk memperlancar dan mempermudah penyampaian produk kepada konsumen. Menurut Suseno Budi Prasetyo (2008) Dibutuhkan suatu proses yang efisien untuk menjamin produk yang didistribusikan sampai ke tangan konsumen secara luas dengan biaya yang minimal. Bila perusahaan memiliki sistem pendistribusian yang efisien, maka perusahaan akan dapat menguasai pasar.

2.5 *Weight Moving Average* (WMA)

Weight Moving Average (WMA) merupakan metode *moving average* atau rata-rata bergerak yang memiliki bobot. Pada *Weight Moving Average* (WMA) terdapat pemberian bobot yang digunakan pada setiap perubahan harga. Nilai bobot yang diberikan ini bisa bernilai berapapun tetapi dengan catatan nilai bobot untuk harga terbaru harus lebih besar daripada nilai bobot pada harga sebelumnya. Menurut Prof. Deepak Kapgate (2014) metode *Weight Moving Average* (WMA) merupakan metode yang cocok digunakan pada data yang bersifat *time-series*, yaitu data yang berubah dari waktu ke waktu. Secara matematis *Weight Moving Average* (WMA) dapat dituliskan seperti pada persamaan 1 dibawah ini :

$$WMA = \frac{\sum(\text{data} \times \text{bobot})}{\sum \text{bobot}} \dots\dots\dots [1]$$

Keterangan :

Data = Nilai sebelumnya

Bobot = Penilaian sesuai dengan panjang periode

2.6 Nilai Ketepatan Peramalan

Menurut Kristien Margi S (2015) ketepatan hasil peramalan adalah yang penting dalam sistem peramalan, yaitu bagaimana mengukur kesesuaian antara data yang sudah ada dengan data peramalan. Ada tiga perhitungan yang biasa digunakan untuk menghitung kesalahan peramalan total diantaranya, *mean absolute deviation* (MAD), *mean squared error* (MSE) dan *mean absolute percentage error* (MAPE) yang secara matematis dapat dituliskan seperti pada persamaan 2, 3 dan 4 dibawah ini:

$$MAD = \frac{\sum |X_t - S_t|}{n} \dots\dots\dots [2]$$

$$MSE = \frac{\sum |X_t - S_t|^2}{n} \dots\dots\dots [3]$$

$$MAPE = \frac{\sum |A_t - F_t|}{A_t} \dots\dots\dots [4]$$

2.7 Perhitungan Metode *Weight Moving Average* (WMA)

Untuk mengetahui perhitungan peramalan dengan menggunakan metode *Weight Moving Average* (WMA) maka digunakan rumus seperti yang dituliskan pada persamaan 5 berikut ini:

$$WMA = \frac{\sum (\text{data} \times \text{bobot})}{\sum \text{bobot}} \dots\dots\dots [5]$$

Contoh perhitungan dengan menggunakan 3 bobot yang diambil dari data penjualan produk X 3 bulan sebelumnya

$$WMA = \frac{((25 \times 3) + (23 \times 2) + (30 \times 1))}{(3 + 2 + 1)} = 25,16 = 25$$

Contoh perhitungan dengan menggunakan 5 bobot yang diambil dari data penjualan produk X 5 bulan sebelumnya

$$WMA = \frac{((10 \times 5) + (18 \times 4) + (25 \times 3) + (23 \times 2) + (30 \times 1))}{(3 + 2 + 1)} = 18.2 = 18$$

Tabel 2.1 merupakan hasil peramalan dengan metode *Weight Moving Average* (WMA) 3 dan 5 bobot.

Tabel 2.1 Hasil Peramalan Metode *Weight Moving Average* dengan 3 dan 5 bobot

No.	Bulan	Data Penjualan	WMA dengan 3 bobot	WMA dengan 5 bobot
1.	Januari	30	-	-
2.	Februari	23	-	-
3.	Maret	25	-	-
4.	April	18	25	-
5.	Mei	10	21	-
6.	Juni	16	15	18
7.	Juli	30	14	16
8.	Agustus	42	22	20
9.	September	34	34	28
10.	Oktober	28	36	31
11.	November	20	32	32
12.	Desember	23	25	28

2.8 Contoh Penentuan Kesalahan Peramalan

Tabel 2.2 berikut ini adalah hasil perhitungan *mean absolute deviation* (MAD), *mean squared error* (MSE) dan *mean absolute percentage error* (MAPE) dengan menggunakan 3 bobot

Tabel 2.2 Hasil Perhitungan *MAD*, *MSE*, *MAPE* dengan 3 bobot

No	Bulan	Data penjualan	WMA dengan 3 bobot				
			Data peramalan	Error	MAD	MSE	MAPE
1.	Januari	30	-	-	-	-	-
2.	Februari	23	-	-	-	-	-
3.	Maret	25	-	-	-	-	-
4.	April	18	25	-7	7	49	0.39
5.	Mei	10	21	-11	11	121	1.1
6.	Juni	16	15	1	7	1	0.062
7.	Juli	30	14	16	16	256	0.53
8.	Agustus	42	22	20	20	400	0.47
9.	September	34	34	0	0	0	0
10.	Oktober	28	36	-8	8	64	0.28

11.	November	20	32	-12	12	144	0.6
12.	Desember	23	25	-2	2	4	0.08
JUMLAH				-3	77	1039	3.53
RATA RATA				-0.33	8.56	115.44	39%

Tabel 2.3 berikut ini adalah hasil perhitungan *mean absolute deviation* (MAD), *mean squared error* (MSE) dan *mean absolute percentage error* (MAPE) dengan menggunakan 5 bobot

Tabel 2.3 Hasil Perhitungan *MAD*, *MSE*, *MAPE* dengan 5 bobot

No	Bulan	Data penjualan	WMA dengan 5 bobot				
			Data peramalan	Error	MAD	MSE	MAPE
1.	Januari	15	-	-	-	-	-
2.	Februari	18	-	-	-	-	-
3.	Maret	10	-	-	-	-	-
4.	April	7	-	-	-	-	-
5.	Mei	10	-	-	-	-	-
6.	Juni	28	18	-2	2	4	0.25
7.	Juli	26	16	14	14	196	0.23
8.	Agustus	12	20	22	22	484	1.16
9.	September	16	28	6	6	36	0.37
10.	Oktober	20	31	-3	3	9	1
11.	November	10	32	-12	12	144	0.2
12.	Desember	12	28	-5	5	25	0.05
JUMLAH				20	64	898	2.21
RATA RATA				2.85	9.14	128.28	31%

BAB. 3 METODOLOGI PENELITIAN

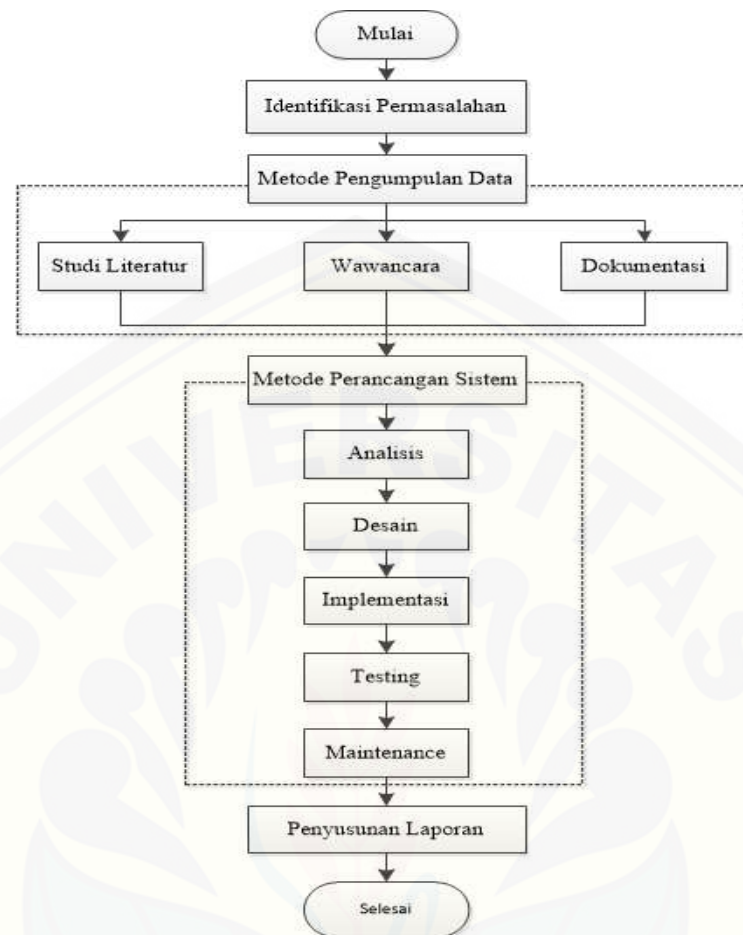
Bab ini menjelaskan mengenai tempat dan waktu penelitian, tahap penelitian, proses pengumpulan data, serta metode yang digunakan pada penelitian ini untuk menyelesaikan pembuatan Sistem Perencanaan dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan dengan Menggunakan Metode *Weight Moving Average* (WMA)

3.1 Tempat dan Waktu Penelitian

Penelitian dilakukan dengan menggunakan data dari UD Purnama Jati. Waktu penelitian yaitu dilaksanakan selama 8 bulan yaitu September 2016 – April 2017.

3.2 Tahap Penelitian

Penelitian ini akan dilakukan dalam beberapa tahapan diantaranya tahap pengumpulan data, tahap analisis, dan tahap pengembangan sistem. Tahapan penelitian ini digunakan untuk mencapai tujuan penelitian yaitu membantu dalam proses peramalan distribusi produk. Tahapan penelitian ini dapat digambarkan dalam bentuk diagram alir seperti pada gambar 3.1.



Gambar 3.1 Diagram Penelitian

(Sumber: Hasil Analisis, 2016)

3.3 Tahap Pengumpulan Data

Adapun tahap tahap pengumpulan data pada penelitian ini dibedakan menjadi tiga tahapan, yaitu tahap studi literature, tahap wawancara, dan tahap dokumentasi

3.3.1 Studi Literatur

Tahap studi literatur merupakan tahap pengumpulan data yang dilakukan oleh peneliti, dimana penelitian yang akan dilakukan ditunjang dengan teori teori terkait yang diperoleh dari beberapa sumber seperti buku, jurnal penelitian maupun

dari informasi yang telah tersedia. Tidak hanya itu penelitian yang akan dilakukan juga mengacu pada referensi yang didasarkan pada penelitian terdahulu.

3.3.2 Tahap Wawancara

Tahap wawancara merupakan tahap pengumpulan data yang dilakukan dengan melakukan komunikasi secara langsung dengan narasumber terkait untuk memperoleh data-data yang dibutuhkan untuk penelitian. Data-data yang dibutuhkan dalam wawancara ini berupa data data yang tidak didokumentasikan seperti data keterangan mengenai alur pendistribusian produk kepada 6 *outlet* yang telah bekerja sama.

Pada penelitian ini penulis melakukan wawancara secara langsung kepada Ibu Dina selaku marketing pada Pusat Oleh-oleh Purnama Jati. Pertanyaan yang diajukan meliputi bagaimana prosedur pencacatan, penafsiran serta alur pendistribusian produk.

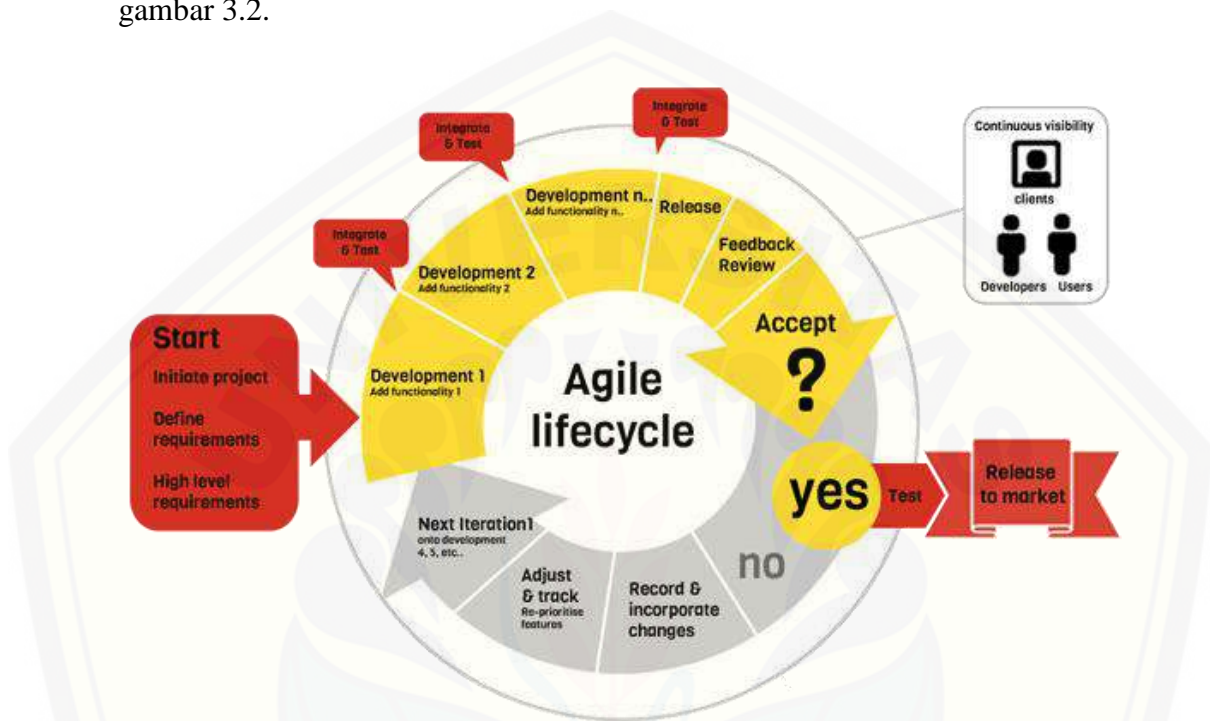
3.3.3 Tahap Dokumentasi

Tahap Dokumentasi merupakan tahap pengumpulan data yang dilakukan sama seperti tahap wawancara hanya yang membedakan adalah data yang diperoleh. Jika pada tahap wawancara data yang diperoleh berupa data yang tidak tertulis, maka pada tahap dokumentasi ini data yang diperoleh berupa data yang telah didokumentasikan oleh pihak terkait yaitu Pusat Oleh-oleh Purnama Jati. Adapun data yang diperoleh pada tahap ini yaitu data *Outlet*, dan data hasil penjualan pada 6 *Outlet* rekanan yang telah bekerja sama dengan Pusat Oleh-oleh Purnama Jati.

3.4 Tahap Perancangan Sistem

Perancangan sistem dibangun berdasarkan analisis dan pengumpulan data untuk membangun sistem yang sesuai dengan kebutuhan pada Pusat Oleh-oleh Purnama Jati. Metode yang digunakan dalam perancangan sistem ini yaitu metode *Agile*. Metode *Agile* merupakan suatu metodologi pengembangan perangkat lunak, dimana penerapan metode ini lebih *flexible* atau dalam arti lebih tanggap terhadap

perubahan. Menurut Sheetal sharma, et al (2012) proses kerja dari metode *Agile* ini merupakan proses berulang dimana perubahan dapat dibuat sesuai dengan kepuasan pelanggan, atau dalam arti fitur fitur baru dapat ditambahkan dengan mudah dengan menggunakan beberapa iterasi. Alur perancangan model *Agile* dapat dilihat pada gambar 3.2.



Gambar 3.2 Tahapan Metode *Agile*

(Sumber : AbuZaid, 2015)

3.4.1 Analisis Kebutuhan

Analisis kebutuhan adalah proses perancangan perangkat lunak yang dimulai dari proses merumuskan solusi dari setiap permasalahan yang muncul. Tahapan yang dilakukan dalam proses analisis kebutuhan meliputi, tahap pengumpulan data, analisis data, serta menentukan kebutuhan fungsional dan non fungsional dari sistem yang akan dibangun.

3.4.2 Desain Sistem

Pembuatan desain sistem menggunakan *Unified Modeling Language (UML)* yang dirancang menggunakan konsep *Object Oriented Programming (OOP)*, Berikut pemodelan UML yang digunakan antara lain:

1. *Business Process* digunakan untuk mendefinisikan aktifitas dan proses yang ada.
2. *Use Case Diagram* digunakan untuk mendefinisikan dengan tepat kebutuhan fungsional yang harus disediakan oleh sistem.
3. *Scenario* digunakan untuk menjelaskan fitur fitur yang ada pada *use case diagram*.
4. *Activity Diagram* digunakan untuk mendefinisikan aktifitas didalam sistem yang dirancang.
5. *Sequence Diagram* digunakan untuk menggambarkan rangkaian pesan yang dikirim antar object juga interaksi yang terjadi didalamnya.
6. *Class Diagram* Digunakan untuk menggambarkan struktur statis class yang ada dalam sistem.
7. *Entity Relationship Diagram* digunakan untuk menunjukkan relasi antar object.

3.4.3 Implementasi

Implementasi perangkat lunak ini dilakukan dengan mengacu pada perancangan desain sistem yang telah dibuat sebelumnya. Implementasi perangkat lunak ini nantinya menggunakan bahasa pemrograman PHP yang akan digunakan untuk membangun sistem berbasis web dan manajemen basis data dengan menggunakan *DBMS MySQL*.

3.4.4 Tahap Pengujian

Tahap pengujian ini dilakukan ketika sistem yang dibangun sudah siap untuk digunakan oleh pengguna. Tujuan dari adanya tahap pengujian ini adalah untuk mengetahui sejauh mana data data yang telah diperoleh berhasil dikelola dan diimplemetasikan dengan baik menggunakan metode yang digunakan. Tahap

pengujian itu sendiri dilakukan untuk mencari kesalahan kesalahan yang mungkin terjadi ketika sistem digunakan oleh pengguna, sehingga sistem dapat diperbaiki dan disempurnakan sebelum sistem digunakan oleh pengguna. Menurut Refi Meisadri, et al (2013) ada dua tahap yang dilakukan dalam melakukan pengujian sistem, yaitu pengujian alpha (fungsional) dan pengujian beta. Pengujian alpha berfokus pada persyaratan fungsional perangkat lunak. Pada tahap pengujian alpha terdapat dua teknik pengujian yang biasa digunakan untuk melakukan pengujian antara lain yaitu *Black Box testing* dan *White Box testing*.

Black box testing merupakan suatu metode yang digunakan untuk mencari kesalahan dan mendemonstrasikan fitur fitur yang ada pada sistem tersebut, sehingga dapat diketahui bahwa input diterima dengan benar dan output yang dihasilkan sesuai dengan yang diharapkan. Sedangkan *White Box testing* merupakan cara pengujian dengan melihat kedalam modul untuk meneliti kode kode program yang ada, kemudian menganalisis apakah ada kesalahan atau tidak. Jika ada modul yang menghasilkan output yang tidak sesuai dengan proses bisnis yang dilakukan, maka baris-baris program, variabel, dan parameter yang terlibat pada unit tersebut akan dicek satu persatu dan diperbaiki, kemudian di-compile ulang.

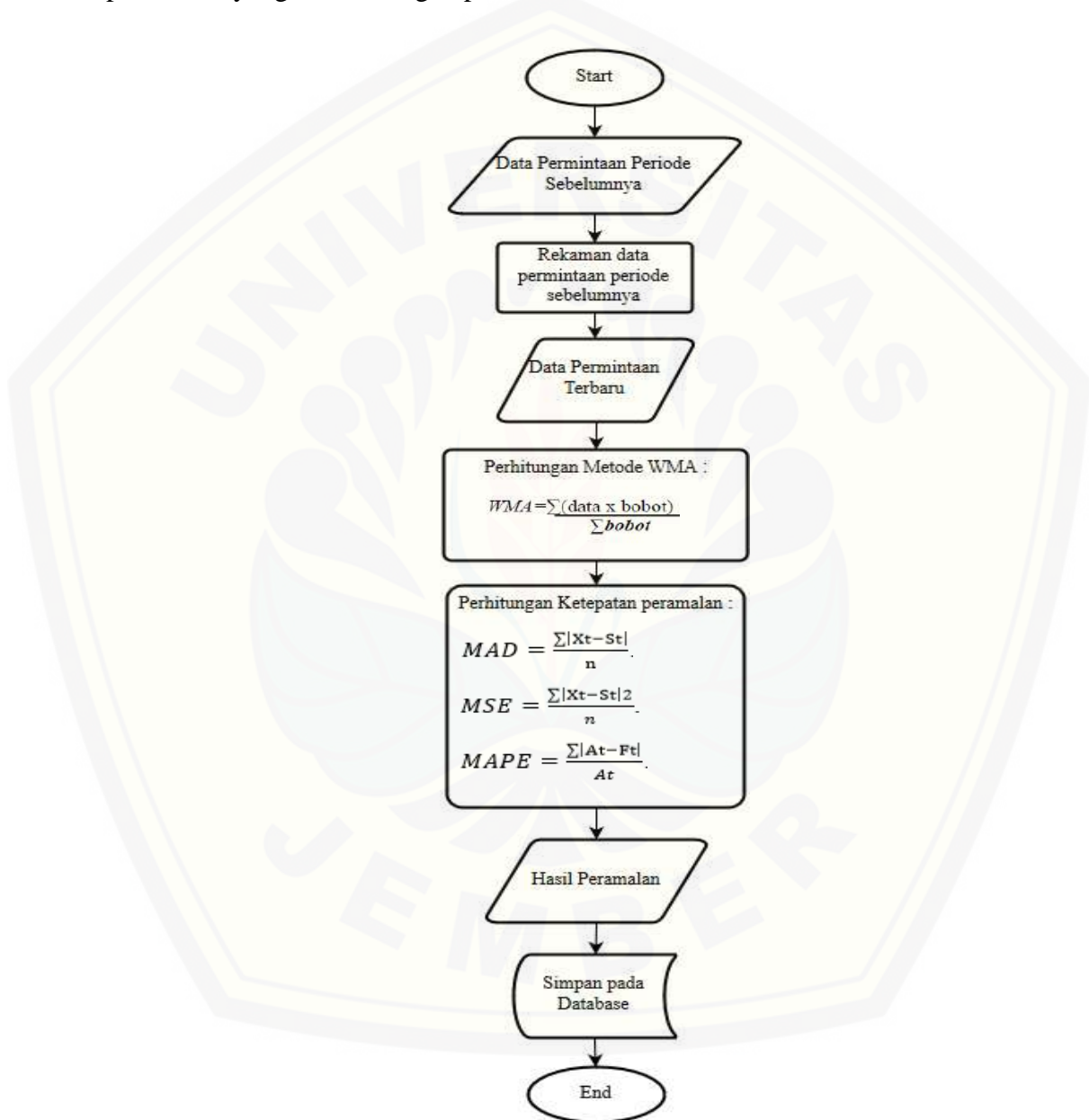
3.5 Perawatan (*Maintenance*)

Perawatan merupakan tahap terakhir dalam metode *Agile*. Sistem yang berhasil dibuat tersebut masih membutuhkan adanya pemeliharaan. Pemeliharaan yang dimaksud dalam hal ini adalah bertujuan untuk peningkatan kinerja dari sistem yang digunakan sebagai kebutuhan baru. Sehingga apabila nantinya user menemukan adanya *bug* pada sistem tersebut, *user* dapat melaporkan secara langsung kepada pihak *developer* untuk segera ditangani.

3.6 Gambaran Sistem

Sistem yang akan dibangun pada penelitian ini yaitu sistem perencanaan dan peramalan distribusi produk dengan menggunakan metode *Weight Moving Average* (WMA) pada Pusat Oleh-oleh Purnama Jati. Sistem ini nantinya digunakan untuk

meramalkan berapa jumlah produk yang harus didistribusikan kepada 6 *outlet* yang tersebar di Kabupaten Jember, agar produk yang didistribusikan tidak berlebihan dan tidak kekurangan. Metode *Weight Moving Average* (WMA) digunakan untuk mengolah data permintaan pada periode periode sebelumnya agar diperoleh hasil peramalan yang sesuai dengan permintaan konsumen.



Gambar 3.3 *Flowchart* Sistem

(Sumber : Hasil Analisis, 2016)

Detail dari gambar 3.3 mengenai *flowchart* sistem akan dijabarkan dalam beberapa tahapan sebagai berikut :

1. Melakukan penyimpanan data data penjualan pada periode sebelumnya untuk dijadikan acuan dalam melakukan peramalan penjualan pada periode selanjutnya.
2. Untuk menghitung hasil peramalan maka proses pertama yang dilakukan yaitu memasukkan data penjualan terbaru, kemudian data tersebut dihitung dengan menggunakan pembobotan yang disesuaikan dengan periode yang digunakan. Pada penelitian ini jumlah bobot yang digunakan mengacu pada jumlah periode yang akan diinputkan oleh pengguna, sehingga sistem akan secara otomatis menentukan bobot untuk masing masing data sejumlah periode yang telah diinputkan dengan catatan pemberian bobot paling besar yaitu pada periode terakhir. Penjelasan lebih detail dapat dilihat pada Bab 2 Kajian Teori.
3. Setelah hasil peramalan ditemukan, maka tahap selanjutnya yang dilakukan yaitu menghitung error dari hasil peramalan. Perhitungan error ini nantinya akan dijadikan acuan pengguna untuk menggunakan hasil peramalan tersebut. Semakin kecil nilai perhitungan error maka hasil peramalan semakin akurat.
4. Tahap terakhir yaitu menyimpan hasil peramalan pada database sebagai hasil rekapan peramalan.

BAB. 4 ANALISIS DAN PERANCANGAN SISTEM

Bab ini akan membahas mengenai analisis dan pengembangan sistem untuk mengimplementasikan metode *Weight Moving Average* (WMA) pada Sistem Perencanaan dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan. Proses perancangan sistem ini berdasarkan metode Agile yang dimulai dari analisis kebutuhan sistem, kemudian pembuatan desain sistem yang terdiri dari *Business Process*, *Usecase Diagram*, *Scenario*, *Sequence Diagram*, *Activity Diagram*, *Class Diagram*, dan *Entity Relationship Diagram* (ERD), penulisan kode program, serta pengujian sistem.

4.1 Analisis Kebutuhan

Tahap analisis kebutuhan sistem merupakan tahapan yang penting dalam pengembangan sebuah sistem. Kebutuhan sistem itu sendiri dibedakan menjadi dua yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional merupakan kebutuhan yang harus disediakan oleh sistem dan berkaitan dengan fungsi dari sistem. Sedangkan kebutuhan non-fungsional merupakan kebutuhan yang dibutuhkan sistem untuk mendukung aktivitas sistem sesuai dengan kebutuhan fungsional yang telah dibuat.

4.1.1 Kebutuhan Fungsional

Kebutuhan Fungsional dari Sistem Perencanaan dan Peramalan Distribusi Produk adalah sebagai berikut:

1. Sistem mampu mengelolah data pengguna yang meliputi *view*, *insert*, *edit*, *delete*.
2. Sistem mampu mengelolah data *outlet* yang meliputi *view*, *insert*, *edit*, *delete*.
3. Sistem mampu mengelolah data peramalan yang meliputi *view*, *insert*.
4. Sistem mampu mengelolah data penjualan pada masing masing *outlet* yang meliputi *view*, *edit*.
5. Sistem mampu mengelolah data pengiriman yang meliputi *view*, *edit*.

6. Sistem mampu melakukan transaksi penjualan, merekap, dan menyimpan hasil transaksi penjualan

4.1.2 Kebutuhan Non-Fungsional

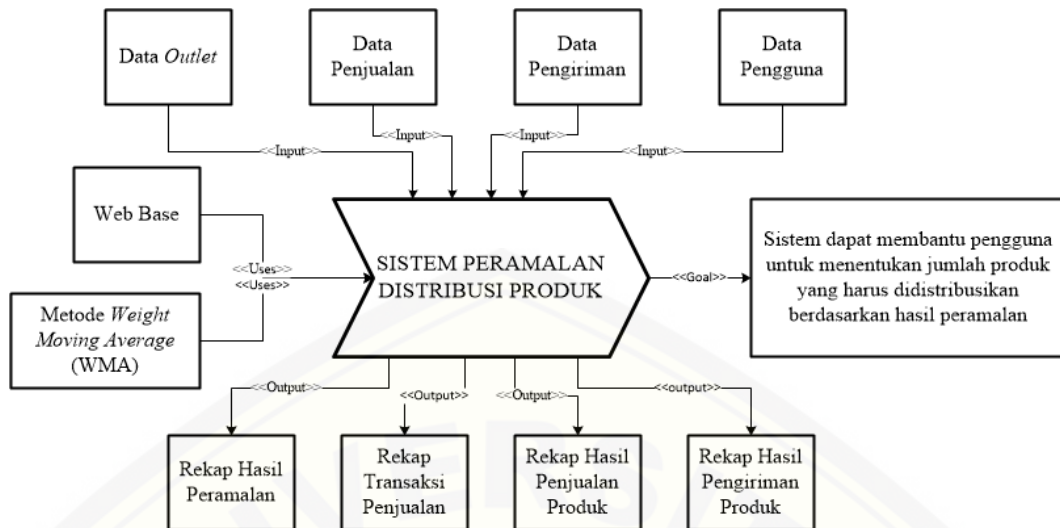
1. Sistem menggunakan *username* dan *password* untuk autentifikasi akses terhadap sistem
2. Tampilan sistem mudah dipahami oleh pengguna sehingga pengguna tidak akan mengalami kesulitan ketika menggunakan sistem.
3. Hanya admin yang memiliki hak akses untuk membuat akun pengguna baru, melakukan update data pengguna, dan menghapus data pengguna.

4.2 Desain Sistem

Desain sistem yang dibuat meliputi *bussines process*, *use case diagram*, *use case scenario*, *sequence diagram*, *activity diagram*, *class diagram*, dan *entity relationship diagram* (ERD).

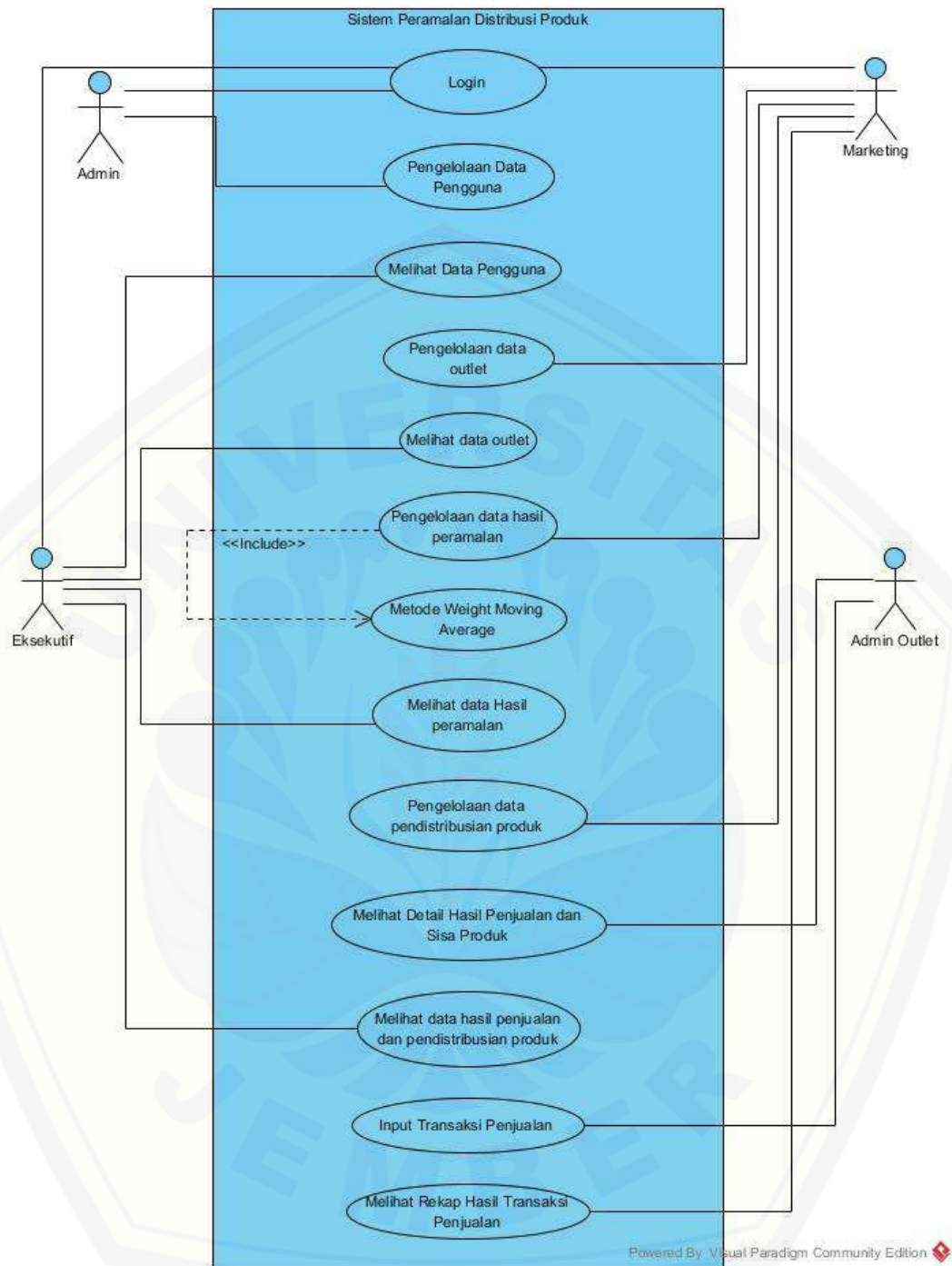
4.2.1 Business Process

Business Process merupakan suatu kumpulan dari aktivitas atau pekerjaan terstruktur yang saling berhubungan untuk menghasilkan suatu produk. Ada beberapa komponen yang terdapat dalam *business Process* diantaranya data yang menjadi masukan sistem (*input*), data masukan yang telah diolah sehingga menghasilkan data keluaran (*output*), media yang digunakan (*uses*), serta tujuan yang ingin dicapai (*goal*). *Business process* pada sistem ini dapat dilihat pada gambar 4.1

Gambar 4.1 *Business Process*

4.2.2 Use Case Diagram

Use Case Diagram merupakan gambaran fungsional dari sistem yang dapat menunjukkan fitur-fitur yang disediakan oleh sistem serta aktor yang dapat mengakses fitur tersebut pada "Sistem Perencanaan dan Peramalan Distribusi Produk Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus: Pusat Oleh-oleh Purnama Jati). Use Case Diagram pada sistem ini dapat dilihat pada Gambar 4.2.



Gambar 4.2 Use Case Diagram

Penjelasan tentang definisi aktor dan definisi *use case diagram* pada gambar 4.2 akan dijelaskan di bawah ini.

1. Definisi Aktor

Definisi aktor yaitu penjelasan mengenai aktor-aktor sebagai pengguna dari “Sistem Perencanaan dan Peramalan Distribusi Produk Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus: Pusat Oleh-oleh Purnama Jati) yang akan dibangun. Terdapat 4 Aktor seperti yang dijelaskan pada tabel 4.1

Tabel 4.1 Definisi Aktor

No	Aktor	Deskripsi
1.	Admin	Aktor Admin pada sistem ini memiliki hak akses untuk mengelolah data pengguna yang dapat mengakses sistem ini. Apabila nantinya ada pergantian pengguna maka admin dapat menghapus akun pengguna lama dan mengganti dengan akun pengguna yang baru.
2.	Marketing	Aktor marketing pada sistem ini memiliki hak akses untuk mengelolah data pendistribusian produk pada masing masing outlet, mengelolah data outlet, serta mengelolah data hasil peramalan.
3.	Eksekutif	Aktor Eksekutif pada sistem ini memiliki peranan untuk mengontrol/mengawasi pengelolaan data data yang meliputi data pengguna, data outlet, data penjualan, data peramalan, dan data pendistribusian produk
4.	Admin <i>Outlet</i>	Aktor Admin <i>Outlet</i> pada sistem ini memiliki perananan untuk mengelolah data penjualan untuk setiap <i>outlet</i> .

2. Definisi Use Case

Definisi use case yaitu penjelasan dari masing-masing use case atau fitur-fitur yang terdapat pada “Sistem Perencanaan dan Peramalan Distribusi Produk Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus: Pusat Oleh-oleh Purnama Jati) yang akan dibangun. Terdapat 10 use case seperti yang dijelaskan pada Tabel 4.2.

Tabel 4.2 Definisi *Use Case*

No.	<i>Use Case</i>	Deskripsi
1.	<i>Login</i>	Menggambarkan proses autentifikasi yang digunakan untuk masuk ke sistem.
2.	Pengelolaan Data Pengguna	Menggambarkan proses untuk menambah, mengedit, menghapus, serta melihat data pengguna.
3.	Melihat Data Pengguna	Menggambarkan proses untuk melihat data pengguna pada aktor aktor tertentu yang memiliki hak akses untuk melihat data pengguna.
4.	Pengelolaan Data <i>Outlet</i>	Menggambarkan proses untuk menambah, mengedit, menghapus, serta melihat data <i>outlet</i> .
5.	Melihat Data <i>Outlet</i>	Menggambarkan proses untuk melihat data <i>outlet</i> pada aktor-aktor tertentu yang memiliki hak akses untuk melihat data <i>outlet</i> .
6.	Pengelolaan Data Hasil Peramalan	Menggambarkan proses untuk menambah dan melihat data hasil peramalan produk

		untuk periode selanjutnya pada masing-masing <i>outlet</i> .
7.	Melihat Hasil Peramalan	Menggambarkan proses untuk melihat data hasil peramalan produk untuk periode selanjutnya pada aktor-aktor tertentu yang memiliki hak akses untuk melihat data hasil peramalan produk pada masing-masing <i>outlet</i> .
8.	Pengelolaan Data Pendistribusian Produk	Menggambarkan proses untuk mengupdate, dan melihat data pendistribusian produk pada masing-masing <i>outlet</i> .
9.	Melihat Detail Hasil Penjualan dan Sisa Produk	Menggambarkan proses untuk melihat data hasil penjualan beserta detail produk yang tersisa. Fitur ini dapat diakses oleh setiap admin pada masing-masing <i>outlet</i> .
10.	Melihat Data Hasil Penjualan dan Pendistribusian Produk	Menggambarkan proses untuk melihat data hasil penjualan dan pendistribusian produk pada aktor-aktor tertentu yang memiliki hak akses untuk melihat data hasil penjualan dan pendistribusian produk.
11.	Input Transaksi Penjualan	Menggambarkan proses untuk melakukan transaksi penjualan yang dilakukan oleh setiap admin pada masing-masing <i>outlet</i> .
12.	Melihat Rekap Hasil Transaksi Penjualan	Menggambarkan Proses untuk melihat rekap data hasil penjualan yang dilakukan oleh aktor-aktor tertentu yang memiliki

		hak akses untuk melihat rekap data hasil transaksi penjualan.
--	--	---

4.2.3 Use Case Skenario

Use Case Scenario digunakan untuk menjelaskan alur sistem sesuai dengan kebutuhan fungsional yang terdapat pada *use case diagram* seperti pada gambar 4.2.

1. Skenario *Use Case Login*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case login* dijelaskan pada lampiran A.

2. Skenario *Use Case Pengelolaan Data Pengguna*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* pengelolaan data pengguna dijelaskan pada lampiran A.

3. Skenario *Use Case Melihat Data Pengguna*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* melihat data pengguna dijelaskan pada lampiran A.

4. Skenario *Use Case Pengelolaan Data Outlet*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* pengelolaan data *outlet* dijelaskan pada lampiran A.

5. Skenario *Use Case Melihat Data Outlet*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* melihat data *outlet* dijelaskan pada lampiran A.

6. Skenario *Use Case Pengelolaan Data Hasil Peramalan*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* pengelolaan data hasil peramalan dijelaskan pada Tabel 4.3.

Tabel 4.3 Skenario *Use Case* Pengelolaan Data Peramalan Distribusi Produk

ID	USC_6
Nama Use Case	Pengelolaan Data Hasil Peramalan
Aktor	Marketing
Deskripsi Singkat	Marketing akan mengelolah data hasil peramalan pada masing masing <i>outlet</i>
PreKondisi	Marketing memilih submenu data peramalan pada menu penjualan
PostKondisi	Marketing berhasil melihat dan menghapus data hasil peramalan
<i>Flow Events</i>	
Skenario Normal : Menambah data penjualan	
Aktor	Reaksi Sistem
1. Klik menu penjualan	
2. Klik sub menu data peramalan	
	<p>3. Menampilkan halaman data peramalan yang meliputi</p> <ul style="list-style-type: none"> • Form cek hasil peramalan yang terdiri dari : <ul style="list-style-type: none"> - <i>Drop down</i> daftar jenis produk - Daftar <i>outlet</i> - <i>Input</i> jumlah periode - Tombol submit • <i>Dropdown</i> nama <i>outlet</i> • <i>Dropdown</i> jenis produk • Tombol tampilkan • Tabel peramalan yang digunakan untuk

	<p>menampilkan data hasil penjualan. Tabel peramalan terdiri dari</p> <ul style="list-style-type: none"> - No. - Tanggal - Jumlah Penjualan - Hasil peramalan - MAPE - Akurasi
<p>4. Mengisi form yang berisi</p> <ul style="list-style-type: none"> • Nama <i>outlet</i> • Jenis produk • Jumlah periode 	
5. Klik tombol submit	
	<p>6. Perhitungan hasil peramalan dengan metode <i>weight moving average</i> sebagai berikut :</p> $WMA = \frac{\sum(\text{data} \times \text{bobot})}{\sum \text{bobot}}$
	7. Menyimpan ke database
	<p>8. Menampilkan halaman data peramalan dengan jumlah penjualan tercantum “data belum diinputkan”, nilai MAPE tertulis 100%, Akurasi tercantum 0% dan data hasil peramalan sudah tercantum pada tabel</p>
Skenario alternative : Jumlah Penjualan belum diinputkan	
5. Klik tombol submit	
	6. Menampilkan span “Jumlah Penjualan Belum Diinputkan”

Skenario alternative : Harus diisi dengan angka	
5. Klik tombol submit	
	6. Menampilkan span “Please enter a number”

7. Skenario *Use Case* Melihat Data Hasil Peramalan.
Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* melihat data hasil peramalan dijelaskan pada lampiran A.
8. Skenario *Use Case* Pengelolaan Data Pendistribusian Produk.
Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* pengelolaan data pendistribusian produk dijelaskan pada lampiran A.
9. Skenario *Use Case* Melihat Detail Hasil Penjualan dan Sisa Produk
Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* melihat detail hasil penjualan dan sisa produk dijelaskan pada lampiran A.
10. Skenario *Use Case* Melihat Data Hasil Penjualan dan Pendistribusian Produk.
Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* melihat data hasil penjualan dan pendistribusian produk dijelaskan pada lampiran A.
11. Skenario *Use Case* Input Transaksi Penjualan.
Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* Input transaksi penjualan dijelaskan pada lampiran A.
12. Skenario *Use Case* Melihat Rekap Hasil Transaksi Penjualan.
Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif. Skenario *use case* melihat rekap hasil transaksi penjualan dijelaskan pada lampiran A.

4.2.4 *Sequence Diagram*

Sequence diagram adalah diagram yang digunakan untuk menggambarkan interaksi yang terjadi antar objek di dalam sistem yang disusun pada sebuah urutan dan rangkaian waktu pada Sistem Perencanaan dan Peramalan Distribusi Produk Berdasarkan Jumlah Permintaan dengan Menggunakan Metode *Weight Moving Average* (WMA) (Studi kasus: Pusat Oleh-oleh Purnama Jati).

1. *Sequence Diagram Login*

Penggambaran *sequence diagram login* digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

2. *Sequence Diagram Pengelolaan Data Pengguna*

Penggambaran *sequence diagram* pengelolaan data pengguna digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

3. *Sequence Diagram Melihat Data Pengguna*

Penggambaran *sequence diagram* melihat data pengguna digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

4. *Sequence Diagram Pengelolaan Data Outlet*

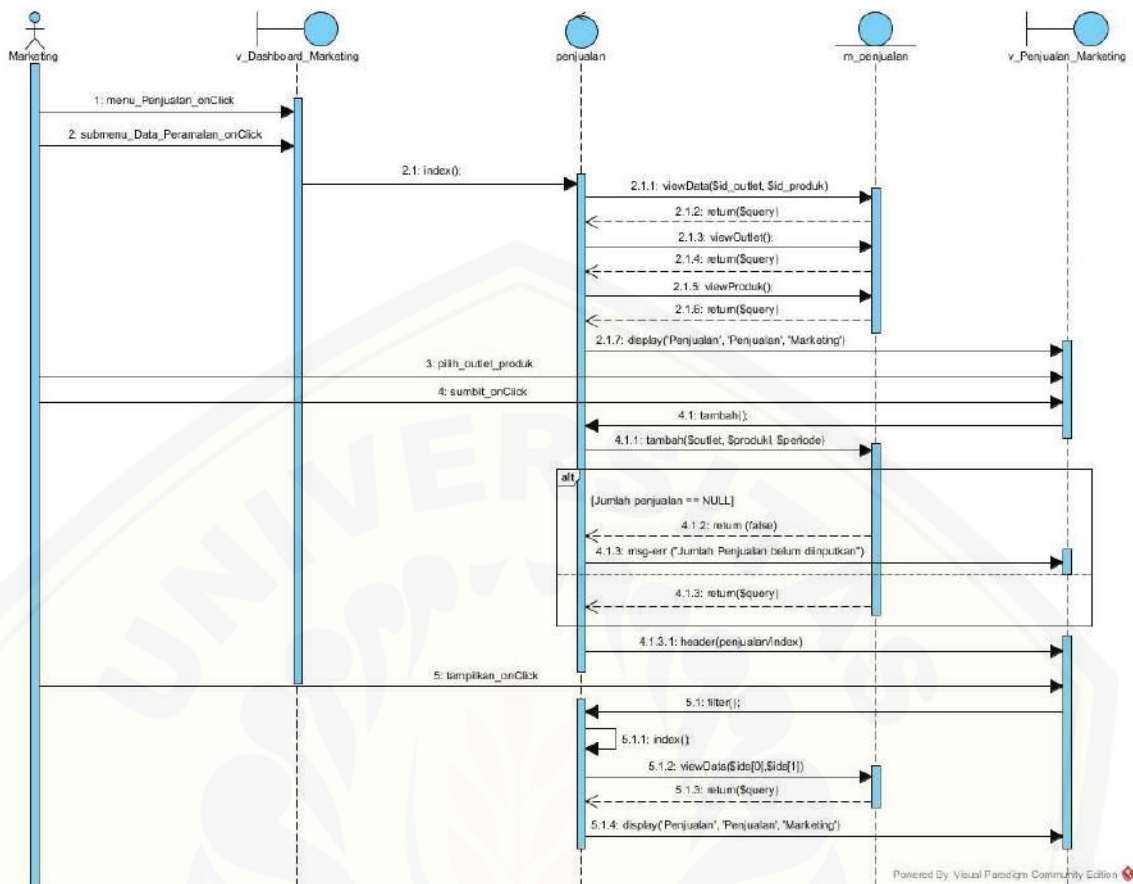
Penggambaran *sequence diagram* pengelolaan data *outlet* digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

5. *Sequence Diagram Melihat Data Outlet*

Penggambaran *sequence diagram* melihat data *outlet* digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

6. *Sequence Diagram Pengelolaan Data Hasil Peramalan*

Penggambaran *sequence diagram* pengelolaan data hasil peramalan digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar 4.3.



Gambar 4.3 *Sequence* Pengelolaan Data Hasil Peramalan

7. *Sequence Diagram* Melihat Data Hasil Peramalan

Penggambaran *sequence diagram* melihat data hasil peramalan digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

8. *Sequence Diagram* Pengelolaan Data Pendistribusian Produk.

Penggambaran *sequence diagram* pengelolaan data pendistribusian produk digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

9. *Sequence Diagram* Melihat Detail Hasil Penjualan dan Sisa Produk

Penggambaran *sequence diagram* melihat detail hasil penjualan dan sisa produk digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

10. *Sequence Diagram* Melihat Data Hasil Penjualan dan Pendistribusian Produk.

Penggambaran *sequence diagram* melihat data hasil penjualan dan pendistribusian produk digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

11. *Sequence Diagram* Input Transaksi Penjualan.

Penggambaran *sequence diagram* input transaksi penjualan digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

12. *Sequence Diagram* Melihat Rekap Data Hasil Transaksi Penjualan.

Penggambaran *sequence diagram* melihat rekap data hasil transaksi penjualan digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada lampiran B.

4.2.5 *Activity Diagram*

Activity Diagram menggambarkan alur aktivitas dalam sistem peramalan distribusi produk yang akan dibangun.

1. *Activity Diagram Login.*

Activity Diagram Login dapat dilihat pada lampiran C.

2. *Activity Diagram* Pengelolaan Data Pengguna.

Activity Diagram Pengelolaan Data pengguna dapat dilihat pada lampiran C.

3. *Activity Diagram* Melihat Data Pengguna.

Activity Diagram melihat data pengguna dapat dilihat pada lampiran C.

4. *Activity Diagram* Pengelolaan Data *Outlet*.

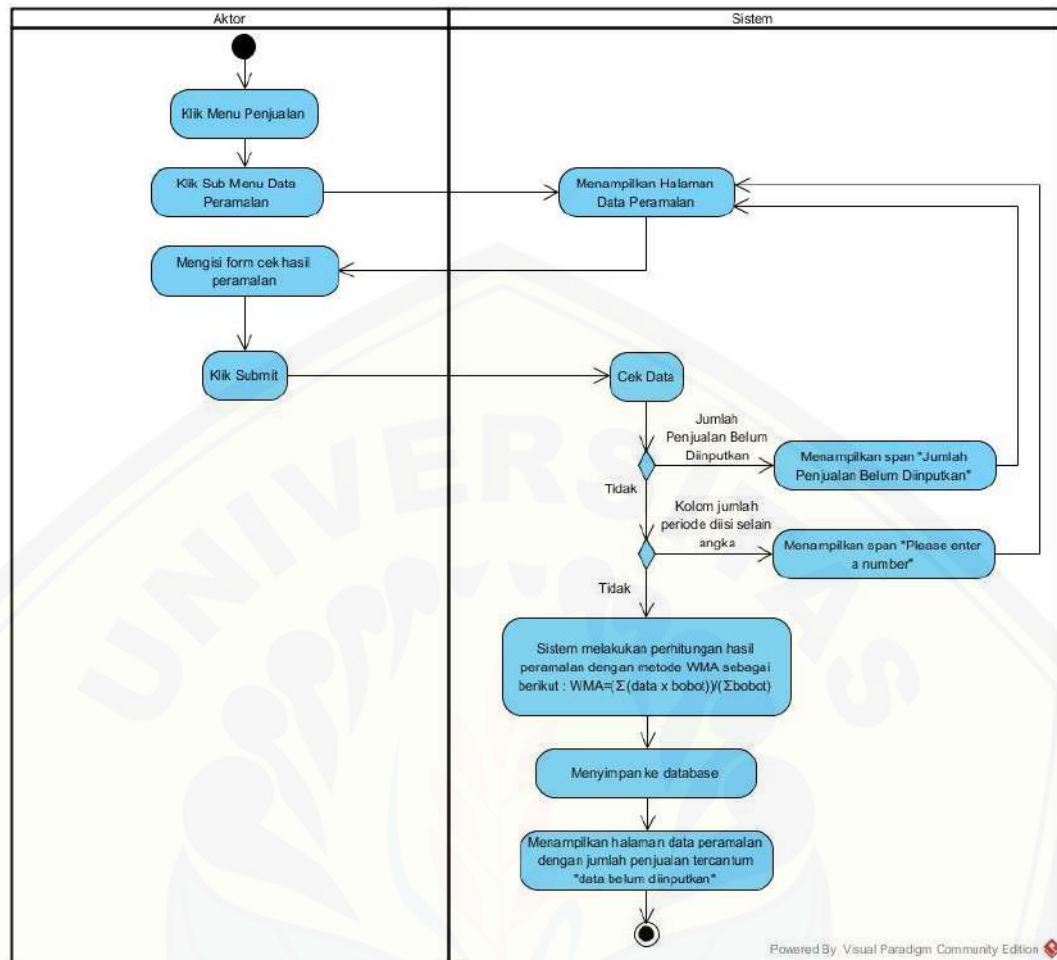
Activity Diagram pengelolaan data *outlet* dapat dilihat pada lampiran C.

5. *Activity Diagram* Melihat Data *Outlet*.

Activity Diagram melihat data *outlet* dapat dilihat pada lampiran C.

6. *Activity Diagram* Pengelolaan Data Hasil Peramalan.

Activity Diagram pengelolaan data hasil peramalan dapat dilihat pada gambar 4.4



Gambar 4.4 Activity Diagram Pengelolaan Data Hasil Peramalan

7. *Activity Diagram* Melihat Data Hasil Peramalan.
Activity Diagram melihat data hasil peramalan dapat dilihat pada lampiran C.
8. *Activity Diagram* Pengelolaan Data Pendistribusian Produk.
Activity Diagram pengelolaan data pendistribusian produk dapat dilihat pada lampiran C.
9. *Activity Diagram* Melihat Detail Hasil Penjualan dan Sisa Produk
Activity Diagram melihat detail hasil penjualan dan sisa produk dapat dilihat pada lampiran C.
10. *Activity Diagram* Melihat Data Hasil Penjualan dan Pendistribusian Produk.

Activity Diagram melihat data hasil penjualan dan pendistribusian produk dapat dilihat pada lampiran C.

11. *Activity Diagram* Input Transaksi Penjualan.

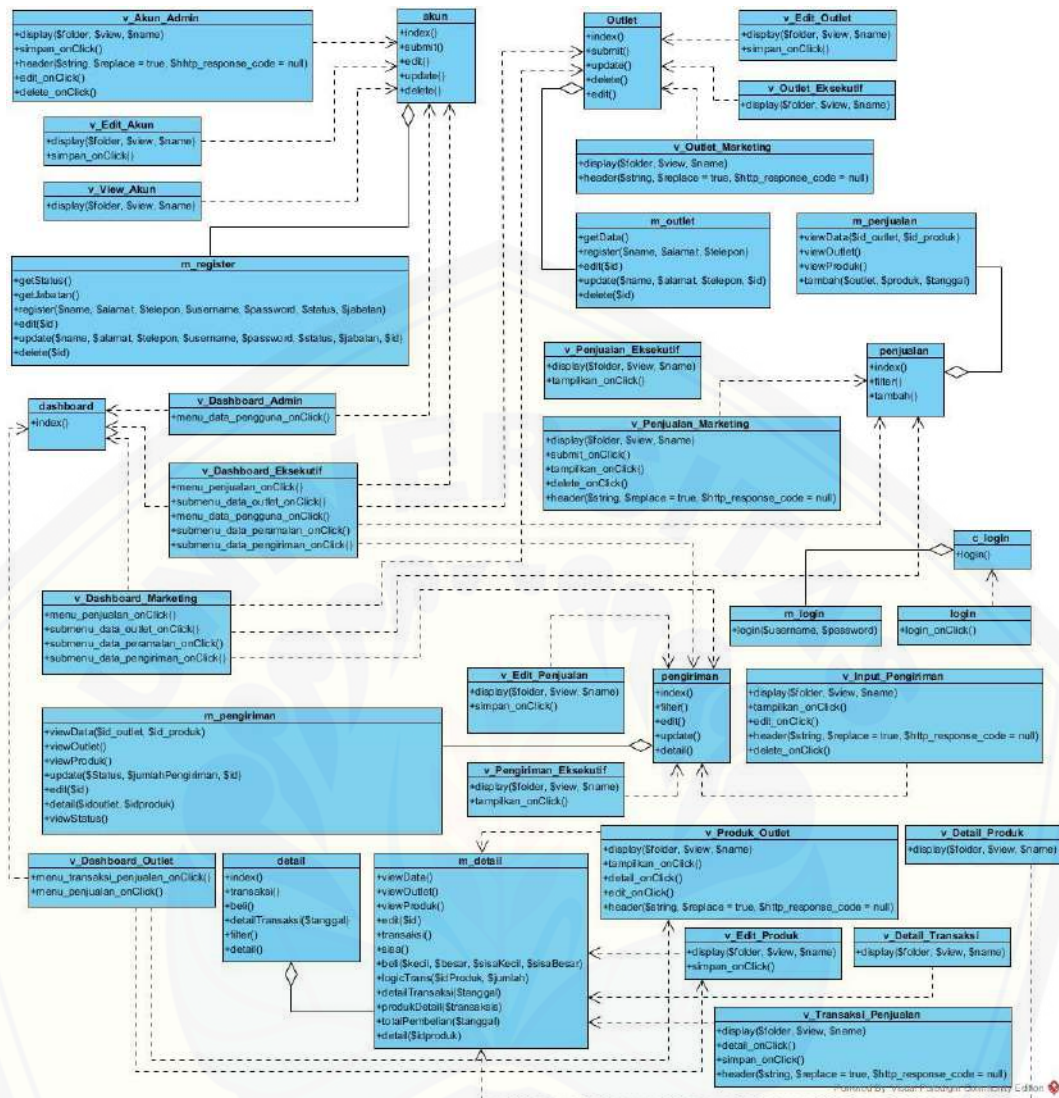
Activity Diagram input transaksi penjualan dapat dilihat pada lampiran C.

12. *Activity Diagram* Melihat Rekap Data Hasil Transaksi Penjualan.

Activity Diagram melihat rekap data hasil transaksi penjualan dapat dilihat pada lampiran C.

4.2.6 *Class Diagram*

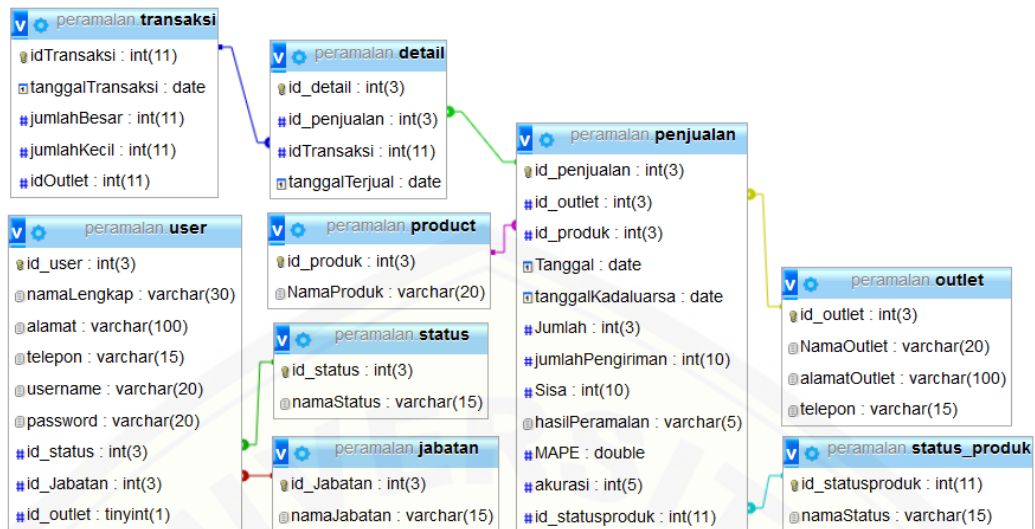
Class Diagram menggambarkan hubungan antarkelas yang digunakan untuk membangun suatu sistem. *Class Diagram* Sistem Perencanaan dan Peramalan Distribusi Produk Menggunakan Metode *Weight Moving Average* (WMA) (Studi Kasus: Pusat Oleh-oleh Purnama Jati) dapat dilihat pada Gambar 4.5.



Gambar 4.5 Class Diagram

4.2.7 Entity Relationship Diagram (ERD)

ERD merupakan gambaran komponen dan struktur database yang digunakan dalam pembangunan sistem. ERD sistem perencanaan dan peramalan distribusi produk menggunakan metode *Weight Moving Average (WMA)* (Studi Kasus: Pusat Oleh-oleh Purnama Jati) dapat dilihat pada Gambar 4.6



Gambar 4.6 Entity Relationship Diagram

4.3 Penulisan Kode Program

Desain yang telah dibuat akan diimplementasikan ke dalam kode program. Penulisan kode program untuk setiap fitur seperti yang telah digambarkan pada *use case diagram* Gambar 4.2 ditulis dalam 3 bagian kelas, meliputi kelas *view*, *controller*, dan *model*.

4.3.1 Kode Program *Login*

Kode program *login* terletak pada kelas *login*, *c_login*, dan *m_login*. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.2 Kode Program Pengelolaan Data Pengguna

Kode program pengelolaan data pengguna terletak pada kelas *v_akun_admin*, *akun*, *m_register*. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.3 Kode Program Melihat Data Pengguna

Kode program melihat data pengguna terletak pada kelas *v_view_akun*, *akun*, *m_register*. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.4 Kode Program Pengelolaan Data *Outlet*

Kode program pengelolaan data *outlet* terletak pada kelas *v_outlet_marketing*, *outlet*, *m_outlet*. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.5 Kode Program Melihat Data *Outlet*

Kode program melihat data *outlet* terletak pada kelas *v_outlet_eksekutif*, *outlet*, *m_outlet*. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.6 Kode Program Pengelolaan Data Hasil Peramalan

Kode program pengelolaan data hasil peramalan terletak pada kelas *v_penjualan_marketing*, *penjualan*, *m_penjualan*. Penulisan kode program ini dapat dilihat pada lampiran D.

Tabel 4.4 Kode Program *Function* tambah

1.	function tambah(\$outlet, \$produk, \$periode){
2.	\$peramalan = \$this->db->selectWhere("SELECT * from penjualan
3.	where id_outlet = \$outlet and id_produk = \$produk order by id_penjualan
4.	desc limit \$periode");
5.	\$tanggal = \$this->db->selectWhere("SELECT Tanggal +
6.	INTERVAL 1 DAY as tanggal, Tanggal + INTERVAL 8 DAY as
7.	tanggalKadaluarsa from penjualan
8.	where id_outlet = \$outlet and id_produk = \$produk order by
9.	tanggal desc limit 1");
10.	
11.	\$p = \$peramalan['data'][0];
12.	\$jmlPenjualan = \$p['Jumlah'];
13.	if(\$jmlPenjualan == NULL){
14.	\$_SESSION['msg-err'] = "Jumlah Penjualan Produk Belum
15.	Diinputkan";
16.	return fasle;
17.	} else {
18.	\$tanggal1= \$tanggal['data'][0]['tanggal'];
19.	\$tanggalKadaluarsa1 = \$tanggal['data'][0]['tanggalKadaluarsa'];
20.	\$jumlah = 0;
21.	for (\$i=\$periode; \$i > 0 ; \$i--) {

22.	<code>\$ramal[] = \$peramalan['data'][\$periode-\$i]['Jumlah']*\$i;</code>
23.	<code>\$jumlah += \$i;</code>
24.	<code>}</code>
25.	<code>\$total = round(array_sum(\$ramal) / \$jumlah);</code>
26.	<code>\$stmt = \$this->db->prepare("INSERT INTO penjualan</code>
27.	<code>(id_penjualan, id_outlet, id_produk, Tanggal, tanggalKadaluarsa,</code>
28.	<code>hasilPeramalan)</code>
29.	<code>VALUES (NULL, :id_outlet, :id_produk, :tanggal,</code>
30.	<code>:tanggalKadaluarsa, :hasilPeramalan)");</code>
31.	<code>\$stmt->bindParam(':id_outlet', \$outlet, PDO::PARAM_STR);</code>
32.	<code>\$stmt->bindParam(':id_produk', \$produk, PDO::PARAM_STR);</code>
33.	<code>\$stmt->bindParam(':tanggal', \$tanggal1, PDO::PARAM_STR);//terus</code>
34.	<code>dimasukin di querynya</code>
35.	<code>\$stmt->bindParam(':tanggalKadaluarsa', \$tanggalKadaluarsa1,</code>
36.	<code>PDO::PARAM_STR);</code>
37.	<code>\$stmt->bindParam(':hasilPeramalan', \$total, PDO::PARAM_STR);</code>
38.	<code>\$stmt->execute();</code>
39.	<code>}</code>
40.	<code>}</code>

Tabel 4.5 Kode Program *Function* tambah

1.	<code>function tambah(){</code>
2.	<code>\$outlet = \$_POST['outlet'];</code>
3.	<code>\$produk = \$_POST['produk'];</code>
4.	<code>\$periode = \$_POST['periode'];</code>
5.	<code>\$hasil = \$this->tambah->tambah(\$outlet, \$produk, \$periode);</code>
6.	<code>header('location:'.URL.'/penjualan/index/1-1');</code>
7.	<code>}</code>

4.3.7 Kode Program Melihat Data Hasil Peramalan

Kode program melihat data hasil peramalan terletak pada kelas `v_penjualan_eksekutif`, `penjualan`, `m_penjualan`. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.8 Kode Program Pengelolaan Data Pendistribusian Produk

Kode program pengelolaan data pendistribusian produk terletak pada kelas `v_input_pengiriman`, `pengiriman`, `m_pengiriman`. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.9 Kode Program Melihat Detail Hasil Penjualan dan Sisa Produk

Kode program melihat data hasil penjualan dan pendistribusian produk terletak pada kelas `v_produk_outlet`, `v_detail_produk`, `detail`, `m_detail`. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.10 Kode Program Melihat Data Hasil Penjualan dan Pendistribusian Produk.

Kode program melihat data hasil penjualan dan pendistribusian produk terletak pada kelas `v_input_pengiriman`, `pengiriman`, `m_pengiriman`. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.11 Kode Program Input Transaksi Penjualan.

Kode program input transaksi penjualan terletak pada kelas `v_transaksi_penjualan`, `detail`, `m_detail`. Penulisan kode program ini dapat dilihat pada lampiran D.

4.3.12 Kode Program Melihat Rekap Data Hasil Transaksi Penjualan.

Kode Program melihat rekap data hasil transaksi penjualan terletak pada kelas `v_detail_transaksi`, `detail`, `m_detail`. Penulisan kode program ini dapat dilihat pada lampiran D.

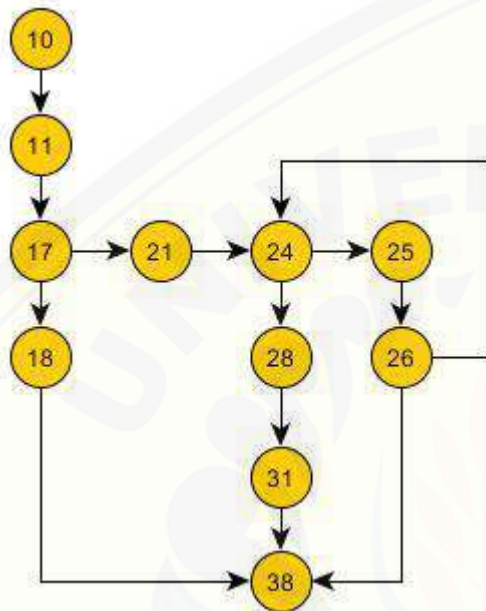
4.4 Pengujian Sistem

Pengujian dilakukan untuk mengevaluasi aplikasi yang telah dibuat. Proses pengujian dilakukan melalui dua tahapan pengujian pertama yaitu *white box* kemudian dilanjutkan dengan pengujian *black box*.

4.4.1 Pengujian White Box

Pengujian *white box* pada “sistem perencanaan dan peramalan distribusi produk berdasarkan jumlah permintaan dengan metode *Weight Moving Average*

(WMA)” (Studi kasus: Pusat Oleh-oleh Purnama Jati) ini dengan cara menggambar diagram alir, menghitung kompleksitas siklomatiknya (CC), dan membuat tabel pengujian *test case*. Pengujian kompleksitas siklomatik pada fitur pengelolaan data peramalan dapat dilihat pada tabel 4.6. Sedangkan untuk fitur yang lain dapat dilihat pada lampiran E.



Gambar 4.7 *Function tambah*

$$CC = E - N + 2 = 13 - 11 + 2 = 4$$

Tabel 4.6 *Test Case Function Tambah*

Test Case 1	Jika data penjualan terakhir tidak sama dengan null
Target yang diharapkan	Sistem menentukan tanggal kadaluarsa berdasarkan tanggal produksi
Hasil pengujian	Benar
Pah/Jalur	10, 11, 17, 21, 24, 28, 31, 38
Test Case 2	Jika data penjualan terakhir sama dengan null

Target yang diharapkan	Tidak berhasil menampilkan hasil peramalan dalam satu periode
Hasil pengujian	Benar
Pah/Jalur	10, 11, 17, 18, 38
Test Case 3	
Target yang diharapkan	Sistem melakukan perulangan perkalian bobot pada data sebelumnya sesuai dengan inputan periode
Hasil pengujian	Benar
Pah/Jalur	10, 11, 17, 21, 24, 25, 26, 24
Test Case 4	
Target yang diharapkan	Sistem menampilkan hasil peramalan untuk periode selanjutnya
Hasil pengujian	Benar
Pah/Jalur	10, 11, 17, 21, 24, 28, 31, 38

4.4.2 Pengujian Black Box

Pengujian *Black box* merupakan pengujian yang dilakukan oleh pengguna dengan menjalankan program secara langsung kemudian menganalisa *input* dan *output* yang dihasilkan aplikasi. Pengujian *black box* untuk fitur pengelolaan data hasil peramalan dapat dilihat pada tabel 4.7. Sedangkan untuk pengujian fitur yang lain dapat dilihat pada lampiran F.

Tabel 4.7 Pengujian *Black Box* Pengelolaan Data Hasil Peramalan

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data hasil peramalan	Memilih submenu data peramalan pada menu penjualan	Menampilkan tabel daftar hasil peramalan	V	
2.	Menambah data peramalan	Memilih tombol submit	Menyimpan data pada <i>database</i>	V	

BAB. 6 PENUTUP

Bab ini berisi mengenai kesimpulan dan saran dari peneliti tentang penelitian yang telah dilakukan. Kesimpulan dan saran tersebut diharapkan dapat digunakan sebagai acuan pada penelitian selanjutnya.

6.1 Kesimpulan

Kesimpulan dari penelitian yang dilakukan adalah:

1. Penelitian ini menghasilkan sebuah sistem perencanaan dan peramalan distribusi produk pada Pusat Oleh-oleh Purnama Jati dengan menggunakan metode *Weight Moving Average* (WMA). Sistem perencanaan dan peramalan distribusi produk ini dapat meramalkan jumlah produk yang akan didistribusikan pada masing-masing *outlet* untuk periode selanjutnya berdasarkan data permintaan produk pada periode sebelumnya.
2. Berdasarkan hasil uji coba diperoleh hasil peramalan dengan tingkat akurasi yang berbeda-beda untuk setiap *outlet*. Nilai akurasi untuk masing masing *outlet* secara keseluruhan berkisar antara 71% -76%, dan nilai MAPE berkisar antara 24% -29% sehingga sistem perencanaan dan peramalan distribusi produk termasuk dalam kategori cukup layak untuk digunakan. Sistem ini dikatakan cukup layak untuk digunakan karena nilai error yang relatif besar. Hal ini disebabkan karena pola data penjualan yang digunakan mengalami pelonjakan dan penurunan yang cukup *significant* sehingga kurang sesuai untuk diolah dengan menggunakan metode *Weight Moving Average* (WMA).

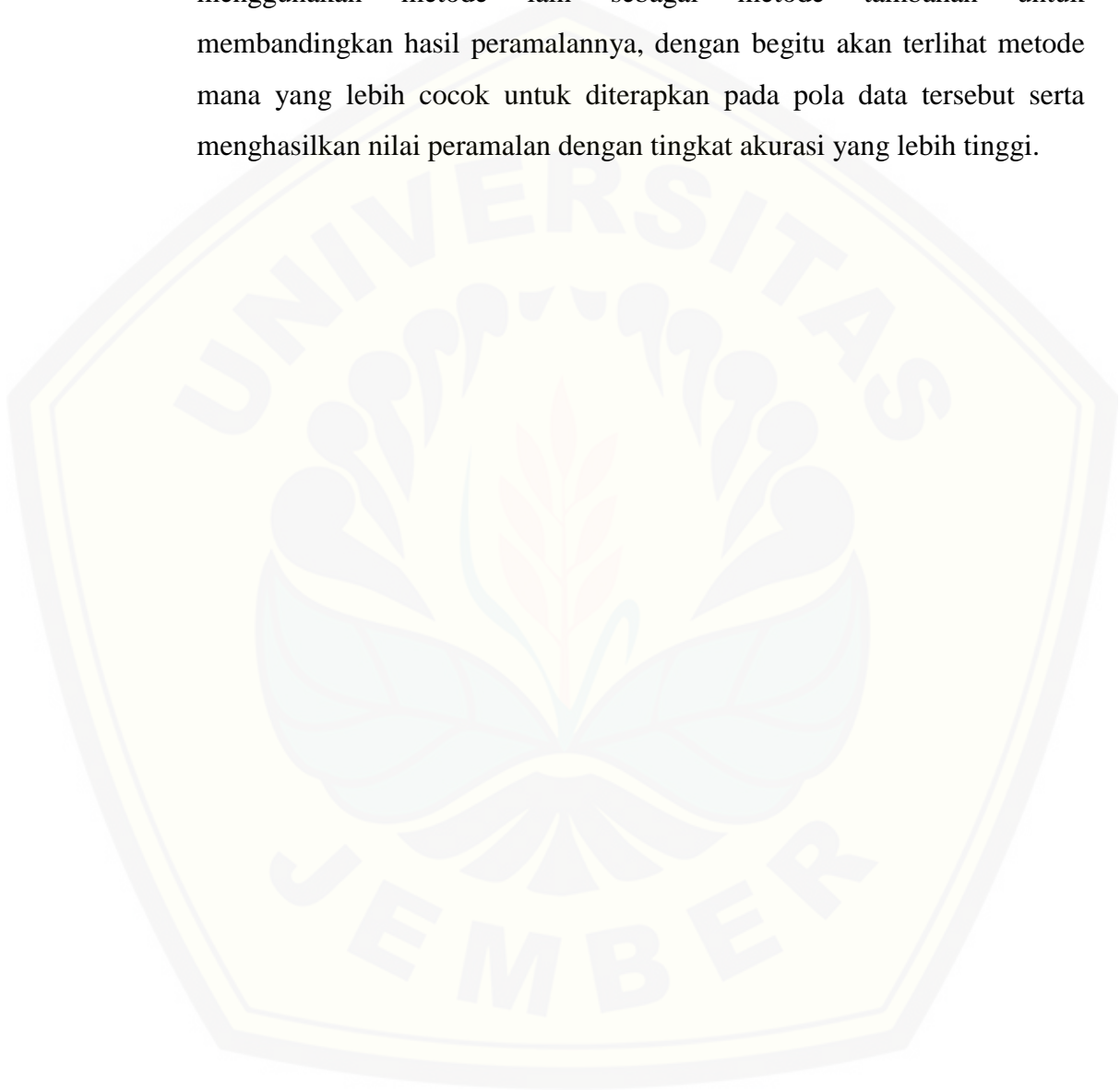
6.2 Saran

Beberapa saran berikut diharapkan dapat memberikan perbaikan dalam penelitian selanjutnya, yaitu:

1. Sebaiknya dalam menentukan metode harus benar-benar memperhatikan kondisi pola data yang akan diramalkan. Alangkah lebih baik apabila penentuan metode dilakukan ketika sudah mendapatkan data *real* dari objek

penelitian yang akan dibahas, sehingga dapat diketahui terlebih dahulu bagaimana pola data yang akan diramalkan dan metode yang cocok untuk mengolah data peramalannya.

2. Apabila penelitian ini dikembangkan nantinya, sebaiknya peneliti menggunakan metode lain sebagai metode tambahan untuk membandingkan hasil peramalannya, dengan begitu akan terlihat metode mana yang lebih cocok untuk diterapkan pada pola data tersebut serta menghasilkan nilai peramalan dengan tingkat akurasi yang lebih tinggi.



DAFTAR PUSTAKA

- AbuZaid, H. (2015, Maret 25). *Top Effective Project Management Methodologies and When to Use Them*. Retrieved from <https://www.linkedin.com/pulse/top-effective-project-management-methodologies-when-use-abuzaid>
- Kapgate, P. D. (2014). Weighted Moving Average Forecast Model based Prediction Service Broker Algorithm for Cloud Computing. *International Journal of Computer Science and Mobile Computing*, 71-79.
- Kristien Margi S, S. P. (2015). Analisa dan Penerapan Metode Single Exponential Smoothing untuk Prediksi Penjualan pada Periode Tertentu (Studi Kasus : PT. Media Cemara Kreasi). *Prosiding SNATIF*, 259-266.
- Maftahatul Hakimah, R. R. (2015). Rancang Bangun Aplikasi Peramalan Persediaan Barang Dengan Metode Trend Projection. *Jurnal SimanteC*, 37-48.
- Muhammad Asnawi, S. H. (2013). Karakteristik Tape Ubi Kayu (Manihot utilissima) Melalui Proses Pematangan Dengan Penggunaan Pengontrol Suhu. *Bioproses Komoditas Tropis*, 56-66.
- Prasetyo, S. B. (2008). Analisis Efisiensi Distribusi Pemasaran Produk dengan Metode Data Envelopment Analysis (DEA). *Jurnal Penelitian Ilmu Teknik*, 120-128.
- Refi Meisadri, N. I. (2013). Pembangunan Game First Person Shooter 3d Alien Hunter. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 1-6.
- Sheetal Sharma, D. S. (2012). Agile Processes and Methodologies: A Conceptual Study . *International Journal on Computer Science and Engineering (IJCSE)*, 892-898.
- Shinta Siti Sundari, S. W. (2015). Sistem Peramalan Persediaan Barang Dengan Weight Moving Average Di Toko The Kids 24. *Konferensi Nasional Sistem & Informatika*, 598-603.
- Tri Bowo Atmojo, R. P. (2013). Pengembangan Model Peramalan Permintaan Kebutuhan Reseller Menggunakan Extreme Learning Machine dalam Konteks Intelligent Warehouse Management System (IWMS). *SemnasIF*, A258-A263.

LAMPIRAN

A. Use Case Skenario

A.1 Skenario Use Case Login

Skenario *Use Case Login* menghubungkan dengan 4 aktor, yaitu admin, marketing, eksekutif, dan admin *outlet*. Dimana terdapat 6 admin *outlet* yang berbeda. Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternative *use case login* dijelaskan pada Tabel A.1 sampai dengan tabel A.9.

Tabel A. 1 Skenario *Use Case Login* Admin

ID	USC_1
Nama Use Case	Login
Aktor	Admin
Deskripsi singkat	Admin akan mengakses sistem
PreKondisi	Admin berada pada halaman login
PostKondisi	Admin berhasil login dan masuk kedalam sistem
<i>Flow Events</i>	
Skenario Normal : Login	
Aktor	Sistem
1. Mengisi username dan password	
2. Klik login	
	3. Menampilkan halaman home admin
Skenario alternative : username dan password salah	
2. Klik login	
	3. Menampilkan span "Username dan Password Salah"
Skenario alternative : username atau password kosong	

2. Klik login	
	3. Menampilkan span “ <i>Please Fill Out This Field</i> ”

Tabel A. 2 Skenario *Use Case Login Marketing*

ID	USC_1	
Nama Use Case	Login	
Aktor	Marketing	
Deskripsi singkat	Marketing akan mengakses sistem	
PreKondisi	Marketing berada pada halaman login	
PostKondisi	Marketing berhasil login dan masuk kedalam sistem	
Flow Events		
Skenario Normal : Login		
Aktor	Sistem	
1. Mengisi username dan password		
2. Klik login		
	3. Menampilkan halaman home admin	
Skenario alternative : username dan password salah		
2. Klik login		
	3. Menampilkan span “Username dan Password Salah”	
Skenario alternative : username atau password kosong		
2. Klik login		
	3. Menampilkan span “ <i>Please Fill Out This Field</i> ”	

Tabel A. 3 Skenario *Use Case Login* Eksekutif

ID	USC_1
Nama Use Case	Login
Aktor	Eksekutif
Deskripsi singkat	Eksekutif akan mengakses sistem
PreKondisi	Eksekutif berada pada halaman login
PostKondisi	Eksekutif berhasil login dan masuk ke dalam sistem
<i>Flow Events</i>	
Skenario Normal : Login	
Aktor	Sistem
1. Mengisi username dan password	
2. Klik login	
	3. Menampilkan halaman home admin
Skenario alternative : username dan password salah	
2. Klik login	
	3. Menampilkan span "Username dan Password Salah"
Skenario alternative : username atau password kosong	
2. Klik login	
	3. Menampilkan span " <i>Please Fill Out This Field</i> "

Tabel A. 4 Skenario *Use Case Login Admin Outlet A*

ID	USC_1
Nama Use Case	Login
Aktor	<i>Admin Outlet</i>
Deskripsi singkat	<i>Admin Outlet</i> akan mengakses sistem

PreKondisi	Admin <i>Outlet</i> berada pada halaman login
PostKondisi	Admin <i>Outlet</i> berhasil login dan masuk kedalam sistem
Flow Events	
Skenario Normal : Login	
Aktor	Sistem
1. Mengisi username dan password	
2. Klik login	
	3. Menampilkan halaman home admin
Skenario alternative : username dan password salah	
2. Klik login	
	3. Menampilkan span "Username dan Password Salah"
Skenario alternative : username atau password kosong	
2. Klik login	
	3. Menampilkan span " <i>Please Fill Out This Field</i> "

Tabel A. 5 Skenario *Use Case* Login Admin *Outlet* B

ID	USC_1
Nama Use Case	Login
Aktor	Admin <i>Outlet</i>
Deskripsi singkat	Admin <i>Outlet</i> akan mengakses sistem
PreKondisi	Admin <i>Outlet</i> berada pada halaman login
PostKondisi	Admin <i>Outlet</i> berhasil login dan masuk kedalam sistem
Flow Events	
Skenario Normal : Login	

Aktor	Sistem
1. Mengisi username dan password	
2. Klik login	
	3. Menampilkan halaman home admin
Skenario alternative : username dan password salah	
2. Klik login	
	3. Menampilkan span "Username dan Password Salah"
Skenario alternative : username atau password kosong	
2. Klik login	
	3. Menampilkan span " <i>Please Fill Out This Field</i> "

Tabel A. 6 Skenario Use Case Login Admin Outlet C

ID	USC_1
Nama Use Case	Login
Aktor	<i>Admin Outlet</i>
Deskripsi singkat	<i>Admin Outlet</i> akan mengakses sistem
PreKondisi	<i>Admin Outlet</i> berada pada halaman login
PostKondisi	<i>Admin Outlet</i> berhasil login dan masuk kedalam sistem
Flow Events	
Skenario Normal : Login	
Aktor	Sistem
1. Mengisi username dan password	
2. Klik login	
	3. Menampilkan halaman home admin

Skenario alternative : username dan password salah	
2. Klik login	
	3. Menampilkan span “Username dan Password Salah”
Skenario alternative : username atau password kosong	
2. Klik login	
	3. Menampilkan span “ <i>Please Fill Out This Field</i> ”

Tabel A. 7 Skenario Use Case Login Admin Outlet D

ID	USC_1
Nama Use Case	Login
Aktor	Admin <i>Outlet</i>
Deskripsi singkat	Admin <i>Outlet</i> akan mengakses sistem
PreKondisi	Admin <i>Outlet</i> berada pada halaman login
PostKondisi	Admin <i>Outlet</i> berhasil login dan masuk kedalam sistem
Flow Events	
Skenario Normal : Login	
Aktor	Sistem
1. Mengisi username dan password	
2. Klik login	
	3. Menampilkan halaman home admin
Skenario alternative : username dan password salah	
2. Klik login	
	3. Menampilkan span “Username dan Password Salah”
Skenario alternative : username atau password kosong	

2. Klik login	
	3. Menampilkan span “ <i>Please Fill Out This Field</i> ”

Tabel A. 8 Skenario *Use Case Login Admin Outlet E*

ID	USC_1
Nama Use Case	Login
Aktor	<i>Admin Outlet</i>
Deskripsi singkat	<i>Admin Outlet</i> akan mengakses sistem
PreKondisi	<i>Admin Outlet</i> berada pada halaman login
PostKondisi	<i>Admin Outlet</i> berhasil login dan masuk kedalam sistem
Flow Events	
Skenario Normal : Login	
Aktor	Sistem
1. Mengisi username dan password	
2. Klik login	
	3. Menampilkan halaman home admin
Skenario alternative : username dan password salah	
2. Klik login	
	3. Menampilkan span “Username dan Password Salah”
Skenario alternative : username atau password kosong	
2. Klik login	
	3. Menampilkan span “ <i>Please Fill Out This Field</i> ”

Tabel A. 9 Skenario *Use Case Login Admin Outlet F*

ID	USC_1	
Nama Use Case	Login	
Aktor	Admin <i>Outlet</i>	
Deskripsi singkat	Admin <i>Outlet</i> akan mengakses sistem	
PreKondisi	Admin <i>Outlet</i> berada pada halaman login	
PostKondisi	Admin <i>Outlet</i> berhasil login dan masuk ke dalam sistem	
Flow Events		
Skenario Normal : Login		
Aktor	Sistem	
4. Mengisi username dan password		
5. Klik login		
	6. Menampilkan halaman home admin	
Skenario alternative : username dan password salah		
4. Klik login		
	5. Menampilkan span "Username dan Password Salah"	
Skenario alternative : username atau password kosong		
4. Klik login		
	5. Menampilkan span " <i>Please Fill Out This Field</i> "	

A.2 Skenario *Use Case* Pengelolaan Data Pengguna

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* pengelolaan data pengguna dapat dilihat pada tabel A.10.

Tabel A. 10 Skenario *Use Case* Pengelolaan Data Pengguna

ID	USC_2
Nama Use Case	Pengelolaan Data Pengguna
Aktor	Admin
Deskripsi Singkat	Admin akan mengelolah data pengguna
PreKondisi	Marketing memilih menu data pengguna
PostKondisi	Marketing berhasil melihat, menambahkan, mengupdate, dan menghapus data pengguna
Flow Events	
Skenario Normal : Menambah data user	
Aktor	Reaksi Sistem
1. Klik menu data pengguna	<p>2. Menampilkan halaman data pengguna yang meliputi</p> <ul style="list-style-type: none"> • Form tambah akun untuk memasukkan data pengguna yang berisi <ul style="list-style-type: none"> - Nama Lengkap - Alamat - Telepon - <i>Username</i> - <i>Password</i> - Status - Jabatan • Tabel data pengguna untuk menampilkan data pengguna <ul style="list-style-type: none"> - No - Nama Pegawai - Alamat

	<ul style="list-style-type: none"> - Telepon - <i>Username</i> - <i>Password</i> - Status Jabatan - <i>Action</i> • Tombol edit dan <i>delete</i>
<p>3. Mengisi form tambah akun yang berisi</p> <ul style="list-style-type: none"> • Nama Lengkap • Alamat • Telepon • <i>Username</i> • <i>Password</i> • Status • Jabatan 	
4. Klik tombol simpan	
	5. Menyimpan ke database
	6. Menampilkan halaman data pengguna dengan kondisi data yang telah diinputkan terapat pada tabel data pengguna
Skenario alternative : Kolom pada form kosong	
4. Klik tombol simpan	
	5. Menampilkan span "please fill out this field"
Skenario alternative : Kolom telepon harus diisi angka	
1. Klik tombol simpan	
	2. Menampilkan span "please enter a number"
Skenario Normal : Mengubah data <i>user</i>	

Aktor	Reaksi Sistem
7. Klik tombol edit	
	8. Menampilkan data pengguna pada form sesuai dengan data pengguna yang dipilih pada tabel. Form edit akun berisi <ul style="list-style-type: none"> • Nama Lengkap • Alamat • Telepon • <i>Username</i> • <i>Password</i> • Status • Jabatan
9. Mengubah data pada form data pengguna	
10. Klik tombol simpan	
	11. Menyimpan ke database
	12. Menampilkan halaman data pengguna dengan kondisi data pengguna yang diedit sudah terupdate pada tabel data pengguna
Skenario alternative : Kolom pada form kosong	
10. Klik tombol simpan	
	11. Menampilkan span “please fill out this field”
Skenario alternative : Kolom telepon harus diisi angka	
10. Klik tombol simpan	
	12. Menampilkan span “please enter a number”

Skenario Normal : Menghapus data <i>user</i>	
Aktor	Reaksi Sistem
7. Klik tombol delete	
	8. Menampilkan popup “apakah anda yakin menghapus data pengguna?”
9. Klik Ya	
	10. Menghapus data dari database
	11. Menampilkan halaman data pengguna

A.3 Skenario *Use Case* Melihat Data Pengguna

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* melihat data pengguna dapat dilihat pada tabel A.11.

Tabel A. 11 Skenario *Use Case* Melihat Data Pengguna

ID	USC_3
Nama Use Case	Melihat Data Pengguna
Aktor	Eksekutif
Deskripsi singkat	Eksekutif akan melihat data pengguna
PreKondisi	Eksekutif memilih menu data pengguna
PostKondisi	Eksekutif berhasil melihat data pengguna
Flow Events	
Skenario Normal : Melihat Data User	
Aktor	Sistem
1. Klik menu data pengguna	
	2. Menampilkan halaman data pengguna yang meliputi tabel data pengguna yang terdiri dari : - No

	<ul style="list-style-type: none"> - Nama Lengkap - Alamat - Telepon - Jabatan - Status
--	--

A.4 Skenario *Use Case* Pengelolaan Data *Outlet*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* pengelolaan data *outlet* dapat dilihat pada tabel A.12.

Tabel A. 12 Skenario *Use Case* Pengelolaan Data *Outlet*

ID	USC_4
Nama Use Case	Pengelolaan Data <i>Outlet</i>
Aktor	Marketing
Deskripsi Singkat	Marketing akan mengelolah data <i>outlet</i>
PreKondisi	Marketing memilih submenu data <i>outlet</i> pada menu penjualan
PostKondisi	Marketing berhasil menambahkan, mengupdate, menghapus, dan melihat data <i>outlet</i>
<i>Flow Events</i>	
Skenario Normal : Menambah data <i>outlet</i>	
Aktor	Reaksi Sistem
1. Klik menu penjualan	
2. Klik sub menu data <i>outlet</i>	
	3. Menampilkan halaman data <i>outlet</i> yang meliputi

	<ul style="list-style-type: none"> • Form tambah <i>outlet</i> untuk memasukkan data <i>outlet</i> yang berisi: <ul style="list-style-type: none"> - Nama <i>Outlet</i> - Alamat - Telepon • Tabel data <i>outlet</i> untuk menampilkan data <i>outlet</i> yang berisi : <ul style="list-style-type: none"> - Nama <i>Outlet</i> - Alamat - Telepon - <i>Action</i> • Tombol edit dan <i>delete</i>
4. Mengisi form yang berisi <ul style="list-style-type: none"> - Nama <i>outlet</i> - Alamat - Telepon 	
5. Klik tombol simpan	
	6. Menyimpan ke database
	7. Menampilkan halaman data <i>outlet</i> dengan kondisi data <i>outlet</i> yang telah ditambahkan tercantum pada tabel data <i>outlet</i>
Skenario alternative : Kolom pada form kosong	
5. Klik tombol simpan	
	6. Menampilkan span “please fill out this field”
Skenario alternative : Kolom telepon harus diisi angka	
4. Klik tombol simpan	

	5. Menampilkan span “Please enter a number”
Skenario Normal : Mengubah data outlet	
Aktor	Reaksi Sistem
8. Klik tombol edit	
	9. Menampilkan data <i>outlet</i> pada form tambah <i>outlet</i> sesuai dengan data <i>outlet</i> yang dipilih pada tabel. Form tambah <i>outlet</i> berisi: <ul style="list-style-type: none"> • Nama <i>Outlet</i> • Alamat • Telepon
10. Mengubah data pada form tambah <i>outlet</i>	
11. Klik tombol simpan	
	12. Menyimpan ke database
	13. Menampilkan halaman data <i>outlet</i> dengan kondisi data <i>outlet</i> yang telah diedit sudah terupdate pada tabel data <i>outlet</i>
Skenario alternative : Kolom pada form kosong	
5. Klik tombol simpan	
	6. Menampilkan span “please fill out this field”
Skenario alternative : Kolom telepon harus diisi angka	
5. Klik tombol simpan	
	12. Menampilkan span “please enter a number”
Skenario Normal : Menghapus data outlet	
Aktor	Reaksi Sistem

8. Klik tombol delete	
	9. Menampilkan popup “apakah anda yakin menghapus data <i>outlet</i> ?”
10. Klik Ya	
	11. Menghapus data dari database
	12. Menampilkan halaman data <i>outlet</i>

A.5 Skenario *Use Case* Melihat Data *Outlet*

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* melihat data *outlet* dapat dilihat pada tabel A.13.

Tabel A. 13 Skenario *Use Case* Melihat Data *Outlet*

ID	USC_5	
Nama Use Case	Melihat Data <i>Outlet</i>	
Aktor	Eksekutif	
Deskripsi singkat	Eksekutif akan melihat data <i>outlet</i>	
PreKondisi	Eksekutif memilih submenu data <i>outlet</i> pada menu penjualan	
PostKondisi	Eksekutif berhasil melihat data <i>outlet</i>	
Flow Events		
Skenario Normal : Melihat Data <i>Outlet</i>		
Aktor	Sistem	
1. Klik menu Penjualan		
2. Klik sub menu data <i>outlet</i>		
	3. Menampilkan halaman data <i>outlet</i> yang berisi tabel data <i>outlet</i> yang terdiri dari : <ul style="list-style-type: none"> - Id <i>outlet</i> - Nama <i>outlet</i> 	

	<ul style="list-style-type: none"> - Alamat - Telepon
--	---

A.6 Skenario *Use Case* Melihat Data Hasil Peramalan

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* melihat data hasil peramalan dapat dilihat pada tabel A.14.

Tabel A. 14 Skenario *Use Case* Melihat Data Hasil Peramalan

ID	USC_7
Nama Use Case	Melihat Data Hasil Peramalan
Aktor	Eksekutif
Deskripsi singkat	Eksekutif akan melihat data hasil peramalan
PreKondisi	Eksekutif memilih submenu data peramalan pada menu penjualan
PostKondisi	Eksekutif berhasil melihat data hasil peramalan
Flow Events	
Skenario Normal : Melihat Data <i>Outlet</i>	
Aktor	Sistem
1. Klik menu Penjualan	
2. Klik sub menu data pengiriman	
	3. Menampilkan halaman data pengiriman yang berisi <ul style="list-style-type: none"> • <i>Dropdown</i> nama <i>outlet</i> • <i>Dropdown</i> jenis produk • Tombol tampilkan

	<ul style="list-style-type: none"> • Tabel data <i>outlet</i> yang terdiri dari: <ul style="list-style-type: none"> - No - Tanggal - Hasil Penjualan - Hasil Peramalan - MAPE - Akurasi
4. Memilih data yang akan ditampilkan melalui <i>dropdown</i>	
	5. Menampilkan data pada tabel peramalan

A.7 Skenario *Use Case* Pengelolaan Data Pendistribusian Produk

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* pengelolaan data pendistribusian produk dapat dilihat pada tabel A.15.

Tabel A. 15 Skenario *Use Case* Pengelolaan Data Pendistribusian Produk

ID	USC_8
Nama Use Case	Pengelolaan Data Pendistribusian Produk
Aktor	Marketing
Deskripsi Singkat	Marketing akan mengelolah data pendistribusian produk pada masing masing <i>outlet</i>
PreKondisi	Marketing memilih submenu data pengiriman pada menu penjualan

PostKondisi	Marketing berhasil melihat dan mengupdate data hasil penjualan dan pendistribusian produk
<i>Flow Events</i>	
Skenario Normal : Menambah data pendistribusian produk	
Aktor	Reaksi Sistem
1. Klik menu penjualan	
2. Klik sub menu data pengiriman	
	<p>3. Menampilkan halaman data pengiriman yang meliputi</p> <ul style="list-style-type: none"> • <i>Dropdown</i> nama <i>outlet</i> • <i>Dropdown</i> jenis produk • Tombol tampilkan • Tabel pengiriman yang digunakan untuk menampilkan data pengiriman. Tabel pengiriman terdiri dari : <ul style="list-style-type: none"> - No. - Tanggal Produksi - Tanggal Kadaluarsa - Jumlah Pengiriman - Hasil Penjualan - Sisa - Status - <i>Button</i> Edit • <i>Button</i> Detail
4. Pilih <i>dropdown</i> edit pada kolom <i>action</i>	

	<p>5. Menampilkan form tambah data pengiriman yang digunakan untuk menginputkan jumlah produk yang dikirimkan dan jumlah produk yang terjual. Form tambah data pengiriman terdiri dari:</p> <ul style="list-style-type: none"> • Nama <i>outlet</i> • Nama produk • Tanggal Produksi • Tanggal Kadaluarsa • Jumlah pengiriman • Tombol Simpan
6. Mengisi data jumlah pengiriman	
7. Klik tombol simpan	
	8. Sistem mengupdate status produk menjadi terkirim, dan menampilkan jumlah sisa sesuai dengan jumlah pengiriman
	9. Menyimpan ke database
	10. Menampilkan halaman data pengiriman dengan jumlah pengiriman, sisa dan status tercantum
Skenario alternative : Kolom pada form kosong	
7. Klik tombol simpan	
	8. Menampilkan span “please fill out this field”
Skenario alternative : Kolom jumlah pengiriman harus diisi angka	
7. Klik tombol simpan	

	8. Menampilkan span “please enter a number”
Skenario Normal : Melihat Detail Produk	
11. Klik <i>button</i> detail	
	12. Menampilkan halaman detail produk yang tersisa di setiap <i>outlet</i> pada tabel detail produk yang berisi: <ul style="list-style-type: none"> • No • ID Produk • Tanggal Produksi • Tanggal Kadaluarsa

A.8 Skenario *Use Case* Melihat Detail Hasil Penjualan dan Sisa Produk

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* melihat detail hasil penjualan dan sisa produk dapat dilihat pada tabel 16.

Tabel A. 16 Skenario *Use Case* Melihat Detail Hasil Penjualan dan Sisa Produk

ID	USC_9
Nama Use Case	Melihat Detail Hasil Penjualan dan Sisa Produk
Aktor	<i>Admin Outlet</i>
Deskripsi Singkat	<i>Admin Outlet</i> akan melihat detail hasil penjualan dan sisa produk
PreKondisi	<i>Admin Outlet</i> memilih menu data penjualan
PostKondisi	<i>Admin Outlet</i> berhasil melihat tabel hasil penjualan dan detail produk yang tersisa
Flow Events	

Skenario Normal : Melihat Detail Hasil Penjualan dan Sisa Produk	
Aktor	Reaksi Sistem
1. Klik menu data penjualan	
	<p>2. Menampilkan halaman data penjualan yang berisi :</p> <ul style="list-style-type: none">• <i>Dropdown</i> jenis produk• Tabel Hasil Penjualan yang terdiri dari:<ul style="list-style-type: none">- No- Tanggal Produksi- Tanggal Kadaluarsa- Hasil Penjualan- Sisa• <i>Button</i> Detail
3. Klik <i>button</i> detail	
	<p>4. Menampilkan halaman detail produk yang tersisa pada tabel detail produk yang belum terjual. Tabel tersebut terdiri dari:</p> <ul style="list-style-type: none">• No• ID Produk• Tanggal Produksi• Tanggal Kadaluarsa

A.9 Skenario *Use Case* Melihat Data Hasil Penjualan dan Pendistribusian Produk

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* melihat data hasil penjualan dan pendistribusian produk dapat dilihat pada tabel 17.

Tabel A. 17 Skenario *Use Case* Melihat Data Hasil Penjualan dan Pendistribusian Produk

ID	USC_10
Nama Use Case	Melihat data hasil penjualan dan pendistribusian produk
Aktor	Eksekutif
Deskripsi singkat	Eksekutif akan melihat data hasil penjualan dan pendistribusian produk
PreKondisi	Eksekutif memilih submenu data pengiriman pada menu penjualan
PostKondisi	Eksekutif berhasil melihat data hasil penjualan dan pendistribusian produk
<i>Flow Events</i>	
Skenario Normal : Melihat Data Penjualan dan Pendistribusian Produk	
Aktor	Sistem
1. Klik menu Penjualan	
2. Klik sub menu data pengiriman	
	3. Menampilkan halaman data pengiriman yang meliputi <ul style="list-style-type: none"> • <i>Dropdown</i> daftar jenis produk • <i>Dropdown</i> daftar <i>outlet</i>. • Tombol tampilkan

	<ul style="list-style-type: none"> • Tabel pengiriman yang terdiri dari : <ul style="list-style-type: none"> - No - Tanggal Produksi - Tanggal Kadaluarsa - Jumlah pengiriman - Hasil penjualan - Sisa - Status
4. Memilih data yang akan ditampilkan melalui <i>dropdown</i>	
	5. Menampilkan data pada tabel pengiriman

A.10 Skenario Input Transaksi Penjualan

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* input transaksi penjualan dapat dilihat pada tabel A.18.

Tabel A. 18 Skenario *Use Case* Pengelolaan Data Transaksi Penjualan

ID	USC_11
Nama Use Case	Pengelolaan Data Transaksi Penjualan
Aktor	<i>Admin Outlet</i>
Deskripsi Singkat	<i>Admin Outlet</i> akan mengelolah data transaksi penjualan untuk setiap <i>outlet</i>
PreKondisi	<i>Admin Outlet</i> memilih menu transaksi penjualan
PostKondisi	<i>Admin Outlet</i> berhasil melakukan transaksi penjualan

<i>Flow Events</i>	
Skenario Normal : Input Transaksi Penjualan	
Aktor	Reaksi Sistem
1. Klik menu transaksi penjualan	
	<p>2. Menampilkan halaman transaksi penjualan yang meliputi:</p> <ul style="list-style-type: none"> • Tanggal transaksi • Jumlah pembelian produk proll tape besar • Jumlah pembelian produk proll tape kecil • Jumlah sisa produk proll tape besar • Jumlah sisa produk proll tape kecil • <i>Button</i> simpan • Tabel detail transaksi digunakan untuk menampilkan rekap transaksi penjualan. Tabel detail transaksi terdiri dari : <ul style="list-style-type: none"> - Nomer - Tanggal Transaksi - Detail - <i>Button</i> detail
3. Mengisi data transaksi penjualan	
4. Klik <i>button simpan</i>	

	5. Sistem mengupdate jumlah hasil penjualan dan sisa dengan sistem <i>first in first out</i>
	6. Menyimpan data transaksi pada database dan menampilkan pada tabel detail transaksi.
Skenario alternative : Kolom pada form kosong	
4. Klik tombol simpan	
	5. Menampilkan span “please fill out this field”
Skenario alternative : Kolom jumlah penjualan harus diisi angka	
4. Klik tombol simpan	
	5. Menampilkan span “please enter e number”

A.11 Skenario Melihat Rekap Hasil Transaksi Penjualan

Penjelasan urutan aksi aktor dan reaksi sistem pada skenario normal dan skenario alternatif dari skenario *use case* melihat rekap transaksi penjualan dapat dilihat pada tabel A.19.

Tabel A. 19 Skenario Use Case Melihat Rekap Hasil Transaksi Penjualan

ID	USC_12
Nama Use Case	Melihat Rekap Hasil Transaksi Penjualan
Aktor	Admin <i>Outlet</i>
Deskripsi Singkat	Admin <i>Outlet</i> akan melihat rekap hasil transaksi penjualan
PreKondisi	Admin <i>Outlet</i> memilih menu transaksi penjualan
PostKondisi	Admin <i>Outlet</i> berhasil melihat rekap hasil transaksi penjualan

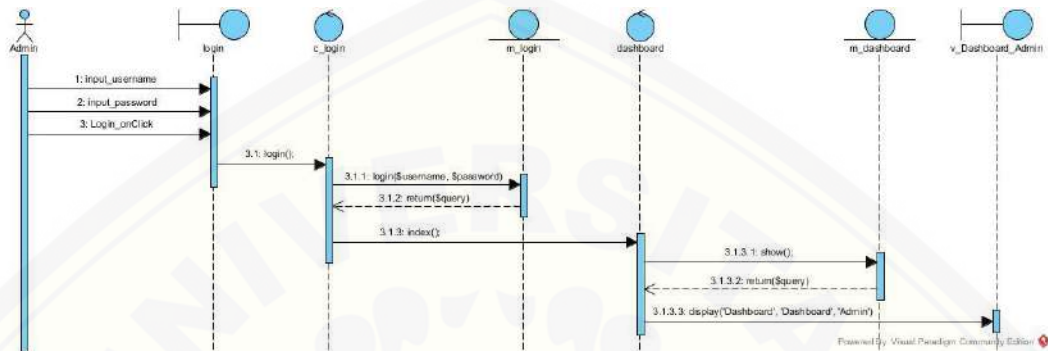
<i>Flow Events</i>	
Skenario Normal : Melihat Rekap Hasil Transaksi Penjualan	
Aktor	Reaksi Sistem
1. Klik menu transaksi penjualan	
	<p>2. Menampilkan halaman transaksi penjualan yang meliputi:</p> <ul style="list-style-type: none">• Tanggal transaksi• Jumlah pembelian produk proll tape besar• Jumlah pembelian produk proll tape kecil• Jumlah sisa produk proll tape besar• Jumlah sisa produk proll tape kecil• <i>Button</i> simpan• Tabel detail transaksi digunakan untuk menampilkan rekap transaksi penjualan. Tabel detail transaksi terdiri dari :<ul style="list-style-type: none">- Nomer- Tanggal Transaksi- Detail- <i>Button</i> detail
3. Klik <i>button</i> detail	

	<p>4. Menampilkan halaman Detail Transaksi Penjualan yang terdiri dari</p> <ul style="list-style-type: none">• Total pembelian produk proll tape besar• Total pembelian produk proll tape kecil• tabel untuk setiap transaksi yang berisi<ul style="list-style-type: none">- Jumlah pembelian produk proll tape besar- Jumlah pembelian produk proll tape kecil- Tanggal Produksi- ID. Produk
--	--

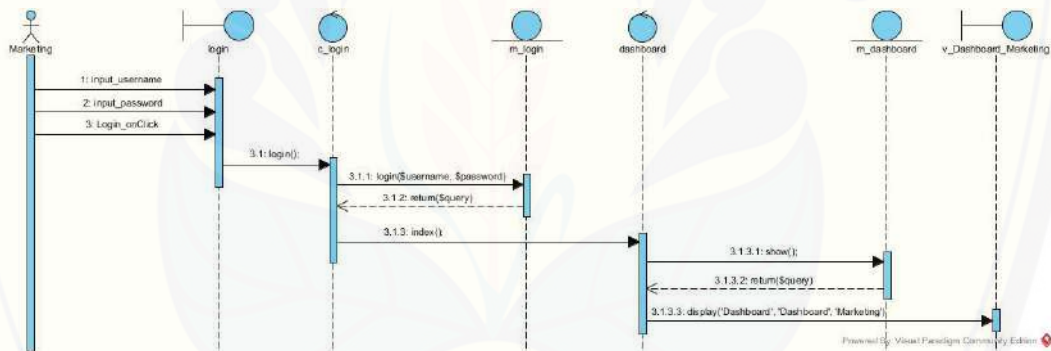
B. Sequence Diagram

B.1 Sequence Diagram Login

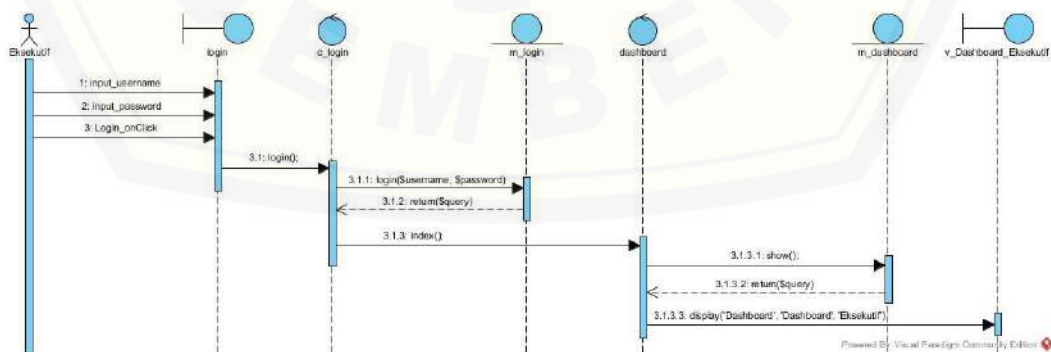
Penggambaran *sequence diagram login* digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.1 sampai B.9.



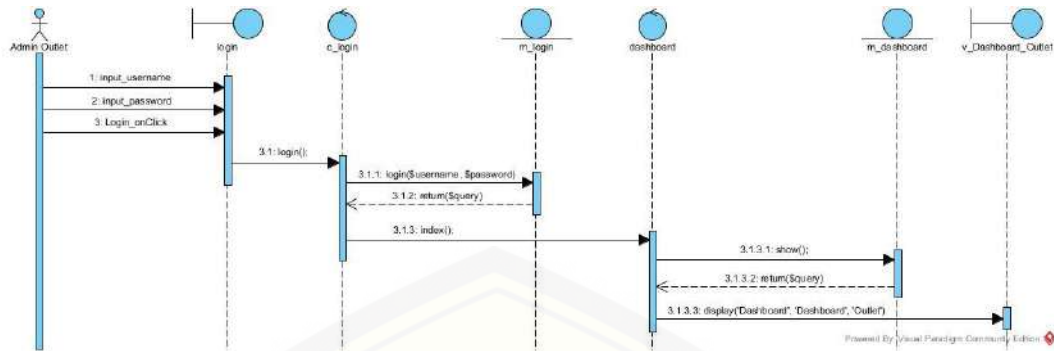
Gambar B. 1 Sequence Diagram Login Admin



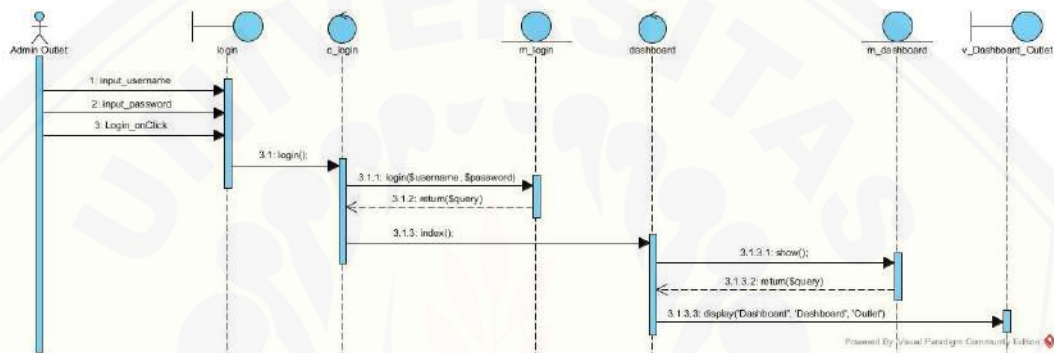
Gambar B. 2 Sequence Diagram Login Admin



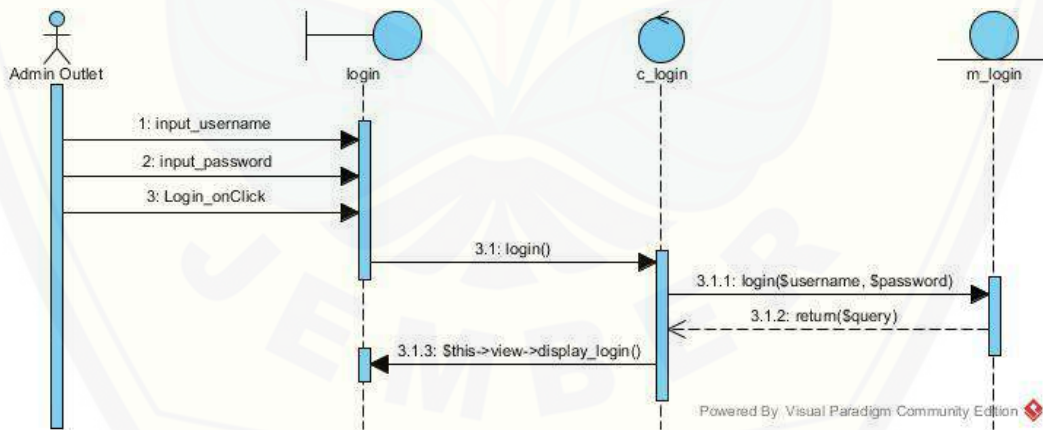
Gambar B. 3 Sequence Diagram Login Eksekutif



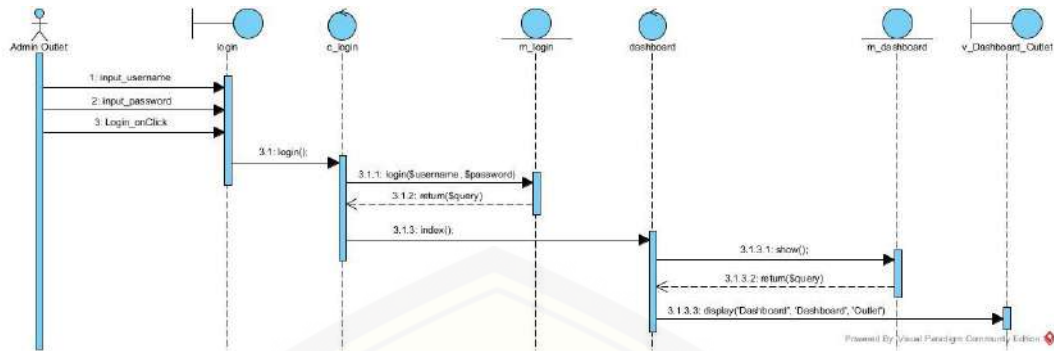
Gambar B. 4 Sequence Diagram Admin Outlet A



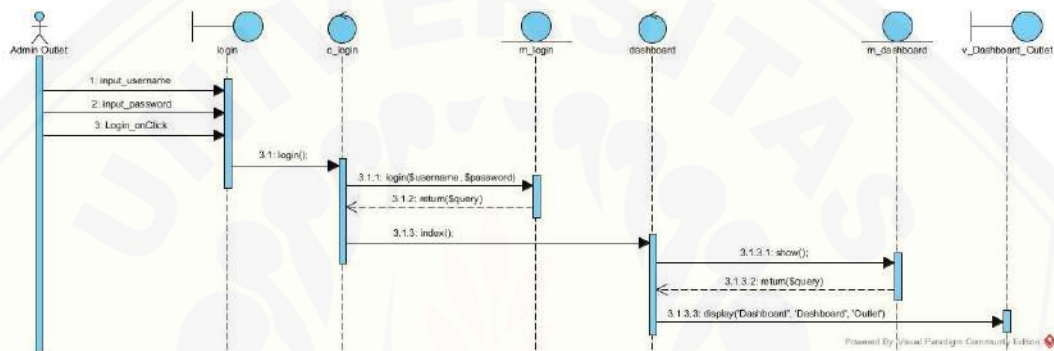
Gambar B. 5 Sequence Diagram Admin Outlet B



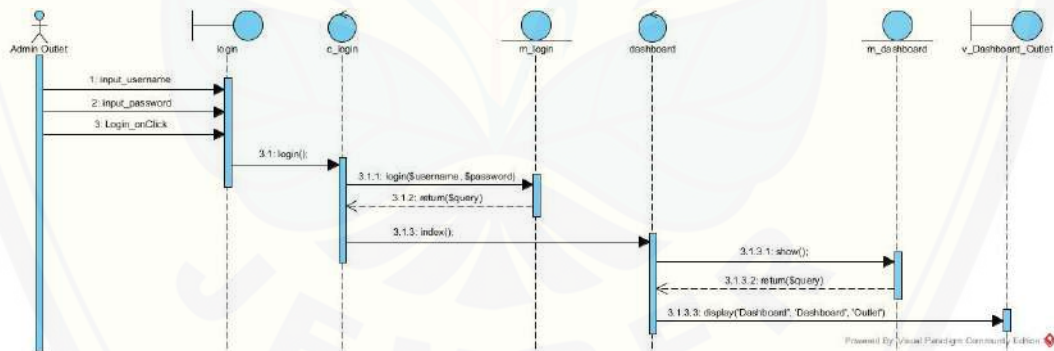
Gambar B. 6 Sequence Diagram Admin Outlet C



Gambar B. 7 Sequence Diagram Admin Outlet D



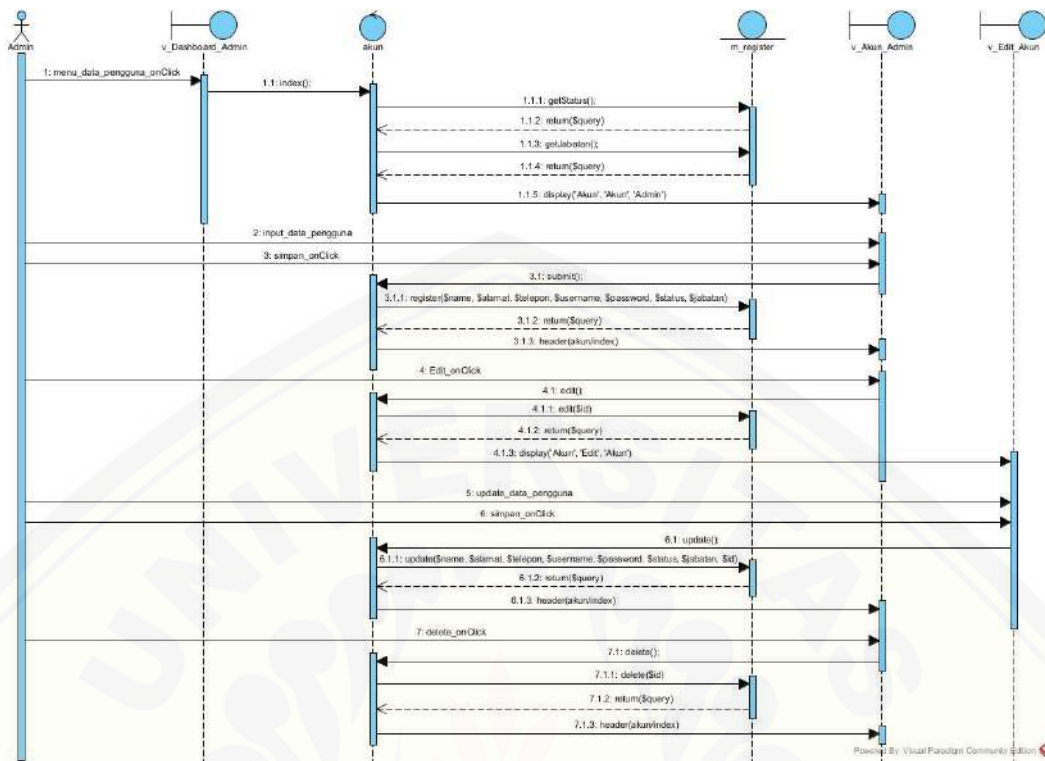
Gambar B. 8 Sequence Diagram Admin Outlet E



Gambar B. 9 Sequence Diagram Admin Outlet F

B.2 Sequence Diagram Pengelolaan Data Pengguna

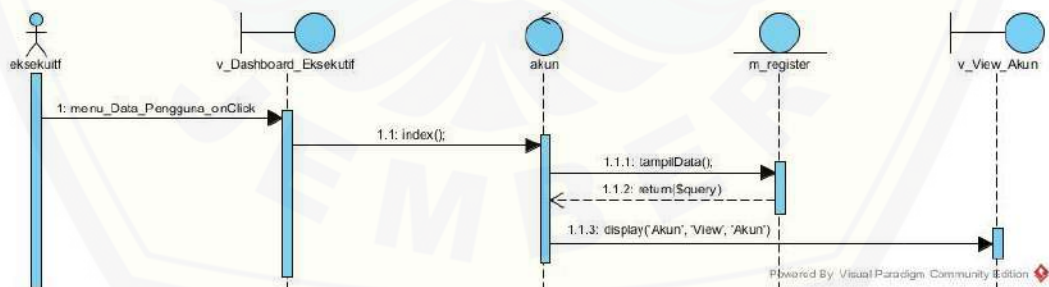
Penggambaran *sequence diagram* pengelolaan data pengguna digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.10.



Gambar B. 10 Sequence Diagram Pengelolaan Data Pengguna

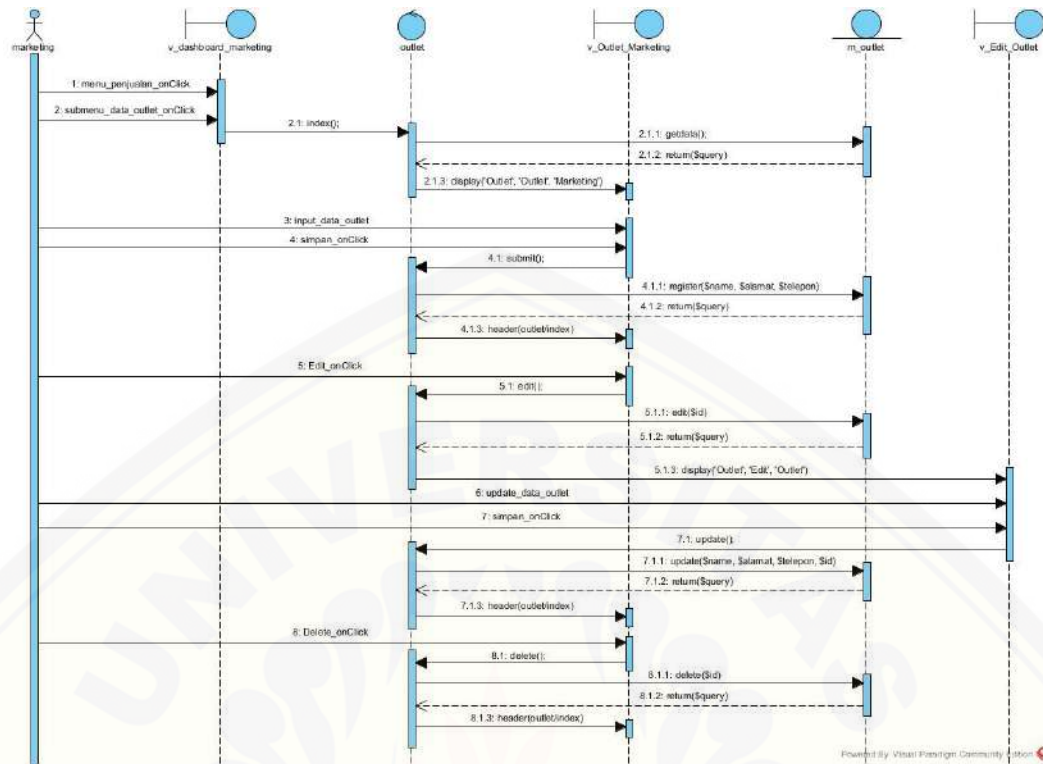
B.3 Sequence Diagram Melihat Data Pengguna

Penggambaran *sequence diagram* melihat data pengguna digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.11.



Gambar B. 11 Sequence Diagram Melihat Data Pengguna

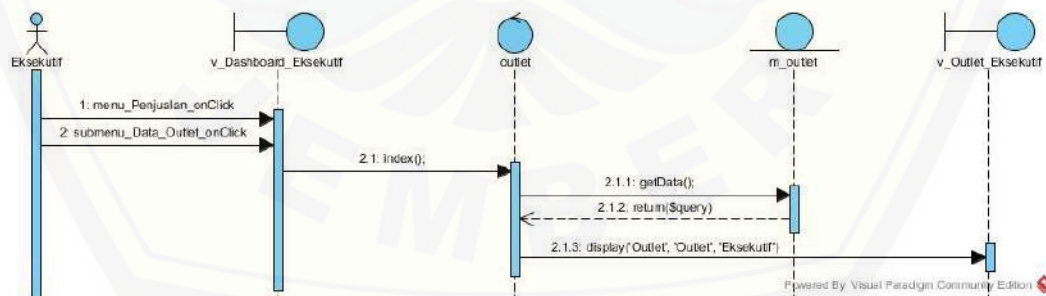
Penggambaran *sequence diagram* pengelolaan data outlet digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.12.



Gambar B. 12 Sequence Diagram Pengelolaan Data Outlet

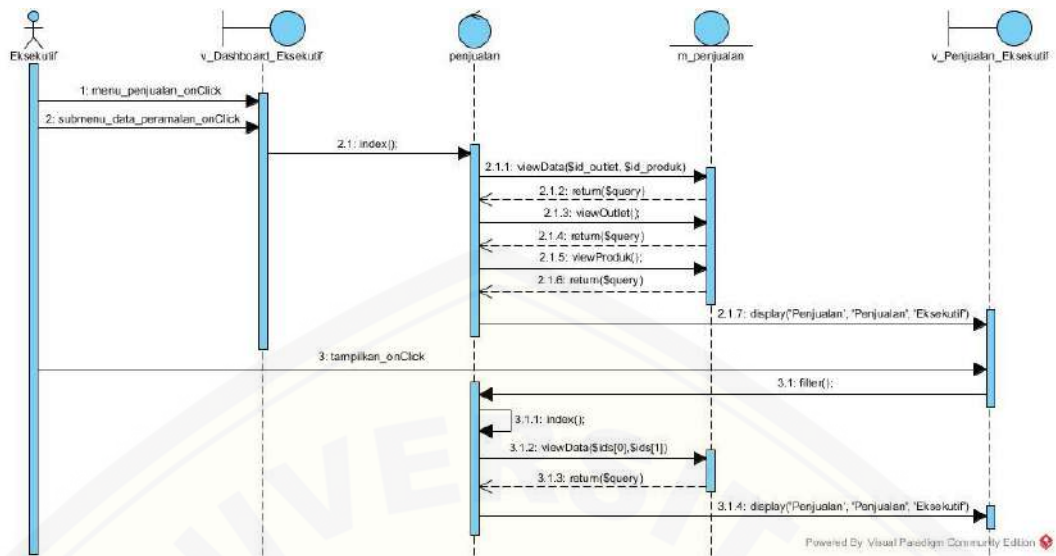
B.5 Sequence Diagram Melihat Data Outlet

Penggambaran *sequence diagram* melihat data *outlet* digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.13.



Gambar B. 13 Sequence Diagram Melihat Data Outlet

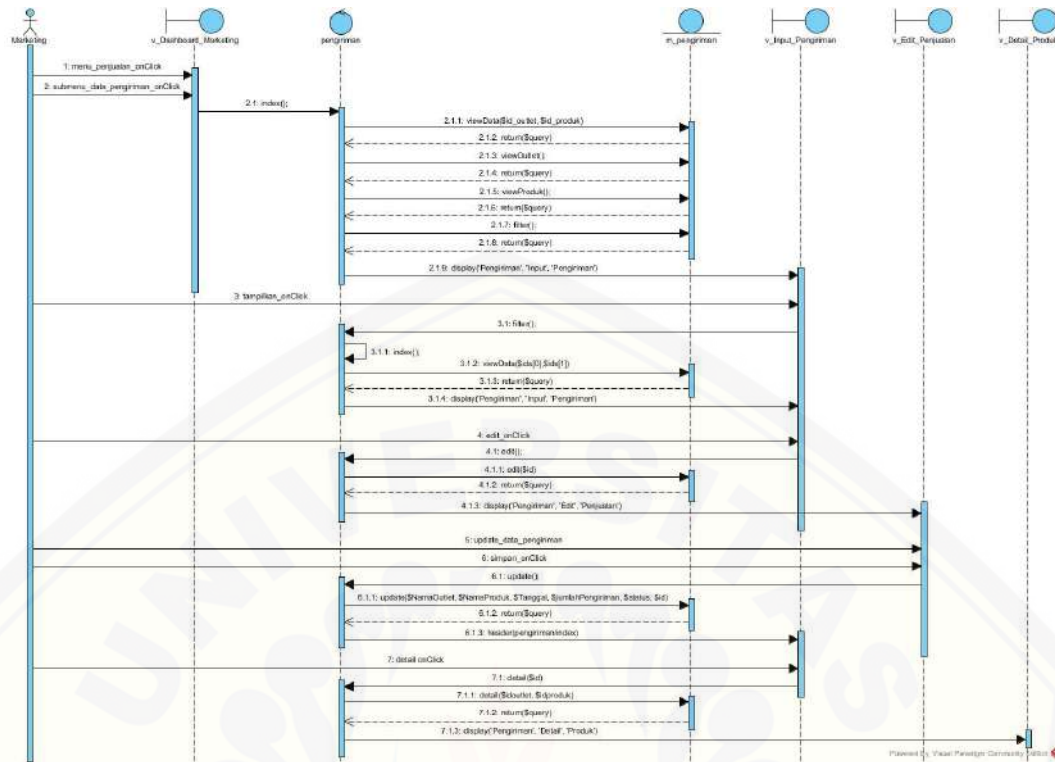
Penggambaran *sequence diagram* melihat data hasil peramalan digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.14.



Gambar B. 14 *Sequence Diagram* Melihat Data Hasil Peramalan

B.7 *Sequence Diagram* Pengelolaan Data Pendistribusian

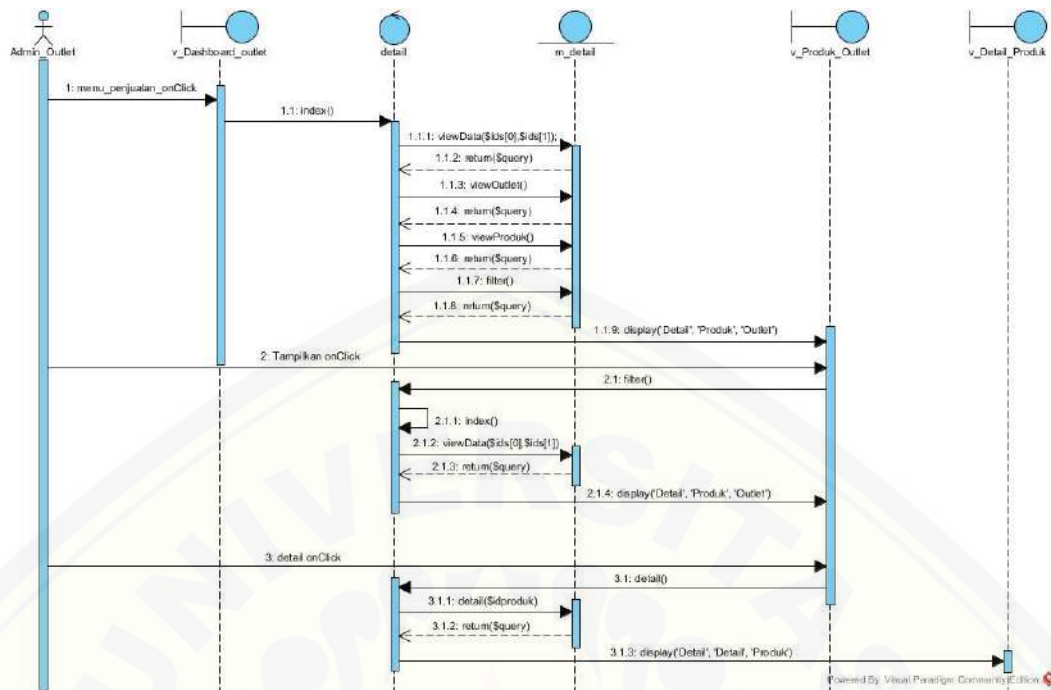
Penggambaran *sequence diagram* pengelolaan data pendistribusian produk digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.15.



Gambar B. 15 *Sequence Diagram* Pengelolaan Data Pendistribusian Produk

B.8 *Sequence Diagram* Melihat Detail Hasil Penjualan dan Sisa Produk

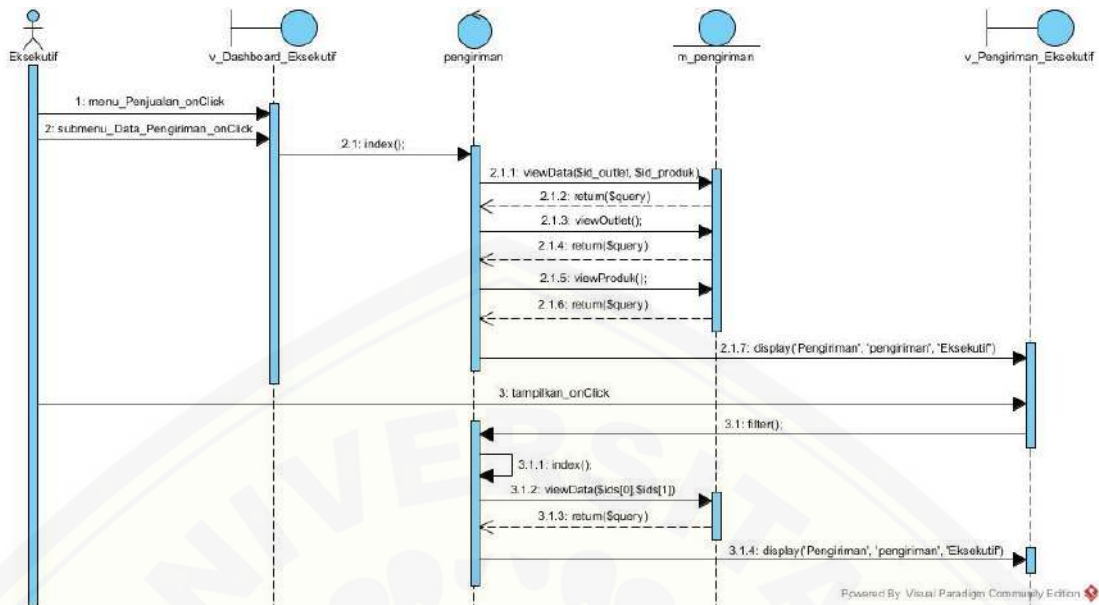
Penggambaran *sequence diagram* melihat detail hasil penjualan dan sisa produk digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.16.



Gambar B. 16 *Sequence Diagram* Melihat Detail Hasil Penjualan dan Sisa Produk

B.9 *Sequence Diagram* Melihat Data Hasil Penjualan dan Pendistribusian Produk

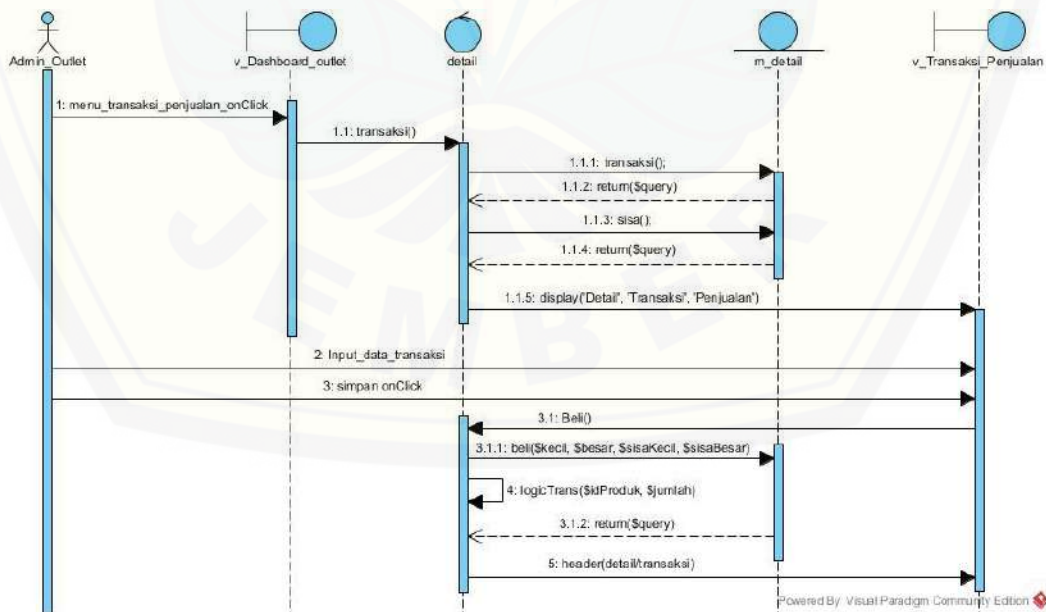
Penggambaran *sequence diagram* melihat data hasil penjualan dan pendistribusian produk digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.17.



Gambar B. 17 Sequence Diagram Melihat Data Hasil Penjualan dan Pendistribusian Produk

B.10 Sequence Diagram Input Transaksi Penjualan

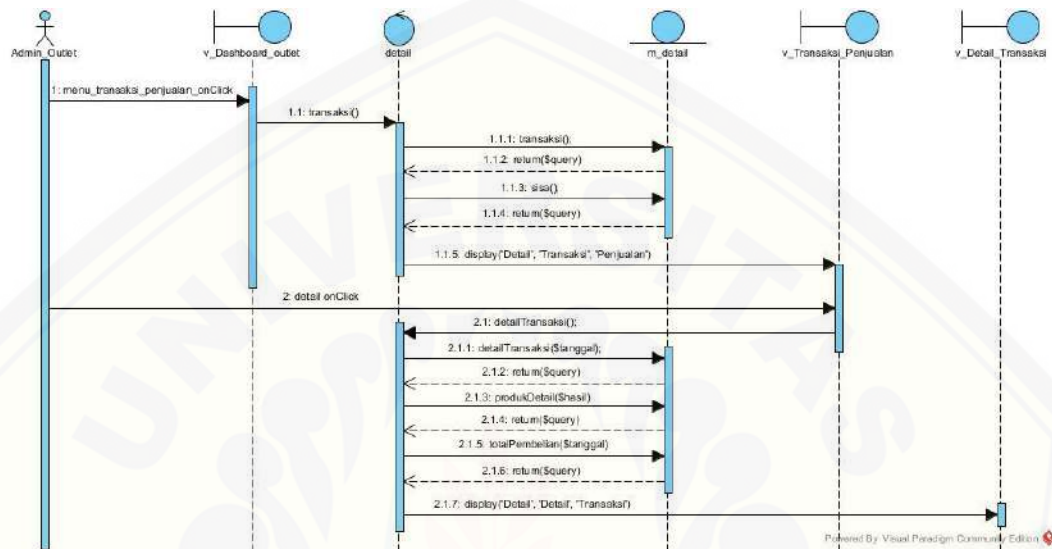
Penggambaran *sequence diagram* input transaksi penjualan digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.18.



Gambar B. 18 Sequence Diagram Input Transaksi Penjualan

B.11 Sequence Diagram Melihat Rekap Hasil Transaksi Penjualan

Penggambaran *sequence diagram* melihat rekap hasil transaksi penjualan digunakan untuk menjelaskan fungsi atau method yang akan dibuat seperti yang ditunjukkan pada gambar B.19.

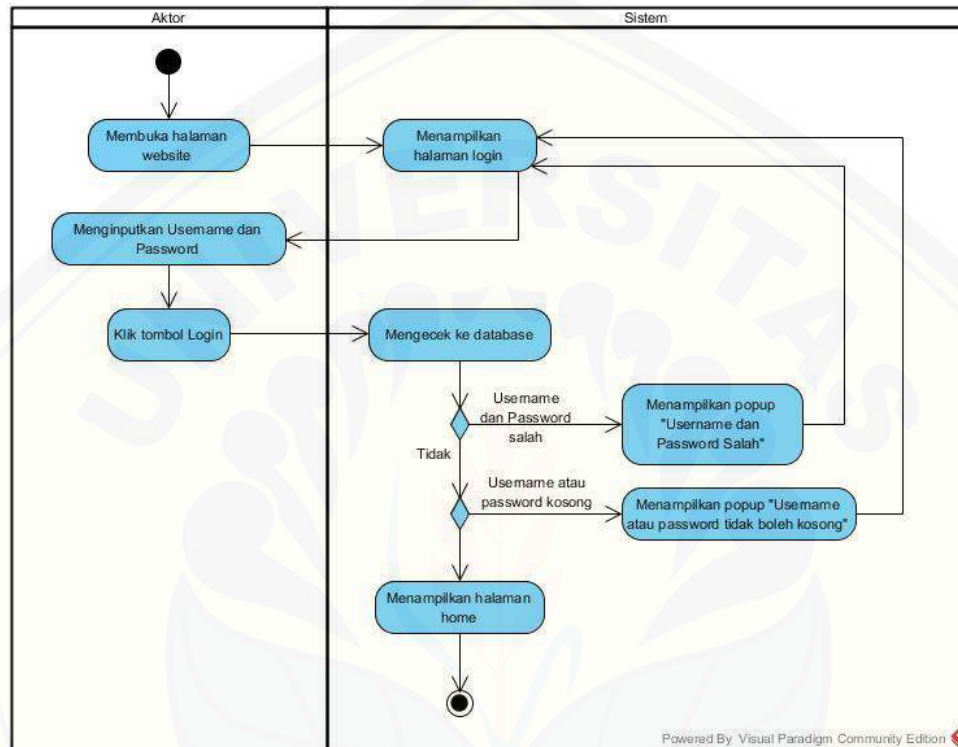


Gambar B. 19 *Sequence Diagram* Melihat Rekap Hasil Transaksi Penjualan

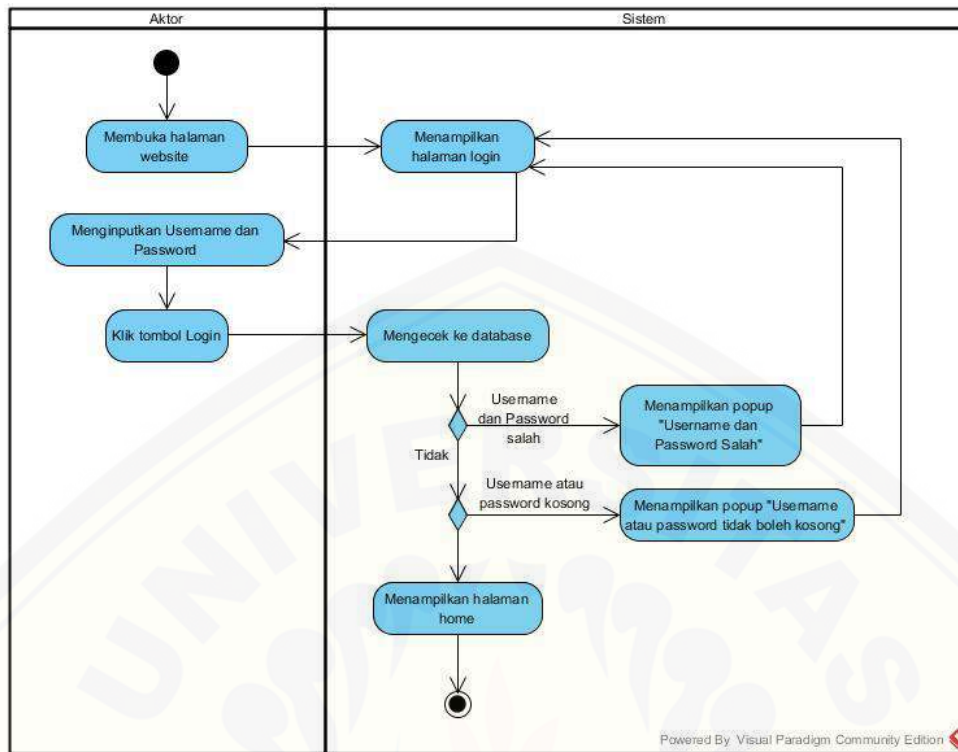
C. Activity Diagram

C.1 Activity Diagram Login

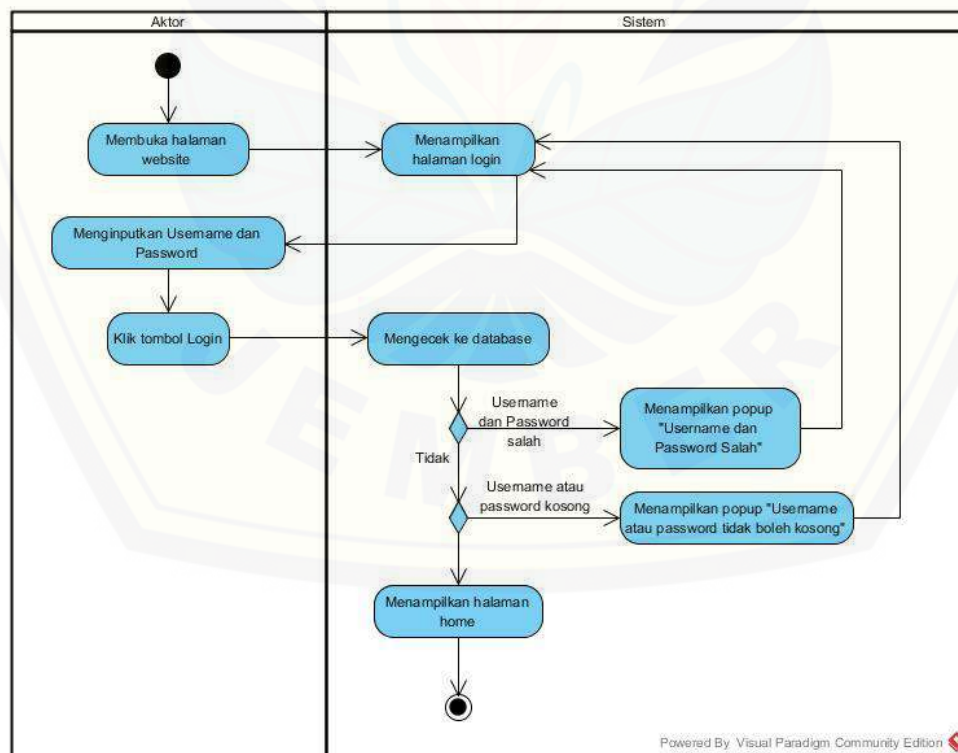
Activity diagram login dapat dilihat pada pada gambar C.1 sampai dengan gambar C.9.



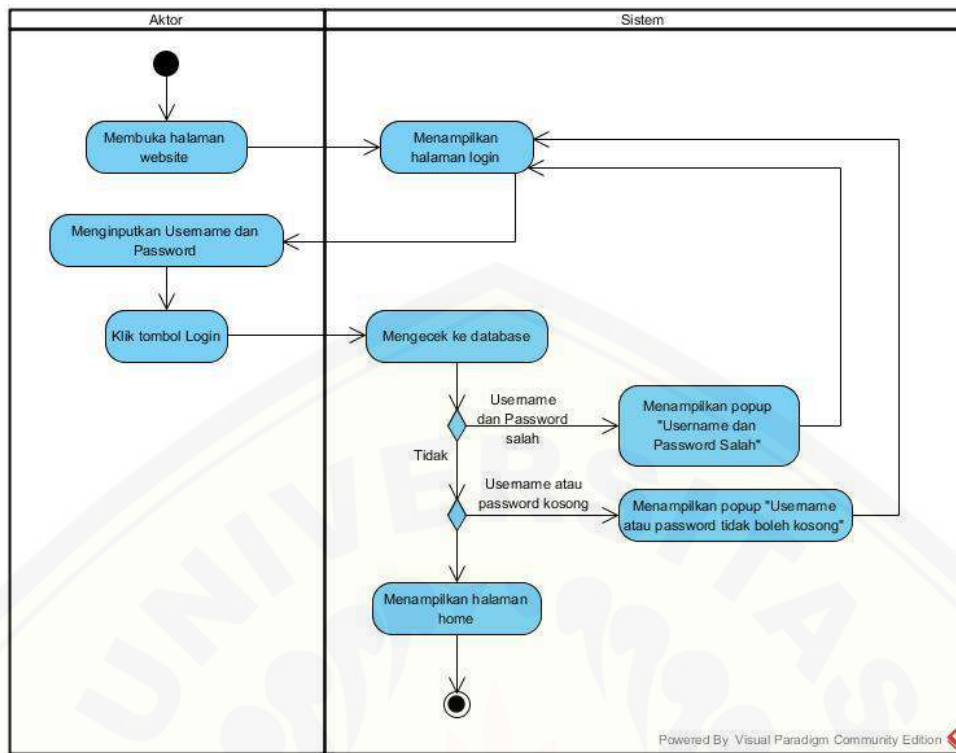
Gambar C. 1 Activity Diagram Login Admin



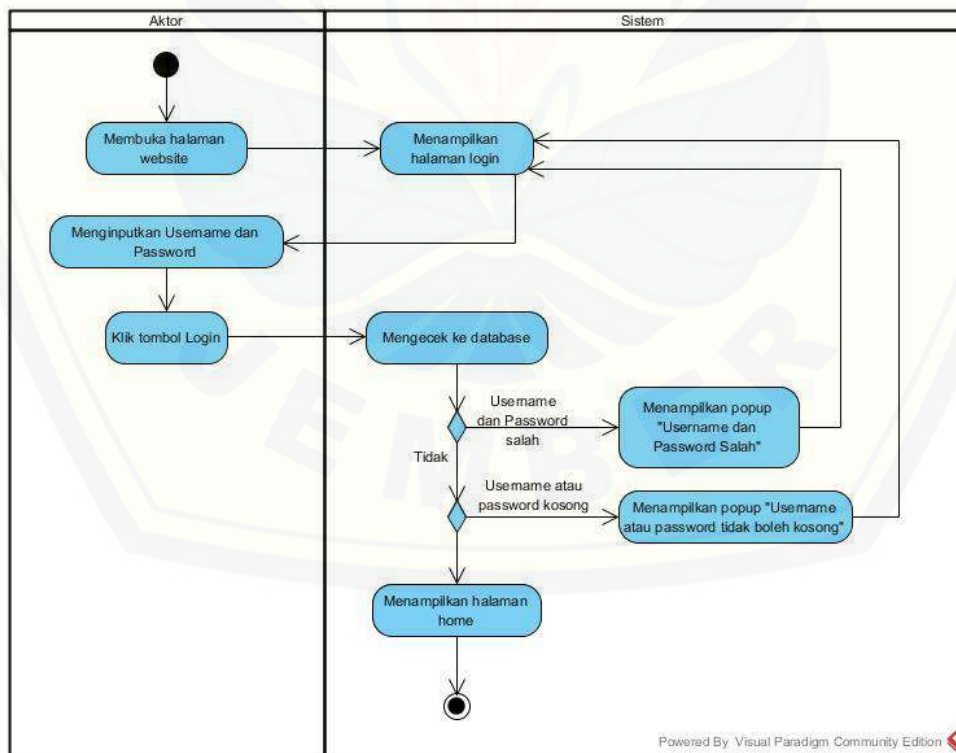
Gambar C. 2 Activity Diagram Login Marketing



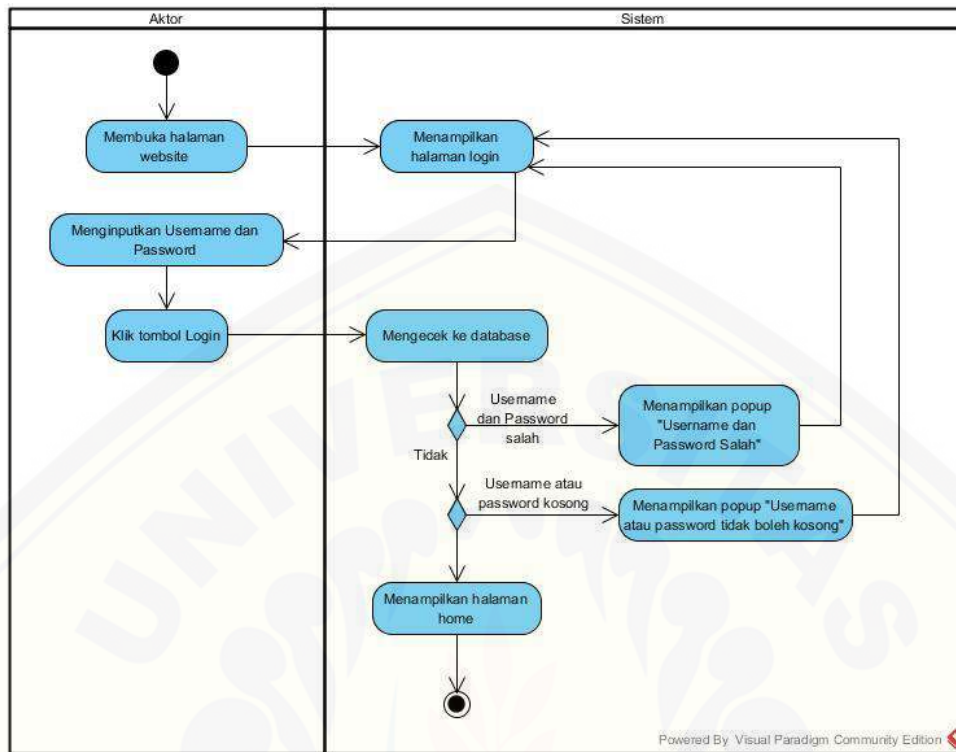
Gambar C. 3 Activity Diagram Login Eksekutif



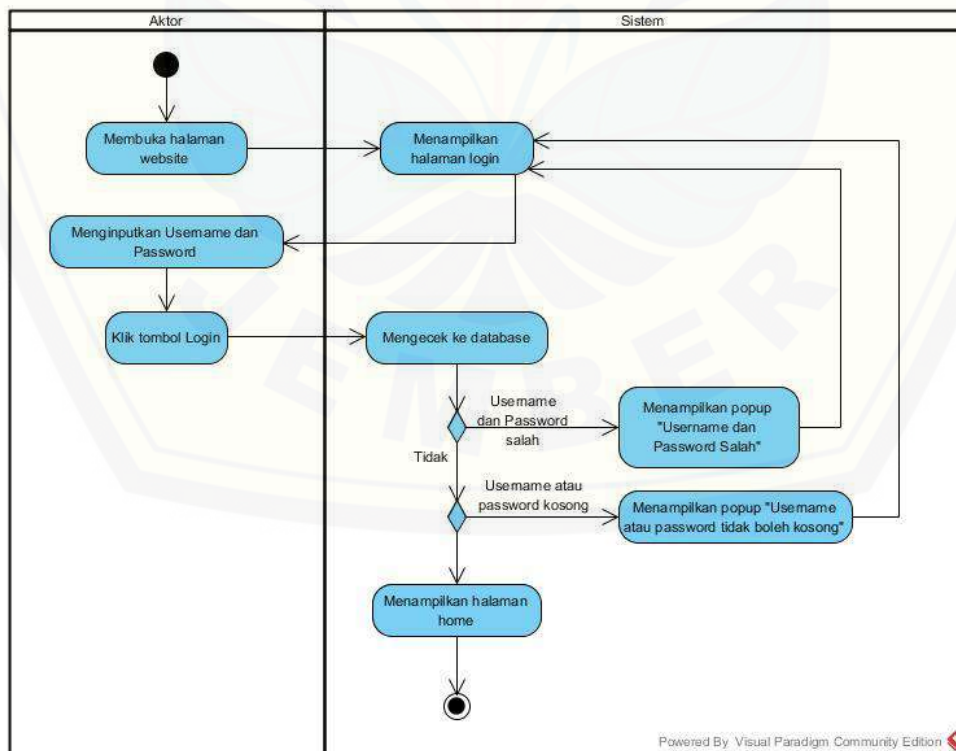
Gambar C. 4 Activity Diagram Login Admin Outlet A



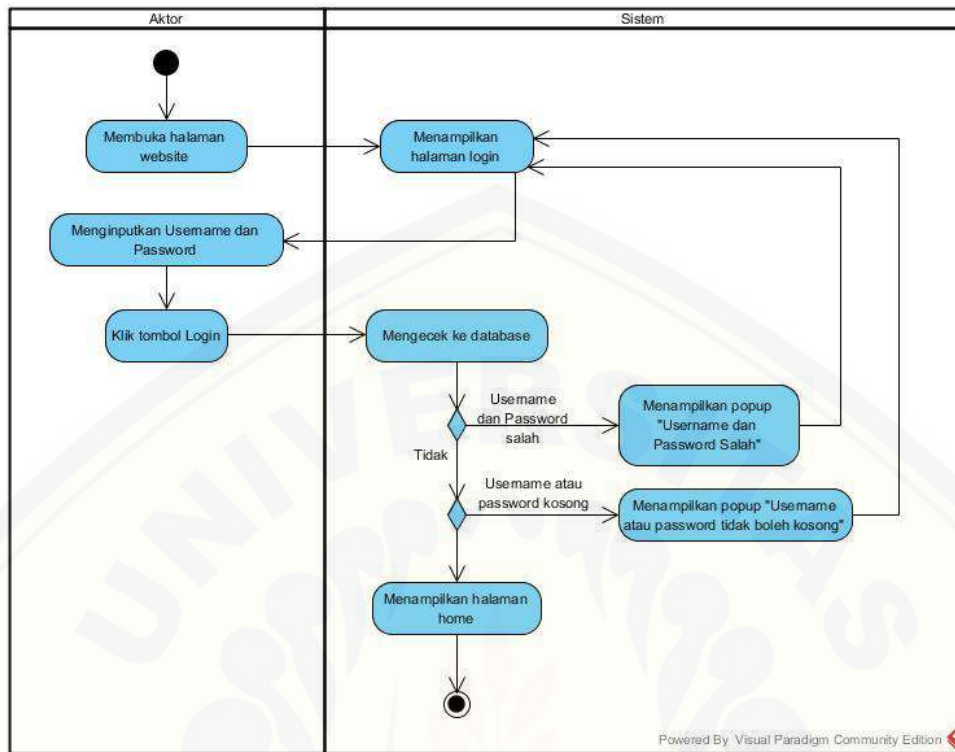
Gambar C. 5 Activity Diagram Login Admin Outlet B



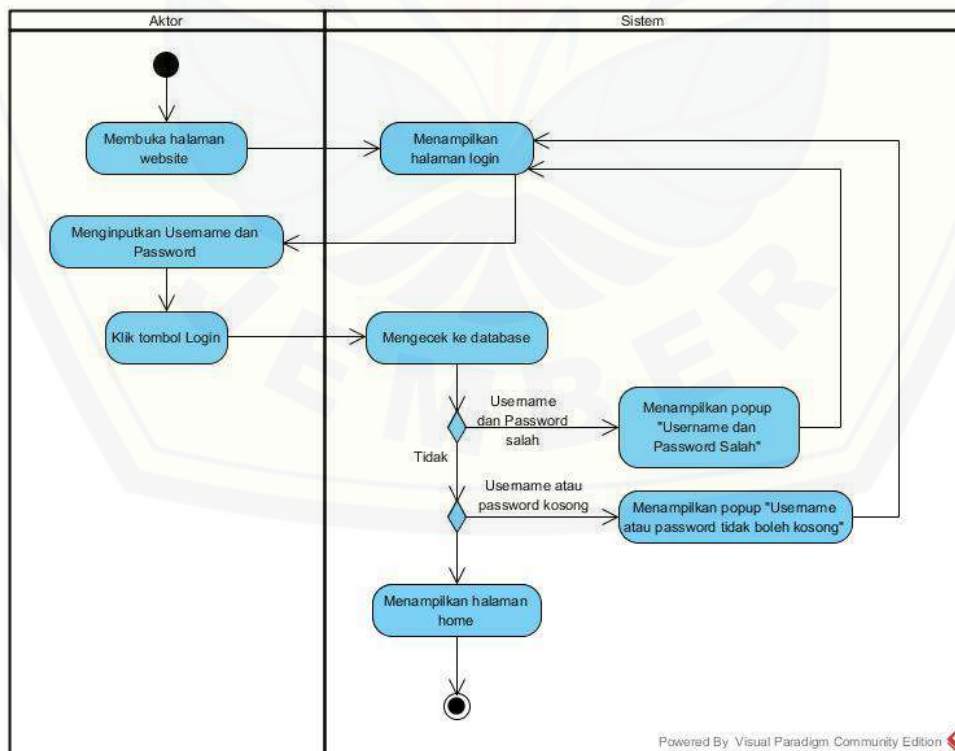
Gambar C. 6 Activity Diagram Login Admin Outlet C



Gambar C. 7 Activity Diagram Login Admin Outlet C



Gambar C. 8 Activity Diagram Login Admin Outlet E

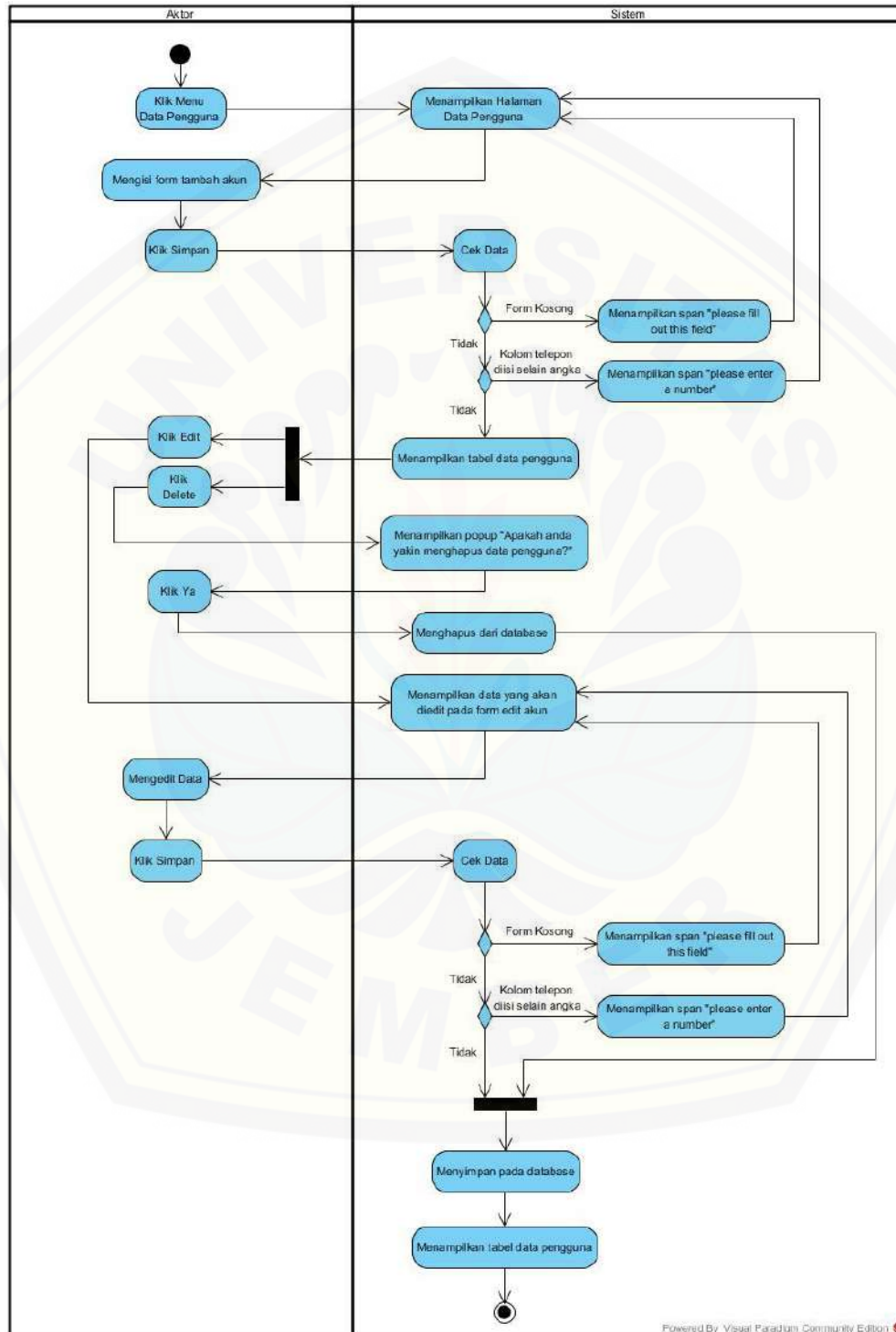


Gambar C. 9 Activity Diagram Login Admin Outlet F

C.2 Activity Diagram Pengelolaan Data Pengguna

Activity diagram pengelolaan data pengguna dapat dilihat pada pada gambar

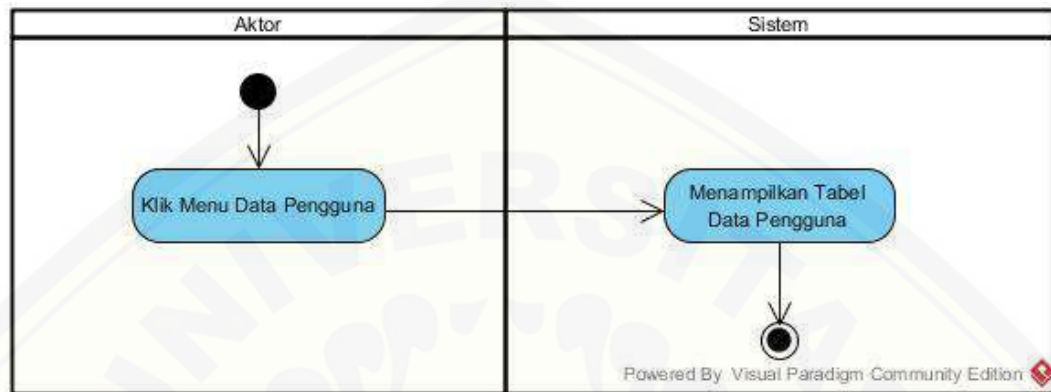
C.10.



Gambar C. 10 Activity Diagram Pengelolaan Data Pengguna

C.3 *Activity Diagram* Melihat Data Pengguna

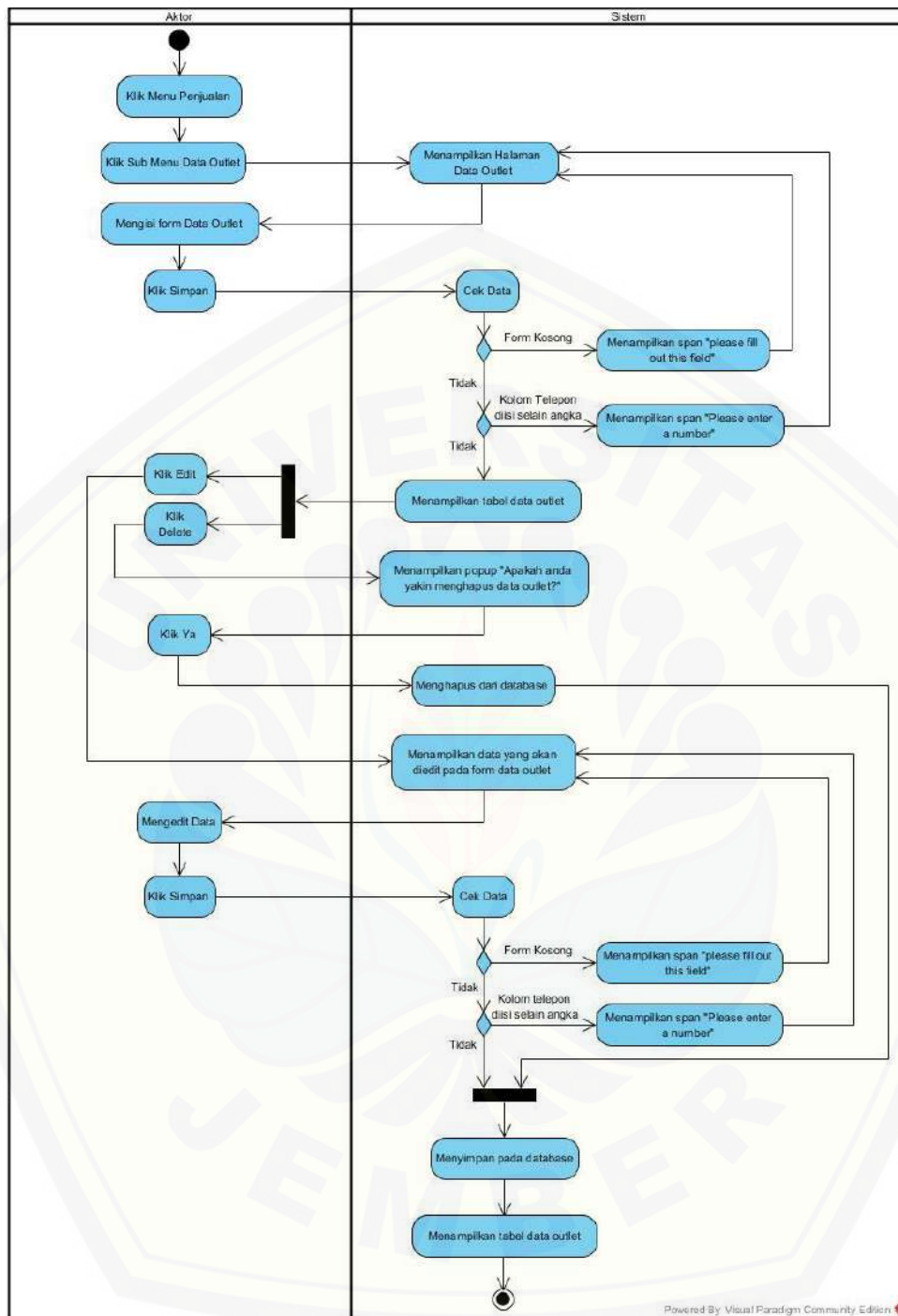
Activity diagram melihat data pengguna dapat dilihat pada pada gambar C.11.



Gambar C. 11 *Activity Diagram* Melihat Data Pengguna

C.4 *Activity Diagram* Pengelolaan Data Outlet

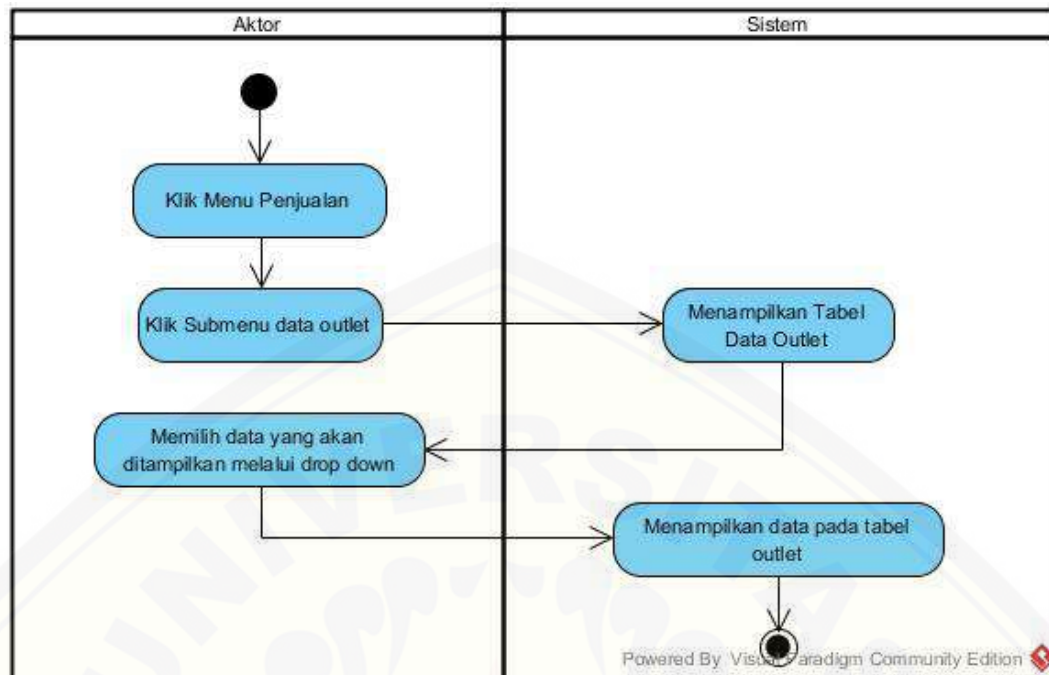
Activity diagram pengelolaan data outlet dapat dilihat pada pada gambar C.12.



Gambar C. 12 Activity Diagram Pengelolaan Data Outlet

C.5 Activity Diagram Melihat Data Outlet

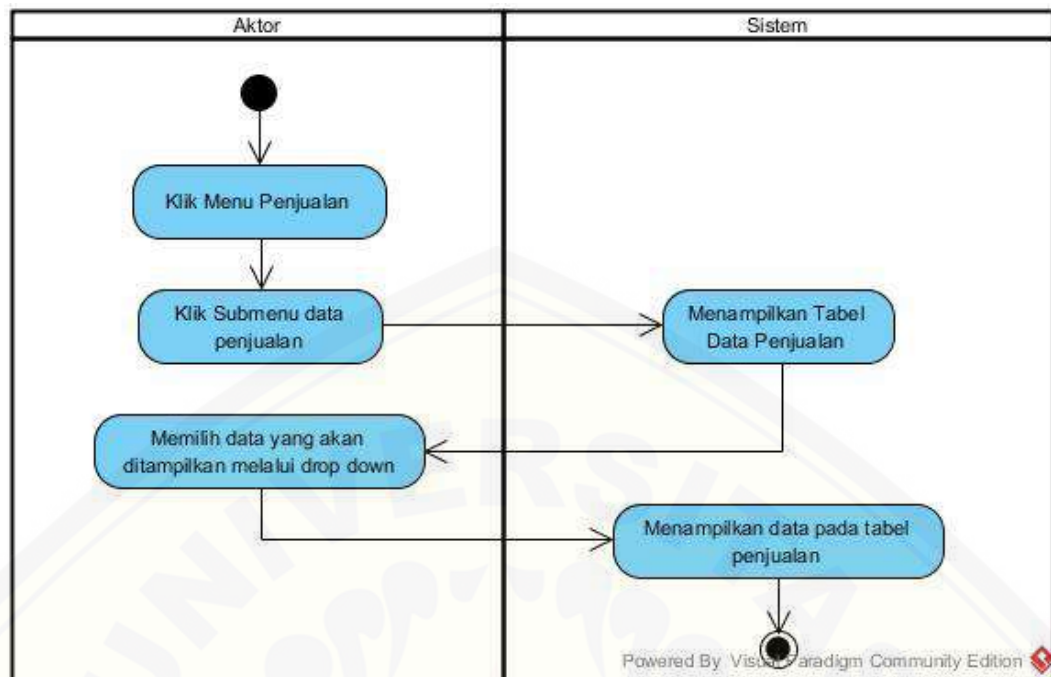
Activity diagram melihat data outlet dapat dilihat pada pada gambar C.13.



Gambar C. 13 *Activity Diagram* Melihat Data Outlet

C.6 *Activity Diagram* Melihat Data Hasil Peramalan

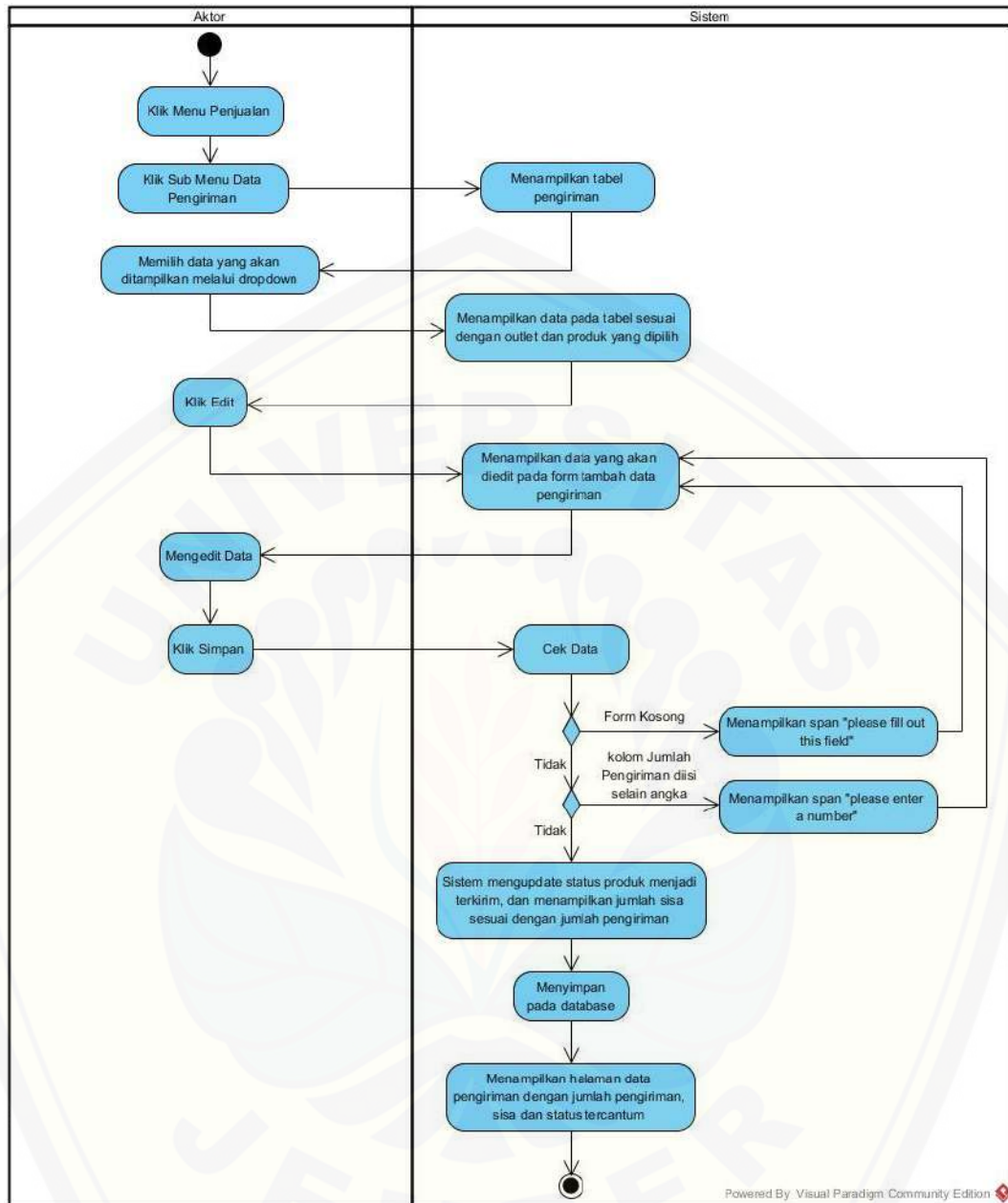
Activity diagram melihat data hasil peramalan dapat dilihat pada pada gambar C.14.



Gambar C. 14 *Activity Diagram* Melihat Data Hasil Peramalan

C.7 *Activity Diagram* Pengelolaan Data Pendistribusian Produk

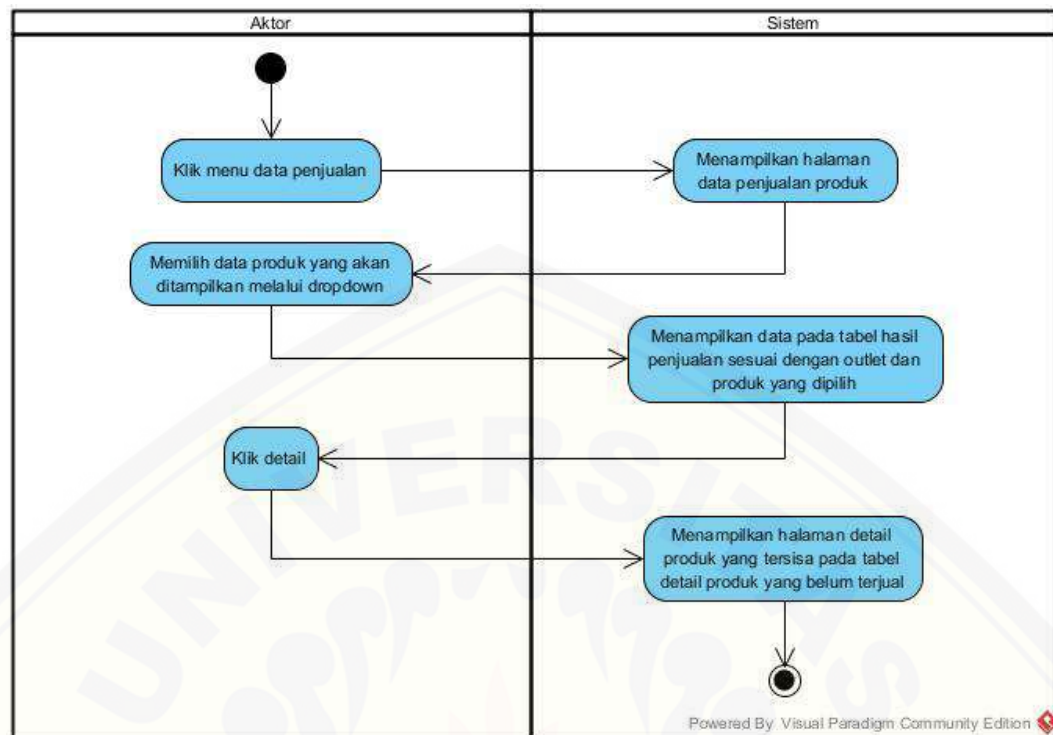
Activity diagram pengelolaan data pendistribusian produk dapat dilihat pada pada gambar C.15.



Gambar C. 15 Activity Diagram Pengelolaan Data Pendistribusian Produk

C.8 Activity Diagram Melihat Detail Hasil Penjualan dan Sisa Produk

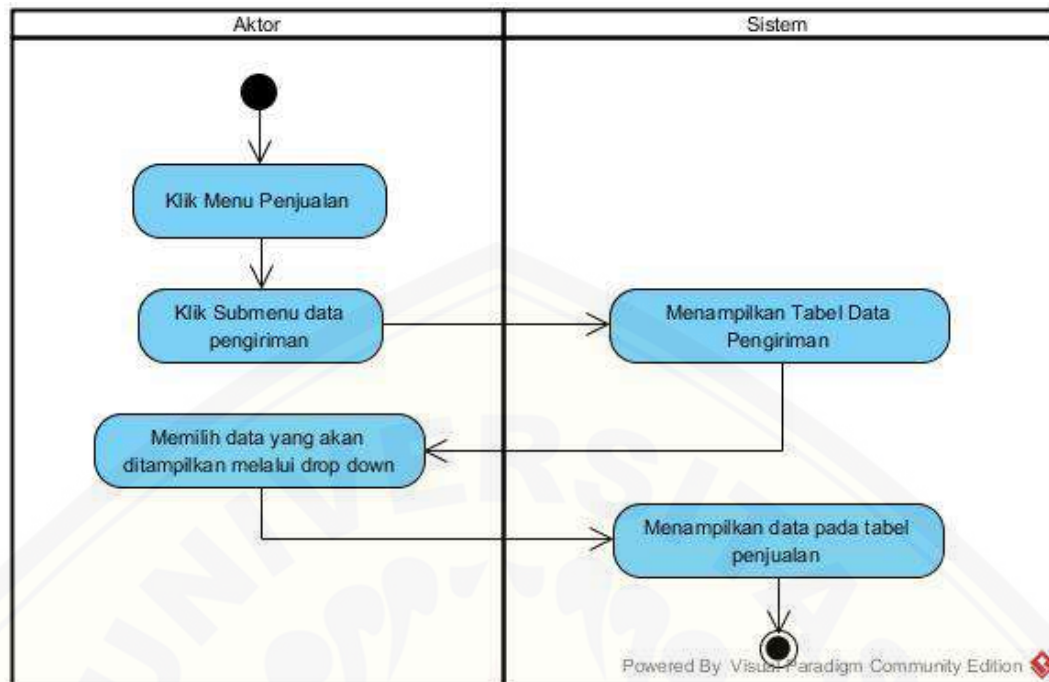
Activity diagram melihat detail hasil penjualan dan sisa produk dapat dilihat pada pada gambar C.16.



Gambar C. 16 Activity Diagram Melihat Detail Hasil Penjualan dan Sisa Produk

C.9 Activity Diagram Melihat Data Hasil Penjualan dan Pendistribusian Produk

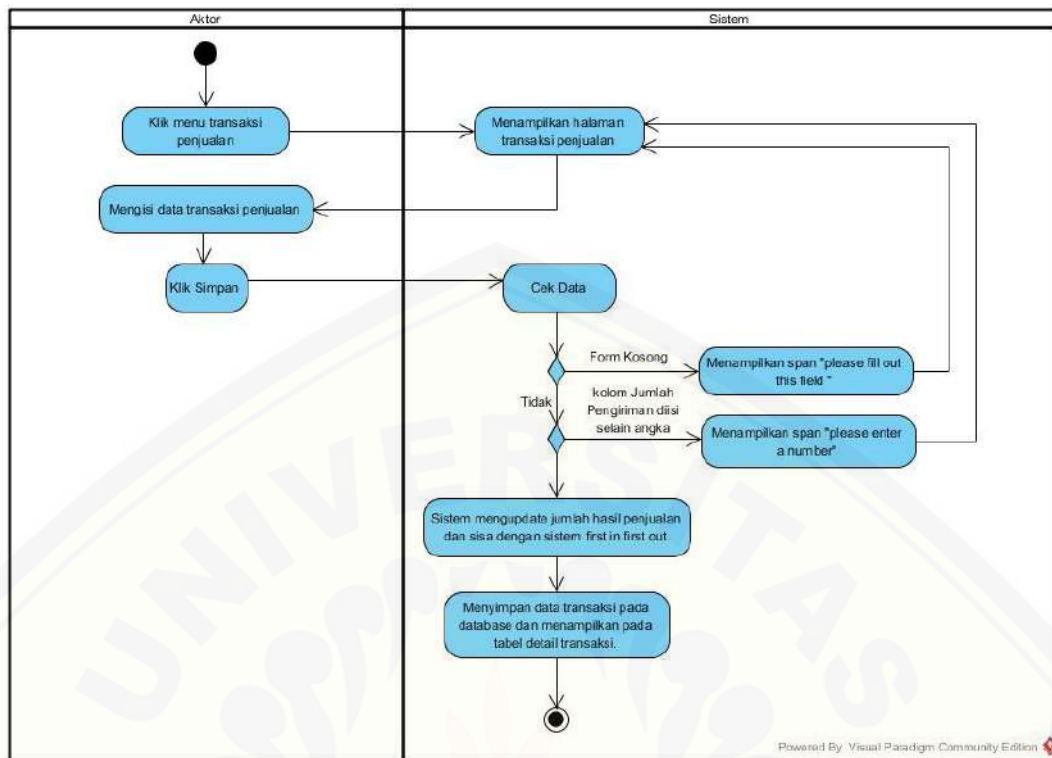
Activity diagram melihat data hasil penjualan dan pendistribusian produk dapat dilihat pada pada gambar C.17.



Gambar C. 17 *Activity Diagram* Melihat Data Hasil Penjualan dan Pendistribusian Produk

C.10 *Activity Diagram* Input Transaksi Penjualan

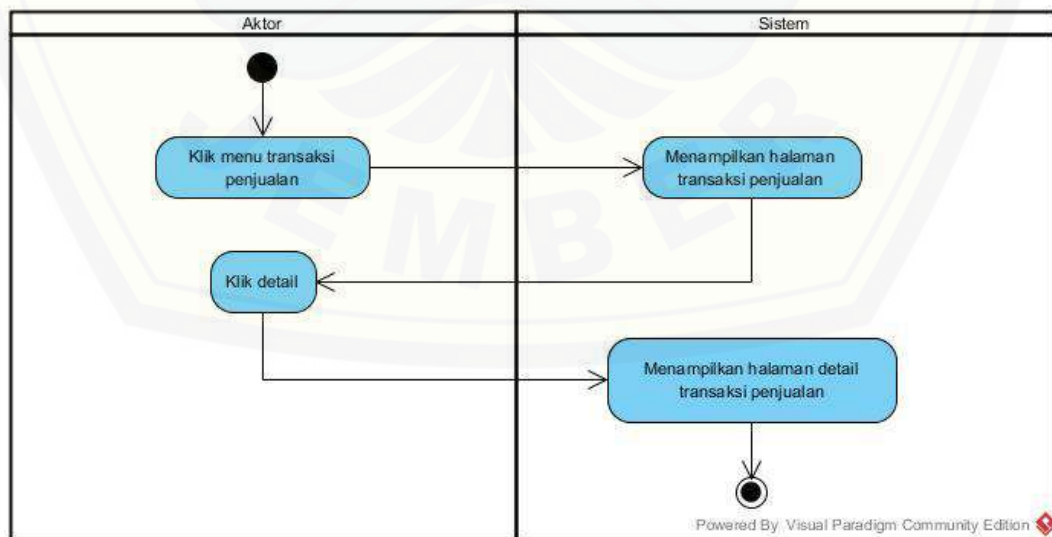
Activity diagram input transaksi penjualan dapat dilihat pada pada gambar C.18.



Gambar C. 18 Activity Diagram Input Transaksi Penjualan

C.11 Activity Diagram Melihat Rekap Hasil Transaksi Penjualan

Activity diagram melihat rekap hasil transaksi penjualan dapat dilihat pada pada gambar C.19.



Gambar C. 19 Activity Diagram Melihat Rekap Hasil Transaksi Penjualan

D. Kode Program

D.1 Kode Program *Login*

Kode Program *login* terletak pada class *m_login* dan *c_login*.

1. Kelas *m_login*

Penulisan kode program *m_login* dapat dilihat pada tabel D.1.

Tabel D. 1 Kode Program *Function Login*

1.	<code>function login(\$username, \$password){</code>
2.	<code> \$stmt = \$this->db->prepare("SELECT * FROM user u JOIN</code>
3.	<code>jabatan j ON u.id_Jabatan = j.id_Jabatan</code>
4.	<code> WHERE username = '\$username' AND password = '\$password'</code>
5.	<code> LIMIT 1");</code>
6.	
7.	<code> \$stmt->execute();</code>
8.	
9.	<code> \$result = \$stmt->fetch(PDO::FETCH_ASSOC);</code>
10.	
11.	<code> if(\$stmt->rowCount() > 0){</code>
12.	<code> return array('cek' => true, 'data' => \$result);</code>
13.	<code> } else {</code>
14.	<code> echo "";</code>
15.	<code> }</code>
16.	<code>}</code>

2. Kelas *c_login*

Penulisan kode program *c_login* dapat dilihat pada tabel D.2 sampai dengan D.5.

Tabel D. 2 Kode Program *Function Construct*

1.	<code>function __construct() {</code>
2.	<code> parent::__construct();</code>
3.	<code> \$this->model = \$this->model->load('login');</code>
4.	
5.	<code> \$logged = session::get('loggedIn');</code>
6.	<code> if (\$logged == true) {</code>
7.	<code> \$this->redirect('dashboard');</code>
8.	<code> }</code>
9.	<code>}</code>

Tabel D. 3 Kode Program *Function Index*

1.	function index() {
2.	\$this->view->display_login();
3.	}

Tabel D. 4 Kode Program *Function Login*

1.	function login() {
2.	if (isset(\$_POST['submit'])) {
3.	\$username = \$_POST['username'];
4.	\$password = \$_POST['password'];
5.	
6.	\$hasil = \$this->model->login(\$username, \$password);
7.	\$data = \$hasil['data'];
8.	
9.	if (\$hasil['cek'] == true) {
10.	session::set('username', \$username);
11.	session::set('jabatan', \$data['namaJabatan']);
12.	session::set('idoutlet', \$data['id_outlet']);
13.	session::set('loggedIn', true);
14.	\$this->redirect('dashboard');
15.	} else {
16.	echo "username dan password salah";
17.	}
18.	} else {
19.	echo "error";
20.	}
21.	}

Tabel D. 5 Kode Program *Function Logout*

1.	function logout() {
2.	session::destroy();
3.	header('Location: ' . URL . ');
4.	exit;
5.	}

D.2 Kode Program Pengelolaan Data Pengguna

1. Kelas m_register

Penulisan kode program m_register dapat dilihat pada tabel D.6 sampai dengan D.13.

Tabel D. 6 Kode Program *Function Register*

1.	function register(\$name, \$alamat, \$telepon, \$username, \$password,
2.	\$status, \$jabatan, \$outlet){
3.	\$stmt = \$this->db->prepare("INSERT INTO user (id_user,
4.	namaLengkap, alamat, telepon, username, password, id_status,
5.	id_Jabatan, id_outlet)
6.	VALUES (NULL, :namaLengkap, :alamat, :telepon, :username,
7.	:password, :id_status, :id_Jabatan, :id_outlet)");
8.	\$stmt->bindParam(':namaLengkap', \$name, PDO::PARAM_STR);
9.	\$stmt->bindParam(':alamat', \$alamat, PDO::PARAM_STR);
10.	\$stmt->bindParam(':telepon', \$telepon, PDO::PARAM_STR);
11.	\$stmt->bindParam(':username', \$username, PDO::PARAM_STR);
12.	\$stmt->bindParam(':password', \$password, PDO::PARAM_STR);
13.	\$stmt->bindParam(':id_status', \$status, PDO::PARAM_STR);
14.	\$stmt->bindParam(':id_Jabatan', \$jabatan, PDO::PARAM_STR);
15.	\$stmt->bindParam(':id_outlet', \$outlet, PDO::PARAM_STR);
16.	\$stmt->execute();
17.	
18.	}

Tabel D. 7 Kode Program *Function TampilData*

1.	function tampilData(){
2.	\$dataPegawai = \$this->db->selectWhere('SELECT id_user,
3.	namaLengkap, alamat, telepon, username,password, namaStatus,
4.	namaJabatan
5.	FROM user u join jabatan j on u.id_Jabatan = j.id_Jabatan join
6.	status s on u.id_status = s.id_status ORDER BY id_user ASC ');
7.	return \$dataPegawai['data'];
8.	}

Tabel D. 8 Kode Program *Function getStatus*

1.	function getStatus(){
2.	\$status = \$this->db->selectWhere('SELECT namaStatus, id_status
3.	from status');
4.	return \$status['data'];
5.	}

Tabel D. 9 Kode Program *Function getJabatan*

1.	function getJabatan(){
2.	

3.	\$jabatan = \$this->db->selectWhere('SELECT namaJabatan,
4.	id_Jabatan from jabatan');
5.	return \$jabatan['data'];
	}

Tabel D. 10 Kode Program *Function* getOutlet

1.	function getOutlet(){
2.	\$outlet = \$this->db->selectWhere('SELECT NamaOutlet, id_outlet
3.	from outlet');
4.	return \$outlet['data'];
5.	}

Tabel D. 11 Kode Program *Function* edit

1.	function edit(\$id){
2.	\$editAkun = \$this->db->selectWhere('select * from user where
3.	id_user = :id_user', array ('id_user' => \$id));
4.	return \$editAkun['data'];
5.	}

Tabel D. 12 Kode Program *Function* Update

1.	function update(\$name, \$alamat, \$telepon, \$username, \$password,
2.	\$status, \$jabatan, \$outlet, \$id){
3.	\$this->db->update('UPDATE `user` SET `namaLengkap`=
4.	:nama, `alamat` = :alamat,
5.	`telepon` = :telepon, `username` = :user, `password` =
6.	:pass, `id_status` = :status, `id_Jabatan` = :jabatan, `id_outlet` = :outlet
7.	WHERE id_user = :id',
8.	array(
9.	'nama' => \$name,
10.	'alamat' => \$alamat,
11.	'telepon' => \$telepon,
12.	'user' => \$username,
13.	'pass' => \$password,
14.	'status' => \$status,
15.	'jabatan' => \$jabatan,
16.	'outlet' => \$outlet,
17.	'id' => \$id
18.));
19.	}

Tabel D. 13 Kode Program *Function delete*

1.	function delete(\$id){
2.	\$this->db->delete('Delete from user where id_user = :id_user',
3.	array('id_user' => \$id));
4.	}

2. Kelas akun

Penulisan kode program akun dapat dilihat pada tabel D.14 sampai dengan D.18.

Tabel D. 14 Kode Program *Function index*

1.	function index(\$id) {
2.	\$sids = split("-", \$id);
3.	\$this->view->display = \$this->register->tampilData();
4.	\$this->view->status = \$this->register->getStatus();
5.	\$this->view->jabatan = \$this->register->getJabatan();
6.	\$this->view->outlet = \$this->register->getOutlet();
7.	\$this->view->idstatus = \$sids[0];
8.	\$this->view->idjabatan = \$sids[1];
9.	\$jabatan = session::get('jabatan');
10.	
11.	if (\$jabatan == 'Admin') {
12.	\$this->view->display('Akun', 'Akun', 'Admin');
13.	} else if (\$jabatan == 'Eksekutif') {
14.	\$this->view->display('Akun', 'View', 'Akun');
15.	}
16.	
17.	}

Tabel D. 15 Kode Program *Function submit*

1.	function submit() {
2.	if (isset(\$_POST['submit'])) {
3.	\$name = \$_POST['name'];
4.	\$alamat = \$_POST['alamat'];
5.	\$telepon = \$_POST['telepon'];
6.	\$no = trim(\$_POST ['telepon']);
7.	if (is_numeric(\$no) == true){
8.	\$telepon = \$_POST['telepon'];
9.	\$username = \$_POST['username'];
10.	\$password = \$_POST['password'];
11.	\$status = \$_POST['status'];

12.	<code>\$jabatan = \$_POST['Jabatan'];</code>
13.	<code>if(isset(\$_POST['outlet']))){</code>
14.	<code> \$outlet = \$_POST['outlet'];</code>
15.	<code> }else{</code>
16.	<code> \$outlet = 0;</code>
17.	<code> }</code>
18.	<code> \$hasil = \$this->register->register(\$name, \$alamat, \$telepon,</code>
19.	<code> \$username, \$password, \$status, \$jabatan, \$outlet);</code>
20.	<code> } else {</code>
21.	<code> \$_SESSION['msg-err']['telepon'] = "anda harus memasukkan</code>
22.	<code> angka pada nomer telepon";</code>
23.	<code> \$_SESSION['post'] = \$_POST;</code>
24.	
25.	<code> }</code>
26.	<code> header('location:'.URL.'/akun/index/1-1');</code>
27.	<code> }</code>
28.	<code>}</code>

Tabel D. 16 Kode Program *Function* edit

1.	<code>function edit(\$id){</code>
2.	<code> \$this->view->akun = \$this->register->edit(\$id);</code>
3.	<code> \$this->view->status = \$this->register->getStatus();</code>
4.	<code> \$this->view->jabatan = \$this->register->getJabatan();</code>
5.	<code> \$this->view->outlet = \$this->register->getOutlet();</code>
6.	<code> \$this->view->display('Akun', 'Edit', 'Akun');</code>
7.	<code>}</code>

Tabel D. 17 Kode Program *Function* update

1.	<code>function update(){</code>
2.	<code> \$id = \$_POST['id'];</code>
3.	<code> \$name = \$_POST['name'];</code>
4.	<code> \$alamat = \$_POST['alamat'];</code>
5.	<code> \$telepon = \$_POST['telepon'];</code>
6.	<code> \$username = \$_POST['username'];</code>
7.	<code> \$password = \$_POST['password'];</code>
8.	<code> \$status = \$_POST['status'];</code>
9.	<code> \$jabatan = \$_POST['Jabatan'];</code>
10.	<code> if(isset(\$_POST['outlet']))){</code>
11.	<code> \$outlet = \$_POST['outlet'];</code>
12.	<code> }else{</code>
13.	<code> \$outlet = 0;</code>
14.	<code> }</code>
15.	<code>}</code>

16.	<code>\$hasil = \$this->register->update(\$name, \$alamat, \$telepon,</code>
17.	<code>\$username, \$password, \$status, \$jabatan, \$outlet, \$id);</code>
18.	<code>header('location:'.URL.'/akun/index/1-1');</code>
19.	<code>}</code>

Tabel D. 18 Kode Program *Function delete*

1.	<code>function delete(\$id){</code>
2.	<code>\$this->register->delete(\$id);</code>
3.	<code>header('location:'.URL.'/akun/index/1-1');</code>
4.	<code>}</code>

D.3 Kode Program Pengelolaan Data *Outlet*

1. Kelas `m_outlet`

Penulisan kode program `m_register` dapat dilihat pada tabel D.19 sampai dengan D.23.

Tabel D. 19 Kode Program *Function Register*

1.	<code>function register(\$name, \$alamat, \$telepon){</code>
2.	<code>\$stmt = \$this->db->prepare("INSERT INTO outlet (id_outlet,</code>
3.	<code>NamaOutlet, alamatOutlet, telepon)</code>
4.	<code>VALUES (NULL, :NamaOutlet, :alamatOutlet, :telepon));</code>
5.	<code>\$stmt->bindParam(':NamaOutlet', \$name, PDO::PARAM_STR);</code>
6.	<code>\$stmt->bindParam(':alamatOutlet', \$alamat, PDO::PARAM_STR);</code>
7.	<code>\$stmt->bindParam(':telepon', \$telepon, PDO::PARAM_STR);</code>
8.	<code>\$stmt->execute();</code>
9.	
10.	<code>}</code>

Tabel D. 20 Kode Program *Function getData*

1.	<code>function getData(){</code>
2.	<code>\$query = \$this->db->selectAll("SELECT * FROM outlet ORDER</code>
3.	<code>BY id_outlet ASC");</code>
4.	<code>return \$query;</code>
5.	<code>}</code>

Tabel D. 21 Kode Program *Function edit*

1.	<code>function edit(\$id){</code>
2.	

3.	\$editOutlet = \$this->db->selectWhere('select * from outlet where
4.	id_outlet = :id_outlet', array ('id_outlet' => \$id));
5.	return \$editOutlet['data'];
	}

Tabel D. 22 Kode Program *Function update*

1.	function update(\$name, \$alamat, \$telepon, \$id){
2.	\$this->db->update('UPDATE `outlet` SET `NamaOutlet`=
3.	:nama,`alamatOutlet`= :alamat,`telepon`= :telepon
4.	WHERE id_outlet = :id',
5.	array(
6.	'nama' => \$name,
7.	'alamat' => \$alamat,
8.	'telepon' => \$telepon,
9.	'id' => \$id
10.));
11.	
12.	}

Tabel D. 23 Kode Program *Function delete*

1.	function delete(\$id){
2.	\$this->db->delete('Delete from outlet where id_outlet = :id_outlet',
3.	array('id_outlet' => \$id));
4.	}

2. Kelas outlet

Penulisan kode program outlet dapat dilihat pada tabel D.24 sampai dengan D.28.

Tabel D. 24 Kode Program *Function index*

1.	function index() {
2.	\$this->view->display = \$this->Model->getData();
3.	\$jabatan = session::get('jabatan');
4.	
5.	if (\$jabatan == 'Marketing') {
6.	\$this->view->display('Outlet', 'Outlet', 'Marketing');
7.	} else if (\$jabatan == 'Eksekutif') {
8.	\$this->view->display('Outlet', 'Outlet', 'Eksekutif');
9.	}
10.	}

Tabel D. 25 Kode Program *Function submit*

1.	function submit() {
2.	if (isset(\$_POST['submit'])) {
3.	\$name = \$_POST['nama-outlet'];
4.	\$alamat = \$_POST['alamat-outlet'];
5.	\$telepon = \$_POST['phone'];
6.	\$hasil = \$this->Model->register(\$name, \$alamat, \$telepon);
7.	header('location:'.URL.'/Outlet/index');
8.	}else{
9.	echo "error";
10.	}
11.	}

Tabel D. 26 Kode Program *Function edit*

1.	function edit(\$id){
2.	\$this->view->outlet = \$this->Model->edit(\$id);
3.	\$this->view->display('Outlet', 'Edit', 'Outlet');
4.	}

Tabel D. 27 Kode Program *Function update*

1.	function update(){
2.	\$id = \$_POST['id'];
3.	\$name = \$_POST['name'];
4.	\$alamat = \$_POST['alamat'];
5.	\$telepon = \$_POST['telepon'];
6.	\$hasil = \$this->Model->update(\$name, \$alamat, \$telepon, \$id);
7.	header('location:'.URL.'/outlet/index');
8.	}

Tabel D. 28 Kode Program *Function delete*

1.	function delete(\$id){
2.	\$this->Model->delete(\$id);
3.	header('location:'.URL.'/Outlet/index');
4.	}

D.4 Kode Program Pengelolaan Data Peramalan

1. Kelas m_penjualan

Penulisan kode program m_penjualan dapat dilihat pada tabel D.29 sampai dengan D.31.

Tabel D. 29 Kode Program *Function* viewData

1.	function viewData(\$id_outlet, \$id_produk){
2.	\$dataPenjualan = \$this->db->selectWhere("SELECT * from
3.	penjualan where id_outlet = \$id_outlet and id_produk = \$id_produk");
4.	return \$dataPenjualan['data'];
5.	}

Tabel D. 30 Kode Program *Function* viewOutlet

1.	function viewOutlet(){
2.	\$dataOutlet = \$this->db->selectWhere('SELECT NamaOutlet,
3.	id_outlet from outlet');
4.	return \$dataOutlet['data'];
5.	}

Tabel D. 31 Kode Program *Function* viewProduk

1.	function viewProduk(){
2.	\$dataProduk = \$this->db->selectWhere('SELECT NamaProduk,
3.	id_produk from product');
4.	return \$dataProduk['data'];
5.	}

2. Kelas penjualan

Penulisan kode program penjualan dapat dilihat pada tabel D.32 sampai dengan D.34.

Tabel D. 32 Kode Program *Function* index

1.	function index(\$id) {
2.	\$sids = split("-", \$id);
3.	\$this->view->penjualan = \$this->tambah-
4.	>viewData(\$sids[0], \$sids[1]);
5.	\$this->view->outlet = \$this->tambah->viewOutlet();
6.	\$this->view->produk = \$this->tambah->viewProduk();
7.	\$this->view->idoutlet = \$sids[0];
8.	\$this->view->idproduk = \$sids[1];
9.	
10.	\$jabatan = session::get('jabatan');

11.	
12.	if (\$jabatan == 'Marketing') {
13.	\$this->view->display('Penjualan', 'Penjualan', 'Marketing');
14.	} else if (\$jabatan == 'Eksekutif') {
15.	\$this->view->display('Penjualan', 'Penjualan', 'Eksekutif');
16.	}
17.	}

Tabel D. 33 Kode Program Function filter

1.	function filter(){
2.	\$id = \$_POST['idoutlet'].'-'. \$_POST['idproduk'];
3.	\$this->index(\$id);
4.	}

Tabel D. 34 Kode Program *Function* tambah

1.	function tambah(){
2.	\$outlet = \$_POST['outlet'];
3.	\$produk = \$_POST['produk'];
4.	\$hasil = \$this->tambah->tambah(\$outlet, \$produk);
5.	header('location:'.URL.'/penjualan/index/1-1');
6.	}

D.5 Kode Program Pengelolaan Data Pendistribusian Produk

1. Kelas m_pengiriman

Penulisan kode program m_pengiriman dapat dilihat pada tabel D.35 sampai dengan D.40.

Tabel D. 35 Kode Program *Function* viewData

1.	function viewData(\$id_outlet, \$id_produk){
2.	\$dataPenjualan = \$this->db->selectWhere("SELECT * from
3.	penjualan p left join status_produk s on p.id_statusproduk =
4.	s.id_statusproduk where id_outlet = \$id_outlet and id_produk =
5.	\$id_produk order by tanggal");
6.	return \$dataPenjualan['data'];
7.	}

Tabel D. 36 Kode Program *Function* viewOutlet

1.	function viewOutlet(){
2.	\$dataOutlet = \$this->db->selectWhere('SELECT NamaOutlet,
3.	id_outlet from outlet');
4.	return \$dataOutlet['data'];
5.	}

Tabel D. 37 Kode Program *Function* viewProduk

1.	function viewProduk(){
2.	\$dataProduk = \$this->db->selectWhere('SELECT NamaProduk,
3.	id_produk from product');
4.	return \$dataProduk['data'];
5.	}

Tabel D. 38 Kode Program *Function* viewStatus

1.	function viewStatus(){
2.	\$dataStatus= \$this->db->selectWhere('SELECT namaStatus,
3.	id_statusproduk from status_produk');
4.	return \$dataStatus['data'];
5.	}

Tabel D. 39 Kode Program *Function* edit

1.	function edit(\$id){
2.	\$editOutlet = \$this->db->selectWhere('select * from penjualan p
3.	join product pr on p.id_produk = pr.id_produk
4.	join outlet o on p.id_outlet = o.id_outlet where id_penjualan =
5.	:id_penjualan',
6.	array ('id_penjualan' => \$id));
7.	return \$editOutlet['data'];
8.	}

Tabel D. 40 Kode Program *Function* update

1.	function update(\$jumlahPengiriman, \$status, \$id){
2.	\$this->db->update('UPDATE `penjualan` SET `jumlahPengiriman`
3.	= :jumlahPengiriman, `id_statusproduk` = :status, Sisa = :sisa

4.	WHERE id_penjualan = :id',
5.	array(
6.	'jumlahPengiriman' => \$jumlahPengiriman,
7.	'status' => \$status,
8.	'sisa' => \$jumlahPengiriman,
9.	'id' => \$id
10.));
11.	}

2. Kelas Pengiriman

Penulisan kode program pengiriman dapat dilihat pada tabel D.41 sampai dengan D.44.

Tabel D. 41 Kode Program *Function index*

1.	function index(\$id) {
2.	\$sids = split("-", \$id);
3.	\$this->view->pengiriman = \$this-> kirim-
4.	->viewData(\$sids[0], \$sids[1]);
5.	\$this->view->outlet = \$this-> kirim->viewOutlet();
6.	\$this->view->produk = \$this-> kirim->viewProduk();
7.	\$this->view->status = \$this-> kirim->viewStatus();
8.	\$this->view->idoutlet = \$sids[0];
9.	\$this->view->idproduk = \$sids[1];
10.	\$jabatan = session::get('jabatan');
11.	if (\$jabatan == 'Marketing') {
12.	\$this->view->display('Pengiriman', 'Input', 'Pengiriman');
13.	} else if (\$jabatan == 'Eksekutif') {
14.	\$this->view->display('Pengiriman', 'pengiriman', 'Eksekutif');
15.	}
16.	}

Tabel D. 42 Kode Program *Function filter*

1.	function filter(){
2.	\$id = \$_POST['idoutlet'].'-'.\$_POST['idproduk'];
3.	\$this->index(\$id);
4.	}

Tabel D. 43 Kode Program *Function edit*

1.	function edit(\$id){
2.	\$this->view->data = \$this-> kirim->edit(\$id);

3.	<code>\$this->view->display('Pengiriman', 'Edit', 'Penjualan');</code>
4.	<code>}</code>

Tabel D. 44 Kode Program *Function update*

1.	<code>function update(){</code>
2.	<code> \$id = \$_POST['id'];</code>
3.	<code> \$JumlahPengiriman = \$_POST['JumlahPengiriman'];</code>
4.	<code> \$status = 1;</code>
5.	<code> \$hasil = \$this->kirim->update(\$JumlahPengiriman, \$status, \$id);</code>
6.	<code> header('location:'.URL.'/pengiriman/index/1-1');</code>
7.	<code>}</code>

D.6 Kode Program Pengelolaan Data Pendistribusian Produk

1. Kelas `m_pengiriman`

Penulisan kode program `m_pengiriman` dapat dilihat pada tabel D.45 sampai dengan D.51.

Tabel D. 45 Kode Program *function viewData*

1.	<code>function viewData(\$id_outlet, \$id_produk){</code>
2.	<code> \$dataPenjualan = \$this->db->selectWhere("SELECT * from</code>
3.	<code>penjualan p left join status_produk s on p.id_statusproduk =</code>
4.	<code>s.id_statusproduk</code>
5.	<code> where id_outlet = \$id_outlet and id_produk = \$id_produk order</code>
6.	<code>by tanggal");</code>
7.	<code> return \$dataPenjualan['data'];</code>
8.	<code>}</code>

Tabel D. 46 Kode Program *Function ViewOutlet*

1.	<code>function viewOutlet(){</code>
2.	<code> \$dataOutlet = \$this->db->selectWhere('SELECT NamaOutlet,</code>
3.	<code>id_outlet from outlet');</code>
4.	<code> return \$dataOutlet['data'];</code>
5.	<code>}</code>

Tabel D. 47 Kode Program *Function* ViewProduk

1.	function viewProduk(){
2.	\$dataProduk = \$this->db->selectWhere('SELECT NamaProduk,
3.	id_produk from product');
4.	return \$dataProduk['data'];
5.	}

Tabel D. 48 Kode Program *Function* ViewStatus

1.	function viewStatus(){
2.	\$dataStatus= \$this->db->selectWhere('SELECT namaStatus,
3.	id_statusproduk from status_produk');
4.	return \$dataStatus['data'];
5.	}

Tabel D. 49 Kode Program *Function* Edit

1.	function edit(\$id){
2.	\$editOutlet = \$this->db->selectWhere('select * from penjualan p
3.	join product pr on p.id_produk = pr.id_produk
4.	join outlet o on p.id_outlet = o.id_outlet where id_penjualan =
5.	:id_penjualan',
6.	array ('id_penjualan' => \$id));
7.	return \$editOutlet['data'];
8.	}

Tabel D. 50 Kode Program *Function* Update

1.	function update(\$jumlahPengiriman, \$status, \$id){
2.	\$this->db->update('UPDATE `penjualan` SET `jumlahPengiriman`
3.	= :jumlahPengiriman, `id_statusproduk` = :status, Sisa = :sisa
4.	WHERE id_penjualan = :id',
5.	array(
6.	'jumlahPengiriman' => \$jumlahPengiriman,
7.	'status' => \$status,
8.	'sisa' => \$jumlahPengiriman,
9.	'id' => \$id
10.));
11.	
12.	for (\$i=0; \$i < \$jumlahPengiriman; \$i++) {
13.	\$stmt = \$this->db->prepare("INSERT INTO `detail`(`id_detail`,
14.	`id_penjualan`, tanggalTerjual) VALUES (NULL, \$id, NULL)");
15.	\$stmt->execute();

16.	}
17.	}

Tabel D. 51 Kode Program *Function* detail

1.	function detail(\$idoutlet, \$idproduk){
2.	\$detailProduk = \$this->db->selectWhere("SELECT * from detail d
3.	join penjualan p on d.id_penjualan = p.id_penjualan
4.	where tanggalTerjual is NULL and id_outlet = \$idoutlet and
5.	id_produk = \$idproduk");
6.	return \$detailProduk['data'];
7.	}

2. Kelas Pengiriman

Penulisan kode program pengiriman dapat dilihat pada tabel D.52 sampai dengan D.56.

Tabel D. 52 Kode Program *Function* Index

1.	function index(\$id) {
2.	\$sids = explode("-", \$id);
3.	\$this->view->pengiriman = \$this-> kirim-
4.	>viewData(\$sids[0], \$sids[1]);
5.	\$this->view->outlet = \$this-> kirim->viewOutlet();
6.	\$this->view->produk = \$this-> kirim->viewProduk();
7.	\$this->view->status = \$this-> kirim->viewStatus();
8.	\$this->view->idoutlet = \$sids[0];
9.	\$this->view->idproduk = \$sids[1];
10.	\$jabatan = session::get('jabatan');
11.	if (\$jabatan == 'Marketing') {
12.	\$this->view->display('Pengiriman', 'Input', 'Pengiriman');
13.	} else if (\$jabatan == 'Eksekutif') {
14.	\$this->view->display('Pengiriman', 'pengiriman', 'Eksekutif');
15.	}
16.	}

Tabel D. 53 Kode Program *Function* Filter

1.	function filter(){
2.	\$id = \$_POST['idoutlet'].'-'.\$_POST['idproduk'];
3.	\$this->index(\$id);
4.	}

Tabel D. 54 Kode Program *Function* Edit

1.	<code>function edit(\$id){</code>
2.	<code> \$this->view->data = \$this-> kirim->edit(\$id);</code>
3.	<code> \$this->view->display('Pengiriman', 'Edit', 'Penjualan');</code>
4.	<code>}</code>

Tabel D. 55 Kode Program *Function* Update

1.	<code>function update(){</code>
2.	<code> \$id = \$_POST['id'];</code>
3.	<code> \$JumlahPengiriman = \$_POST['JumlahPengiriman'];</code>
4.	<code> \$status = 1;</code>
5.	<code> \$hasil = \$this->kirim->update(\$JumlahPengiriman, \$status, \$id);</code>
6.	<code> header('location:'.URL.'/pengiriman/index/1-1');</code>
7.	<code>}</code>

Tabel D. 56 Kode Program *Function* Detail

1.	<code>function detail(\$id){</code>
2.	<code> \$sids = explode("-", \$id);</code>
3.	<code> \$this->view->detail = \$this->kirim->detail(\$sids[0], \$sids[1]);</code>
4.	<code> \$this->view->display('Pengiriman', 'Detail', 'Produk');</code>
5.	<code>}</code>

D.7 Kode Program Pengelolaan Data Penjualan Produk

1. Kelas `m_detail`

Penulisan kode program `m_detail` dapat dilihat pada tabel D.57 sampai dengan D.68.

Tabel D. 57 Kode Program *function* `viewData`

1.	<code>function viewData(\$id_outlet, \$id_produk){</code>
2.	<code> \$dataPenjualan = \$this->db->selectWhere("SELECT * from</code>
3.	<code>penjualan where id_outlet = \$id_outlet and id_produk = \$id_produk</code>
4.	<code>order by tanggal");</code>
5.	<code> return \$dataPenjualan['data'];</code>
6.	<code>}</code>

Tabel D. 58 Kode Program *Function* viewOutlet

1.	function viewOutlet(){
2.	\$dataOutlet = \$this->db->selectWhere('SELECT NamaOutlet,
3.	id_outlet from outlet');
4.	return \$dataOutlet['data'];
5.	}

Tabel D. 59 Kode Program *Function* viewProduk

1.	function viewProduk(){
2.	\$dataProduk = \$this->db->selectWhere('SELECT NamaProduk,
3.	id_produk from product');
4.	return \$dataProduk['data'];
5.	}

Tabel D. 60 Kode Program *Function* edit

1.	function edit(\$id){
2.	\$editOutlet = \$this->db->selectWhere('select * from penjualan p
3.	join product pr on p.id_produk = pr.id_produk
4.	join outlet o on p.id_outlet = o.id_outlet where id_penjualan =
5.	:id_penjualan',
6.	array ('id_penjualan' => \$id));
7.	return \$editOutlet['data'];
8.	}

Tabel D. 61 Kode Program *Function* Transaksi

1.	function transaksi(){
2.	\$idoutlet = \$_SESSION['idoutlet'];
3.	\$traksaksi = \$this->db->selectWhere("select * from transaksi
4.	where idOutlet = \$idoutlet GROUP by tanggalTransaksi");
5.	return \$traksaksi['data'];
6.	}

Tabel D. 62 Kode Program *Function* Sisa

1.	function sisa(){
2.	\$idoutlet = \$_SESSION['idoutlet'];
3.	\$sisaBesar = \$this->db->selectWhere("select sum(Sisa) as sisa
4.	from penjualan where id_outlet = \$idoutlet and id_produk = 1 ");
5.	\$sisaKecil = \$this->db->selectWhere("select sum(Sisa) as sisa
6.	from penjualan where id_outlet = \$idoutlet and id_produk = 2 ");

7.	\$sisabesar['besar'] = \$sisabesar['data'][0];
8.	\$sisakecil['kecil'] = \$sisakecil['data'][0];
9.	return \$sisabesar;
10.	}

Tabel D. 63 Kode Program *Function Beli*

1.	function beli(\$kecil, \$besar, \$sisakecil, \$sisabesar){
2.	\$idoutlet = \$_SESSION['idoutlet'];
3.	if (\$kecil > \$sisakecil \$besar > \$sisabesar) {
4.	return "Barang Tidak Cukup!";
5.	}
6.	\$stmt = \$this->db->prepare("INSERT INTO transaksi
7.	(idTransaksi, tanggalTransaksi, jumlahBesar, jumlahKecil, idOutlet)
8.	VALUES (NULL, CURRENT_DATE(), \$besar, \$kecil,
9.	\$idoutlet)");
10.	\$stmt->execute();
11.	\$this->logicTrans(1, \$besar);
12.	\$this->logicTrans(2, \$kecil);
13.	return "Transaksi Berhasil";
14.	}

Tabel D. 64 Kode Program *Function LogicTrans*

1.	function logicTrans(\$idProduk, \$jumlah){
2.	if(\$jumlah != 0){
3.	\$idoutlet = \$_SESSION['idoutlet'];
4.	\$tn = \$this->db->selectWhere("select * from transaksi where
5.	tanggalTransaksi = CURRENT_DATE() and idOutlet = \$idoutlet
6.	order by idTransaksi desc limit 1");
7.	\$idtn = \$tn['data'][0]['idTransaksi'];
8.	
9.	for (\$i = \$jumlah; \$i > 0; \$i--) {
10.	\$d = \$this->db->selectWhere("SELECT id_detail FROM detail
11.	d join penjualan p on d.id_penjualan = p.id_penjualan
12.	WHERE p.id_outlet = \$idoutlet AND tanggalTerjual is NULL
13.	and id_produk = \$idProduk order by id_detail asc limit 1");
14.	\$iddd = \$d['data'][0]['id_detail'];
15.	\$this->db->update("UPDATE detail SET idTransaksi= \$idtn,
16.	tanggalTerjual = CURRENT_DATE()
17.	WHERE id_detail = \$iddd");
18.	
19.	\$cari = \$this->db->selectWhere("SELECT * from detail where
20.	id_detail = \$iddd");
21.	\$idcari = \$cari['data'][0]['id_penjualan'];

22.	\$carisisa = \$this->db->selectWhere("SELECT count(*) as
23.	jumlah from detail where
24.	id_penjualan = \$idcari and tanggalTerjual is null");
25.	\$sisisa = \$carisisa['data'][0]['jumlah'];
26.	\$this->db->update("UPDATE `penjualan` SET `Sisa` = \$sisisa
27.	WHERE id_penjualan = \$idcari");
28.	
29.	}
30.	
31.	\$j = \$this->db->selectWhere("SELECT count(*) as jumlah from
32.	detail d join penjualan p on d.id_penjualan = p.id_penjualan
33.	where tanggalTerjual = CURRENT_DATE() and id_outlet =
34.	\$idoutlet and id_produk = \$idProduk");
35.	\$jumlah = \$j['data'][0]['jumlah'];
36.	\$ambil = \$this->db->selectWhere("SELECT * from penjualan
37.	where id_outlet = \$idoutlet
38.	and id_produk = \$idProduk and Tanggal =
39.	CURRENT_DATE());
40.	\$p = \$ambil['data'][0];
41.	\$hasilPeramalan = \$p['hasilPeramalan'];
42.	\$error = \$jumlah - \$hasilPeramalan;
43.	\$mad = abs(\$error);
44.	\$mape = (\$mad / \$jumlah) / 0.01;
45.	\$akurasi = 100 - \$mape;
46.	\$this->db->update("UPDATE `penjualan` SET `Jumlah` =
47.	\$jumlah, MAPE = \$mape, akurasi = \$akurasi where id_outlet =
48.	\$idoutlet
49.	and id_produk = \$idProduk and Tanggal =
50.	CURRENT_DATE());
51.	
52.	}
53.	}

Tabel D. 65 Kode Program *Function* detailTransaksi

1.	function detailTransaksi(\$tanggal) {
2.	\$idoutlet = \$_SESSION['idoutlet'];
3.	\$tr = \$this->db->selectWhere("select * from transaksi where
4.	tanggalTransaksi = '\$tanggal' and idOutlet = \$idoutlet");
5.	return \$tr['data'];
6.	}

Tabel D. 66 Kode Program *Function* ProdukDetail

1.	function produkDetail(\$transaksis){
2.	\$idoutlet = \$_SESSION['idoutlet'];
3.	foreach(\$transaksis as \$transaksi){
4.	\$idtransaksi = \$transaksi['idTransaksi'];
5.	\$pb = \$this->db->selectWhere("select * from detail d join
6.	penjualan p on d.id_penjualan = p.id_penjualan
7.	where idTransaksi = \$idtransaksi and id_produk = 1");
8.	\$pk = \$this->db->selectWhere("select * from detail d join
9.	penjualan p on d.id_penjualan = p.id_penjualan
10.	where idTransaksi = \$idtransaksi and id_produk = 2");
11.	\$pro['besar'][\$idtransaksi] = \$pb['data'];
12.	\$pro['kecil'][\$idtransaksi] = \$pk['data'];
13.	}
14.	return \$pro;
15.	}

Tabel D. 67 Kode Program *Function* totalPembelian

1.	function totalPembelian(\$tanggal){
2.	\$idoutlet = \$_SESSION['idoutlet'];
3.	\$tb = \$this->db->selectWhere("select sum(jumlahBesar) as jumlah
4.	from transaksi
5.	where tanggalTransaksi = '\$tanggal' and idOutlet = \$idoutlet");
6.	\$tk = \$this->db->selectWhere("select sum(jumlahKecil) as jumlah
7.	from transaksi
8.	where tanggalTransaksi = '\$tanggal' and idOutlet = \$idoutlet");
9.	\$tot['besar'] = \$tb['data'][0]['jumlah'];
10.	\$tot['kecil'] = \$tk['data'][0]['jumlah'];
11.	return \$tot;
12.	}

Tabel D. 68 Kode Program *Function* detail

1.	function detail(\$idproduk){
2.	\$idoutlet = \$_SESSION['idoutlet'];
3.	\$detailProduk = \$this->db->selectWhere("SELECT * from detail d
4.	join penjualan p on d.id_penjualan = p.id_penjualan
5.	where tanggalTerjual is NULL and id_outlet = \$idoutlet and
6.	id_produk = \$idproduk");
7.	return \$detailProduk['data'];
8.	}

2. Kelas detail

Penulisan kode program detail dapat dilihat pada tabel D.69 sampai dengan D.65.

Tabel D. 69 Kode Program *Function* index

1.	function index(\$id) {
2.	\$sids = split("-", \$id);
3.	\$this->view->penjualan = \$this->detail-
4.	>viewData(\$sids[0], \$sids[1]);
5.	\$this->view->outlet = \$this->detail->viewOutlet();
6.	\$this->view->produk = \$this->detail->viewProduk();
7.	\$this->view->idoutlet = \$sids[0];
8.	\$this->view->idproduk = \$sids[1];
9.	\$jabatan = session::get('jabatan');
10.	if (\$jabatan == 'Admin Outlet') {
11.	\$this->view->display('Detail', 'Produk', 'Outlet');
12.	}
13.	}

Tabel D. 70 Kode Program *Function* Transaksi

1.	function transaksi(){
2.	\$this->view->transaksi = \$this->detail->transaksi();
3.	\$this->view->sisa = \$this->detail->sisa();
4.	\$this->view->display('Detail', 'Transaksi', 'Penjualan');
5.	}

Tabel D. 71 Kode Program *Function* Beli

1.	function beli(){
2.	\$kecil = empty(\$_POST['kecil']) ? 0 : \$_POST['kecil'];
3.	\$besar = empty(\$_POST['besar']) ? 0 : \$_POST['besar'];
4.	\$sisaBesar = \$_POST['sisaBesar'];
5.	\$sisaKecil = \$_POST['sisaKecil'];
6.	\$hasil = \$this->detail->beli(\$kecil, \$besar, \$sisaKecil, \$sisaBesar);
7.	\$_SESSION['msg-err'] = \$hasil;
8.	header('location:'.URL.'/detail/transaksi');
9.	}

Tabel D. 72 Kode Program *Function* detailTransaksi

1.	function detailTransaksi(\$tanggal){
2.	\$hasil = \$this->detail->detailTransaksi(\$tanggal);
3.	\$this->view->produk = \$this->detail->produkDetail(\$hasil);
4.	\$this->view->total = \$this->detail->totalPembelian(\$tanggal);
5.	\$this->view->transaksi = \$hasil;
6.	\$this->view->display('Detail', 'Detail', 'Transaksi');
7.	}

Tabel D. 73 Kode Program *Function* filter

1.	function filter(){
2.	\$id = \$_POST['idoutlet'].'-'. \$_POST['idproduk'];
3.	\$this->index(\$id);
4.	}

Tabel D. 74 Kode Program *Function* edit

1.	function edit(\$id){
2.	\$this->view->data = \$this->detail->edit(\$id);
3.	\$this->view->display('Detail', 'Edit', 'Produk');
4.	}

Tabel D. 75 Kode Program *Function* update

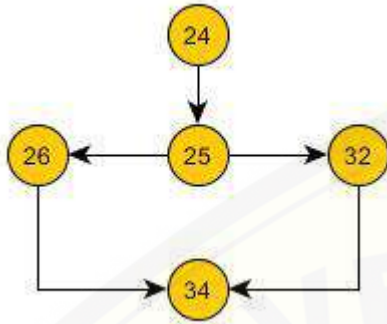
1.	function update(){
2.	\$id = \$_POST['id'];
3.	\$outlet = \$_POST['outlet'];
4.	\$produk = \$_POST['produk'];
5.	\$jumlah = \$_POST['jumlah'];
6.	\$hasil = \$this->detail->update(\$outlet, \$produk, \$jumlah, \$id);
7.	header('location:'.URL.'/detail/index/1-1');
8.	}

Tabel D. 76 Kode Program *Function* Detail

1.	function detail(\$idproduk){
2.	\$this->view->detail = \$this->detail->detail(\$idproduk);
3.	\$this->view->display('Detail', 'Detail', 'Produk');
4.	}

E. Pengujian *White Box*

E.1 Kelas Controller

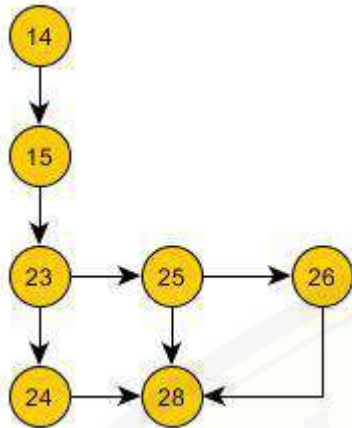


Gambar E. 1 *Function Submit (outlet)*

$$CC = E - N + 2 = 5 - 5 + 2 = 2$$

Tabel E. 1 Test case Function Submit (Outlet)

Test Case 1	Jika berhasil menambahkan data <i>outlet</i> baru
Target yang diharapkan	Berhasil menambahkan atau menginputkan data <i>outlet</i> baru
Hasil pengujian	Benar
Pah/Jalur	24, 25, 26, 34
Test Case 2	Jika tidak berhasil menambahkan data <i>outlet</i> baru
Target yang diharapkan	Tidak berhasil menambahkan data <i>outlet</i> baru dan menampilkan pesan error
Hasil pengujian	Benar
Pah/Jalur	24, 25, 32, 34

Gambar E. 2 *Function Index* (pengiriman)

$$CC = E - N + 2 = 8 - 7 + 2 = 3$$

Tabel E. 2 Test Case Function Index (Pengiriman)

Test Case 1	Jika berhasil menampilkan halaman input data pengiriman oleh marketing
Target yang diharapkan	Berhasil menampilkan halaman input data pengiriman oleh marketing
Hasil pengujian	Benar
Pah/Jalur	14, 15, 23, 24, 28
Test Case 2	Jika berhasil menampilkan halaman pengiriman oleh eksekutif
Target yang diharapkan	berhasil menampilkan halaman pengiriman oleh eksekutif
Hasil pengujian	Benar
Pah/Jalur	14, 15, 23, 25, 26, 28
Test Case 3	Jika tidak berhasil menampilkan halaman pengiriman oleh eksekutif dan halaman input data pengiriman oleh marketing

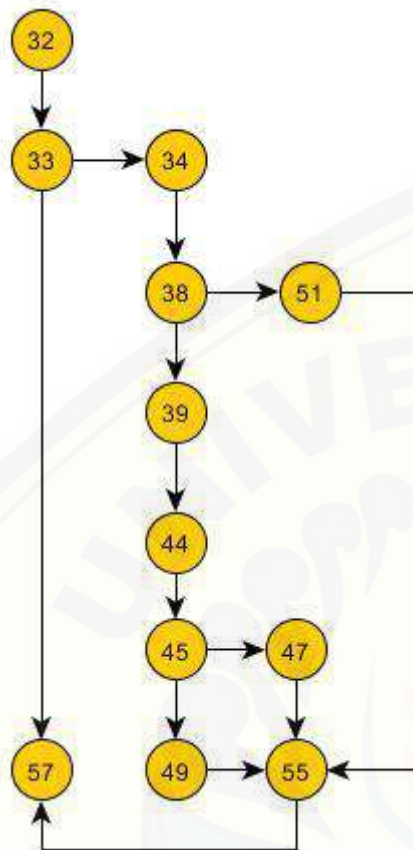
Target yang diharapkan	tidak berhasil menampilkan halaman pengiriman oleh eksekutif dan halaman input data pengiriman oleh marketing
Hasil pengujian	Benar
Pah/Jalur	14, 15, 23, 25, 28

Gambar E. 3 *Function update* (pengiriman)

$$CC = E - N + 2 = 2 - 3 + 2 = 1$$

Tabel E. 3 Test Case Function Update (Pengiriman)

Test Case 1	Jika berhasil mengupdate data pengiriman
Target yang diharapkan	Berhasil mengupdate data pengiriman
Hasil pengujian	Benar
Pah/Jalur	43, 44, 49

Gambar E. 4 *Function Submit* (pengguna)

$$CC = E - N + 2 = 14 - 12 + 2 = 4$$

Tabel E. 4 Test Case Function Submit (Pengguna)

Test Case 1	Jika tidak menekan tombol submit
Target yang diharapkan	Tidak memproses apapun
Hasil pengujian	Benar
Pah/Jalur	32, 33, 57
Test Case 2	Jika berhasil menambahkan data pengguna dengan input nomer telepon benar dan jabatan admin <i>outlet</i>

Target yang diharapkan	Berhasil menambahkan data pengguna dengan input nomer telepon benar dan jabatan admin <i>outlet</i>
Hasil pengujian	Benar
Pah/Jalur	32, 33, 34, 38, 39, 44, 45, 49, 55, 57
Test Case 3	
Target yang diharapkan	berhasil menambahkan data pengguna dengan input nomer telepon benar dan jabatan selain admin <i>outlet</i>
Hasil pengujian	Benar
Pah/Jalur	32, 33, 34, 38, 39, 44, 45, 47, 55, 57
Test Case 3	
Target yang diharapkan	Tidak berhasil menambahkan data pengguna karena input nomer telepon salah
Hasil pengujian	Benar
Pah/Jalur	32, 33, 34, 38, 51, 55, 57

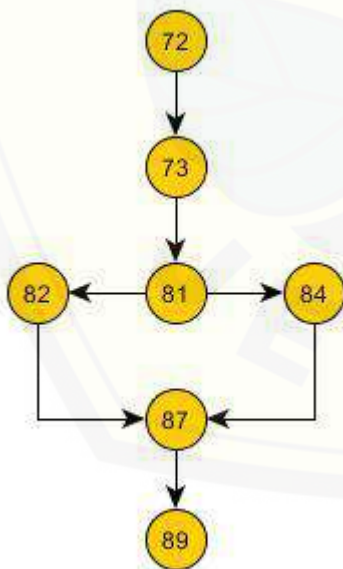


Gambar E. 5 *Function Edit* (pengguna)

$$CC = E - N + 2 = 2 - 3 + 2 = 1$$

Tabel E. 5 Test Case Function Edit (Pengguna)

Test Case 1	Jika berhasil menampilkan data yang akan diedit pada form edit data pengguna
Target yang diharapkan	Berhasil menampilkan data yang akan diedit pada form edit data pengguna
Hasil pengujian	Benar
Pah/Jalur	64, 65, 70

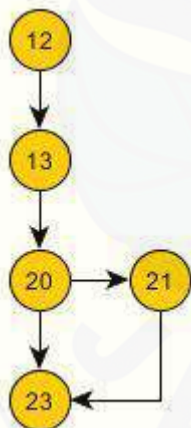


Gambar E. 6 *Function update* (pengguna)

$$CC = E - N + 2 = 7 - 7 + 2 = 2$$

Tabel E. 6 Test Case Function Update (Pengguna)

Test Case 1	Jika berhasil mengupdate data pengguna dengan jabatan admin <i>outlet</i>
Target yang diharapkan	Berhasil mengupdate data pengguna
Hasil pengujian	Benar
Pah/Jalur	72, 73, 81, 82, 87, 89
Test Case 2	Jika berhasil mengupdate data pengguna dengan jabatan selain admin <i>outlet</i>
Target yang diharapkan	berhasil mengupdate data pengguna dengan jabatan selain admin <i>outlet</i>
Hasil pengujian	Benar
Pah/Jalur	72, 73, 81, 84, 87, 89



Gambar E. 7 Function Index (detail)

$$CC = E - N + 2 = 5 - 5 + 2 = 2$$

Tabel E. 7 Test Case Function Index (Detail)

Test Case 1	Jika berhasil menampilkan halaman produk <i>outlet</i> pada admin <i>outlet</i>
Target yang diharapkan	Berhasil menampilkan halaman produk <i>outlet</i> pada admin <i>outlet</i>
Hasil pengujian	Benar
Pah/Jalur	12, 13, 20, 21, 23
Test Case 2	Jika tidak berhasil menampilkan halaman produk <i>outlet</i> pada admin <i>outlet</i>
Target yang diharapkan	Tidak berhasil menampilkan halaman produk <i>outlet</i> pada admin <i>outlet</i>
Hasil pengujian	Benar
Pah/Jalur	12, 13, 20, 23

Gambar E. 8 *Function Update* (detail)

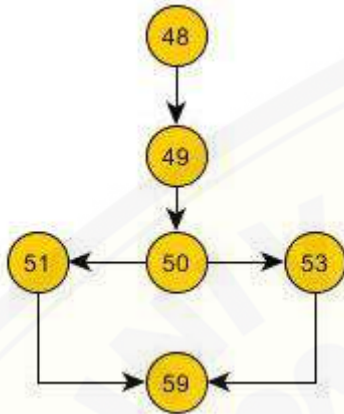
$$CC = E - N + 2 = 2 - 3 + 2 = 1$$

Tabel E. 8 Test Case Function Update (Detail)

Test Case 1	Jika berhasil mengupdate data penjualan pada setiap <i>outlet</i>
Target yang diharapkan	Berhasil mengupdate data penjualan pada setiap <i>outlet</i>

Hasil pengujian	Benar
Pah/Jalur	37, 38, 44

E.2 Kelas Model



Gambar E. 9 Function Beli (Detail)

$$CC = E - N + 2 = 6 - 6 + 2 = 2$$

Tabel E. 9 Test Case Function Beli (Detail)

Test Case 1	Jika jumlah pembelian lebih dari jumlah sisa produk
Target yang diharapkan	Tidak berhasil menambahkan transaksi penjualan dan menampilkan span “Barang tidak cukup”
Hasil pengujian	Benar
Pah/Jalur	48, 49, 50, 51, 59
Test Case 2	Jika jumlah pembelian kurang dari jumlah sisa produk
Target yang diharapkan	berhasil menambahkan transaksi penjualan dan menampilkan span “transaksi berhasil”
Hasil pengujian	Benar
Pah/Jalur	48, 49, 50, 53, 59

F. Pengujian Black Box

F.1 Fitur *Login*

Tabel F. 1 Pengujian *Black Box* Fitur *Login*

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Login	Memilih tombol login	Menampilkan halaman home user	V	

F.2 Fitur Pengelolaan Data Pengguna

Tabel F. 2 Pengujian *Black Box* Fitur Pengelolaan Data Pengguna

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data pengguna	Memilih menu data pengguna	Menampilkan tabel daftar pengguna	V	
2.	Menambah data pengguna	Memilih tombol simpan	Menyimpan data pada <i>database</i>	V	
3.	Mengubah data pengguna	Memilih tombol edit	Menampilkan halaman form edit akun	V	
		Memilih tombol simpan	a. Menyimpan data ke <i>database</i> b. Menampilkan data pengguna pada tabel data pengguna	V	
4.	Menghapus data pengguna	Memilih tombol <i>delete</i>	Menghapus data dari <i>database</i>	V	

F.3 Fitur Melihat Data Pengguna

Tabel F. 3 Pengujian *Black Box* Fitur Melihat Data Pengguna

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data pengguna	Memilih menu data pengguna	Menampilkan tabel daftar data pengguna	V	

F.4 Fitur Pengelolaan Data *Outlet*Tabel F. 4 Pengujian *Black Box* Fitur Pengelolaan Data *Outlet*

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data <i>outlet</i>	Memilih menu data <i>outlet</i>	Menampilkan tabel daftar <i>outlet</i>	V	
2.	Menambah data <i>outlet</i>	Memilih tombol simpan	Menyimpan data pada <i>database</i>	V	
3.	Mengubah data <i>outlet</i>	Memilih tombol edit	Menampilkan halaman form edit data <i>outlet</i>	V	
		Memilih tombol simpan	c. Menyimpan data ke <i>database</i> d. Menampilkan data <i>outlet</i> pada tabel data <i>outlet</i>	V	
4.	Menghapus data <i>outlet</i>	Memilih tombol <i>delete</i>	Menghapus data dari <i>database</i>	V	

F.5 Fitur Melihat Data *Outlet*Tabel F. 5 Pengujian *Black Box* Melihat Data *Outlet*

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data <i>outlet</i>	Memilih menu data <i>outlet</i>	Menampilkan tabel daftar data <i>outlet</i>	V	

F.6 Fitur Melihat Data Hasil Peramalan

Tabel F. 6 Pengujian *Black Box* Melihat Data Hasil Peramalan

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data hasil peramalan	Memilih sub menu data peramalan pada menu penjualan	Menampilkan tabel daftar hasil peramalan	V	

F.7 Fitur Pengelolaan Data Pendistribusian Produk

Tabel F. 7 Pengujian *Black Box* Pengelolaan Data Pendistribusian Produk

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data hasil penjualan dan pendistribusian produk	Memilih menu data pengirimann	Menampilkan tabel pengiriman	V	
2.	Menambah data pengiriman	Memilih tombol edit	Menampilkan halaman form tambah data pengiriman	V	
		Memilih tombol simpan	a. Menyimpan data ke <i>database</i>	V	

			b. Menampilkan data pengiriman dan status produk pada tabel data pengiriman		
--	--	--	---	--	--

F.8 Fitur Pengelolaan Data Penjualan Produk

Tabel F. 8 Pengujian *Black Box* Pengelolaan Data Penjualan Produk

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat data hasil penjualan dan sisa produk	Memilih menu data penjualan	Menampilkan tabel data penjualan produk	V	
2.	Menambah data hasil penjualan	Memilih tombol edit	Menampilkan halaman form tambah data penjualan	V	
		Memilih tombol simpan	a. Menyimpan data ke <i>database</i> b. Menampilkan data penjualan dan sisa pada tabel data penjualan produk	V	

F.9 Fitur Melihat Data Hasil Penjualan dan Pendistribusian Produk

Tabel F. 9 Pengujian *Black Box* Melihat Data Hasil Penjualan dan Pendistribusian Produk

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Pengelolaan data hasil penjualan produk dan pendistribusian produk	Memilih sub menu data pengiriman pada menu penjualan	Menampilkan tabel daftar pengiriman	V	

F.10 Fitur Input Transaksi Penjualan

Tabel F. 10 Fitur Input Transaksi Penjualan

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Input transaksi penjualan	Memilih menu transaksi penjualan	Data transaksi penjualan tersimpan pada database dan ditampilkan pada halaman rekap hasil transaksi penjualan	V	

F.11 Fitur Melihat Rekap Hasil Transaksi Penjualan

Tabel F. 11 Melihat Rekap Hasil Transaksi Penjualan

No	Fitur	Aksi	Hasil	Kesimpulan	
				Berhasil	Tidak
1.	Melihat Rekap Hasil Transaksi Penjualan	Memilih <i>button</i> detail pada kolom tabel data transaksi pada menu transaksi penjualan	Menampilkan rekap keseluruhan hasil transaksi penjualan	V	