



**SISTEM INFORMASI PERAMALAN PENJUALAN PRODUK
KECANTIKAN DENGAN MENGGUNAKAN METODE *LEAST SQUARE*
(STUDI KASUS : LARISSA AESTHETIC CENTER)**

SKRIPSI

oleh

Bayu Islam Septian Wildani

122410101102

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

2019



**SISTEM INFORMASI PERAMALAN PENJUALAN PRODUK
KECANTIKAN DENGAN MENGGUNAKAN METODE *LEAST SQUARE*
(STUDI KASUS : LARISSA AESTHETIC CENTER)**

SKRIPSI

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

oleh

Bayu Islam Septian Wildani

122410101102

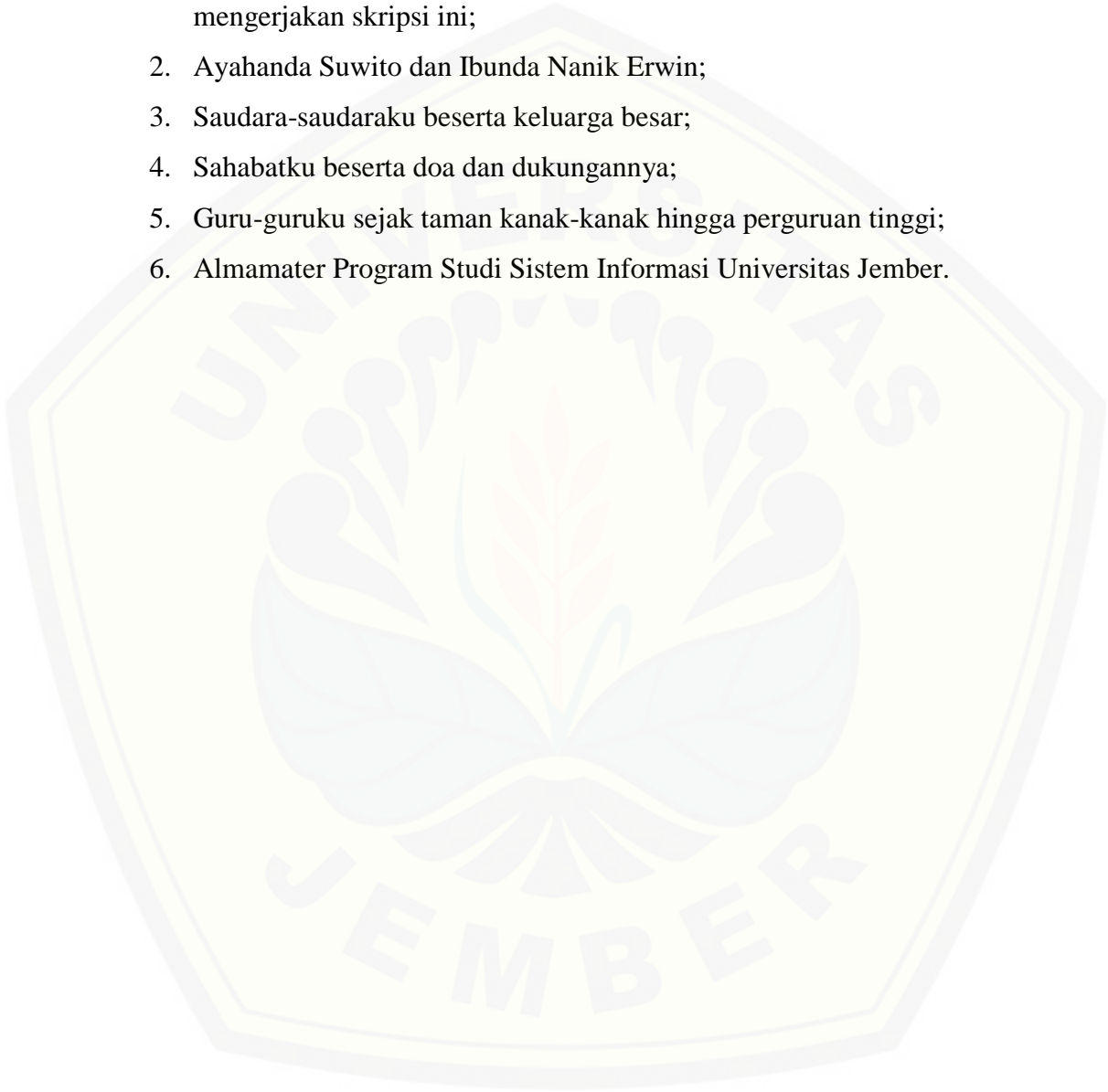
**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS JEMBER**

2019

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT, yang telah memberikan kemudahan dan kelancaran dalam mengerjakan skripsi ini;
2. Ayahanda Suwito dan Ibunda Nanik Erwin;
3. Saudara-saudaraku beserta keluarga besar;
4. Sahabatku beserta doa dan dukungannya;
5. Guru-guruku sejak taman kanak-kanak hingga perguruan tinggi;
6. Almamater Program Studi Sistem Informasi Universitas Jember.



MOTTO

Hai orang-orang yang beriman, mintalah pertolongan(kepada Allah) dengan sabar dan (mengerjakan) shalat, sesungguhnya Allah beserta orang-orang yang sabar. (terjemahan Surat Al-Baqarah ayat 153)¹



¹ Kementerian Agama Republik Indonesia. 2004. Al Qur'an dan Terjemahannya. Semarang: CV. Al Waah

PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Bayu Islam Septian Wildani

NIM : 122410101102

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* (Studi Kasus : Larissa Aesthetic Center)”, adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, belum pernah diajukan pada institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

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

Jember, Mei 2019

Yang menyatakan,

Bayu Islam Septian Wildani

NIM 122410101102

SKRIPSI

**SISTEM INFORMASI PERAMALAN PENJUALAN PRODUK
KECANTIKAN DENGAN MENGGUNAKAN METODE *LEAST SQUARE*
(STUDI KASUS : LARISSA AESTHETIC CENTER)**

Oleh

Bayu Islam Septian Wildani

122410101102

PEMBIMBING

Dosen Pembimbing Utama : M. Arief Hidayat, S. Kom, M.Kom

Dosen pembimbing Pendamping : Gayatri Dwi Santika, S.SI., M.Kom

PENGESAHAN PEMBIMBING

Skripsi berjudul "Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* (Studi Kasus : Larissa Aesthetic Center)", telah diuji dan disahkan pada,

hari, tanggal :

tempat : Program Studi Sistem Informasi Universitas Jember

Disetujui oleh:

Pembimbing I,

Pembimbing II,

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

NIP 198101232010121003

Gayatri Dwi Santika, S.SI., M.Kom

NIP 760017013

PENGESAHAN

Skripsi berjudul "Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* (Studi Kasus : Larissa Aesthetic Center)", telah diuji dan disahkan pada,

hari, tanggal :

tempat : Program Studi Sistem Informasi Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

Nelly Oktavia A, S.Si., MT
NIP 198410242009122008

Januar Adi Putra, S.Kom., M.Kom
NIP 760017015

Mengesahkan

Dekan Fakultas Ilmu Komputer,

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

NIP 196811131994121001

RINGKASAN

Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* (Studi Kasus : Larissa Aesthetic Center); Bayu Islam Septian Wildani, 122410101102; 2019; 136 Halaman; Program Studi Sistem Informasi Universitas Jember.

Klinik kecantikan Larisa Aesthetic Center cabang Jember merupakan salah satu klinik kecantikan yang mengusung konsep “*back to nature*”, bahan-bahan yang digunakan berbahan dasar alami. Dalam pemasarannya produk tersebut pastinya terdapat tanggal kadaluarsa yang telah ditetapkan oleh produsen. Tidak sedikit produk yang sampai batas kadaluarsanya tidak terjual, sehingga banyak di jumpai penumpukan produk. Beda hal ketika permintaan pelanggan melebihi persediaan yang ada, sehingga persediaan stok yang tidak mencukupi untuk memenuhi kebutuhan pelanggan. Keadaan tersebut menyebabkan kerugian pada klinik kecantikan apabila stok produk tidak sesuai dengan permintaan pelanggan.

Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* ini dapat meramalkan penjualan produk pada periode selanjutnya yang dapat digunakan sebagai acuan untuk ketersediaan stok produk yang ada di klinik kecantikan Larissa Aesthetic Center. Penulis membuat sistem peramalan *least square* dengan bahasa pemrograman *PHP* dengan *framework Codeigniter* dan *MySQL* sebagai *database*. Dengan sistem ini diharapkan dapat membantu klinik kecantikan Larissa Aesthetic Center dalam menyesuaikan stok produk untuk periode. Dari hasil penelitian yang dilakukan, aplikasi dapat menjadi sebuah solusi bagi pihak klinik kecantikan Larissa Aesthetic Center.

PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* (Studi Kasus : Larissa Aesthetic Center). Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata Satu (S1) pada Program Studi Sistem Informasi Universitas Jember.

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

1. Prof. Dr. Saiful Bukhori ST., M.Kom., selaku Dekan Fakultas Ilmu Komputer Universitas Jember.
2. M. Arief Hidayat, S. Kom, M. Kom selaku Dosen Pembimbing Utama dan Gayatri Dwi Santika, S.SI., M.Kom selaku Dosen Pembimbing Pendamping yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi.
3. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember.
4. Ibu Nanik Erwin, Tante Susiati dan Kakak Diah Rengganis yang selalu mendukung dan mendoakan.
5. Keluarga penulis yang selama ini memberikan nasehat dan perhatian.
6. Maulida Erni Wati sekeluarga yang selalu memberikan semangat dan memotivasi.
7. Teman – teman yang sudah membantu memberi pengarahan dalam pengerjaan tugas akhir Bagus Windi, Andika Widiarso, Aji Mukti, Rahmatullah, Ardian Sukma, Ahmad Nur Fikri, dan Mas Nip.
8. Keluarga besar UKM Balwana.
9. Teman - teman seperjuangan Formation angkatan 2012.
10. Teman-teman Program Studi Sistem Informasi di semua angkatan.
11. Semua pihak yang tidak dapat disebutkan satu persatu.

DAFTAR ISI

SKRIPSI.....	i
PERSEMBAHAN.....	ii
MOTTO.....	iii
PERNYATAAN.....	iv
PEMBIMBING.....	v
PENGESAHAN PEMBIMBING.....	vi
PENGESAHAN.....	vii
RINGKASAN.....	viii
PRAKATA.....	ix
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL.....	xiv
BAB 1. PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	2
1.3. Tujuan.....	2
1.4. Batasan Masalah.....	3
1.5. Sistematika Penulisan.....	3
BAB 2. TINJAUAN PUSTAKA.....	5
2.1. Penelitian Terdahulu.....	5
2.2. Prediksi atau Peramalan (<i>Forecasting</i>).....	6
2.3. Metode <i>Least Square</i>	6
2.4. <i>Mean Absolute Percentage Error (MAPE)</i>	7
BAB 3. METODE PENELITIAN.....	8
3.1. Jenis Penelitian.....	8
3.2. Waktu Penelitian.....	8
3.3. Tahapan Penelitian.....	8
3.3.1. Analisis Kebutuhan (<i>Requirements Definition</i>).....	9
3.3.2. Sistem dan Desain Software (<i>System and Software Design</i>).....	10
3.3.3. Implementasi Sistem (<i>Implementation and Unit Testing</i>).....	11

3.3.4.	Pengujian Sistem (<i>Integration and System Testing</i>)	11
3.3.5.	Pemeliharaan (<i>Operation and Maintenance</i>)	12
BAB 4. PENGEMBANGAN SISTEM.....		13
4.1	Tahapan Analisis Kebutuhan.....	13
4.1.1	Penerapan <i>Least Square</i>	13
4.1.2	Data Kebutuhan.....	13
4.1.3	SOP (<i>Statement of purpose</i>).....	14
4.1.4	Fungsi Sistem	15
4.1.5	Kebutuhan Fungsional.....	15
4.1.6	Kebutuhan Non-Fungsional	15
4.2	Tahapan Desain Sistem	16
4.2.1	<i>Bussiness Process</i>	16
4.2.2	<i>Usecase Diagram</i>	17
4.2.3	<i>Usecase Scenario</i>	20
4.2.4	<i>Activity Diagram</i>	31
4.2.5	<i>Sequence Diagram</i>	35
4.2.6	<i>Class Diagram</i>	40
4.2.7	<i>Entity Relations Diagram (ERD)</i>	41
4.3	Tahapan Implementasi Sistem.....	41
4.3.1	Kode Program Peramalan	41
4.4	Pengujian	44
4.4.1	<i>White Box</i>	44
4.4.2	<i>Black Box</i>	46
BAB 5. HASIL DAN PEMBAHASAN.....		49
5.1.	Hasil Pengembangan Sistem	49
5.1.1.	Fitur <i>Login</i>	49
5.1.2.	Fitur Pengaturan Akun	50
5.1.3.	Fitur Data Produk.....	50
5.1.4.	Fitur Data Penjualan.....	52
5.1.5.	Fitur Peramalan	53
5.1.6.	Fitur Profil.....	54
5.2.	Implementasi Metode <i>Least Square</i>	55

5.3. Tingkat Kebenaran Peramalan	56
BAB 6. PENUTUP.....	58
6.1 Kesimpulan.....	58
6.2 Saran.....	58
DAFTAR PUSTAKA	60
LAMPIRAN.....	61
LAMPIRAN A. <i>Scenario</i>	61
A1. <i>Use Case Scenario</i> Login	61
A2. <i>Use Case Scenario</i> Mengelola Data Penjualan.....	62
A3. <i>Use Case Scenario</i> Mengelola Data Prediksi	64
A4. <i>Use Case Scenario</i> Logout	65
LAMPIRAN B. <i>Activity Diagram</i>	66
B1. <i>Activity Diagram</i> Mengelola Data Prediksi	66
B1. <i>Activity Diagram</i> Mengubah Data Profil.....	66
B1. <i>Activity Diagram</i> Logout	67
LAMPIRAN C. <i>Sequence Diagram</i>	67
C1. <i>Sequance Diagram</i> Mengelola Data User (Manager).....	67
C2. <i>Sequance Diagram</i> Mengelola Data Produk (Pegawai)	69
C3. <i>Sequance Diagram</i> Mengelola Data Penjualan (Pegawai).....	71
C4. <i>Sequance Diagram</i> Megubah Data Profil.....	73
C5. <i>Sequance Diagram</i> Logout	73
LAMPIRAN E. Kode Program	74
E1. Kode Program <i>Login</i>	74
E2. Kode Program Mengelola Data User	79
E3. Kode Program Mengelola Data Produk	89
E4. Kode Program Mengelola Data Penjualan	103
E5. Kode Program Mengubah Data Profil.....	113
E6. Kode Program <i>Logout</i>	116
LAMPIRAN F. Tabel Jumlah Penjualan Produk Larissa Aesthetic Center Tahun 2017.....	Error! Bookmark not defined.
LAMPIRAN G. Black Box	117

DAFTAR GAMBAR

Gambar 3. 1 Model Waterfall (Kersting, 2017).....	9
Gambar 4. 1 Bussiness Process Sistem Informasi Penjualan Produk Kecantikan	17
Gambar 4. 2 Usecase Diagram Sistem Informasi Penjualan Produk Kecantikan .	18
Gambar 4. 3 Activity Diagram Login	31
Gambar 4. 4 Activity Diagram Mengelola Data User.....	32
Gambar 4. 5 Activity Diagram Mengelola Data Produk.....	33
Gambar 4. 6 Activity Diagram Data Penjualan	34
Gambar 4. 7 Sequence Diagram Login Manager.....	35
Gambar 4. 8 Sequence Diagram Input Data Produk	36
Gambar 4. 9 Sequence Diagram Edit Produk	36
Gambar 4. 10 Sequence Diagram Aktivasi Produk	37
Gambar 4. 11 Sequence Diagram Tambah Data Penjualan	37
Gambar 4. 12 <i>Sequence Diagram</i> Ubah Data Pejualan	38
Gambar 4. 13 Sequence Diagram Menghapus Data Pejualan.....	38
Gambar 4. 14 Sequence Diagram Login Pegawai	39
Gambar 4. 15 Sequence Diagram Peramalan.....	39
Gambar 4. 16 Class Diagram	40
Gambar 4. 17 Entitiy Relationship Diagram (ERD)	41
Gambar 4. 18 Diagram Alir Function prediksi(\$idproduct)	45
Gambar 5. 1 Halaman Login.....	49
Gambar 5. 2 Halaman Pengaturan Akun.....	50
Gambar 5. 3 Halaman Data Produk	50
Gambar 5. 4 Halaman Form Tambah Data Produk.....	51
Gambar 5. 5 Halaman Form Edit Data Produk	51
Gambar 5. 6 Halaman Data penjualan	52
Gambar 5. 7 Halaman Form Tambah Data Penjualan	52
Gambar 5. 8 Halaman Form Edit Data Pejualan.....	53
Gambar 5. 9 Halaman Data Peramalan	54
Gambar 5. 10 Halaman Profil	54

DAFTAR TABEL

Tabel 4. 1 Data Jumlah Penjualan Produk Larissa Aesthetic Center Tahun 2017	14
Tabel 4. 2 Deskripsi Aktor	19
Tabel 4. 3 Deskripsi Usecase	19
Tabel 4. 4 Usecase Scenario Mengelola Data User	20
Tabel 4. 5 Usecase Scenario Mengelola Data Produk	25
Tabel 4. 6 Kode Program Pada Class Model m_hitung	41
Tabel 4. 7 Hasil Pengujian Black Box	46
Tabel 5. 1 Perhitungan nilai analisis metode least square.....	55
Tabel 5. 2 Perhitungan Nilai MAPE	56

BAB 1. PENDAHULUAN

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

1.1. Latar Belakang

Menurut Prasetya (2009) peramalan adalah seni dan ilmu untuk memperkirakan kejadian di masa depan. Hal ini dapat dilakukan dengan pengambilan data masa lalu dan menempatkannya ke masa yang akan datang dengan suatu model matematis. Peramalan kerap digunakan sebagai perencanaan dan operasi control dalam berbagai bidang salah satunya pada manajemen produksi. Peramalan jumlah penjualan di masa depan dimaksud untuk mengendalikan jumlah stock barang yang ada, dengan demikian dapat dihindari kekurangan atau kehabisan barang. Sehingga permintaan konsumen dapat terpenuhi sesuai dengan waktu yang dijadwalkan.

Larisa Aesthetic Center merupakan klinik kecantikan estetika pratama yang mengusung konsep “Natural Ingredient with Hight Technology” yaitu sistem perawatan wajah, perawatan rambut hingga perawatan tubuh dengan memakai bahan alami yang disinergikan penggunaan teknologi modern dengan harga terjangkau (Larissa Aesthetic Center, 2019). Dalam pemasarannya produk tersebut pastinya terdapat tanggal kadaluarsa yang telah ditetapkan oleh produsen. Tidak sedikit produk yang sampai batas kadaluarsanya tidak terjual, sehingga banyak di jumpai penumpukan produk. Beda hal ketika permintaan pelanggan melebihi persediaan yang ada, sehingga persediaan stok yang tidak mencukupi untuk memenuhi kebutuhan pelanggan. Keadaan tersebut menyebabkan kerugian pada klinik kecantikan apabila stok produk tidak sesuai dengan permintaan pelanggan.

Penjualan pada tahun 2017 dengan produk Apple Milk Cleanser, L Milk Cleanser Tea Tree, Green Tea Milk Cleanser, Chamomile Milk Cleanser mengalami penjualan naik turun yang dapat di lihat pada Tabel 4.1 Jumlah Penjualan Produk Larissa Aesthetic Center Tahun 2017 . Ketidakstabilan tersebut

dapat menyebabkan kekurangan atau kelebihan stok produk sehingga diperlukan prediksi atau peramalan jumlah persediaan produk di klinik kecantikan Larisa Aesthetic Center cabang Jember. Metode yang akan digunakan untuk prediksi stok persediaan ini menggunakan metode *Least Square* karena terbukti dari penelitian yang dilakukan Kuriawan dan Nurdioyono pada tahun 2015, mengenai komparasi antara metode *Double Exponential Smoothing* dan *Least Square* dalam menganalisis pendapatan retribusi uji kendaraan bermotor menunjukkan tingkat akurasi metode *Least Square* lebih baik dibandingkan metode *Double Exponential Smoothing*. Metode tersebut dimaksudkan untuk mengendalikan jumlah stok produk yang ada, dengan demikian dapat dihindari kekurangan atau kehabisan produk. Sehingga permintaan konsumen dapat terpenuhi sesuai waktu yang telah dijadwalkan.

1.2. Rumusan Masalah

Dengan mempertimbangkan latar belakang masalah di atas, dapat dirumuskan masalah yaitu :

1. Bagaimana menerapkan metode *Least Square* dalam memprediksikan banyaknya jumlah penjualan produk kecantikan di Larisa Aesthetic Center cabang Jember?
2. Bagaimana merancang dan membangun sebuah sistem informasi dengan menerapkan metode *Least Square* dalam proses prediksi?

1.3. Tujuan

Tujuan dan manfaat yang dipaparkan merupakan berbagai hasil yang ingin dicapai dan dapat digunakan dari beberapa aspek penelitian ini terselesaikan antara lain :

1. Menerapkan metode *Least Square* dalam menentukan peramalan penjualan produk.
2. Merancang dan membangun sebuah sistem informasi dengan menerapkan metode *Least Square* pada klinik kecantikan Larissa Aesthetic Center.

1.4. Batasan Masalah

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

1. Sistem ini diterapkan di Larissa Aesthetic Center cabang Jember.
2. Sistem ini menerapkan metode *Least Square* untuk persediaan jumlah stok produk.
3. Merancang dan membangun sistem peramalan (Forecasting) jumlah persediaan Produk Larissa Aesthetic Center cabang Jember.
4. Sistem yang dibangun berbasis *website*.
5. Sistem ini tidak menginput data dari transaksi melainkan data rekap perbulannya.
6. Sistem ini menggunakan 4 jenis produk kecantikan yang ada di Larissa Aesthetic Center.
7. Sistem ini menggunakan peramalan jangka pendek.

1.5. Sistematika Penulisan

Sistematika penulisan dan urutan skripsi ini disusun sebagai berikut:

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

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

6. Penutup

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



BAB 2. TINJAUAN PUSTAKA

Bab ini menjelaskan tentang kajian-kajian teori terkait dengan penelitian yang dapat mendukung penelitian yang dilakukan. Tinjauan pustaka merupakan teori-teori dan pustaka yang akan dipakai dalam penelitian. Teori-teori ini berupa teori dari buku literatur dan jurnal.

2.1. Penelitian Terdahulu

Penelitian terdahulu yang dilakukan Muhammad Ihsan Fauzi Rambe (2014) dengan judul “Perancangan Aplikasi Peramalan Persediaan Obat – Obatan Menggunakan Metode *Least Square* (Studi Kasus : Apotik Mutiara Hati)”. Berdasarkan dari hasil penelitian, analisis, perencanaan sistem dan pembuatan program dapat disimpulkan bahwa analisis peramalan menggunakan metode *least square* dapat dipergunakan untuk meramalkan penjualan obat diperiode yang akan datang berdasarkan metode data penjualan tahun sebelumnya selain itu dari aplikasi peramalan dapat menghasilkan hasil ramalan dan telah meminimumkan kesalahan meramal (*forecast error*) tingkat penjualan obat-obatan pada apotek.

Adapun penelitian Rachmad Budi Setiawan (2016) berjudul “Perbandingan Metode Setengah Rata- Rata dan Metode Kuadrat Terkecil Untuk Peramalan Pendapatan Perusahaan di BLU UPTD Terminal Mangkang Semarang”. Dalam penelitian di atas penulis ingin mengetahui seberapa bagus keakuratan peramalan antara metode *Least Square* dan metode *Semi Average*. Dari hasil yang ada dimana dikatakan bahwa *Least Square* lebih unggul di bandingkan metode *Semi Average* dilihat dari perhitungan evaluasi peramalan menggunakan RMSE (*Root Mean Square Error*) dan MAPE (*Mean Absolute Percentage Error*).

Dari penelitian tersebut dapat memberikan masukan serta terapan metode yang dapat digunakan untuk penelitian yang akan datang. Perbedaan dari penelitian terdahulu terdapat pada periode peramalan. Peramalan yang akan di pakai untuk penelitian menggunakan peramalan jangka pendek dengan periode setiap bulan.

2.2. Prediksi atau Peramalan (*Forecasting*)

Peramalan adalah kegiatan memperkirakan tingkat permintaan produk yang diharapkan untuk suatu produk atau beberapa produk dalam periode waktu tertentu di masa yang akan datang (Biegel, 1999). Data masa lalu merupakan dasar untuk meramalkan keadaan mendatang.

Peramalan biasanya diklarifikasikan berdasarkan waktu horizon masa depan yang dibagi menjadi beberapa kategori :

- a. Peramalan jangka pendek. Peramalan yang mencakup sampel data dalam rentan waktu harian, mingguan, dan bulanan.
- b. Peramalan jangka menengah. Peramalan jangka menengah atau *intermediate*, umumnya mencakup sampel data dalam rentan waktu 1 bulan hingga setahun.
- c. Peramalan jangka panjang. Peramalan yang mencakup *sample* data dalam rentan waktu lebih dari 1 tahun.

Peramalan yang digunakan pada penelitian ini yaitu peramalan jangka pendek karena periode yang di pakai dalam rentan waktu setiap bulan.

2.3. Metode *Least Square*

Metode *Least Square* (Sugiarto,2000) merupakan metode yang paling umum digunakan dalam peramalan untuk data time series. Dalam metode ini kita harus menentukan parameter x, apabila parameter x telah terbentuk dan di jumlahkan, parameter tersebut harus berjumlah 0 meskipun data historis berjumlah ganjil dan genap. Metode ini mempunyai keunikan meski rumus yang digunakan sederhana tetapi mempunyai hasil *forecast* yang bagus.

Rumus untuk perhitungan peramalan penjualan ditunjukkan pada persamaan berikut :

a. Perhitungan pemulusan total :

$$Y = a + bX \quad (2.1)$$

b. Perhitungan pencarian nilai a :

$$a = \Sigma Y / N \quad (2.2)$$

c. Perhitungan pencarian nilai b :

$$b = \frac{\sum XY}{\sum X^2} \quad (2.3)$$

Dimana,

X = Periode Waktu

Y = Jumlah Penjualan Pada periode X

a = Bilangan Konstan

b = Koefisien Kecondongan garis tren

N = Banyaknya Peirode

2.4. Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error merupakan rata – rata dari keseluruhan presentase kesalahan antara data aktual dengan data hasil peramalan. Tingkat kesalahan ini sering disebut dengan error atau residual. Kesalahan dalam proyeksi bisa disebabkan karena nilai proyeksi terlalu kecil atau terlalu besar. Nilai proyeksi yang terlalu besar maupun terlalu kecil merupakan sebuah kesalahan. Oleh karena itu kita tidak dapat langsung menjumlahkan atau merata-rata nilai kesalahan karena kesalahan ada yang bertanda positif dan ada yang bertanda negative. Suatu model mempunyai kinerja sangat bagus jika nilai MAPE berada di bawah 10%, dan mempunyai kinerja bagus jika nilai MAPE berada di antara 10% dan 20% (Zainun dan Majid,2003). Berikut ini adalah persamaan dalam menghitung nilai MAPE

$$MAPE = \sum_{t=1}^n \frac{|PE_t|}{n} \quad (2.4)$$

Dimana :

PE_t = kesalahan persentase = $\frac{X_t - F_t}{X_t} * 100$

E_t = kesalah periode $t = X_t - F_t$

X_t = data aktual periode t

n = banyak periode t

BAB 3. METODE PENELITIAN

Pada bagian ini dijelaskan mengenai metode penelitian untuk mendapatkan gambaran tahapan yang sistematis, yang dilakukan untuk menganalisa data dan mengembangkan sistem. Sehingga dapat digunakan untuk menjawab perumusan masalah dan dapat mencapai tujuan dari penelitian.

3.1. Jenis Penelitian

Jenis penelitian yang digunakan dalam membangun sistem peramalan penjualan produk kecantikan merupakan penelitian pengembangan, karena tujuan penelitian adalah untuk membangun sebuah sistem peramalan penjualan di periode selanjutnya. Penelitian pengembangan bertujuan untuk membuat dan mengembangkan suatu produk yang efektif untuk digunakan. Penelitian ini bukan jenis penelitian yang ditujukan untuk menemukan teori atau menguji kebenaran dari suatu teori dalam bentuk eksperimentasi.

3.2. Waktu Penelitian

Tempat pelaksanaan penelitian ini adalah Larissa Aesthetic Center di kabupaten Jember. Waktu pelaksanaan penelitian adalah selama dua bulan, terhitung dari bulan Januari 2018 hingga bulan Februari 2018.

3.3. Tahapan Penelitian

Tahap pengembangan sistem dilakukan setelah analisis data telah selesai dilakukan serta dijadikan bahan untuk membangun sistem sesuai kebutuhan yang ada. Pengembangan perangkat lunak pada penelitian ini dengan menggunakan model *waterfall*. Menurut Pressman (2002) model *waterfall* melakukan pendekatan pengembangan perangkat lunak yang sistematis dan sekuensial dimulai pada tingkat kemajuan sistem sampai analisis, desain, kode, pengujian, dan pemeliharaan. Sebagaimana ditunjukkan pada Gambar 3.1



Gambar 3. 1 Model *Waterfall* (Kersting, 2017)

3.3.1. Analisis Kebutuhan (*Requirements Definition*)

Tahap analisis merupakan langkah menganalisis kebutuhan-kebutuhan sistem informasi yang dibangun. Kebutuhan sistem informasi dibagi menjadi 2 yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan yang didapat berasal dari data yang telah terkumpulkan dan telah diolah yang berhubungan dengan perencanaan aktivitas distribusi pada perusahaan. Data-data tersebut kemudian dikelompokkan menjadi kebutuhan fungsional dan kebutuhan non-fungsional. Teknik pengumpulan data yang dilakukan antara lain:

1. Studi Pustaka

Studi pustaka ini dilakukan dengan tujuan sebagai dasar pembahasan penyusunan dasar teori yang digunakan dalam penelitian. Sumber yang digunakan sebagai studi pustaka berupa buku, jurnal, dan karya ilmiah dan penelitian yang sejenis sebelumnya.

2. Wawancara

Wawancara merupakan teknik pengumpulan data survei berupa tanya jawab peneliti dengan narasumber. Wawancara tersebut berupa percakapan langsung antar dua pihak atau lebih untuk mendapatkan informasi secara lisan dengan tujuan untuk memperoleh data yang dapat menjelaskan ataupun menjawab suatu permasalahan penelitian (Hasibuan, 2007). Peneliti melakukan wawancara kepada *manager* Larissa Aesthetic Center untuk mengetahui permasalahan yang ada dan memperoleh data – data produk yang nantinya akan diolah dalam penelitian ini. Data dan informasi yang diperoleh dari tahapan ini data penjualan produk.

3. Pengolahan Data

Tahap ini dimulai dengan menelaah data secara keseluruhan, pada tahap ini juga dilakukan penentuan kriteria apa saja yang layak untuk dijadikan parameter dari data yang sudah dikumpulkan, dan menentukan kebutuhan sistem berdasarkan studi literatur dan wawancara yang telah dilakukan.

3.3.2. Sistem dan Desain Software (*System and Software Design*)

Desain sistem informasi merupakan tahap pembuatan desain sistem informasi, dimana pada penelitian ini menggunakan *Unified Modeling Language* (UML) yang dirancang dengan konsep *Object-Oriented Programming* (OOP). Pemodelan UML yang digunakan sebagai berikut:

3.3.2.1. *Business Process*

Business process merupakan diagram yang menggambarkan proses dari sebuah sistem yang meliputi data apa yang diperlukan lalu data diolah untuk menghasilkan *output* yang diinginkan.

3.3.2.2. *Usecase Diagram*

Usecase diagram merupakan diagram yang menggambarkan fungsionalitas interaksi antara *user* dengan sistem informasi. *Usecase diagram* juga dapat menggambarkan hak akses dari *user*.

3.3.2.3. *Usecase Scenario*

Usecase Scenario berfungsi untuk menjelaskan alur sistem dari fitur yang ada pada *usecase diagram*, skenario juga berfungsi menjelaskan reaksi yang akan

terjadi pada sistem setelah menerima aksi dari aktor. *Usecase Scenario* menggambarkan keadaan normal saat setiap aktor mengakses sistem dan keadaan alternatif yang terjadi pada kondisi tertentu.

3.3.2.4. *Sequence Diagram*

Sequence Diagram menggambarkan aliran logika dan interaksi yang terjadi di dalam sistem sesuai urutan dijalankannya sistem tersebut. Interaksi tersebut meliputi, aktor yang mengakses fitur sesuai dengan *usecase diagram*, tampilan sistem, *controller*, model, dan pesan yang disampaikan jika terjadi suatu kondisi tertentu.

3.3.2.5. *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan urutan aktivitas dalam sebuah proses. Aktivitas tersebut sesuai dengan *scenario* yang berisi tugas user dan reaksi sistem dan digambarkan dalam bentuk diagram.

3.3.2.6. *Class Diagram*

Class diagram merupakan gambaran grafis dari struktur objek statis dari sebuah sistem informasi yang menunjukkan kelas objek yang tersusun dari hubungan antar kelas-kelas objek yang lain.

3.3.2.7. *Entity Relationship Diagram*

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

3.3.3. Implementasi Sistem (*Implementation and Unit Testing*)

Setelah desain sistem selesai dilakukan maka akan dilanjutkan ketahap pembuatan system dan implementasi. Pembuatan sistem meliputi penulisan kode program dan pembuatan basis data. Penulisan kode dilakukan menggunakan *tools* *Sublime Text 2* sebagai *editor* dengan bahasa pemograman HTML, PHP, CSS yang terintegrasi dengan DBMS MySQL sebagai manajemen basis data.

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

Tahap pengujian sistem bertujuan untuk mengetahui sejauh mana sistem ini dapat berjalan. *Integration Testing* berfungsi untuk mengetahui apakah sistem ini dapat berfungsi dengan baik sesuai dengan yang diharapkan, serta untuk

mengetahui letak kekurangan pada sistem. Penelitian ini melakukan pengujian sistem dengan cara sebagai berikut:

3.3.4.1. *White Box Testing*

White box testing merupakan cara pengujian dengan melihat modul yang telah dibuat dan program-program yang ada. Pengujian ini dilakukan oleh pembuat program (*developer*). Jika ada modul yang menghasilkan output yang tidak sesuai, maka baris-baris program, variable dan parameter yang terlibat pada unit tersebut satu persatu akan dicek dan diperbaiki, kemudian di compile ulang (Pressman, 2001).

3.3.4.2. *Black Box Testing*

Berbeda dengan *white box testing*, *Black box testing* merupakan metode pengujian perangkat lunak yang memeriksa fungsionalitas aplikasi. Pengujian ini lebih difokuskan kepada kerja program. Pengujian dilakukan dengan melibatkan mahasiswa secara acak sebagai pengguna.

3.3.5. *Pemeliharaan (Operation and Maintenance)*

Perangkat lunak yang sudah selesai akan mengalami perubahan. Perubahan biasanya berupa error sehingga diperlukan perbaikan dan pemeliharaan kepada sistem. Perubahan ini dilakukan agar sistem bersifat dinamis.

BAB 4. PENGEMBANGAN SISTEM

Bab ini menjelaskan mengenai tahapan-tahapan dalam perancangan sistem dengan menerapkan model *waterfall*. Tahapan yang dilakukan meliputi analisis kebutuhan fungsional dan non fungsional sistem, merancang desain sistem yaitu *business process*, *usecase diagram*, *scenario*, *activity diagram*, *sequence diagram*, *class diagram* dan *entity relationship diagram* (ERD), kemudian penulisan kode program, lalu pengujian sistem.

4.1 Tahapan Analisis Kebutuhan

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

Berdasarkan hasil studi pustaka, wawancara dan pengolahan data, tahap pertama yang dilakukan adalah mengelola data produk yang didapat dari hasil wawancara di Larissa Aesthetic Center. Tahap kedua yaitu menghitung prediksi kebutuhan produk periode selanjutnya menggunakan metode *Least Square*. Tahap ketiga yaitu mendeskripsikan SOP dari sistem yang akan dibangun. Tahap keempat menentukan kebutuhan fungsional dari sistem yang akan dibangun. Tahap kelima menentukan kebutuhan non-fungsional sistem yang akan dibangun.

4.1.1 Penerapan *Least Square*

Berdasarkan hasil studi literature yang diperoleh, metode *Least Square* digunakan untuk melakukan prediksi untuk kebutuhan produk di periode selanjutnya. Penerapan metode *Least Square* dapat dilihat pada persamaan 2.3

4.1.2 Data Kebutuhan

Berdasarkan hasil wawancara dengan manager Larissa Aesthetic Center diperoleh data kebutuhan produk kecantikan selama satu tahun terakhir yang dapat dilihat pada tabel 4.1.

Tabel 4. 1 Data Jumlah Penjualan Produk Larissa Aesthetic Center Tahun 2017

No	Bulan	Nama Barang			
		Apple	Milk	Green Tea	Chamomile
		Milk Cleanser	Cleanser Tea Tree	Milk Cleanser	Milk Cleanser
1	Januari	62	225	240	51
2	Februari	54	270	196	82
3	Maret	68	306	131	98
4	April	72	261	195	112
5	Mei	83	210	198	72
6	Juni	92	214	205	74
7	Juli	64	221	182	63
8	Agustus	50	247	188	80
9	September	73	154	215	98
10	Oktober	89	170	234	120
11	November	93	206	194	70
12	Desember	82	186	205	86

4.1.3 SOP (*Statement of purpose*)

Sistem Informasi Penjualan Produk Kecantikan Dengan Fitur Peramalan Menggunakan Metode *Least Square* merupakan sebuah sistem yang mampu mengendalikan persediaan produk. Ada beberapa data yang dibutuhkan agar sistem dapat menghasilkan persediaan produk yang optimal. Data yang dibutuhkan untuk sistem ini yaitu data penjualan setiap bulannya. Dari data tersebut, dilakukan perhitungan menggunakan metode *least square*. Dari hasil perhitungan tersebut akan diperoleh kebutuhan stok yang optimal untuk order pada periode berikutnya.

Selain itu sistem ini juga mampu mengelola data harga produk dan data penjualan. Dengan menggunakan sistem ini maka persediaan produk dapat dikendalikan dan dimonitoring sehingga tidak melebihi kapasitas gudang.

4.1.4 Fungsi Sistem

Fungsi sistem menggunakan fitur login yang dapat menentukan hak akses dari setiap user. Sistem akan melakukan autentifikasi *username* dan *password*. Selanjutnya sistem akan menampilkan sesuai dengan hak akses pengguna yang meliputi :

a. Manager

Manager ketika login berhasil, sistem akan menampilkan halaman home. Manager dapat mengakses beberapa fitur yaitu mengelola data user, melihat data produk, melihat data penjualan dan mengolah data prediksi.

b. Pegawai

Pegawai ketika login berhasil maka sistem akan menampilkan halaman home. Pegawai dapat mengakses beberapa fitur yaitu mengelola data produk dan mengelola data penjualan.

4.1.5 Kebutuhan Fungsional

Kebutuhan fungsional berisi fitur-fitur yang harus dipenuhi dalam sistem agar sistem berfungsi sesuai dengan tujuan dan kebutuhan pengguna. Kebutuhan fungsional dari Sistem Informasi Penjualan Produk Kecantikan Dengan Fitur Peramalan Menggunakan Metode *Least Square* yaitu :

- a. Sistem mampu mengelola data user untuk menyimpan data user dan hak akses sistem (tambah user, edit user dan hapus user)
- b. Sistem mampu mengelola data produk (tambah produk, edit produk dan hapus produk)
- c. Sistem mampu mengelola data penjualan (tambah data, edit data dan hapus data)
- d. Sistem mampu memprediksi penjualan di periode berikutnya
- e. Sistem mampu mengelola data user (tambah user, edit user dan hapus user)

4.1.6 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan fitur-fitur yang dimiliki untuk mendukung fungsionalitas sistem dalam memenuhi kebutuhan dari pengguna. Kebutuhan non-fungsional dari Sistem Informasi Penjualan Produk Kecantikan Dengan Fitur Peramalan Menggunakan Metode *Least Square* yaitu sebagai berikut :

- a. Sistem menggunakan username dan password untuk akses pengguna terhadap sistem.
- b. Sistem menggunakan tampilan yang mudah dimengerti oleh user untuk mempermudah pengoperasian.

4.2 Tahapan Desain Sistem

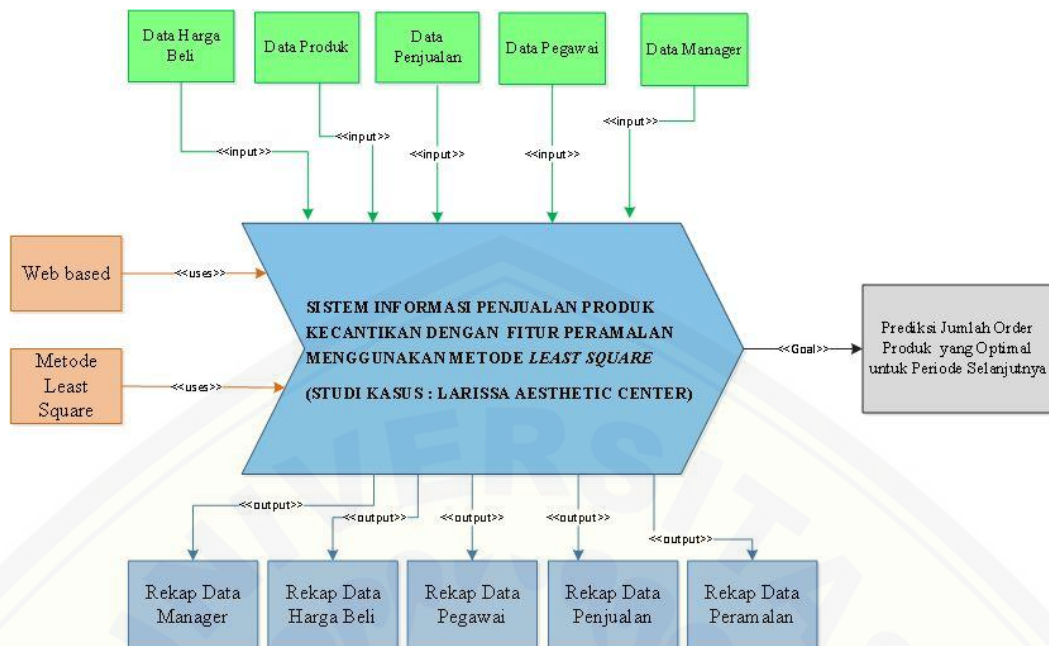
Desain sistem yang dibuat pada Sistem Penjualan Produk Kecantikan meliputi *business process*, *usecase diagram*, *scenario*, *sequence diagram*, *activity diagram*, *class diagram* dan *entity relation diagram*.

4.2.1 *Bussiness Process*

Bussiness process merupakan diagram yang menggambarkan data yang dibutuhkan oleh sistem berupa *input*, *output* dan *goal sistem* yang dibangun.

Business process terdiri dari :

- a. *Input* : Data yang dimasukkan ke dalam sistem
- b. *Output* : Data yang dihasilkan oleh sistem
- c. *Goal* : Tujuan sistem
- d. *Uses* : Platform yang digunakan pada sistem
- e. *Process* : Sistem yang bekerja

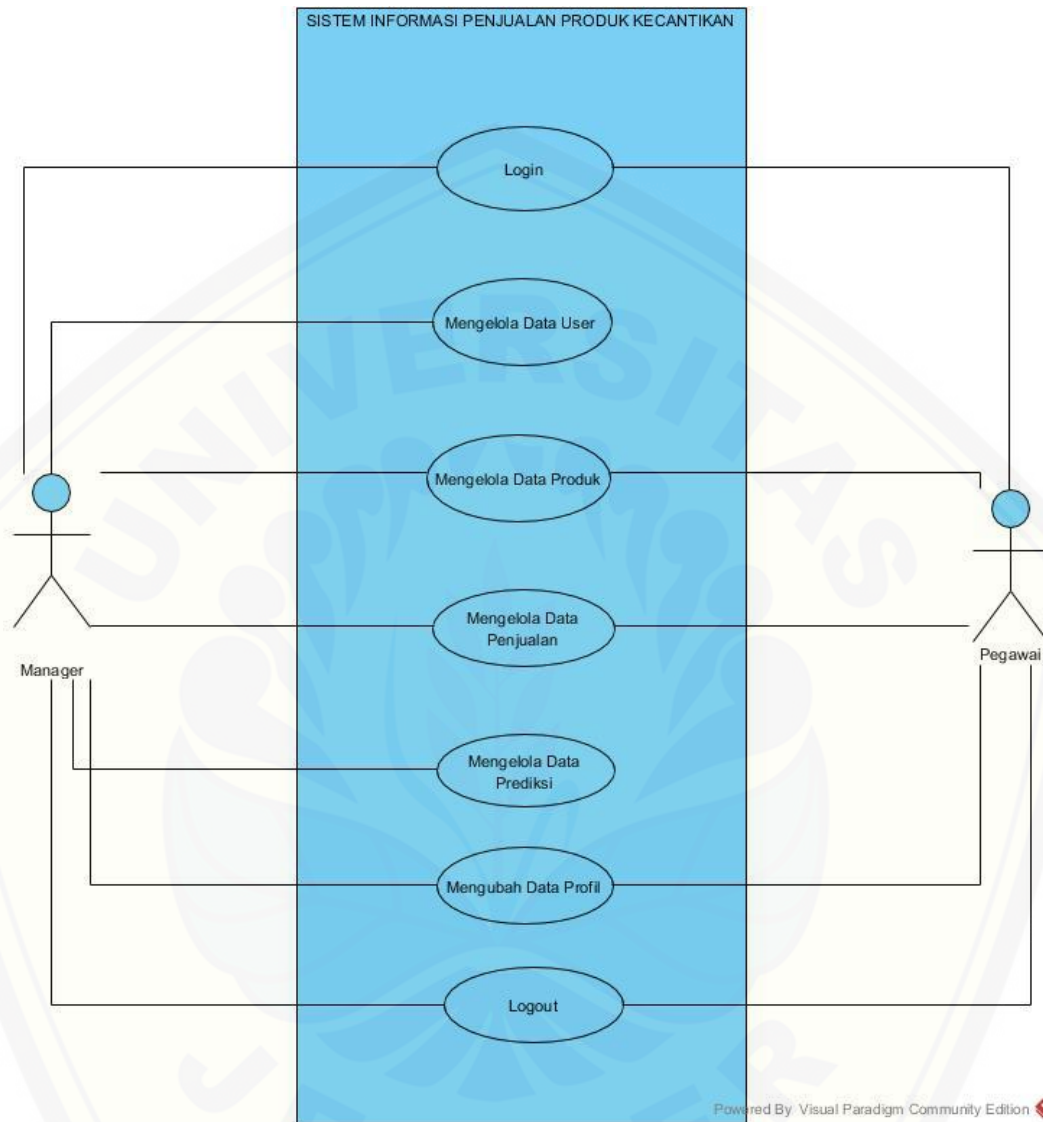


Gambar 4. 1 *Business Process* Sistem Informasi Penjualan Produk Kecantikan

Gambar 4.1 merupakan *business process* dari Sistem Informasi Penjualan Produk Kecantikan Dengan Fitur Peramalan Menggunakan Metode *Least Square*. *Business process* menjelaskan proses *input*, *output*, *goal* dan *uses* pada sistem. *Input* berupa data harga beli, data produk, data penjualan, data pegawai dan data manager. *Output* berupa rekap data manager, rekap data harga beli, rekap data pegawai, rekap data penjualan dan rekap data peramalan. *Uses* yang dipakai berupa web based dan metode *least square*. *Goal* dari pembuatan sistem ini adalah Prediksi Jumlah Order Produk yang Optimal untuk Periode Selanjutnya dan process berupa sistem informasi pengendalian persediaan stok produk menggunakan metode *least square*.

4.2.2 *Usecase Diagram*

Usecase diagram merupakan diagram yang menjelaskan interaksi antara actor dengan sistem yang dibangun. Dari *usecase diagram* diketahui fitur-fitur apa saja yang terdapat dalam sistem dan aktor siapa saja yang dapat mengaksesnya. *Usecase diagram* ditunjukkan pada gambar 4.2



Gambar 4. 2 *Usecase Diagram* Sistem Informasi Penjualan Produk Kecantikan

Gambar 4.2 menunjukkan usecase diagram Sistem Informasi Pejualan Produk Kecantikan yang terdiri dari 2 aktor yaitu Manager dan Pegawai. Adapun deskripsi dari masing-masing actor dapat dilihat pada Tabel 4.2

Tabel 4. 2 Deskripsi Aktor

No.	Aktor	Hak Akses
1.	Manager	Manager merupakan aktor yang memiliki hak akses untuk mnegelola data user (tambah, edit dan hapus), melihat data produk, melihat data penjualan dan mengolah data prediksi
2.	Pegawai	Pegawai merupakan aktor yang memiliki hak akses untuk mengelola data produk (tambah, edit dan hapus) dan mengolah data penjualan (tambah, edit dan hapus)

Terdapat 7 *usecase* dalam *usecase diagram*. Deskripsi dari 7 *usecase* tersebut terdapat pada tabel 4.3

Tabel 4. 3 Deskripsi *Usecase*

No	Usecase	Penjelasan
1	Login	Merupakan usecase yang menggambarkan proses login atau autentifikasi untuk masuk ke sistem
2	Mengelola Data User	Merupakan usecase yang menggambarkan proses menampilkan, menambah dan mengubah data user. Hak akses ini hanya dapat digunakan oleh Manager
3	Melihat Data Produk	Merupakan usecase yang menggambarkan proses menampilkan data produk. Hak akses ini hanya dapat digunakan oleh Manager
4	Mengelola Data Produk	Merupakan usecase yang menggambarkan proses menampilkan, mengubah dan menambah data produk. Hak akses ini hanya dapat digunakan oleh Pegawai
5	Mengelola Data Pejualan	Merupakan usecase yang menggambarkan proses menampilkan, mengubah dan menambah data

		penjualan. Hak akses ini hanya dapat digunakan oleh Pegawai
6	Mengelola Data Prediksi	Merupakan usecase yang menggambarkan proses menampilkan data prediksi. Hak akses ini hanya dapat digunakan oleh Manager
7	Logout	Merupakan usecase yang menggambarkan proses keluar dari sistem

4.2.3 Usecase Scenario

Usecase Scenario adalah menjelaskan alur dari aksi aktor dan reaksi sistem sesuai dengan kebutuhan fungsional pada sistem. *Usecase Scenario* sistem informasi penjualan produk kecantikan dengan menggunakan metode *least square* adalah sebagai berikut:

1. Usecase Scenario Mengelola Data User

Penjelasan urutan aksi aktor untuk mengelola data user (menambah, mengubah dan menghapus) dan reaksi sistem pada skenario normal dan *scenario alternatif* terdapat pada tabel 4.4.

Tabel 4. 4 *Usecase Scenario* Mengelola Data User

Nama	Mengelola Data User
Aktor	Manager
Pre-Kondisi	Manager berhasil melakukan login ke sistem
Pra-Kondisi	Manager berhasil menambah, mengedit dan mengubah status user
SKENARIO NORMAL	
“Melihat data user”	
Aktor	Sistem
1. Memilih Pengaturan Akun	
	2. Menampilkan tabel user yang berisi :

-
- a. No
 - b. Nama Akun
 - c. Kata Sandi
 - d. Level
 - e. Status
 - f. Action yang berisi tombol hapus dan tombol edit

Serta terdapat tombol tambah data dan search

SKENARIO NORMAL

“Menambah data user”

Aktor	Sistem
1. Klik tombol Tambah Data	2. Menampilkan top up form tambah data user yang berisi: <ul style="list-style-type: none"> a. Username b. Password
3. Mengisi form tambah data user	Serta terdapat tombol save dan tombol close
4. Klik tombol Save	5. Menyimpan data user kedalam tabel user pada database
	3. Menampilkan tabel user yang berisi : <ul style="list-style-type: none"> a. No b. Nama Akun c. Kata Sandi d. Level

-
- e. Status
 - f. Action yang berisi tombol hapus dan tombol edit

Serta terdapat alert success ! Data berhasil di tambahkan

SCENARIO ALTERNATIF

“Data belum diisi”

Aktor	Sistem
3a. Tidak mengisi form tambah data user	
4a. Klik tombol Simpan	
	5a. Menampilkan notifikasi “ Please fill out this field”

SCENARIO ALTERNATIF

“Batal menambahkan data”

Aktor	Sistem
3b. Mengisi form tambah data user	
4b. Klik tombol Close	
	4b. Menampilkan tabel user yang berisi : <ul style="list-style-type: none"> a. No b. Nama Akun c. Kata Sandi d. Level e. Status f. Action yang berisi tombol hapus dan tombol edit

Serta terdapat tombol tambah data dan search

SKENARIO NORMAL**“Mengedit data user”**

Aktor	Sistem
1. Klik tombol Edit	
	2. Menampilkan pop up form edit data user yang berisi : <ol style="list-style-type: none"> a. Username b. Password <p>Serta terdapat tombol Save dan Close kembali pada halaman form Edit data user</p>
3. Mengubah username dan password	
4. Klik tombol simpan	
	5. Menyimpan data user ke dalam tabel user pada database
	6. Menampilkan tabel user yang berisi : <ol style="list-style-type: none"> a. No b. Nama Akun c. Kata Sandi d. Level e. Status f. Action yang berisi tombol hapus dan tombol edit <p>Serta terdapat tombol tambah data dan search</p>

SCENARIO ALTERNATIF**“Data belum diisi”**

Aktor	Sistem
--------------	---------------

3a. Tidak mengisi username
dan password

4a. Klik tombol Simpan

5a. Menampilkan notifikasi “ Please fill out this
field”

SCENARIO ALTERNATIF

“Batal mengubah data”

Aktor

Sistem

3c. Mengubah username dan password

4c. Klik tombol Kembali

5c. Menampilkan tabel user yang berisi
:

- a. No
- b. Nama Akun
- c. Kata Sandi
- d. Level
- e. Status
- f. Action yang berisi tombol
hapus dan tombol edit

Serta terdapat tombol tambah data dan
search

SKENARIO NORMAL

“Mengubah Status User”

Aktor

Sistem

1. Klik tombol Hapus

2. Menampilkan pop up Delete Data
User, Dengan pesan Apakah anda
yakin akan menghapus user ?

3. Klik tombol Ok

Menampilkan tabel user yang berisi :

- a. No
- b. Nama Akun
- c. Kata Sandi
- d. Level
- e. Status
- f. Action yang berisi tombol hapus dan tombol edit

Serta terdapat alert Succes ! Data Sudah di hapus

2. *Usecase Scenario* Mengelola Data Produk (Manager)

Penjelasan urutan aksi aktor untuk mengelola data produk (menambah, mengubah dan menghapus) dan reaksi sistem pada *scenario* normal dan *scenario* alternatif terdapat pada tabel 4.5 *Usecase Scenario* Mengelola Data Produk (Pegawai)

Tabel 4. 5 *Usecase Scenario* Mengelola Data Produk

Nama	Mengelola Data Produk
Aktor	Pegawai dan Manager
Pre-Kondisi	Pegawai dan Manager melakukan login ke sistem
Pra-Kondisi	Pegawai dan Manager berhasil menambah dan melihat detail data stok bahan baku

SKENARIO NORMAL

“Melihat Data Produk”

Aktor	Sistem
--------------	---------------

1. Memilih menu Data Produk

-
2. Menampilkan tabel data Produk yang berisi :
 - a. No
 - b. Nama Prouct
 - c. Harga Beli
 - d. Status
 - e. Action yang berisi tombol edit dan hapus

Serta terdapat tombol tambah data

<p>SKENARIO NORMAL “Menambah Data Produk”</p>

Aktor	Sistem
<ol style="list-style-type: none"> 1. Klik tombol Tambah 	<ol style="list-style-type: none"> 2. Menampilkan pop up form tambah Data Product yang berisi : <ol style="list-style-type: none"> a. Nama b. Harga Jual c. Harga Beli <p style="text-align: center;">Serta terdapat tombol Close dan save</p>
<ol style="list-style-type: none"> 3. Mengisi nama, Harga Jual dan Harga Beli 	<ol style="list-style-type: none"> 5. Menyimpan data Produk ke tabel product pada database
<ol style="list-style-type: none"> 4. Klik tombol Save 	<ol style="list-style-type: none"> 3. Menampilkan tabel data Produk yang berisi : <ol style="list-style-type: none"> a. No b. Nama Prouct

-
- c. Harga Beli
 - d. Status
 - e. Action yang berisi tombol edit dan hapus

Serta terdapat alert success ! data berhasil ditambahkan

SCENARIO ALTERNATIF

“Data belum diisi”

Aktor	Sistem
3a. Tidak mengisi nama bahan baku dan jumlah	
4a. Klik tombol Simpan	
	5a. Menampilkan notifikasi “Please fill out this field”

SCENARIO ALTERNATIF

“Batal menambah data”

Aktor	Sistem
3b. Mengubah nama, harga jual dan harga beli	
4b. Klik tombol close	
	4. 5b. Menampilkan tabel data Produk yang berisi : <ul style="list-style-type: none"> a. No b. Nama Prouct c. Harga Beli d. Status e. Action yang berisi tombol edit dan hapus

Serta terdapat tombol tambah data

SKENARIO NORMAL

“Edit Data produk”

Aktor

Sistem

1. Klik tombol Edit

6. Menampilkan pop up form Edit Data

Product yang berisi :

- a. Nama
- b. Harga Jual
- c. Harga Beli

Serta terdapat tombol Close dan save

2. Mengubah nama, Harga jual
dan Harga beli

3. Klik tombol save

5. Menampilkan tabel data Produk yang
berisi :

- a. No
- b. Nama Prouct
- c. Harga Beli
- d. Status
- e. Action yang berisi tombol edit dan
hapus

Serta terdapat alert success ! data berhasil
dirubah

SCENARIO ALTERNATIF

“Batal edit data Produk”

Aktor

Sistem

2a. Mengubah nama, harga jual
dan harga beli

4b. Klik tombol close

6. 5b. Menampilkan tabel data Produk yang
berisi :

- f. No
- g. Nama Prouct
- h. Harga Beli
- i. Status
- j. Action yang berisi tombol edit dan
hapus

Serta terdapat tombol tambah data

SCENARIO ALTERNATIF

“Mengubah Status Produk”

Aktor

Sistem

1. Pilih tombol hapus

2. Menampilkan pop up Status Data
product

3. pilih OK

7. Menampilkan tabel data Produk yang
berisi :

- a. No
 - b. Nama Prouct
 - c. Harga Beli
 - d. Status
 - e. Action yang berisi tombol edit dan
hapus
-

Serta terdapat alert success status data berhasil di rubah

SCENARIO ALTERNATIF
“Batal mengubah status Produk”

Aktor	Sistem
3a. pilih close	4a. Menampilkan tabel data Produk yang berisi : <ul style="list-style-type: none"> a. No b. Nama Prouct c. Harga Beli d. Status e. Action yang berisi tombol edit dan hapus Serta terdapat tombol tambah data

3. *Usecase Scenario* Mengelola Data Penjualan (Pegawai)

Penjelasan urutan aksi aktor untuk mengelola data penjualan (menambah, mengubah dan menghapus) dan reaksi sistem pada skenario normal dan skenario alternative terdapat pada Lampiran A (*Usecase Scenario*).

4. *Usecase Scenario Login* (Manager dan Pegawai)

Penjelasan urutan aksi aktor untuk melakukan login dan reaksi sistem pada skenario normal dan skenario alternative terdapat pada Lampiran A(*Usecase Scenario*).

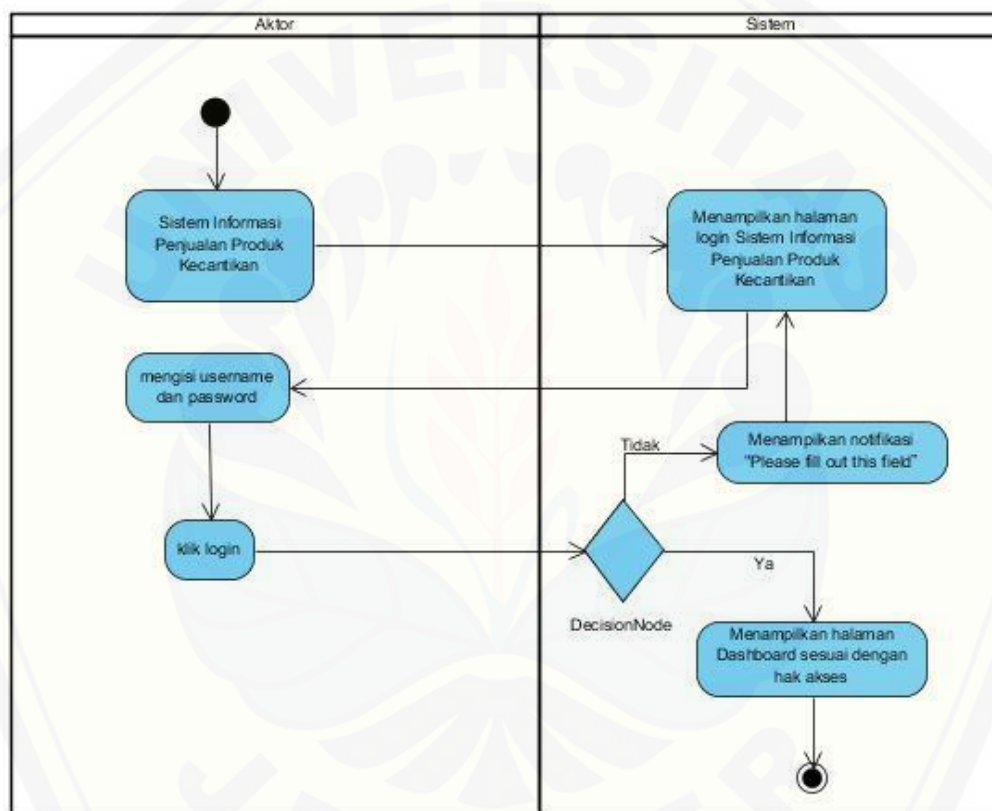
5. *Usecase Scenario Logout* (Manager dan Pegawai)

Penjelasan urutan aksi aktor untuk melakukan login dan reaksi sistem pada skenario normal dan skenario alternative terdapat pada Lampiran A(*Usecase Scenario*).

4.2.4 Activity Diagram

1. Activity Diagram Login

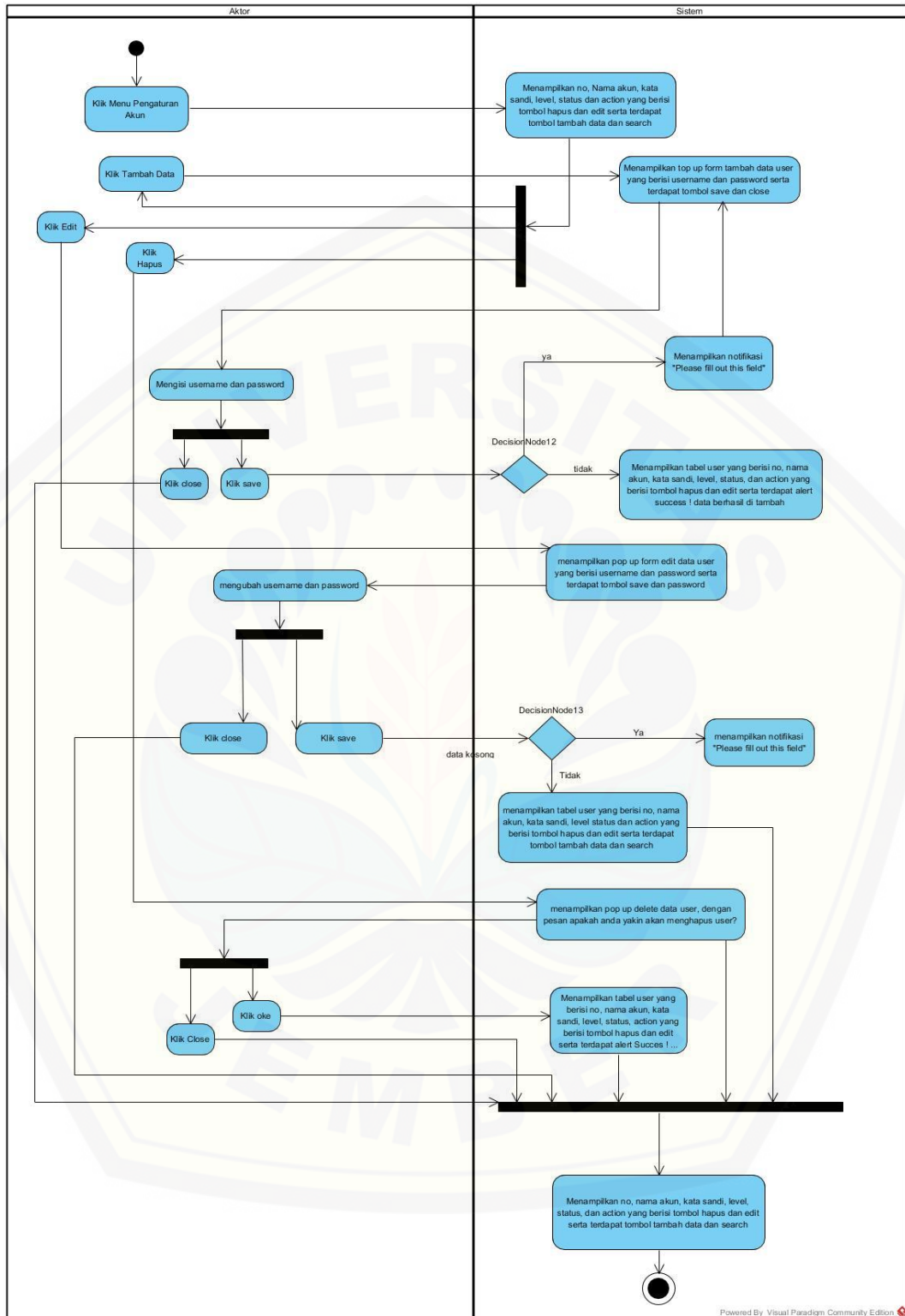
Activity diagram login menggambarkan alur aktifitas dari proses login agar pengguna dapat mengakses fitur sistem. Aktor yang melakukan login pada sistem ini adalah Manager dan Pegawai. Setiap aktor harus memiliki username dan password yang terdaftar pada sistem agar dapat melakan login. *Activity diagram* dijelaskan pada gambar 4.3 berikut.



Gambar 4. 3 Activity Diagram Login

2. Activity Diagram Mengelola Data User

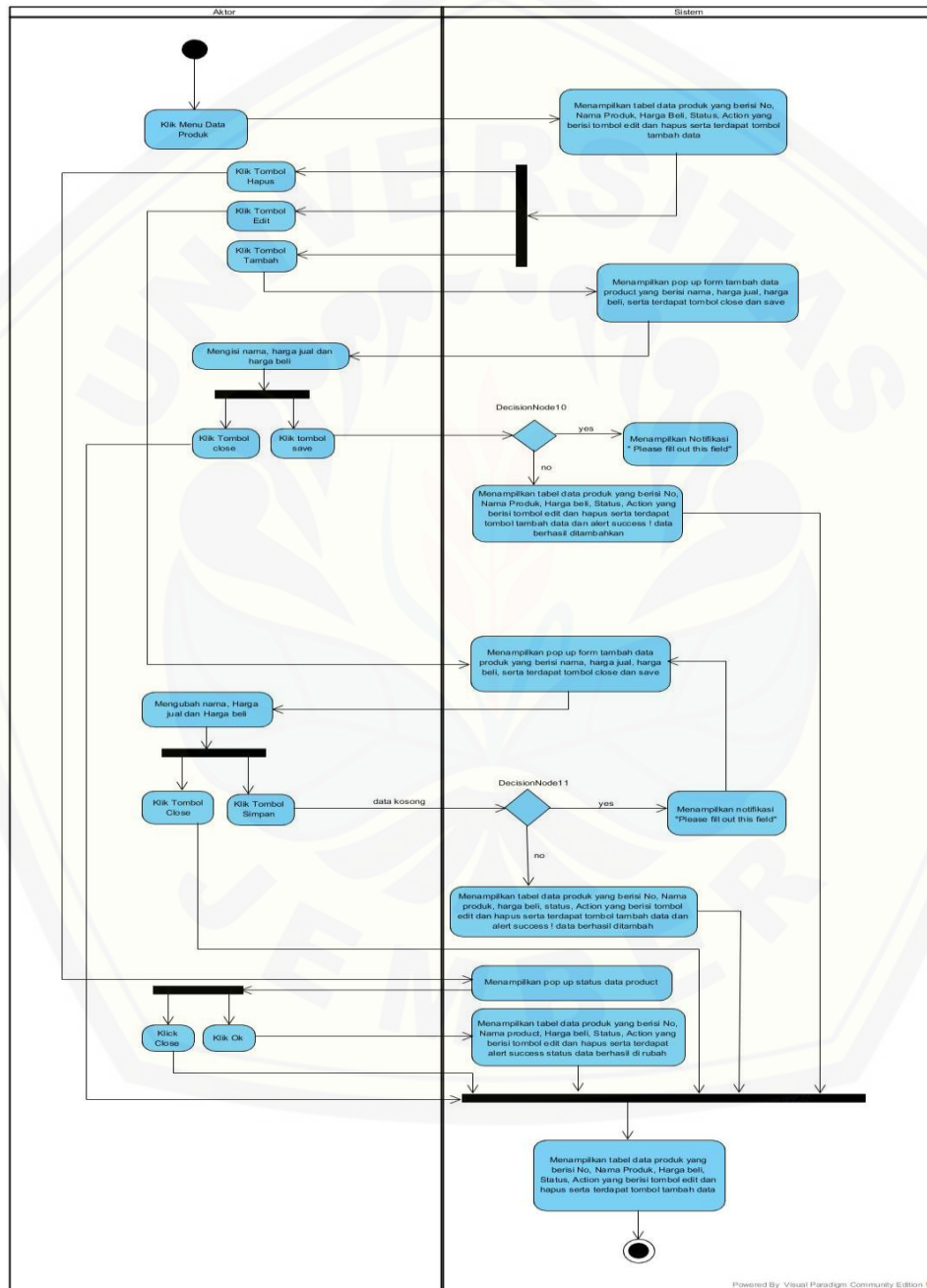
Activity diagram mengelola data user menggambarkan alur aktifitas proses menambah data user, mengubah data user dan menghapus data user yang dilakukan oleh Manager. *Activity diagram* ini menjelaskan mengenai alur yang harus dilakukan Manager untuk menambah, mengubah serta menghapus data user dan reaksi sistem ketika manager memberikan aksi terhadap sistem. *Activity diagram* dijelaskan pada gambar 4.4 berikut.



Gambar 4. 4 Activity Diagram Mengelola Data User

3. Activity Diagram Mengelola Data Produk

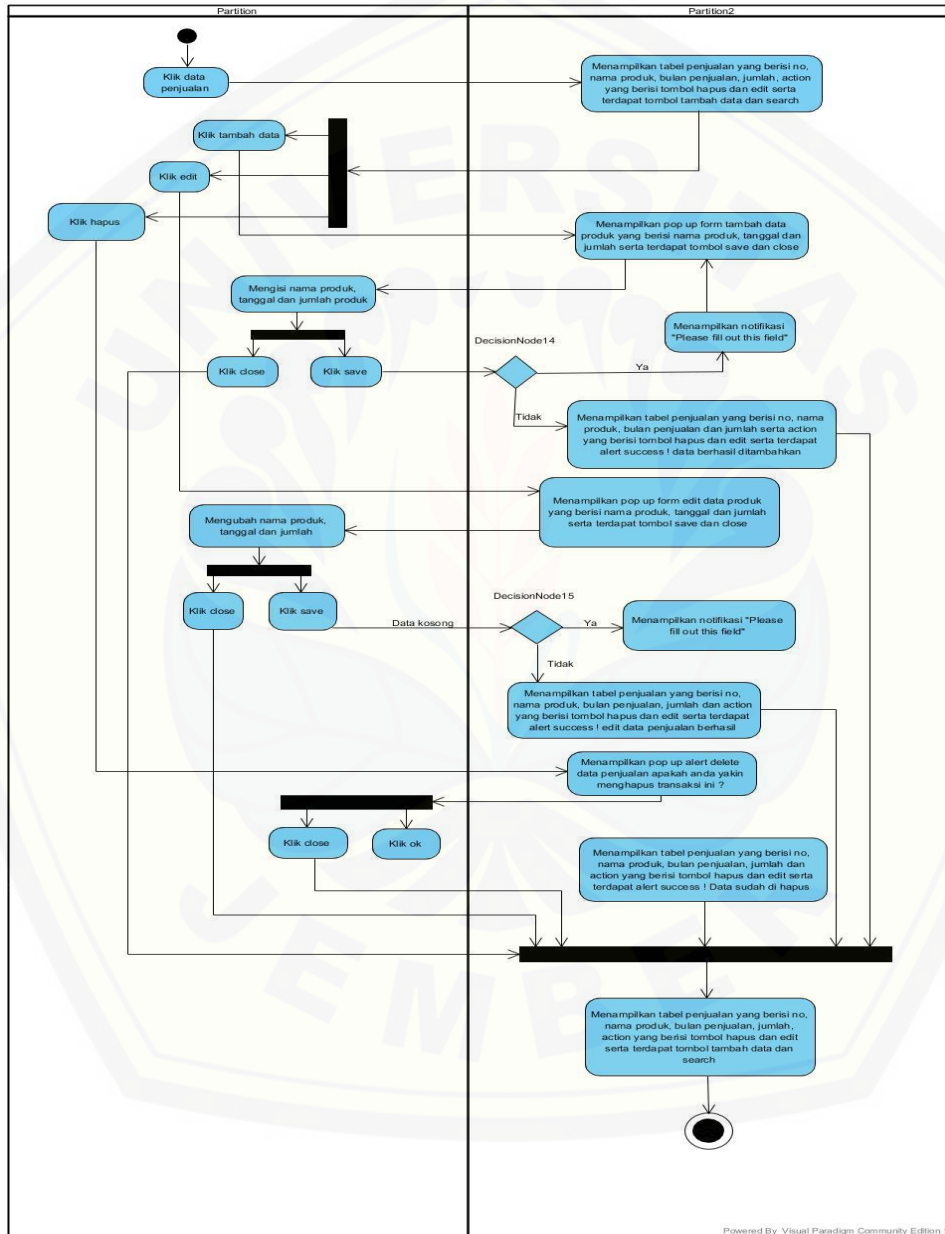
Activity diagram ini menjelaskan mengenai alur yang harus dilakukan pegawai untuk menambah, mengedit serta menghapus data produk dan reaksi sistem ketika pegawai memberikan aksi terhadap sistem. Activity diagram mengelola data produk dijelaskan pada gambar 4.3 berikut.



Gambar 4. 5 Activity Diagram Mengelola Data Produk

4. Activity Diagram Mengelola Data Penjualan

Activity diagram ini menjelaskan mengenai alur yang harus dilakukan pegawai untuk menambah, mengedit serta menghapus data penjualan dan reaksi sistem ketika bagain produksi memberikan aksi terhadap sistem. Activity diagram mengelola data produk terdapat pada gambar 4.4 berikut.



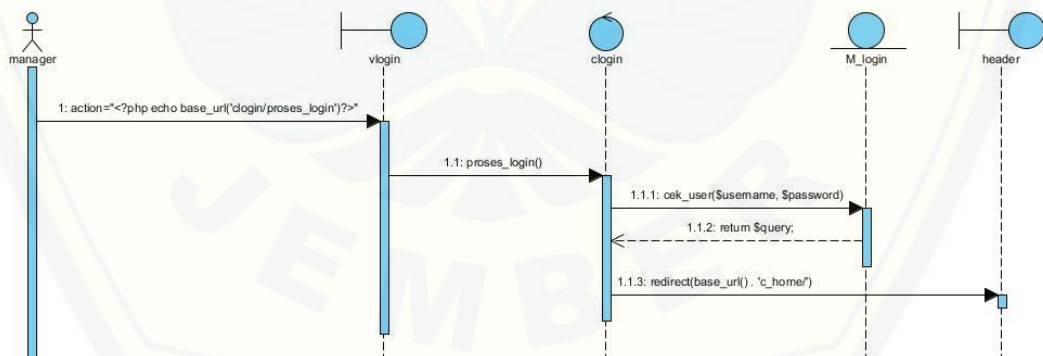
Gambar 4. 6 Activity Diagram Data Penjualan

4.2.5 Sequence Diagram

Sequence diagram adalah dokumentasi berupa diagram yang menampilkan interaksi di dalam sistem sesuai dengan urutan dijalankannya sistem tersebut. Interaksi tersebut meliputi aktor yang mengakses fitur tersebut sesuai dengan *use case diagram*, tampilan sistem, *controller*, model, dan pesan yang disampaikan jika terjadi kondisi tertentu. *Sequence diagram* digunakan untuk menggambarkan skenario dan aliran logika dalam sistem sehingga tahapan yang dibangun jelas dan teratur. *Sequence diagram* dari Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* (Studi Kasus : Larissa Aesthetic Center) adalah sebagai berikut :

1. *Sequence Diagram* Login (Manager)

Sequence diagram ini menggambarkan scenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian atau event untuk melakukan proses login. Actor yang dapat mengelola data laporan penjualan adalah admin. Pada *sequence* ini terdapat *class view login.php* dan *headeradmin.php class controller c_login.php* dan *c_home.php*, *class model m_login.php* serta *method-method* yang dibutuhkan dalam proses ini. *Sequence diagram* login admin dijelaskan pada gambar 4.5 berikut.

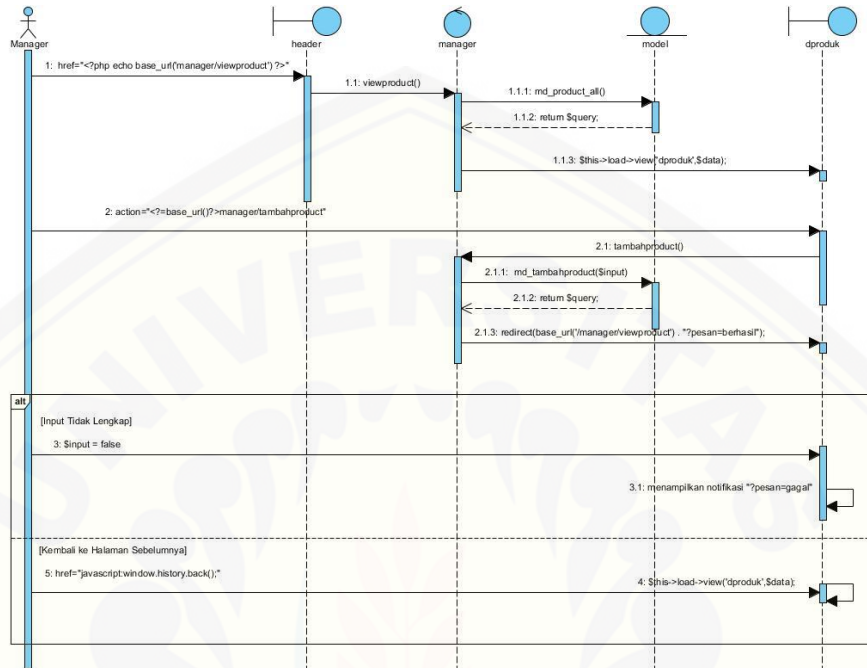


Gambar 4. 7 *Sequence Diagram* Login Manager

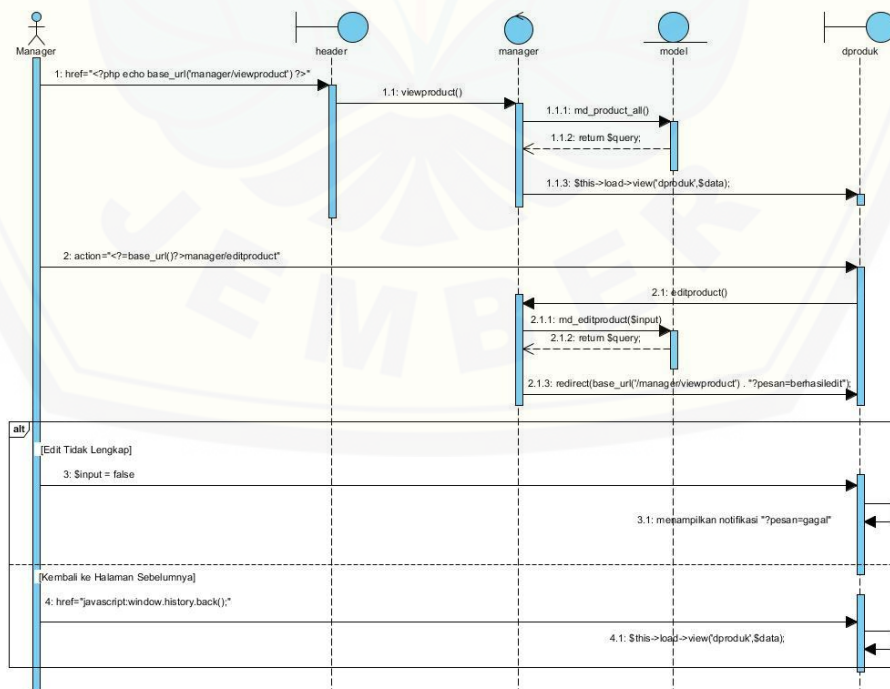
2. *Sequence Diagram* Mengelola Data Produk (Manager)

Sequence diagram ini menggambarkan *scenario* atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian atau event untuk Mengelola data produk yaitu menambah data produk, mengubah data produk dan

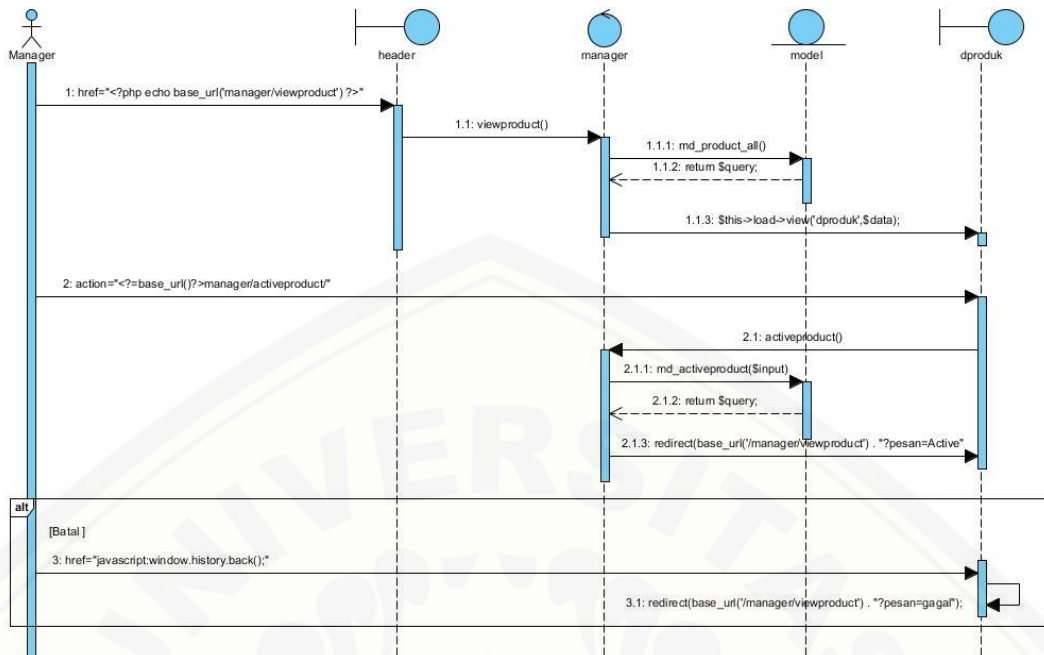
mengubah aktivasi data produk. Aktor yang dapat menginput data produk adalah Manager dan Pegawai. *Sequence diagram* mengelola data produk akan dijelaskan pada gambar 4.6 – 4.8



Gambar 4. 8 *Sequence Diagram* Input Data Produk



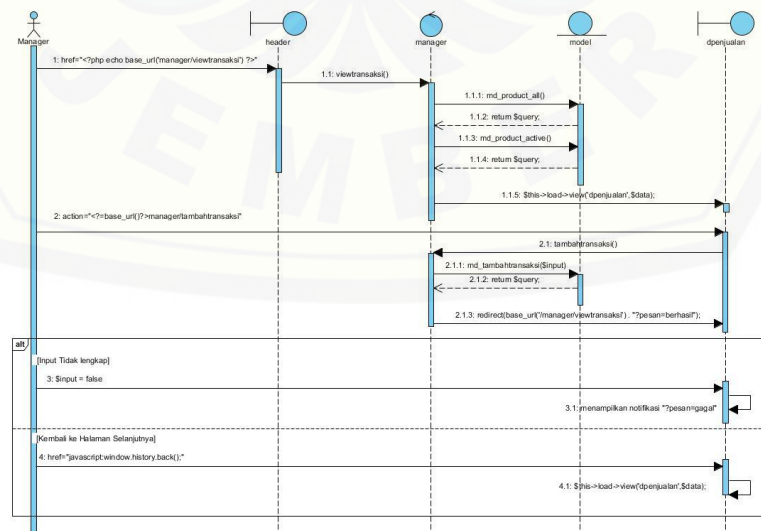
Gambar 4. 9 *Sequence Diagram* Edit Produk



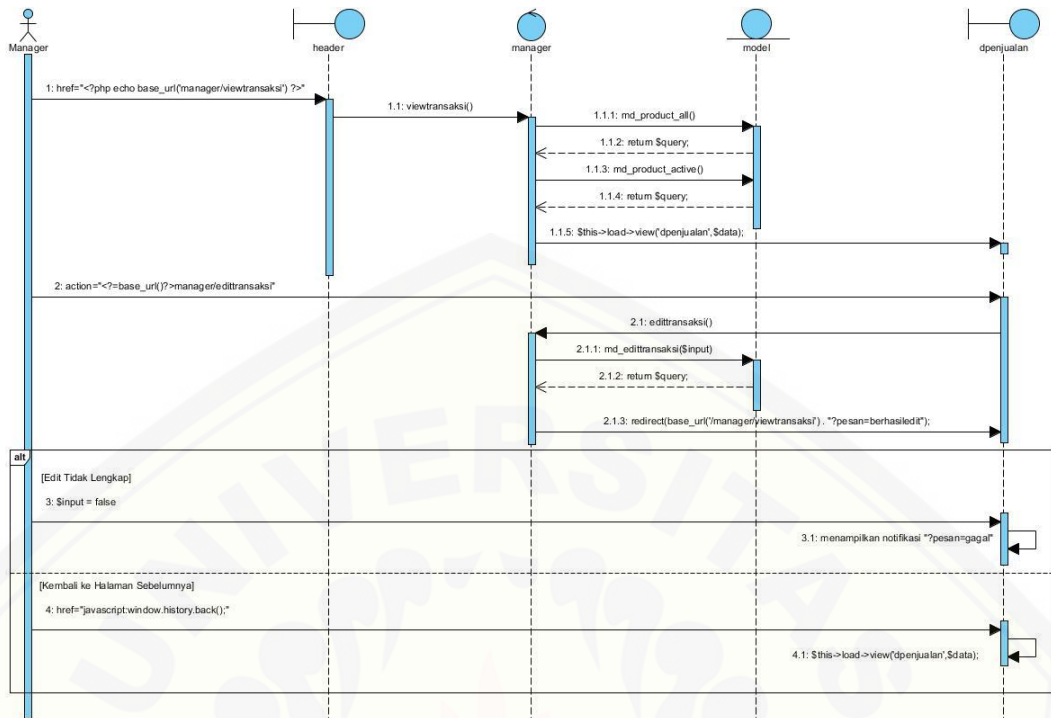
Gambar 4. 10 Sequence Diagram Aktivasi Produk

3. Sequence Diagram Mengelola Data Penjualan (Manager)

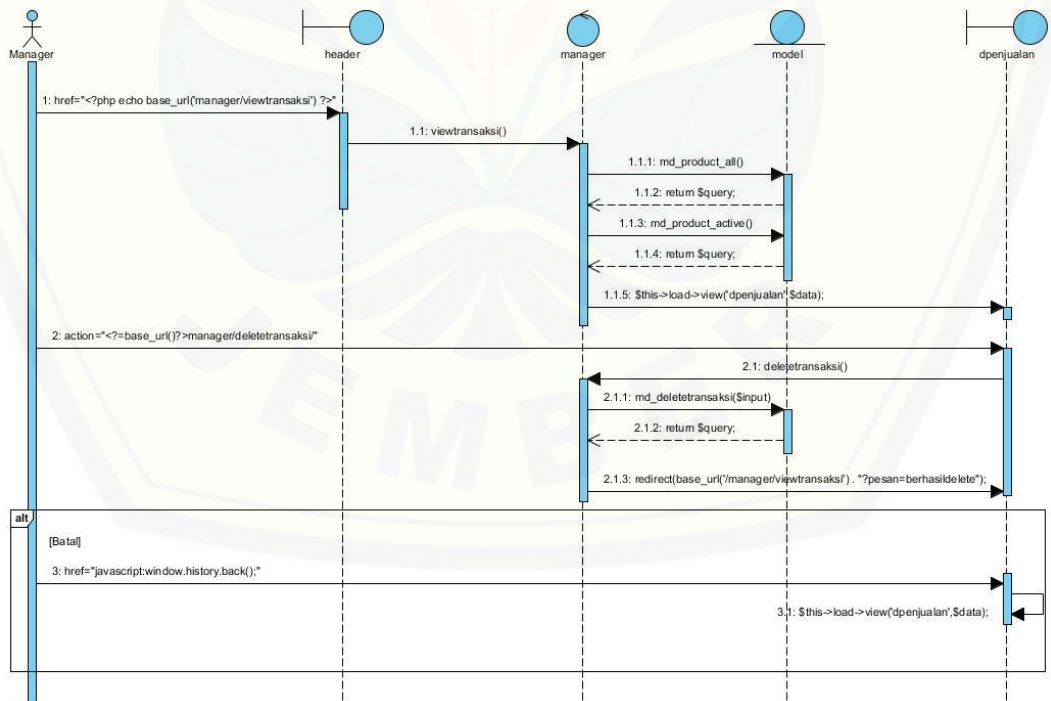
Sequence diagram ini menggambarkan *scenario* atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian atau event untuk mengelola data Penjualan yaitu menambah data penjualan, mengubah data penjualan dan menghapus data penjualan. Aktor yang dapat mengelola data penjualan adalah manager. Sequence diagram mengelola data penjualan dijelaskan pada gambar 4.9 – 4.11 berikut.



Gambar 4. 11 Sequence Diagram Tambah Data Penjualan



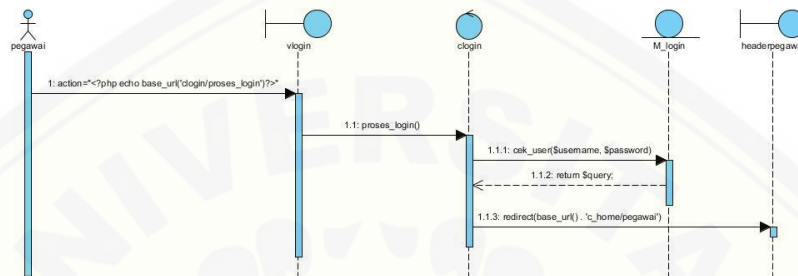
Gambar 4. 12 Sequence Diagram Ubah Data Pejualan



Gambar 4. 13 Sequence Diagram Menghapus Data Pejualan

4. *Sequence Diagram Login (Pegawai)*

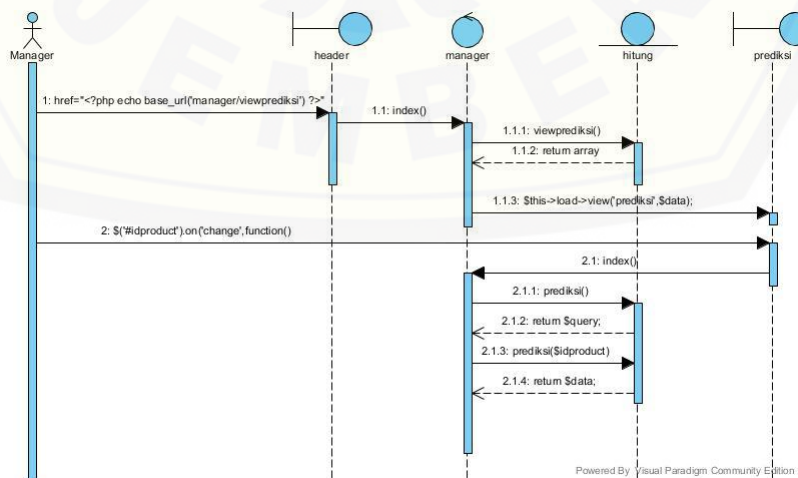
Sequence diagram ini menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian atau event untuk melakukan proses login. Aktor yang dapat mengelola data laporan penjualan adalah Pegawai. *Sequence diagram* login Pegawai akan dijelaskan pada gambar 4.12 berikut.



Gambar 4. 14 *Sequence Diagram Login Pegawai*

5. *Sequence Diagram Peramalan (Manager)*

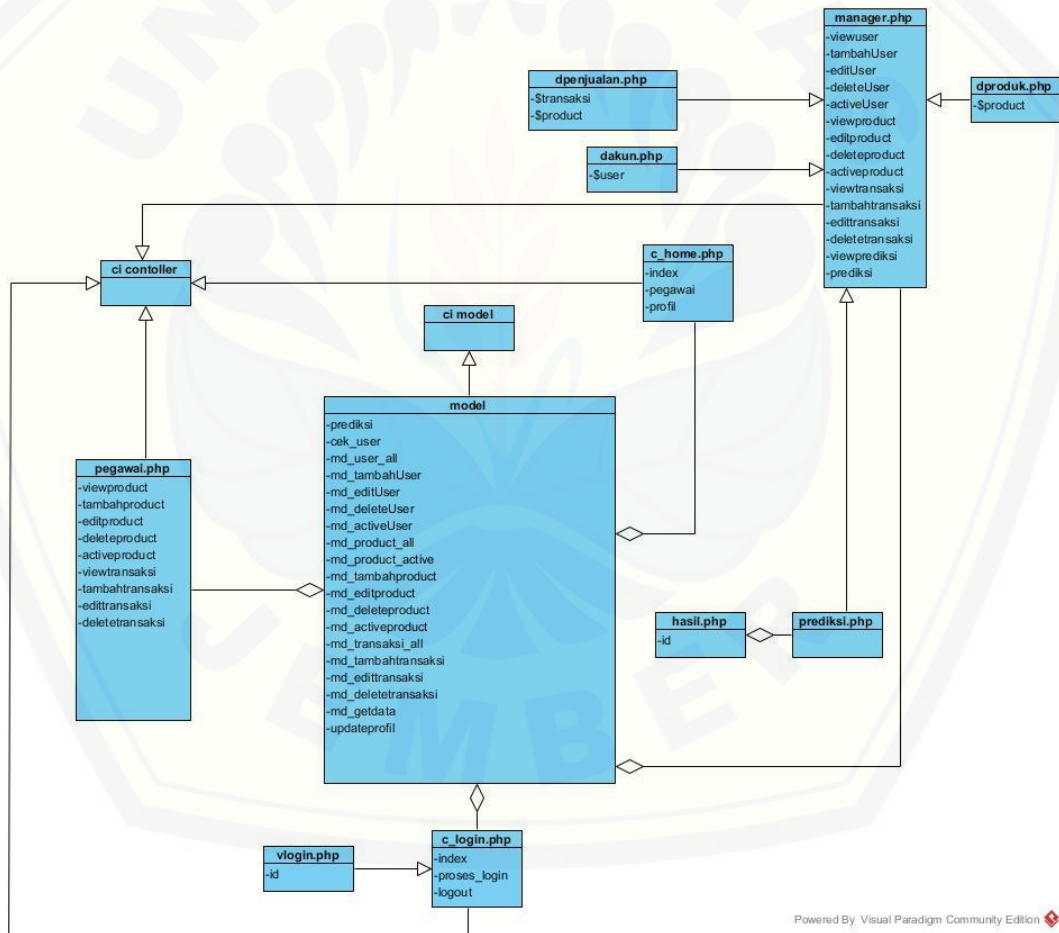
Sequence diagram ini menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian atau event untuk meramalkan produk kecantikan. Aktor yang dapat meramalkan produk kecantikan adalah manager. Pada *sequence* ini terdapat class view prediksi.php, header.php, footer.php, class controller manager.php, class model hitung.php serta method-method yang dibutuhkan dalam proses ini. *Sequence diagram* peramalan terdapat pada gambar 4.13.



Gambar 4. 15 *Sequence Diagram Peramalan*

4.2.6 Class Diagram

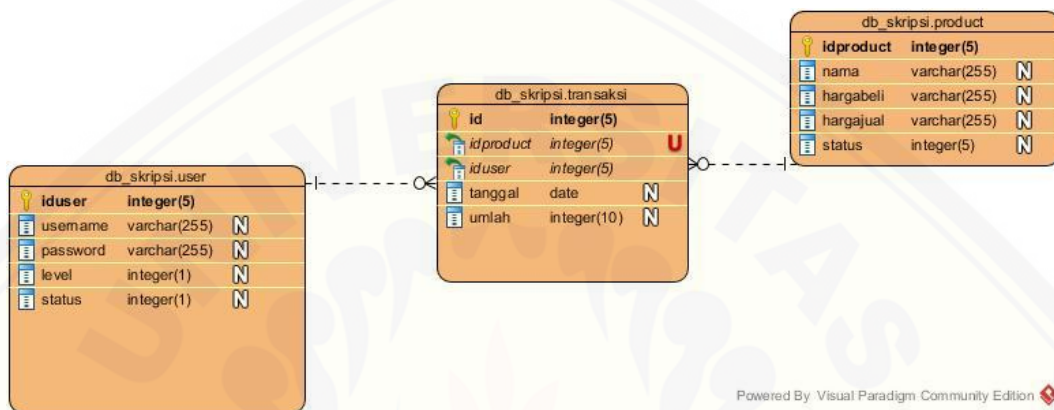
Class diagram menggambarkan hubungan antara kelas yang digunakan untuk membangun suatu sistem. Dalam paradigma OOP (Object Oriented Programming) terdapat 3 jenis kelas yaitu model, view dan controller. Berdasarkan sequence diagram yang telah dibangun, class diagram Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode Least Square (Studi Kasus : Larissa Aesthetic Center) terdiri dari 1 class model dan 2 class controller. Pada class tersebut memiliki fungsi masing-masing, class controller berfungsi untuk mengakses data yang ada pada class Model. Class diagram sistem dapat dilihat pada Gambar 4.13.



Gambar 4. 16 Class Diagram

4.2.7 Entity Relations Diagram (ERD)

Entity relationship diagram (ERD) sistem informasi pengendalian persediaan bahan baku menggunakan metode *least square* merupakan gambaran komponen dan struktur database yang saling berhubungan untuk digunakan dalam pembuatan sistem. ERD yang diimplementasikan pada sistem ini terdiri dari 3 entitas yang dapat dilihat pada gambar 4.14.



Gambar 4. 17 Entity Relationship Diagram (ERD)

4.3 Tahapan Implementasi Sistem

Tahap implementasi merupakan tahap penulisan kode program sesuai desain yang telah dirancang sebelumnya. Bahasa pemrograman yang digunakan dalam sistem ini adalah bahasa Pemrograman PHP dengan *framework Code Igniter* (CI) sedangkan DBMS yang digunakan adalah MySQL. Penulisan kode program ini ditulis dalam 3 bagian kelas yaitu *model*, *view* dan *controller*. Penulisan kode program dijelaskan pada sub sub bab berikut.

4.3.1 Kode Program Peramalan

Tabel 4. 6 Kode Program Pada *Class Model m_hitung*

Kode Program
<pre> <?php if (!defined('BASEPATH')) exit('No direct script access allowed'); </pre>


```
class Hitung extends CI_Model {
    function __construct(){
        parent::__construct();
    }
    public function prediksi($idproduct) {

        $this->db->where('tanggal',date('Y-m-d'));
        $this->db->where('idproduct',$idproduct);
        $query = $this->db->get('transaksi');

        $this->db->where('id',$idproduct);
        $data['product'] = $this->db->get('product')->row_array();
        // var_dump($this->db->last_query());

        $jumlah=$query->num_rows();
        if($jumlah>0){
            $genap=$jumlah%2==0?true:false;
            $tengah=($genap?((($jumlah)/2)+1:((($jumlah+1)/2))-1;
            $datas=$query->result_array();
            $data[$tengah]['x']=$genap?-1:0;
            $x=$data[$tengah]['x'];
            $sx2=pow($x, 2);
            $jumlah = 0;
            $i=0;
            $data['x']=array();
            $data['jumlahx'] = 0;
            $data['jumlahx2'] = 0;
            $data['jumlahxy'] = 0;
            $data['xprediksi'] = 0;

            foreach ($datas as $a) {
                if($genap){
                    if($i<$tengah){
                        $data['x'][$i]= -((2*($tengah-$i))-1);
                        $data['x2'][$i] = pow($data['x'][$i],2);
                        $data['xy'][$i] = $data['x'][$i]*$a['jumlah'];
                    }
                }
            }
        }
    }
}
```

```

    $data['jumlahx'] = $data['jumlahx']+$data['x'][$i];
    $data['jumlahx2'] = $data['jumlahx2']+$data['x2'][$i];
    $data['jumlahxy'] = $data['jumlahxy']+$data['xy'][$i];
}
else{
    $data['x'][$i] = ($i-$tengah)+1;
    $data['x2'][$i] = pow($data['x'][$i],2);
    $data['xy'][$i] = $data['x'][$i]*$a['jumlah'];
    $data['jumlahx'] = $data['jumlahx']+$data['x'][$i];
    $data['jumlahx2'] = $data['jumlahx2']+$data['x2'][$i];
    $data['jumlahxy'] = $data['jumlahxy']+$data['xy'][$i];
    $tengah --;
}

}
else{
    $data['x'][$i] = $i-$tengah;
    $data['x2'][$i] = pow($data['x'][$i],2);
    $data['xy'][$i] = $data['x'][$i]*$a['jumlah'];
    $data['jumlahx'] = $data['jumlahx']+$data['x'][$i];
    $data['jumlahx2'] = $data['jumlahx2']+$data['x2'][$i];
    $data['jumlahxy'] = $data['jumlahxy']+$data['xy'][$i];
}

}

$jumlah += $a['jumlah'];
$data['tanggal'][$i] = $a['tanggal'];
$data['nilai'][$i] = $a['jumlah'];
$i++;
}

$data['totaly'] = $jumlah;
$data['total']=$i;
$genap?$data['xprediksi']=$data['x'][$i-
1]+2:$data['xprediksi']=$data['x'][$i-1]+1;
$data['nilaia'] = $jumlah/$i;
$data['nilaib'] = $data['jumlahxy']/$data['jumlahx2'];
$data['y']
=
$data['nilaia']+(($data['nilaib']*$data['xprediksi']));

```

```

    return $data;
  }else{
    return array();
  }
}
}
}

```

Pada Tabel 4.4 merupakan kode perhitungan metode *Least Square*. Pada kode program tersebut menjelaskan tahapan atau langkah – langkah proses perhitungan metode *Least Square* yang menghasilkan perhitungan peramalan. Hasil perhitungan tersebut adalah peramalan penjualan periode berikutnya.

4.4 Pengujian

4.4.1 *White Box*

Pengujian ini bertujuan untuk mengetahui apakah sistem mampu menghasilkan fungsi, inputan dan keluaran yang sesuai dengan spesifikasi dari kebutuhan sistem. Tahapannya meliputi :

1. Diagram alir

Diagram alir merupakan notasi yang digunakan untuk merepresentasikan aliran kontrol yang digambarkan dari hasil penomoran dari *listing* program. Diagram alir digambarkan dengan *node* (simpul) yang dihubungkan dengan *edge* (garis).

2. Kompleksitas siklomatik (*cyclomatic complexity*)

Kompleksitas siklomatik merupakan metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logis suatu program. Kompleksitas siklomatik mendefinisikan jumlah jalur independen dalam suatu program. Perhitungan kompleksitas siklomatik menggunakan persamaan 4

$$V(G) = E - N + 2 \quad \text{Persamaan (4.1)}$$

Keterangan:

$V(G)$ = Kompleksitas siklomatik

E = Jumlah *edge* (garis)

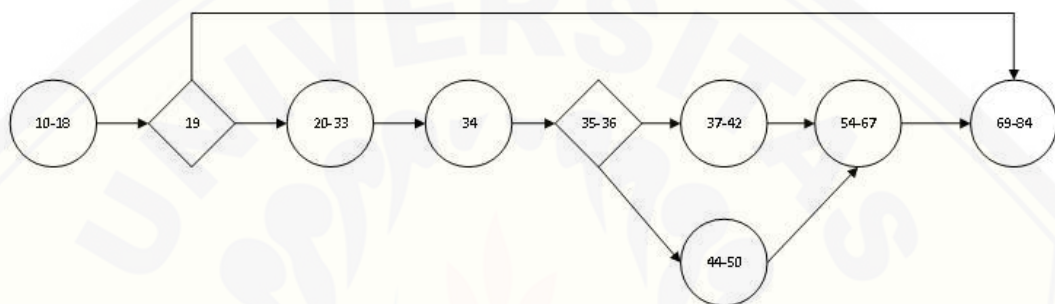
N = Jumlah *node* (simpul)

3. Pengujian Basis Set (*Test Case*)

Pada bagian ini diberikan contoh data yang menggambarkan pelaksanaan jalur di basis set. Data yang dieksekusi dimasukkan ke dalam grafik alir apakah sudah melewati basis set yang tersedia. Sistem telah memenuhi syarat kelayakan perangkat lunak jika salah satu jalur yang dieksekusi setidaknya satu kali.

A. Diagram Alir

Diagram Alir Function prediksi(\$idproduct)



Gambar 4. 18 Diagram Alir Function prediksi(\$idproduct)

B. Kompleksitas Siklomatik (*Cyclomatic Complexity*)

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

$$V(G) = 10 - 9 + 2$$

$$V(G) = 3$$

C. Jalur Independen (*Independent Path*)

Jalur independen merupakan setiap jalur yang melalui program yang memperkenalkan setidaknya satu kondisi baru. Bila dinyatakan dalam grafik alir, jalur independen harus bergerak setidaknya sepanjang satu edge yang belum dilintasi sebelum jalur tersebut didefinisi. Jalur independen fitur prediksi pada Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* (Studi Kasus : Larissa Aesthetic Center) adalah sebagai berikut.

Jalur 1 : (10-18) – 19 – (20-33) – 34 – (37-42) – (54-67) – (69-84)

Jalur 2 : (10-18) – 19 – (20-33) – 34 – (44-50) – (54-67) – (69-84)

Jalur 3 : (10-18) – 19 – (69-84)

4.4.2 Black Box

Black box testing merupakan pengujian perangkat lunak dari segi fungsionalitas *software* yang berkaitan dengan struktur internal atau kerja. Pengujian *black box* pada sistem manajemen persediaan produk dengan fitur prediksi menggunakan metode *least square* dilakukan untuk mengetahui masukan dan keluaran dari sistem telah sesuai dengan kebutuhan fungsional atau tidak. Pengujian dilakukan pada setiap menu. Hasil pengujian *black box* dapat dilihat pada tabel 4. berikut.

Tabel 4. 7 Hasil Pengujian Black Box

No.	Menu	Fungsi	Aksi	Hasil	Ket
1.	Login (Manag er dan Pegawai)	Menu login merupakan fitur yang digunakan Manager dan Pegawai untuk masuk ke sistem	Mengisi <i>username</i> dan <i>password</i> <i>form login</i> kemudian pilih tombol <i>login</i>	Login berhasil dan menampilkan halaman home manager dan Pegawai	√
			<i>Username</i> dan <i>password</i> tidak diisi	Menampilkan halaman login manager dan dengan notifikasi “ <i>Please fill out this field</i> ”	√
2.	Mengel ola Data Produk(Manage	Halaman yang menampilkan beranda, data		Menampilkan beranda, data produk, data penjualan, peramalan,	√

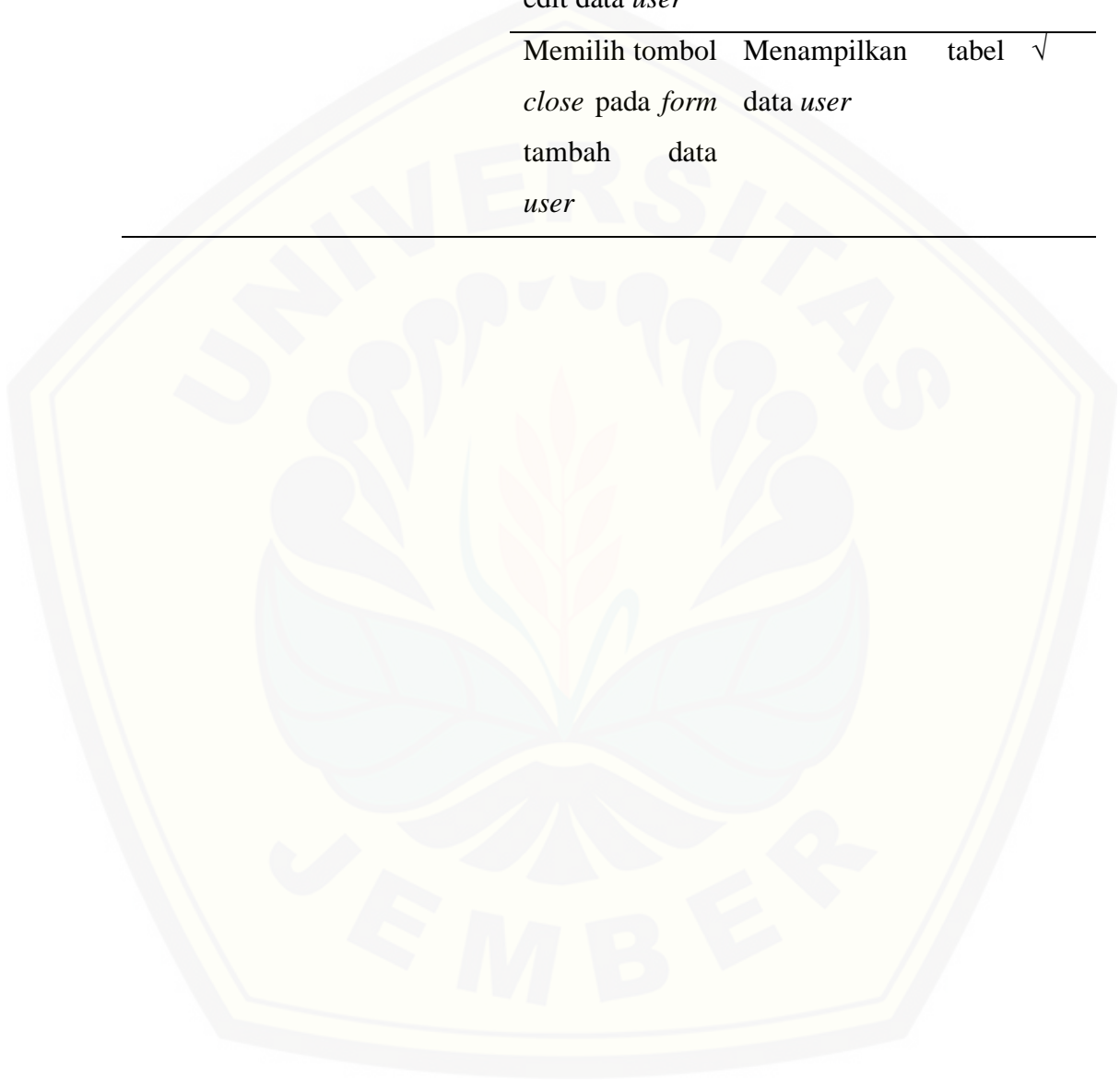
	r dan produk, data Pegawai penjualan,) peramalan, pengaturan akun dan profil				pengaturan akun dan profil
3.	Mengelola Data User (Manager)	Menu data yang manager digunakan untuk menambah data <i>user</i> , mengubah <i>user</i> dan menghapus data <i>user</i>	Memilih menu Pengaturan Akun	Memilih tombol tambah data	Menampilkan tabel user dengan atribut no, nama akun, kata sandi, level status, aksi, tombol tambah data, tombol hapus, tombol edit dan tombol <i>logout</i> pada halaman user
			Memilih tombol simpan pada <i>form</i> tambah data <i>user</i>	Memilih tombol tambah data	Menampilkan form tambah <i>user</i> dengan <i>textfield</i> <i>username</i> dan, tombol simpan dan tombol kembali
			Memilih tombol <i>close</i> pada <i>form</i> tambah <i>user</i>	Memilih tombol tambah data	Menampilkan tabel data <i>user</i>
			Memilih tombol <i>icon edit</i>	Memilih tombol tambah data	Menampilkan form edit data User dengan atribut <i>username</i> dan

password tombol *save*

dan tombol *close*

Memilih tombol *save* pada *form* edit data *user* Menampilkan data *user* tabel \checkmark

Memilih tombol *close* pada *form* tambah data *user* Menampilkan data *user* tabel \checkmark



BAB 6. PENUTUP

Bab ini berisi kesimpulan dan saran dari penelitian yang telah dilakukan. Kesimpulan dan saran ini diharapkan mampu menjadi acuan untuk melakukan penelitian selanjutnya.

6.1 Kesimpulan

Kesimpulan yang didapat dari penelitian yang telah dilakukan adalah sebagai berikut :

1. Metode *Least Square* dalam penerapannya pada Sistem Informasi Peramalan Penjualan Produk Kecantikan menghasilkan jumlah peramalan sebesar 87.05 unit untuk produk Milk Cleanser Apple pada bulan Januari 2018. Untuk produk Milk Cleanser Tea Tree menghasilkan jumlah peramalan sebesar 168.55 unit pada bulan Januari 2018. Produk Milk Cleanser Green Tea menghasilkan jumlah peramalan sebesar 207.2 unit pada bulan Januari 2018. Sedangkan untuk produk Milk Cleanser Chamomile menghasilkan jumlah peramalan sebesar 92.33 unit pada bulan Januari 2018. Metode *Least Square* ini menggunakan MAPE untuk pengujian presentase kesalahan. MAPE yang dihasilkan sebesar 0.215%. Angka tersebut menunjukkan bahwa persentase kesalahan yang dihasilkan rendah.
2. Sistem Informasi Peramalan Penjualan Produk Kecantikan dengan Menggunakan Metode *Least Square* dibangun menggunakan SDLC Waterfall. Tahapan – tahapan yang dilakukan meliputi analisis kebutuhan, desain sistem, implementasi, pengujian sistem dan *maintenance*. Sistem informasi ini memiliki 2 hak akses dengan aktor manager dan pegawai.

6.2 Saran

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

1. Sistem Informasi Peramalan Penjualan Produk Kecantikan Dengan Menggunakan Metode *Least Square* diharapkan pada pengembang selanjutnya memberikan fitur yang lebih lengkap yang ada di Larissa Aesthetic Center.

2. Proses prediksi jumlah persediaan stok dengan metode berbeda atau mengkombinasikan metode *Least Square* dengan metode lainnya.



DAFTAR PUSTAKA

- Roger S. Pressman, 2002. *Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku Satu)*, ANDI Yogyakarta.
- Makridakis, Spyros, Steven C. Wheelwright, Victo E. Mc Gee., (1999). *Metode dan Aplikasi Peramalan dalam Forecasting Methods and applications*. Binarupa Aksara, Jakarta.
- Rambe, Muhammad (2014). *Perancangan Aplikasi Peramalan Persediaan Obat-Obatan Menggunakan Metode Least Square (Studi Kasus : Apotik Mutiara Hati)*. *Jurnal Pelita Informatika Budi Darma*. (VI) : Hayesalaman 49-50.
- Biegel, John E. (1999), *Pengendalian Produksi Suatu Pendekatan Kuantitatif*, Akademika Presindo, Jakarta.
- Prasetya, H dan Lukiasuti, F. 2009. *Manajemen Oprasi*. Jakarta: MedPress.
- Sugiarto, D. S. (2000). *Metode statistika untuk bisnis dan ekonomi* Jakarta: Gramedia Pustaka Utama
- Heizer, Jay dan Barry Render. 2009. *Manajemen Operasi Buku 1 Edisi 9*. Jakarta: Salemba 4
- Departemen Pendidikan Indonesia. (2008). *Kamus Besar Bahasa Indonesia*. Jakarta: Balai Pustaka.
- Hasibuan, M. S. (2007). *Manajemen Sumber Daya Manusia Perusahaan*. Bandung: PT. Bumi Aksa.
- Larissa Aesthetic Center. (2019). *About*. Diperoleh 4 Juli 2019, dari <https://www.larissa.co.id/about>
- Kurniawan, Y.A., Nurhadiyono, B., 2015. *Komparasi Metode Least Square dan Double Exponential Smoothing Untuk Menganalisis Pendapatan Retribusi Uji Kendaraan Bermotor*.
- Zainun, N. Y., dan Majid, M. Z. A. 2003. *Low Cost House Demand Predictor*. Universitas Teknologi Malaysia.

LAMPIRAN

LAMPIRAN A. *Scenario*

A1. *Use Case Scenario Login*

Nama	Login
Aktor	Manager, Pegawai
Pre-Kondisi	Manager, Pegawai masuk ke sistem
Pra-Kondisi	Manager, Pegawai sudah melakukan login

SKENARIO NORMAL
“Login”

Aktor	Sistem
1. Membuka Sistem Informasi Penjualan Produk Kecantikan	
	2. Menampilkan halaman login Sistem Informasi Penjualan Produk Kecantikan
3. Mengisi Username dan Password sesuai dengan hak akses	
4. Klik tombol Login	
	5. Menampilkan halaman Dashboard sesuai dengan hak akses

SCENARIO ALTERNATIF
“Username dan password tidak diisi

Aktor	Sistem
4. Klik tombol Login	
	5. Menampilkan notifikasi “Please fill out this field”

SCENARIO ALTERNATIF
“Username dan password salah”

Aktor	Sistem
3. Mengisi Username dan Password sesuai dengan hak akses	
4. Klik tombol Login	
	5. Menampilkan notifikasi “Maaf, Username atau Password Salah ”

A2. Use Case Scenario Mengelola Data Penjualan

Nama	Mengelola Data Penjualan
Aktor	Manager
Pre-Kondisi	Manager berhasil melakukan login
Pra-Kondisi	Pegawai berhasil menambah, mengedit dan menghapus Data Penjualan

SKENARIO NORMAL
“Melihat data Penjualan”

Aktor	Sistem
1. Memilih Data Penjualan	2. Menampilkan tabel Penjualan yang berisi : a. No b. Nama Produk c. Bulan Penjualan d. Jumlah e. Action yang berisi tombol Hapus Dan Edit
	Serta terdapat tombol Tambah dan Search pada halaman Data Penjualan

SKENARIO NORMAL
“Menambah data Penjualan”

Aktor	Sistem
1. Klik tombol Tambah Data	2. Menampilkan pop up form tambah Data Produk yang berisi : a. Nama Produk b. Tanggal c. Jumlah
	Serta terdapat tombol Save dan Close
3. Mengisi nama produk, tanggal dan jumlah produk	
4. Klik tombol Save	5. Menyimpan data produksi kedalam tabel transaksi pada database
	6. Menampilkan tabel Penjualan yang berisi : a. No b. Nama Produk c. Bulan Penjualan

d. Jumlah
Action yang berisi tombol Hapus Dan Edit
Serta terdapat alert Succes ! Data berhasil di
tamahkan

SCENARIO ALTERNATIF
“Data belum diisi”

Aktor	Sistem
3a. Tidak mengisi nama produk, Tanggal dan Jumlah Produk	
4a. Klik tombol Simpan	5a. Menampilkan notifikasi “Please fill out this field”

SCENARIO ALTERNATIF
“Batal menambahkan data”

Aktor	Sistem
3b. Mengisi nama produk, Tanggal dan Jumlah Produk	
4b. Klik tombol Kembali	5b. Menampilkan tabel Penjualan yang berisi : a. No b. Nama Produk c. Bulan Penjualan d. Jumlah e. Action yang berisi tombol Hapus Dan Edit
	Serta terdapat tombol Tambah dan Search pada halaman Data Penjualan

SKENARIO NORMAL
“Mengedit Data Produksi”

Aktor	Sistem
1. Klik tombol Edit	2. Menampilkan pop up form edit Data Produk yang berisi : d. Nama Produk e. Tanggal f. Jumlah
	Serta terdapat tombol Save dan Close
3. Klik tombol Save	

4. Menyimpan data produksi kedalam tabel transaksi pada database
5. Menampilkan tabel Penjualan yang berisi :
 - a. No
 - b. Nama Produk
 - c. Bulan Penjualan
 - d. Jumlah
 - e. Action yang berisi tombol Hapus Dan Edit

Serta terdapat alert Succes ! Edit Data Penjualan Berhasil

SKENARIO NORMAL
“Menghapus Data Penjualan”

Aktor	Sistem
1. Klik tombol Hapus	
	2. Menampilan pop up Alert Delete data Penjualan Apakah andi yakin akan menghapus Transaksi ?
3. Klik tombol Ok	
	4. Menampilkan tabel Penjualan yang berisi : <ol style="list-style-type: none"> a. No b. Nama Produk c. Bulan Penjualan d. Jumlah e. Action yang berisi tombol Hapus Dan Edit

Serta terdapat alert Succes ! Data Sudah di hapus

A3. *Use Case Scenario* Mengelola Data Prediksi

Nama	Mengelola data Prediksi
Aktor	Manager
Pre-Kondisi	Manager melakukan prediksi
Pra-Kondisi	Manager berhasil melakukan prediksi

SKENARIO NORMAL
“Login”

Aktor	Sistem
1. Memilih Peramalan	

			2. Menampilkan halaman data prediksi, dengan dropbox pilih produk dan tabel prediksi dengan atribut a. No b. Bulan Penjualan c. Jumlah(Y) d. Prediksi (X) e. X^2 f. $X*Y$
3. Memilih Produk	Dropbox	Pilih	
			4. Menampilkan hasil peramalan produk pada tabel prediksi dengan atribut a. No b. Bulan Penjualan c. Jumlah(Y) d. Prediksi (X) e. X^2 f. $X*Y$ Serta hasil perhitungan dengan metode <i>Least Square</i>

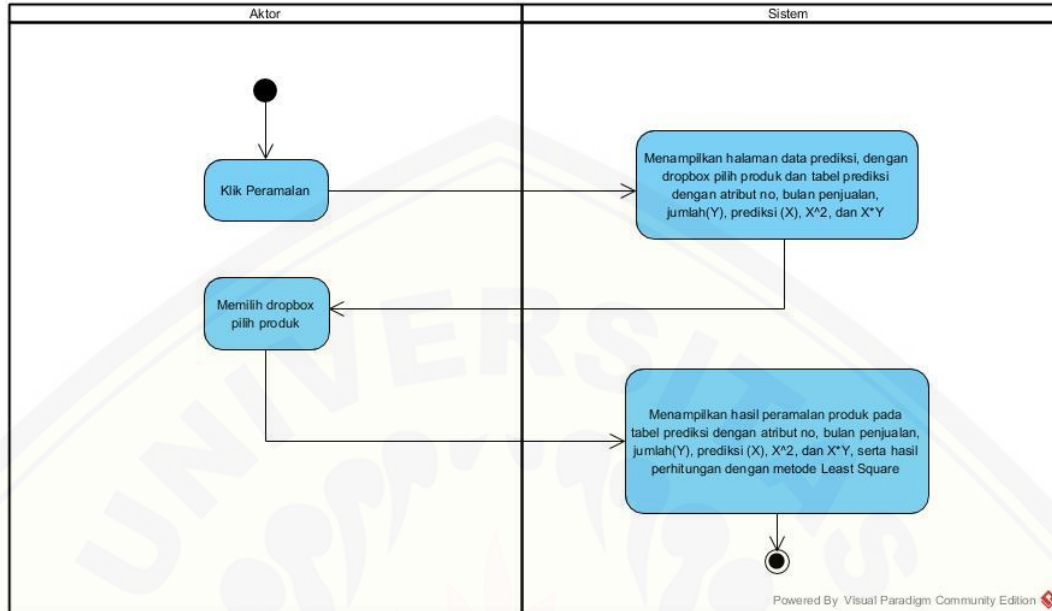
A4. Use Case Scenario Logout

Nama	Logout
Aktor	Manager, Pegawai
Pre-Kondisi	Manager, Pegawai keluar dari sistem
Pra-Kondisi	Manager, Pegawai sudah melakukan logout
SKENARIO NORMAL	
“Login”	

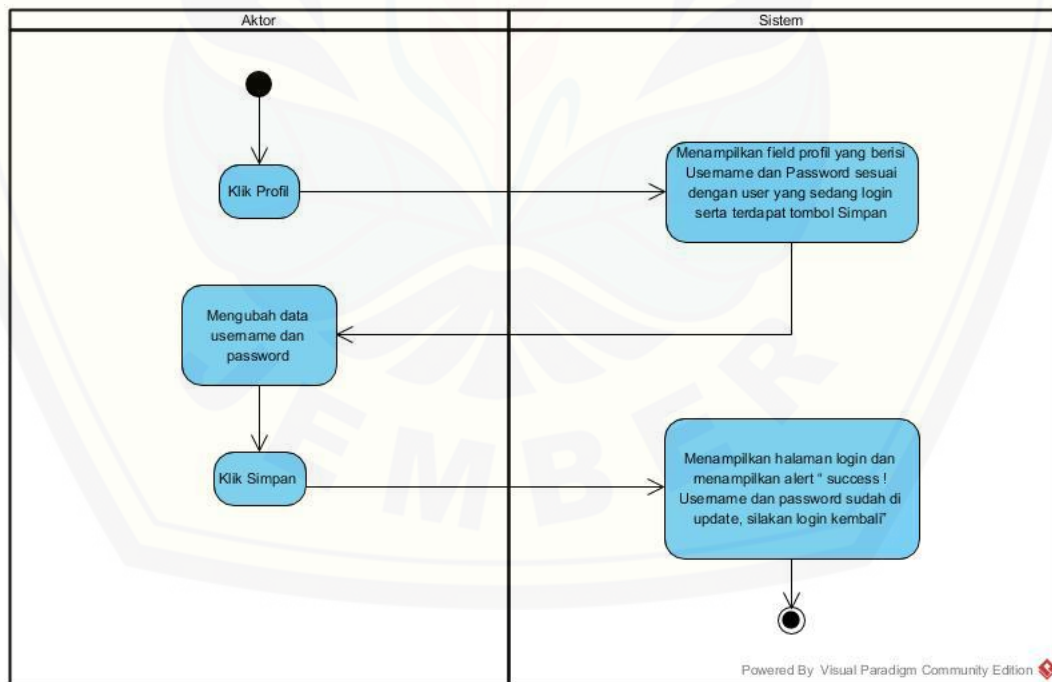
Aktor	Sistem
1. Klik Icon Logout	2. Menampilkan halaman login Sistem Informasi Penjualan Produk Kecantikan

LAMPIRAN B. Activity Diagram

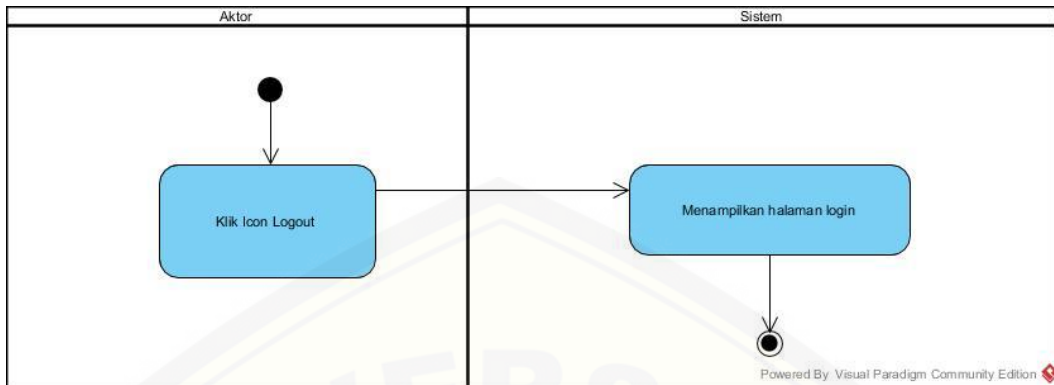
B1. Activity Diagram Mengelola Data Prediksi



B2. Activity Diagram Mengubah Data Profil

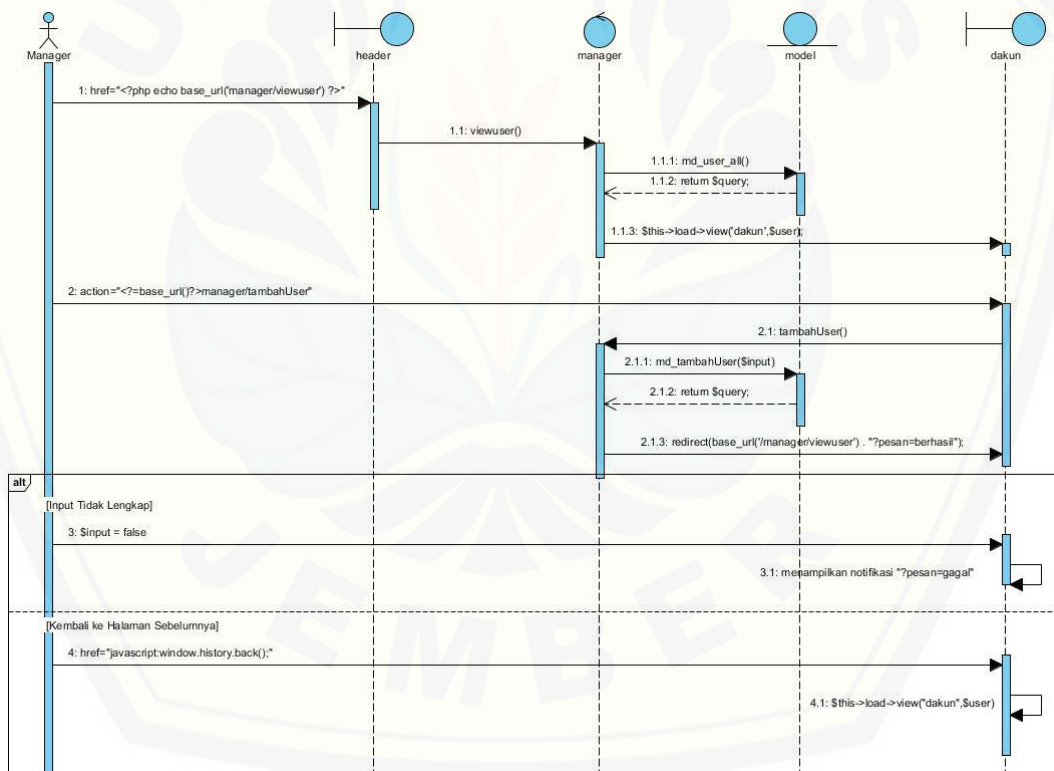


B3. Activity Diagram Logout

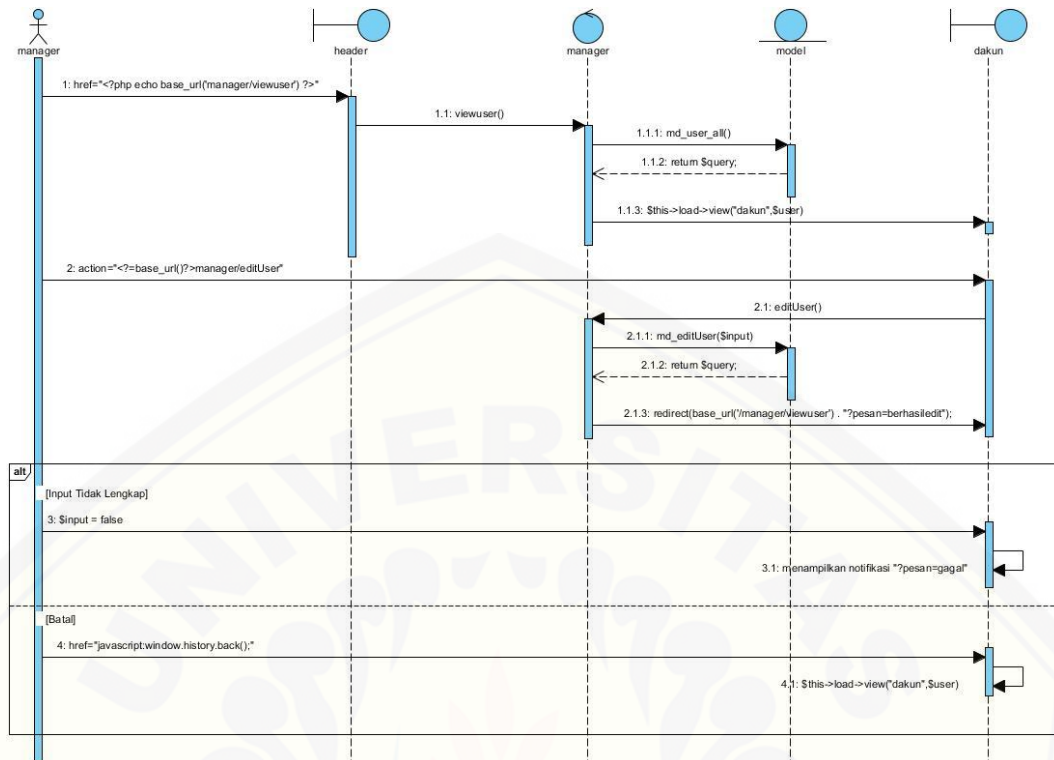


LAMPIRAN C. Sequence Diagram

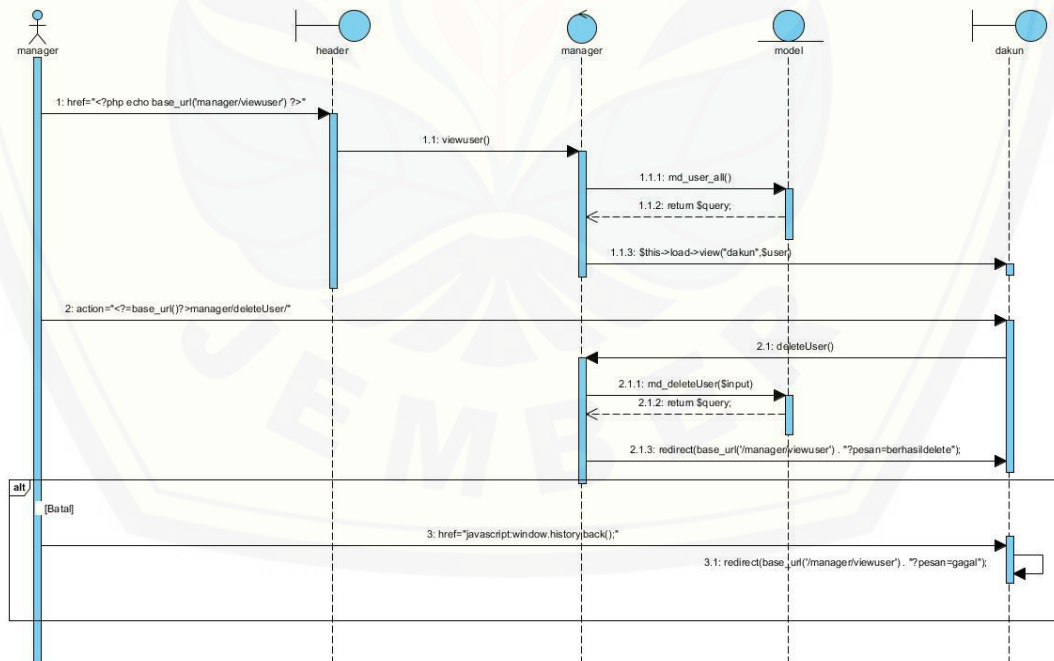
C1. Sequence Diagram Mengelola Data User (Manager)



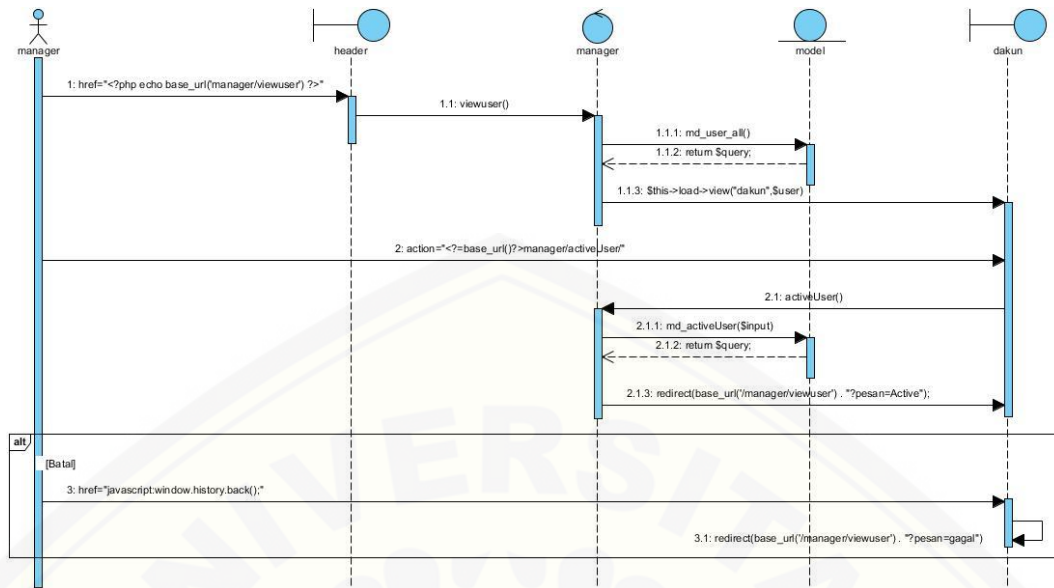
Gambar C1.1 Sequence Diagram Input User



Gambar C1.2 Sequence Diagram Edit User

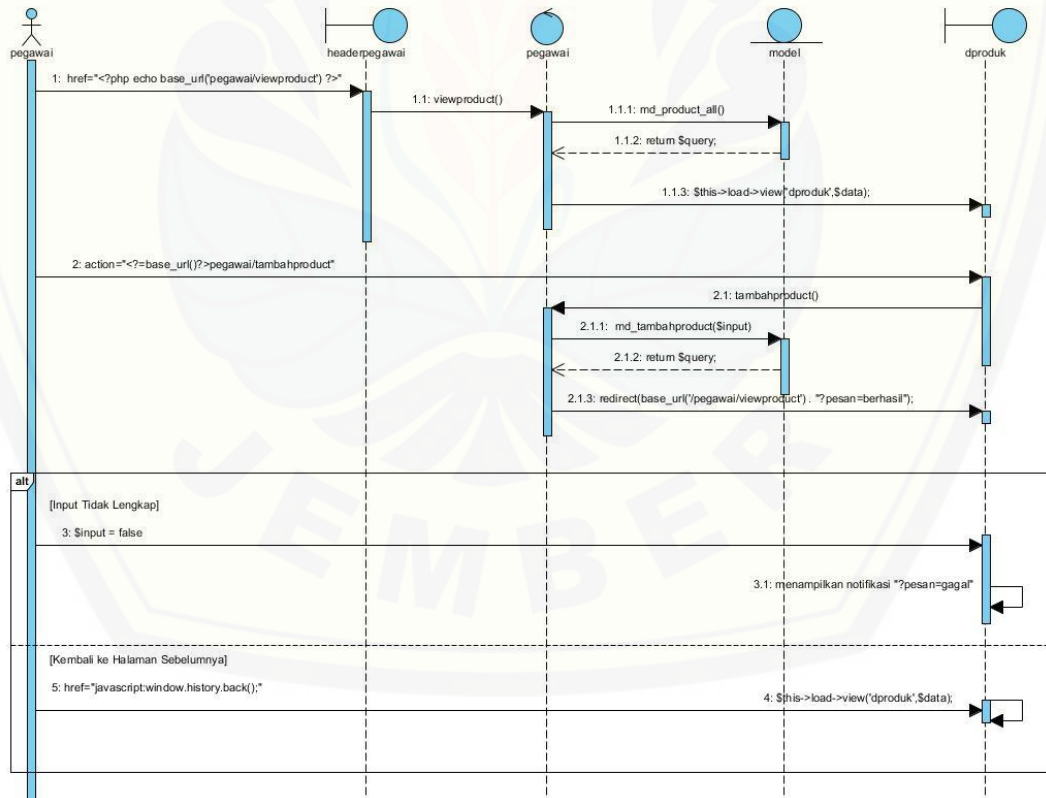


Gambar C1.3 Sequence Diagram Delete User

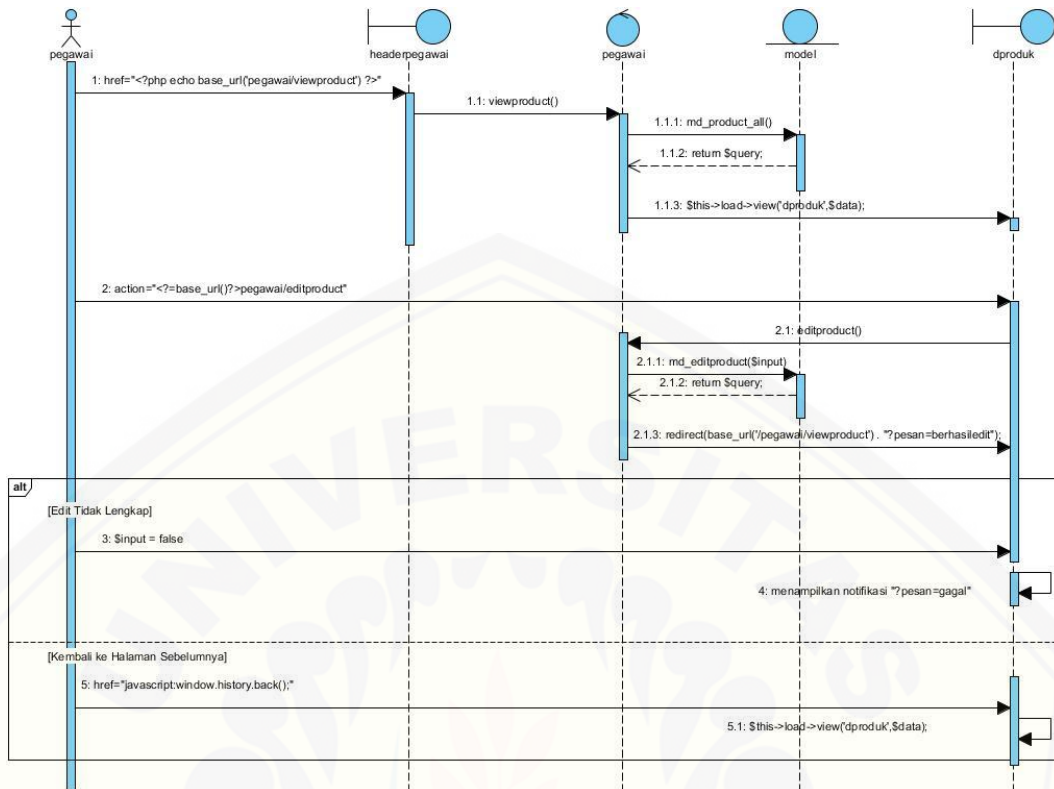


Gambar C1.4 Sequence Diagram Aktivasi User

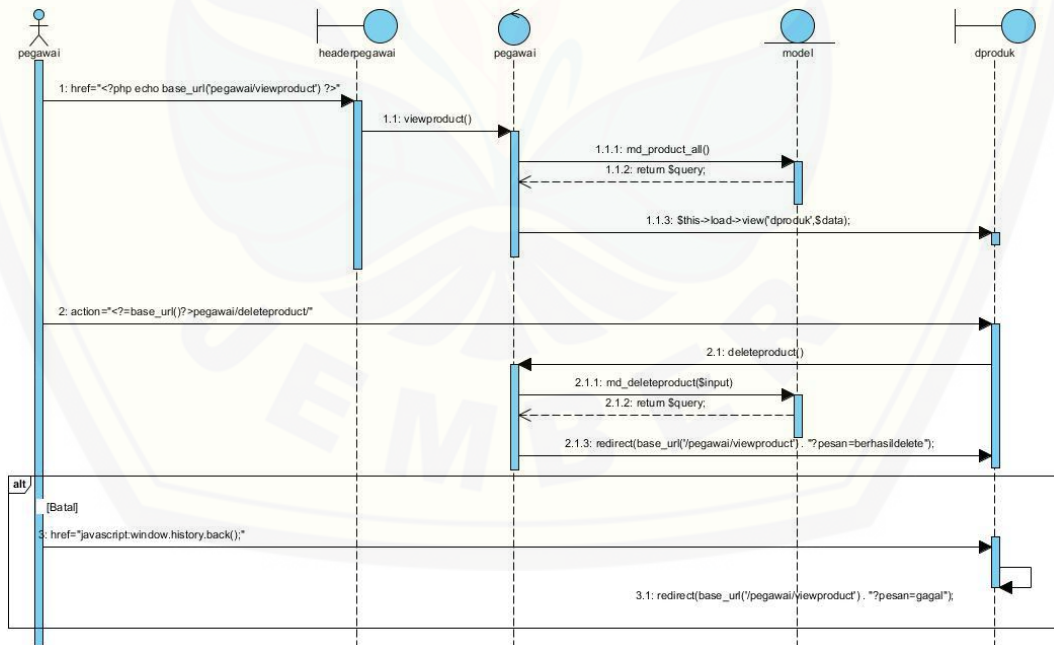
C2. Sequence Diagram Mengelola Data Produk (Pegawai)



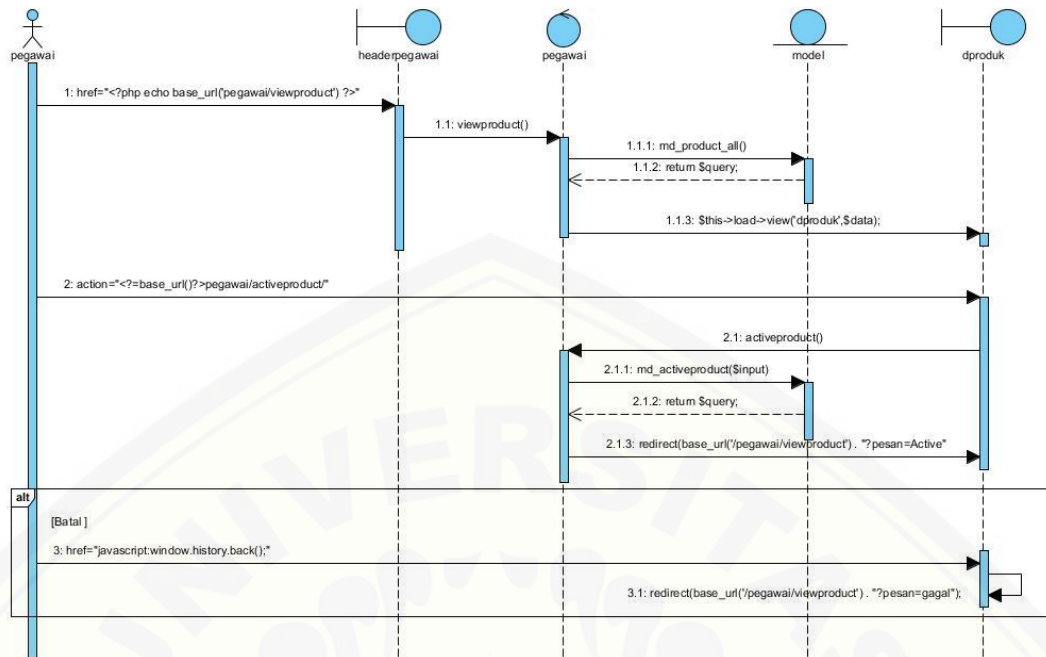
Gambar C2.1 Sequence Diagram Input Data Produk



Gambar C2.2 Sequence Diagram Edit Data Produk

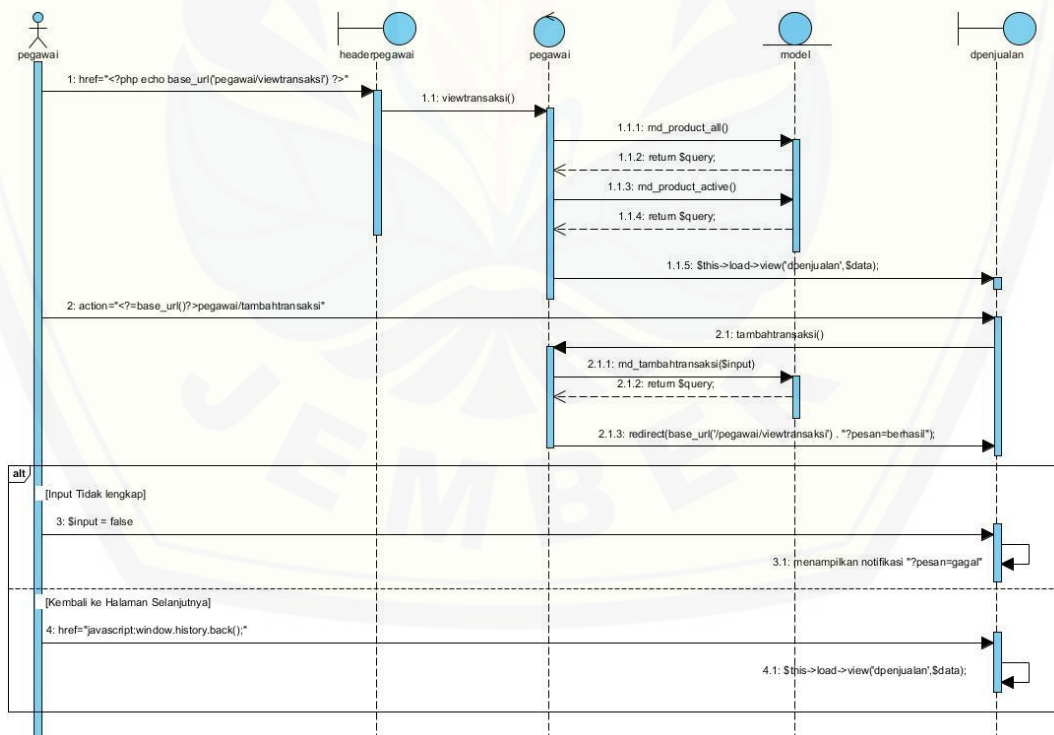


Gambar C2.3 Sequence Diagram Delete Data Produk

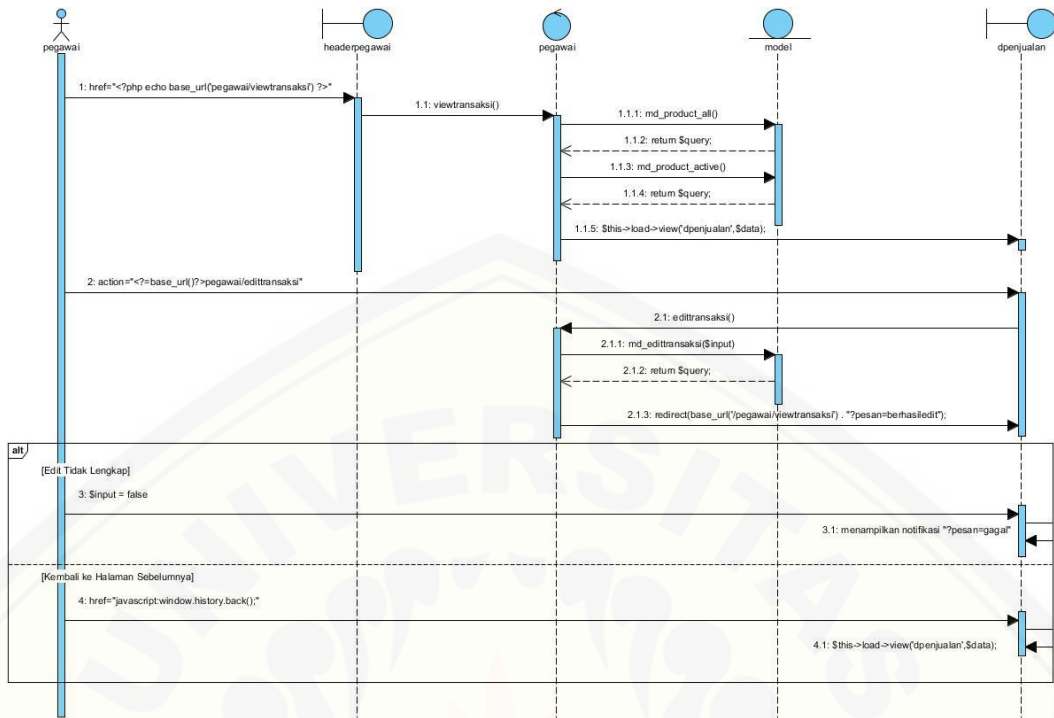


Gambar C2.4 Sequence Diagram Aktivasi Data Produk

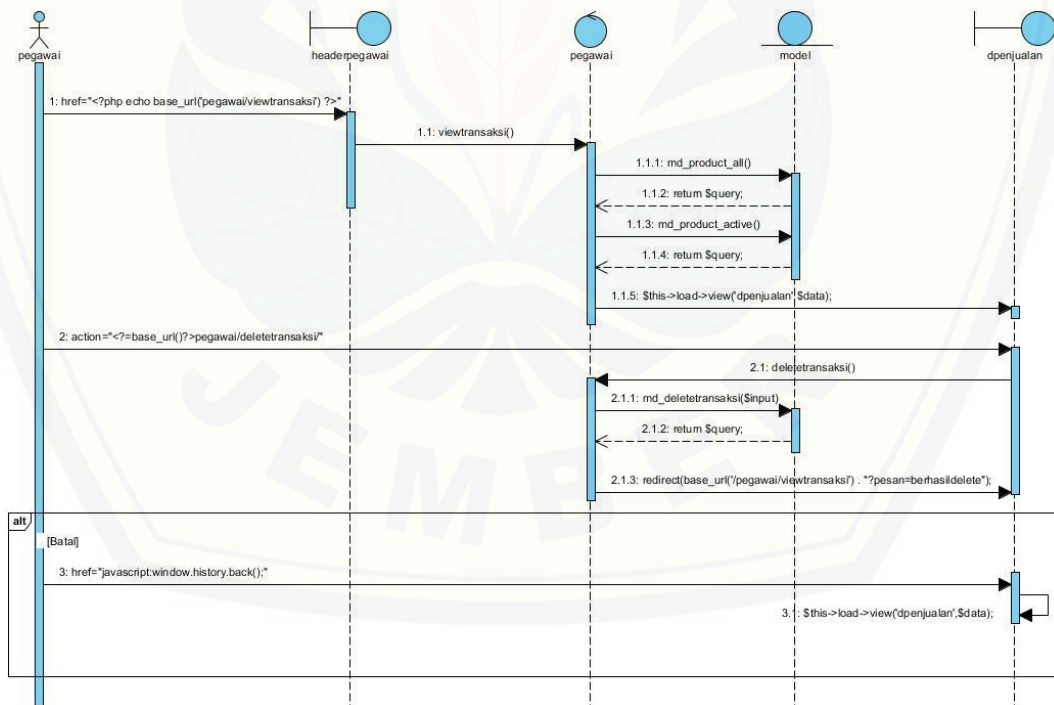
C3. Sequence Diagram Mengelola Data Penjualan (Pegawai)



Gambar C3.1 Sequence Diagram Tambah Data Penjualan

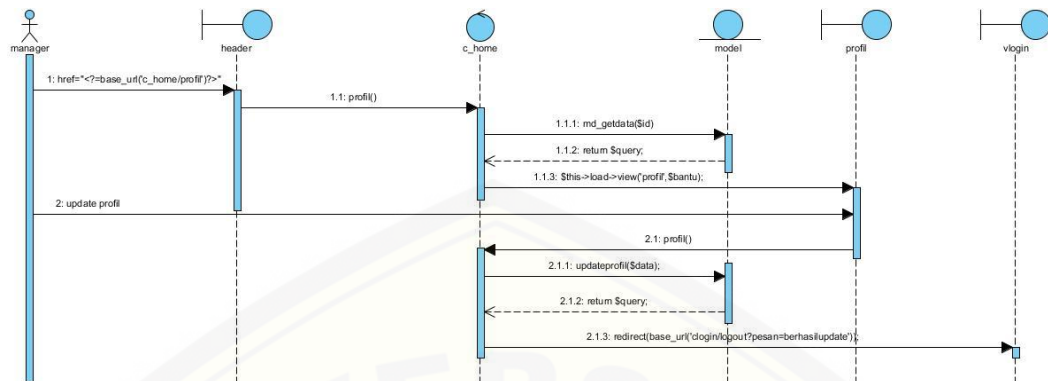


Gambar C3.2 Sequence Diagram Ubah Data Penjualan

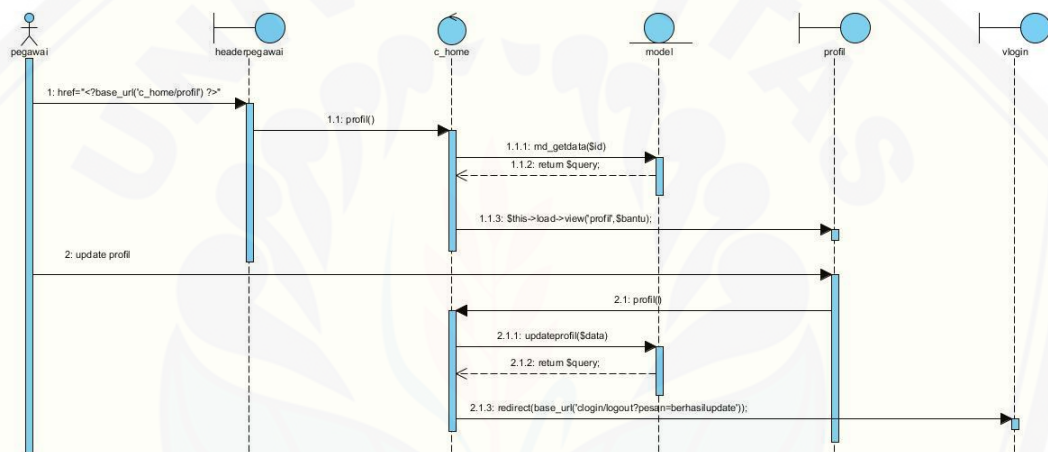


Gambar C3.3 Sequence Diagram Menghapus Data Penjualan

C4. *Sequence Diagram* Mengubah Data Profil

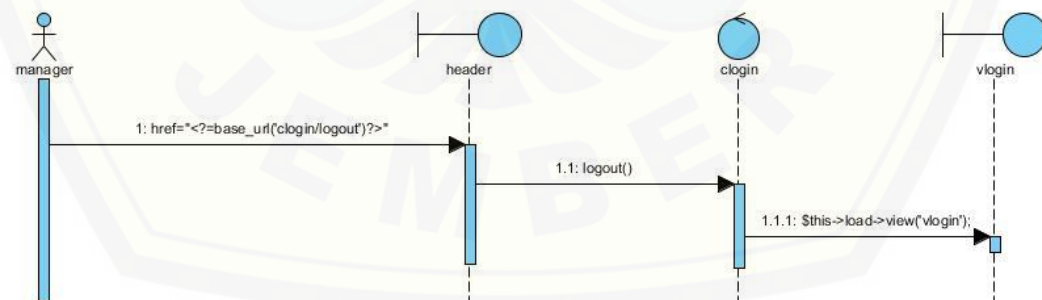


Gambar C4.1 *Sequence Diagram* Mengubah Data Profil (Manager)

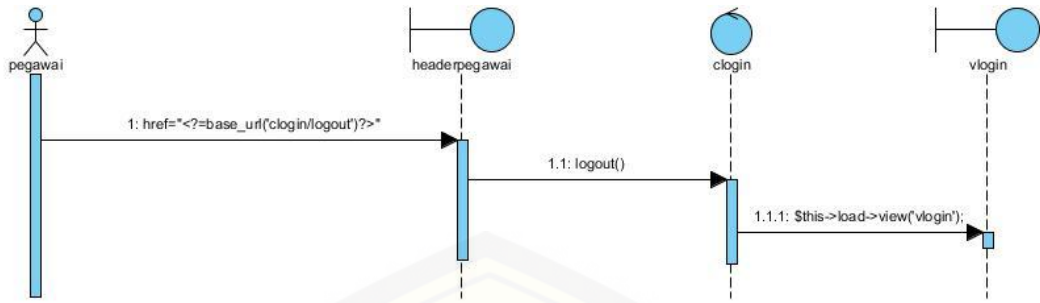


Gambar C4.2 *Sequence Diagram* Mengubah Data Profil (Pegawai)

C5. *Sequence Diagram* Logout



Gambar C5.1 *Sequence Diagram* Logout (Manager)



Gambar C5.2 *Sequence Diagram Logout (Pegawai)*

LAMPIRAN E. Kode Program

E1. Kode Program *Login*

Kode Program <i>View Login</i>
<pre> <!DOCTYPE html> <html lang="en"> <head> <meta charset="utf-8"> <meta http-equiv="X-UA-Compatible" content="IE=edge"> <meta name="viewport" content="width=device-width, initial- scale=1"> <link href=?<?php echo base_url('/assets/plugins/bootstrap/css/bootstrap.min.css') ?>" rel="stylesheet"> <link href=?<?php echo base_url('lite/css/style.css') ?>" rel="stylesheet"> <!-- <link href=?<?php echo base_url('lite/css/colors/blue.css') ?>" id="theme" rel="stylesheet"> --> </head> <body class="login"> <?php if (isset(\$_GET['pesan'])) { if (\$_GET['pesan'] == 'gagal') {?> <div class="alert alert-info" role="alert"> Maaf! Username atau Password Salah </pre>


```
</div>
<?php }else if($_GET['pesan'] == 'gagalstatus'){?>
  <div class="alert alert-danger" role="alert">
    <strong>Maaf!</strong> <a href="#" class="alert-link">Username atau Password Salah/Status Anda non Aktif,Silahkan hubungi Owner</a>
  </div>

  <?php }else if($_GET['pesan'] == 'berhasilupdate'){?>
    <div class="alert alert-success" role="alert">
      <strong>success!</strong> <a href="#" class="alert-link">Username dan Password sudah di update, silahkan login kembali</a>
    </div>

  <?php }?>
  <div class="container-fluid">
    <div class="row align-items-center justify-content-center">
  >
    <div class="col-12 logo-login">
      <div class="text-center">
        <!--GANTI LOGO DISINI -->
        
      </div>
    </div>
    <div class="col-4">
      <div class="logo card">
        <div class="card-block">
          <form class="form-horizontal form-material"
action="<?php echo base_url('clogin/proses_login')?>"
method="post">
            <div class="form-group">
              <label class="col-md-12">username</label>
              <div class="col-md-12">
```

```

        <input type="text" name="username"
id="username" class="form-control form-control-line" required>
    </div>
</div>
<div class="form-group">
    <label class="col-md-12">Password</label>
    <div class="col-md-12">
        <input type="password" value="password"
name="password" class="form-control form-control-line" required>
    </div>
</div>
<div class="form-group">
    <div class="col-sm-12">
        <button type="submit" class="btn btn-login
btn-block btn-flat">Masuk</button>
    </div>
</div>
</form>
</div>
</div>
</div>
</div>
</div>
</body>
</html>

```

Kode Program *Controller Login*

```

<?php
class clogin extends CI_Controller {
    function __construct() {
        parent::__construct();
        $this->load->helper('url');
        $this->load->library('session');
        $this->load->model('M_login');
    }
}

```

```
}
public function index() {
    $this->load->view('vlogin');
}

public function proses_login() {
    $user = $this->input->post('username');
    $pass = $this->input->post('password');

    $login = $this->M_login->cek_user($user, $pass);

    if ($login == true) {
        // login berhasil
        $level = $this->session->userdata('hakAkses');
        if($level == '1'){
            redirect(base_url() . 'c_home');
        }
        else if ($level == '2') {
            redirect(base_url() . 'c_home/pegawai');
        }
    } else {
        // login gagal

        redirect(base_url('?pesan=gagal'));
    }
}

public function logout(){
    if($_GET['pesan']=="gagal"){
        $this->session->sess_destroy();
        redirect(base_url()."?pesan=error");
    }else if($_GET['pesan']=="gagallogin"){
        redirect(base_url()."?pesan=belum");
    }else if($_GET['pesan']=="berhasilupdate"){
```

```
        $this->session->sess_destroy();
        redirect(base_url()."?pesan=berhasilupdate");
    }
    $this->session->sess_destroy();
    redirect(base_url());
}
}
```

Kode Program Model Login

```
<?php

class M_login extends CI_Model {

public function cek_user($username, $password) {
    $this->db->where("username = '$username'");
    $this->db->where('password', $password);
    $query = $this->db->get('user');

    if ($query->num_rows() > 0) {
        foreach ($query->result_array() as $row) {
            $this->session->set_userdata('user_on', TRUE);
            $this->session->set_userdata('username', $row['username']);
            $this->session->set_userdata('password', $row['password']);
            $this->session->set_userdata('hakAkses', $row['level']);
            $this->session->set_userdata('id', $row['id']);

            if ($row['level'] == '1') {
                $this->session->set_userdata('manager', TRUE);
            } else if ($row['hakAkses'] == '2') {
                $this->session->set_userdata('pegawai', TRUE);
            }
            return TRUE;
        }
    } else {
        return FALSE;
    }
}
```

```

    }
  }
}

```

E2. Kode Program Mengelola Data User

Kode Program *View* Mengeola Data *User*

```

<!-- PAGE WRAPPER -->
<div class="page-wrapper">
  <!-- CONTAINER -->
  <div class="container-fluid">
    <div class="row page-titles">
      <div class="col-md-5 col-8 align-self-center">
        <h3 class="text-themecolor m-b-0 m-t-0"><i class="mdi mdi-clipboard-account"></i>Data Akun</h3>
        <ol class="breadcrumb">
          <li class="breadcrumb-item active"><a href="javascript:void(0)">Data Akun</a></li>
        </ol>
      </div>
    </div>

    <!-- PAGE CONTENT -->
    <div class="row">
      <div class="col-12">
        <?php
          if (isset($_GET['pesan'])) {
            if ($_GET['pesan'] == 'berhasil') {?>
              <div class="alert alert-success" role="alert">
                <strong>success!</strong> <a href="#" class="alert-link">Data berhasil di tambahkan</a>
              </div>
            <?php }else if($_GET['pesan'] == 'berhasildelete'){?>
              <div class="alert alert-danger" role="alert">
                <strong>success!</strong> <a href="#" class="alert-link">Data sudah di hapus</a>

```

```
</div>

<?php }else if($_GET['pesan'] == 'berhasiledit'){?>
    <div class="alert alert-danger" role="alert">
        <strong>success!</strong> <a href="#" class="alert-
link">Data berhasil dirubah</a>
    </div>

<?php }?>
<div class="card">
    <div class="card-block">
        <h4 class="card-title">Data Akun</h4>
        <hr>

        <button type="button" class="btn btn-info" data-
toggle="modal" data-target="#myModal" style="margin-Bottom:10px"><i
class="mdi mdi-plus"></i>Tambah data</button>

        <!-- Modal -->
        <div id="myModal" class="modal fade" role="dialog">
            <div class="modal-dialog">

                <!-- Modal content-->
                <div class="modal-content">
                    <div class="modal-header">
                        <button type="button" class="close" data-
dismiss="modal">&times;</button>
                        <h4 class="modal-title">Tambah Data User</h4>
                    </div>
                    <form action="<?==base_url()?>manager/tambahUser"
method="post">

                        <div class="modal-body">
                            <div class=form-group>
                                <input type="text" name="username"
id="username" required="" placeholder="Username" class="form-control">
                            </div>
```

```
        <div class="form-group">
            <input type="password" name="password"
id="password" required="" placeholder="password" class="form-control">
        </div>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-default"
data-dismiss="modal">Close</button>
        <button type="submit" class="btn btn-success"
>Save</button>
    </div>
</form>
</div>

</div>
</div>

<div class="table-responsive">
    <table id="akun" class="table table-striped">
        <thead>
            <tr>
                <th>NO</th>
                <th>Nama Akun</th>
                <th>Kata Sandi</th>
                <th>Level</th>
                <th>Status</th>
                <th>Aksi</th>
            </tr>
        </thead>
        <tbody>
            <?php $i=0;
            foreach ($user as $n) {
                $i++;
                ?>
                <tr>
                    <td><?=$i?></td>
```

```
<td><?=$n['username']?></td>
<td><?=$n['password']?></td>
<td><?=$n['level']==1?'manager':'Pegawai';?>
</td>
<td><?=$n['status']==1?'Active':'Non-
Active';?>
</td>
<td>
<?php if($n['status']!=1){?>
<button type="button" class="btn btn-
success" data-toggle="modal" data-target="#kembalikan-
<?=$n['id']?>"><i class="mdi mdi-check"></i></button>
<?php }else{ ?>
<button type="button" class="btn btn-
danger" data-toggle="modal" data-target="#del-<?=( $n['id'] )?>"><i
class="mdi mdi-delete"></i></button>
<?php } ?>
<button type="button" class="btn btn-
warning" data-toggle="modal" data-target="#edit-<?=$n['id']?>"><i
class="mdi mdi-pencil"></i></button>

</td>
</tr>

<?php }
?>

</tbody>
</table>
</div>
</div>
</div>
</div>
```



```

</div>
<!-- END PAGE CONTENT -->
</div>
<!-- END CONTAINER -->
<?php foreach ($user as $n) {
    if($n['status']==1){?>

    <!-- Modal delete-->
    <div id="del-= $n['id'] ?" class="modal fade" role="dialog">
        <div class="modal-dialog">
            <!-- Modal content-->
            <div class="modal-content">
                <div class="modal-header">
                    <button type="button" class="close" data-
dismiss="modal">&times;</button>
                    <h4 class="modal-title">Delete Data User</h4>
                </div>
                <form action="= base_url() ?manager/deleteUser/"
method="post">
                    <div class="modal-body">
                        <input type="hidden" name="id"
value="= $n['id'] ?">
                        <p>Apakah Anda Yakin akan Menghapus User
<= $n['username'] ?>??</p>
                    </div>

                    <div class="modal-footer">
                        <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
                        <button type="submit" class="btn btn-success"
>Ok</button>
                    </div>
                </form>
            </div>
        </div>
    }
}

```

```
        </div>
    </div>
    <?php }else{ ?>
        <!-- Modal kembalikan-->
        <div id="kembalikan-<?=$n['id']?>" class="modal fade"
role="dialog">
            <div class="modal-dialog">
                <!-- Modal content-->
                <div class="modal-content">
                    <div class="modal-header">
                        <button type="button" class="close" data-
dismiss="modal">&times;</button>
                        <h4 class="modal-title">Aktifkan Data User</h4>
                    </div>
                    <form action="<?=$base_url()>manager/activeUser/"
method="post">
                        <div class="modal-body">
                            <input type="hidden" name="id"
value="<?=$n['id']?>">
                            <p>Apakah Anda Yakin akan mengaktifkan User
<?=$n['username']?>?? </p>
                        </div>
                        <div class="modal-footer">
                            <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
                            <button type="submit" class="btn btn-success"
>Ok</button>
                        </div>
                    </form>
                </div>
            </div>

        </div>
    </div>

    <?php } ?>
    <!-- Modal -->
```

```
<div id="edit-= $n['id'] ?" class="modal fade" role="dialog">
  <div class="modal-dialog">
    <!-- Modal content-->
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <h4 class="modal-title">Edit Data User</h4>
      </div>
      <form action="= base_url() ?manager/editUser"
method="post">
        <div class="modal-body">
          <input type="hidden" name="id" value="= $n['id'] ?" >
          <div class="form-group">
            <input type="text" name="username"
value="= $n['username'] ?" required="" placeholder="Username"
class="form-control">
          </div>
          <div class="form-group">
            <input type="password" name="password"
value="= $n['password'] ?" required="" placeholder="password"
class="form-control">
          </div>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
          <button type="submit" class="btn btn-success"
>Save</button>
        </div>
      </form>
    </div>
  </div>
</div>
<?php } ?>
```

Kode Program *Controller* Mengeola Data *User*

```
public function viewuser()
{
    $id =$this->session->userdata('id');
    $bantu['data'] = $this->model->md_getdata($id);
    $user['user'] = $this->model->md_user_all();
    $this->load->view('header',$bantu);
    $this->load->view('dakun',$user);
    $this->load->view('footer');
}
public function tambahUser() {
    $data['username'] = $this->input->post('username');
    $data['password'] = $this->input->post('password');

    if ($data['username']== null){
        $input = false;
    }else if ($data['password']== null){
        $input = false;
    }else{

        $this->model->md_tambahUser($data);
        $input = true;
    }
    if ($input == true){
        redirect(base_url('/manager/viewuser')
        . "?pesan=berhasil");
    }else{
        redirect(base_url('/manager/viewuser')
        . "?pesan=gagal");
    }
}

public function editUser() {
    $data['id'] = $this->input->post('id');
    $data['username'] = $this->input->post('username');
    $data['password'] = $this->input->post('password');
```

```
        if ($data['username']== null){
            $input = false;
        }else if ($data['password']== null){
            $input = false;
        }else{

            $this->model->md_editUser($data);
            $input = true;
        }
        if ($input == true){
            redirect(base_url('/manager/viewuser')
"?pesan=berhasiledit");
        }else{
            redirect(base_url('/manager/viewuser')
"?pesan=gagal");
        }
    }

    public function deleteUser() {
        $data['id'] = $this->input->post('id');

        if ($data['id']== null){
            $input = false;
        }else{

            $this->model->md_deleteUser($data);
            $input = true;
        }
        if ($input == true){
            redirect(base_url('/manager/viewuser')
"?pesan=berhasildelete");
        }else{
            redirect(base_url('/manager/viewuser')
"?pesan=gagal");
        }
    }
}
```

```

public function activeUser() {
    $data['id'] = $this->input->post('id');
    if ($data['id'] == null){
        $input = false;
    }else{

        $this->model->md_activeUser($data);
        $input = true;
    }
    if ($input == true){
        redirect(base_url('/manager/viewuser')
        . "?pesan=Active");
    }else{
        redirect(base_url('/manager/viewuser')
        . "?pesan=gagal");
    }
}

```

Kode Program *Model* Mengelola Data *User*

```

<?php
class Model extends CI_Model {
    function __construct(){
        parent::__construct();
    }

    public function md_user_all() {
        $query = $this->db->get('user')->result_array();
        return $query;
    }

    public function md_tambahUser($input) {
        $query = $this->db->query("INSERT into `user`
        (`id`,`username`,`password`,`level`,`status`)
        Values('','$input[username]','$input[password]','2','1')");
        return $query;
    }
}

```

```

public function md_editUser($input) {
    $this->db->where('id',$input['id']);
    $query = $this->db->update('user',array('username'=>$input['username'],
        'password'=>$input['password']));

    return $query;
}

public function md_deleteUser($input) {
    $this->db->where('id',$input['id']);
    $query = $this->db->update('user',array('status'=>'0'));
    return $query;
}

public function md_activeUser($input) {
    $this->db->where('id',$input['id']);
    $query = $this->db->update('user',array('status'=>'1'));
    return $query;
}

```

E3. Kode Program Mengelola Data Produk

Kode Program *View* Mengelola Data Produk

```

<!-- PAGE WRAPPER -->
<div class="page-wrapper">
    <!-- CONTAINER -->
    <div class="container-fluid">
        <div class="row page-titles">
            <div class="col-md-5 col-8 align-self-center">
                <h3 class="text-themecolor m-b-0 m-t-0"><i class="mdi mdi-
package"></i>Data Produk</h3>
                <ol class="breadcrumb">
                    <li class="breadcrumb-item active"><a
href="javascript:void(0)">Data Produk</a></li>
                </ol>
            </div>
        </div>
    </div>

```

```
</div>

<!-- PAGE CONTENT -->
<div class="row">
  <div class="col-12">
    <?php
    if (isset($_GET['pesan'])) {
      if ($_GET['pesan'] == 'berhasil') {?>
        <div class="alert alert-success" role="alert">
          <strong>success!</strong> <a href="#" class="alert-
link">Data berhasil di tambahkan</a>
        </div>
        <?php }else if($_GET['pesan'] == 'berhasildelete'){?>
          <div class="alert alert-danger" role="alert">
            <strong>success!</strong> <a href="#" class="alert-
link">Data sudah di hapus</a>
          </div>

          <?php }else if($_GET['pesan'] == 'berhasiledit'){?>
            <div class="alert alert-danger" role="alert">
              <strong>success!</strong> <a href="#" class="alert-
link">Data berhasil dirubah</a>
            </div>

          <?php }?>
          <div class="card">
            <div class="card-block">
              <h4 class="card-title">Data Product</h4>

              <hr>
              <button type="button" class="btn btn-info" data-
toggle="modal" data-target="#myModal" style="margin-Bottom:10px"><i
class="mdi mdi-plus"></i>Tambah Data</button>

            <!-- Modal -->
            <div id="myModal" class="modal fade" role="dialog">
```



```
<div class="modal-dialog">

    <!-- Modal content-->
    <div class="modal-content">
        <div class="modal-header">
            <button type="button" class="close" data-
dismiss="modal">&times;</button>
            <h4 class="modal-title">Tambah Data Product</h4>
        </div>
        <form
action="<?=base_url()?>manager/tambahproduct" method="post">
            <div class="modal-body">
                <div class="form-group">
                    <input type="text" name="nama" required=""
placeholder="Nama" class="form-control">
                </div>
                <div class="form-group">
                    <input type="number" name="hargajual"
required="" placeholder="Harga Jual" class="form-control">
                </div>
                <div class="form-group">
                    <input type="number" name="hargabeli"
required="" placeholder="Harga Beli" class="form-control">
                </div>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-default"
data-dismiss="modal">Close</button>
                <button type="submit" class="btn btn-success"
>Save</button>
            </div>
        </form>
    </div>
</div>
</div>
```

```

<div class="table-responsive">
  <table id="produk" class="table table-striped">
    <thead>
      <tr>
        <th>NO</th>
        <th>Nama Produk</th>
        <th>Harga</th>
        <th>Status</th>
        <th>Aksi</th>
      </tr>
    </thead>
    <tbody>
      <?php $i=0;
      foreach ($product as $n) {
        $i++;
        ?>
        <tr>
          <td><?=$i?></td>
          <td><?=$n['nama']?></td>
          <td><?=$n['hargajual']?></td>
          <td><?=$n['status']==1?'Aktif':'Non-
Aktif';?>
          </td>
          <td>
            <?php if($n['status']!=1){?>
              <button type="button" class="btn btn-
success" data-toggle="modal" data-target="#kembalikan-
<?=$n['id']?>"><i class="mdi mdi-check"></i></button>
            <?php }else{ ?>
              <button type="button" class="btn btn-
danger" data-toggle="modal" data-target="#del-<?=( $n['id'] )?>"><i
class="mdi mdi-delete"></i></button>
            <?php } ?>

```

```
                <button type="button" class="btn btn-
warning" data-toggle="modal" data-target="#edit-= $n['id'] ?"><i
class="mdi mdi-pencil"></i></button>

            </td>
        </tr>

        <?php }
    ?>

    </tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- END PAGE CONTENT -->
</div>
<!-- END CONTAINER -->
<?php foreach ($product as $n) {
    if($n['status']==1){?>

<!-- Modal delete-->
<div id="del-= $n['id'] ?" class="modal fade" role="dialog">
    <div class="modal-dialog">
        <!-- Modal content-->
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 class="modal-title">Delete Data Product</h4>
```

```

        </div>
        <form action="<?=base_url()?>manager/deleteproduct/"
method="post">
            <div class="modal-body">
                <input type="hidden" name="id"
value="<?=$n['id']?>">
                <p>Apakah Anda Yakin akan Menghapus product
<?=$n['nama']?>?</p>
            </div>

            <div class="modal-footer">
                <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
                <button type="submit" class="btn btn-success"
>Ok</button>
            </div>
        </form>
    </div>
</div>
</div>
<?php }else{ ?>
    <!-- Modal kembalikan-->
    <div id="kembalikan-<?=$n['id']?>" class="modal fade"
role="dialog">
        <div class="modal-dialog">
            <!-- Modal content-->
            <div class="modal-content">
                <div class="modal-header">
                    <button type="button" class="close" data-
dismiss="modal">&times;</button>
                    <h4 class="modal-title">Aktifkan Data Product</h4>
                </div>
                <form action="<?=base_url()?>manager/activeproduct/"
method="post">
                    <div class="modal-body">

```

```

        <input type="hidden" name="id"
value="<?=$n['id']?>"
        <p>Apakah Anda Yakin akan mengaktifkan User
<?=$n['nama']?>?</p>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
        <button type="submit" class="btn btn-success"
>Ok</button>
    </div>
</form>
</div>

</div>
</div>

<?php } ?>
<!-- Modal -->
<div id="edit-<?=$n['id']?>" class="modal fade" role="dialog">
    <div class="modal-dialog">
        <!-- Modal content-->
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 class="modal-title">Edit Data Product</h4>
            </div>
            <form action="<?=base_url()?>manager/editproduct"
method="post">
                <div class="modal-body">
                    <input type="hidden" name="id" value="<?=$n['id']?>" >
                    <div class="form-group">
                        <input type="text" name="nama"
value="<?=$n['nama']?>" required="" placeholder="Nama" class="form-
control">

```

```

        </div>
        <div class="form-group">
            <input type="number" name="hargajual"
value="<?=$n['hargajual']?>" required="" placeholder="Harga Jual"
class="form-control">
        </div>
        <div class="form-group">
            <input type="number" name="hargabeli"
value="<?=$n['hargabeli']?>" required="" placeholder="Harga Beli"
class="form-control">
        </div>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
        <button type="submit" class="btn btn-success"
>Save</button>
    </div>
</form>
</div>

</div>
</div>
<?php } ?>

```

Kode Program *Controller* Mengelola Data Produk Manager

```

public function viewproduct()
{
    $id =$this->session->userdata('id');
    $bantu['data'] = $this->model->md_getdata($id);
    $data['product'] = $this->model->md_product_all();

    $this->load->view('header',$bantu);
    $this->load->view('dproduk',$data);
    $this->load->view('footer');
}

```

```
}
public function tambahproduct() {
    $data['nama'] = $this->input->post('nama');
    $data['hargajual'] = $this->input->post('hargajual');
    $data['hargabeli'] = $this->input->post('hargabeli');

    if (empty($data['nama'])) {
        $input = false;
    } else {
        $this->model->md_tambahproduct($data);
        $input = true;
    }
    if ($input == true) {
        redirect(base_url('/manager/viewproduct')
        . "?pesan=berhasil");
    } else {
        redirect(base_url('/manager/viewproduct')
        . "?pesan=gagal");
    }
}

public function editproduct() {
    $data['id'] = $this->input->post('id');
    $data['nama'] = $this->input->post('nama');
    $data['hargajual'] = $this->input->post('hargajual');
    $data['hargabeli'] = $this->input->post('hargabeli');

    if ($data['nama'] == null) {
        $input = false;
    } else {
        $this->model->md_editproduct($data);
        $input = true;
    }
    if ($input == true) {
```

```
        redirect(base_url('/manager/viewproduct')
        .
        "?pesan=berhasiledit");
    }else{
        redirect(base_url('/manager/viewproduct')
        .
        "?pesan=gagal");
    }
}

public function deleteproduct() {
    $data['id'] = $this->input->post('id');

    if ($data['id']== null){
        $input = false;
    }else{

        $this->model->md_deleteproduct($data);
        $input = true;
    }
    if ($input == true){
        redirect(base_url('/manager/viewproduct')
        .
        "?pesan=berhasildelete");
    }else{
        redirect(base_url('/manager/viewproduct')
        .
        "?pesan=gagal");
    }
}

public function activeproduct() {
    $data['id'] = $this->input->post('id');

    if ($data['id']== null){
        $input = false;
    }else{

        $this->model->md_activeproduct($data);
        $input = true;
    }
}
```



```

        if ($input == true){
            redirect(base_url('/manager/viewproduct')
                .
                "?pesan=Active");
        }else{
            redirect(base_url('/manager/viewproduct')
                .
                "?pesan=gagal");
        }
    }
}

```

Kode Program *Controler* Mengelola Data Produk Pegawai

```

public function __construct()
{
    parent::__construct();
    $this->load->helper('url');
    $this->load->model('model');
    $level = $this->session->userdata('hakAkses');
    if($level != 2){
        redirect(base_url('clogin/logout?pesan=gagal'));
    }
}

public function viewproduct()
{
    $id =$this->session->userdata('id');
    $bantu['data'] = $this->model->md_getdata($id);
    $data['product'] = $this->model->md_product_all();
    $this->load->view('headerpegawai',$bantu);
    $this->load->view('dproduk',$data);
    $this->load->view('footer');
}

public function tambahproduct() {
    $data['nama'] = $this->input->post('nama');
    $data['hargajual'] = $this->input->post('hargajual');
    $data['hargabeli'] = $this->input->post('hargabeli');

    if (empty($data['nama'])){

```

```
        $input = false;
    }else{

        $this->model->md_tambahproduct($data);
        $input = true;
    }
    if ($input == true){
        redirect(base_url('/pegawai/viewproduct')
"?pesan=berhasil");
    }else{
        redirect(base_url('/pegawai/viewproduct')
"?pesan=gagal");
    }
}

public function editproduct() {
    $data['id'] = $this->input->post('id');
    $data['nama'] = $this->input->post('nama');
    $data['hargajual'] = $this->input->post('hargajual');
    $data['hargabeli'] = $this->input->post('hargabeli');

    if ($data['nama']== null){
        $input = false;
    }else{

        $this->model->md_editproduct($data);
        $input = true;
    }
    if ($input == true){
        redirect(base_url('/pegawai/viewproduct')
"?pesan=berhasiledit");
    }else{
        redirect(base_url('/pegawai/viewproduct')
"?pesan=gagal");
    }
}

public function deleteproduct() {
```

```
        $data['id'] = $this->input->post('id');

        if ($data['id']== null){
            $input = false;
        }else{

            $this->model->md_deleteproduct($data);
            $input = true;
        }
        if ($input == true){
            redirect(base_url('/pegawai/viewproduct')
"?pesan=berhasildelete");
        }else{
            redirect(base_url('/pegawai/viewproduct')
"?pesan=gagal");
        }
    }

    public function activeproduct() {
        $data['id'] = $this->input->post('id');

        if ($data['id']== null){
            $input = false;
        }else{

            $this->model->md_activeproduct($data);
            $input = true;
        }
        if ($input == true){
            redirect(base_url('/pegawai/viewproduct')
"?pesan=Active");
        }else{
            redirect(base_url('/pegawai/viewproduct')
"?pesan=gagal");
        }
    }
}
```

Kode Program *Model* Mengelola Data Produk

```
public function md_product_all() {
    $query = $this->db->get('product')->result_array();
    return $query;
}

public function md_product_active() {
    $this->db->where('status',1);
    $query = $this->db->get('product')->result_array();
    return $query;
}

public function md_tambahproduct($input) {
    $query=$this->db->insert('product',array(
        'id'=>'',
        'nama'=>$input['nama'],
        'hargajual'=>$input['hargajual'],
        'hargabeli'=>$input['hargabeli'],
        'status'=>1
    ));
    return $query;
}

public function md_editproduct($input) {
    $this->db->where('id',$input['id']);
    $query =$this->db->update('product',array('nama'=>$input['nama'],
        'hargabeli'=>$input['hargabeli'],
        'hargajual'=>$input['hargajual'],
    ));

    return $query;
}

public function md_deleteproduct($input) {
    $this->db->where('id',$input['id']);
    $query =$this->db->update('product',array('status'=>'0'));
}
```

```

    return $query;
}
public function md_activeproduct($input) {
    $this->db->where('id',$input['id']);
$query = $this->db->update('product',array('status'=>'1'));
    return $query;
}

```

E4. Kode Program Mengelola Data Penjualan

Kode Program <i>View</i> Mengelola Data Penjualan
<pre> <!-- PAGE WRAPPER --> <div class="page-wrapper"> <!-- CONTAINER --> <div class="container-fluid"> <div class="row page-titles"> <div class="col-md-5 col-8 align-self-center"> <h3 class="text-themecolor m-b-0 m-t-0"><i class="mdi mdi- package-variant"></i>Data Penjualan</h3> <ol class="breadcrumb"> <li class="breadcrumb-item active">Data Trasaksi </div> </div> </div> <!-- PAGE CONTENT --> <div class="row"> <div class="col-12"> <?php if (isset(\$_GET['pesan'])) { if (\$_GET['pesan'] == 'berhasil') {?> <div class="alert alert-success" role="alert"> success! Data berhasil di tambahkan </div> } } </?php> </div> </div> </pre>

```
<?php }else if($_GET['pesan'] == 'berhasildelete'){?>
    <div class="alert alert-danger" role="alert">
        <strong>success!</strong> <a href="#" class="alert-
link">Data sudah di hapus</a>
    </div>

<?php }else if($_GET['pesan'] == 'berhasiledit'){?>
    <div class="alert alert-danger" role="alert">
        <strong>success!</strong> <a href="#" class="alert-
link">Data berhasil dirubah</a>
    </div>

<?php }?>
<div class="card">
    <div class="card-block">
        <h4 class="card-title">Data Product</h4>

        <hr>
        <button type="button" class="btn btn-info" data-
toggle="modal" data-target="#myModal" style="margin-Bottom:10px"><i
class="mdi mdi-plus"></i>Tambah data</button>

        <!-- Modal -->
        <div id="myModal" class="modal fade" role="dialog">
            <div class="modal-dialog">

                <!-- Modal content-->
                <div class="modal-content">
                    <div class="modal-header">
                        <button type="button" class="close" data-
dismiss="modal">&times;</button>
                        <h4 class="modal-title">Tambah Data Product</h4>
                    </div>
                    <form
action="<?=base_url()?>manager/tambahtransaksi" method="post">
                        <div class="modal-body">
```

```

        <div class=form-group>
            <select name=idproduct class=form-
control">
                <?php foreach ($product as $a) { ?>
                    <option
value="<?=$a['id']?>"><?=$a['nama']?></option>
                <?php } ?>
            </select>
        </div>
        <div class="form-group">
            <input type="date" name="tanggal"
required="" placeholder="tanggal" class="form-control">
        </div>
        <div class="form-group">
            <input type="number" name="jumlah"
required="" placeholder="Jumlah" class="form-control">
        </div>
        <div class="modal-footer">
            <button type="button" class="btn btn-default"
data-dismiss="modal">Close</button>
            <button type="submit" class="btn btn-success"
>Save</button>
        </div>
    </form>
</div>
</div>

<div class="table-responsive">
    <table id="penj" class="table table-striped">
        <thead>
            <tr>
                <th>NO</th>

```

```
<th>Nama Product</th>
<th>Bulan Penjualan</th>
<th>Jumlah</th>
<th>Aksi</th>
</tr>
</thead>
<tbody>
<?php $i=0;
foreach ($transaksi as $n) {
    $i++;
    ?>
    <tr>
        <td><?=$i?></td>
        <td><?=$n['nama']?></td>
        <td><?=
            Date('F Y',strtotime($n['tanggal']));
        ?></td>
        <td><?=$n['jumlah']?></td>
        <td>
            <button type="button" class="btn btn-danger"
data-toggle="modal" data-target="#del-<?=$n['id']?>"><i class="mdi
mdi-delete"></i></button>
            <button type="button" class="btn btn-
warning" data-toggle="modal" data-target="#edit-<?=$n['id']?>"><i
class="mdi mdi-pencil"></i></button>
        </td>
    </tr>
</tbody>
</table>
</?php }
</div>
```



```

        ?>

        </tbody>
    </table>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- END PAGE CONTENT -->
</div>
<!-- END CONTAINER -->
<?php foreach ($transaksi as $n) {

    ?>

    <!-- Modal delete-->
    <div id="del-= $n['id'] ?" class="modal fade" role="dialog">
        <div class="modal-dialog">
            <!-- Modal content-->
            <div class="modal-content">
                <div class="modal-header">
                    <button type="button" class="close" data-
dismiss="modal">&times;</button>
                    <h4 class="modal-title">Delete Data Product</h4>
                </div>
                <form action="= base_url() ?manager/deletetransaksi/"
method="post">
                    <div class="modal-body">
                        <input type="hidden" name="id" value="= $n['id'] ?">
                        <p>Apakah Anda Yakin akan Menghapus transaksi
<= $n['nama'] ?&gt; pada tanggal &lt;<?=
date('d-F-Y', strtotime($n['tanggal'])) ?&gt;?&gt;&lt;/p&gt;
                    &lt;/div&gt;
                &lt;/form&gt;
            &lt;/div&gt;
        &lt;/div&gt;
    &lt;/div&gt;
}
</pre

```

```

        <div class="modal-footer">
            <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
            <button type="submit" class="btn btn-success"
>Ok</button>
        </div>
    </form>
</div>
</div>
</div>
</div>
<!-- Modal -->
<div id="edit-= $n['id'] ?" class="modal fade" role="dialog">
    <div class="modal-dialog">
        <!-- Modal content-->
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 class="modal-title">Edit Data Transaksi</h4>
            </div>
            <form action="= base_url() ?manager/edittransaksi"
method="post">
                <div class="modal-body">
                    <input type="hidden" name="id" value="= $n['id'] ?" >
                    <div class="form-group">
                        <input type="date" name="tanggal"
value="= $n['tanggal'] ?" required="" placeholder="tanggal"
class="form-control">
                    </div>
                    <div class="form-group">
                        <input type="number" name="jumlah"
value="= $n['jumlah'] ?" required="" placeholder="Jumlah"
class="form-control">
                    </div>
                </div>
            </form>
        </div>
    </div>
</div>

```

```

        </div>
        <div class="modal-footer">
            <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
            <button type="submit" class="btn btn-success"
>Save</button>
        </div>
    </form>
</div>

</div>
</div>
<?php } ?>

```

Kode Program *Controller* Mengelola Data Penjualan Manager

```

public function viewtransaksi()
{
    $id =$this->session->userdata('id');
    $bantu['data'] = $this->model->md_getdata($id);
    $data['product'] = $this->model->md_product_active();
    $data['transaksi'] = $this->model->md_transaksi_all();

    $this->load->view('header',$bantu);
    $this->load->view('dpenjualan',$data);
    $this->load->view('footer');
}
public function tambahtransaksi() {

    $data['jumlah'] = $this->input->post('jumlah');
    $data['idproduct'] = $this->input->post('idproduct');
    $data['tanggal'] = $this->input->post('tanggal');

    if (empty($data['idproduct'])) {
        $input = false;
    } else {

```

```
        $this->model->md_tambahtransaksi($data);

        $input = true;
    }
    if ($input == true){
        redirect(base_url('/manager/viewtransaksi')
"?pesan=berhasil");
    }else{
        redirect(base_url('/manager/viewtransaksi')
"?pesan=gagal");
    }
}

public function edittransaksi() {
    $data['id'] = $this->input->post('id');
    $data['jumlah'] = $this->input->post('jumlah');
    $data['tanggal'] = $this->input->post('tanggal');

    if ($data['jumlah']== null){
        $input = false;
    }else{

        $this->model->md_edittransaksi($data);
        $input = true;
    }
    if ($input == true){
        redirect(base_url('/manager/viewtransaksi')
"?pesan=berhasiledit");
    }else{
        redirect(base_url('/manager/viewtransaksi')
"?pesan=gagal");
    }
}

public function deletetransaksi(){
    $data['id'] = $this->input->post('id');
```

```
$this->model->md_deletetransaksi($data);

        redirect(base_url('/manager/viewtransaksi')
                .
                "?pesan=berhasildelete");
```

Kode Program *Controller* Mengelola Data Penjualan Pegawai

```
public function viewtransaksi()
{
    $id =$this->session->userdata('id');
    $bantu['data'] = $this->model->md_getdata($id);
    $data['product'] = $this->model->md_product_active();
    $data['transaksi'] = $this->model->md_transaksi_all();

    $this->load->view('headerpegawai',$bantu);
    $this->load->view('dpenjualan',$data);
    $this->load->view('footer');
}

public function tambahtransaksi() {

    $data['jumlah'] = $this->input->post('jumlah');
    $data['idproduct'] = $this->input->post('idproduct');
    $data['tanggal'] = $this->input->post('tanggal');

    if (empty($data['idproduct'])) {
        $input = false;
    }else{

        $this->model->md_tambahtransaksi($data);

        $input = true;
    }
    if ($input == true){
        redirect(base_url('/pegawai/viewtransaksi')
                .
                "?pesan=berhasil");
    }else{
```

```
        redirect(base_url('/pegawai/viewtransaksi'))
        .
        "?pesan=gagal");
    }
}

public function edittransaksi() {
    $data['id'] = $this->input->post('id');
    $data['jumlah'] = $this->input->post('jumlah');
    $data['tanggal'] = $this->input->post('tanggal');

    if ($data['jumlah']== null){
        $input = false;
    }else{

        $this->model->md_edittransaksi($data);
        $input = true;
    }
    if ($input == true){
        redirect(base_url('/pegawai/viewtransaksi'))
        .
        "?pesan=berhasiledit");
    }else{
        redirect(base_url('/pegawai/viewtransaksi'))
        .
        "?pesan=gagal");
    }
}

public function deletetransaksi(){
    $data['id'] = $this->input->post('id');
    $this->model->md_deletetransaksi($data);

    redirect(base_url('/pegawai/viewtransaksi'))
    .
    "?pesan=berhasildelete");
}
}
```

Kode Program *Model* Mengelola Data Penjualan

```
public function md_transaksi_all() {
$this->db->select('transaksi.id as id, tanggal,jumlah,nama');
$this->db->join('product','product.id=transaksi.idproduct');
$query = $this->db->get('transaksi')->result_array();
    return $query;
}
public function md_tambahtransaksi($input) {
    $query=$this->db->insert('transaksi',array(
        'id'=>'',
        'idproduct'=>$input['idproduct'],
        'tanggal'=>$input['tanggal'],
        'jumlah'=>$input['jumlah'],

    ));
    return $query;
}
public function md_edittransaksi($input) {
$this->db->where('id',$input['id']);
$query = $this->db->update('transaksi',array(
    'tanggal'=>$input['tanggal'],
    'jumlah'=>$input['jumlah']
));

    return $query;
}
public function md_deletetransaksi($input){
    $this->db->where('id',$input['id']);
    $query = $this->db->delete('transaksi');
    return $query;
}
```

E5. Kode Program Mengubah Data Profil

Kode Program <i>View</i> Mengubah Data Profil
<!-- PAGE WRAPPER -->
<div class="page-wrapper">

```

<!-- CONTAINER -->
<div class="container-fluid">
  <div class="row page-titles">
    <div class="col-md-5 col-8 align-self-center">
      <h3 class="text-themecolor m-b-0 m-t-0"><i class="mdi mdi-account-location"></i>Profil</h3>
      <ol class="breadcrumb">
        <li class="breadcrumb-item active"><a href="javascript:void(0)">Data Profil</a></li>
      </ol>
    </div>
  </div>
<!-- PAGE CONTENT -->
<div class="row">
  <div class="col-12">
    <div class="card">
      <div class="card-block">
        <h4 class="card-title">Profil</h4>
        <form action="" method="post">
          <div class="form-group">
            <p>Username</p>
            <input type="text" name="username" value="<?=$data['username']?>" class="form-control col-xs-6" required="required">
          </div>
          <div class="form-group">
            <p>Password</p>
            <input type="text" name="password" value="<?=$data['password']?>" class="form-control col-xs-6" required="required">
          </div>
          <button type="submit" class="btn btn-info">Simpan</button>
        </form>
      </div>
    </div>
  </div>
</div>

```



```

    </div>
  </div>
</div>
</div>

```

Kode Program *Controller* Mengubah Data Profil

```

public function profil(){
    $id =$this->session->userdata('id');
    $bantu['data'] = $this->model->md_getdata($id);
    $level = $this->session->userdata('hakAkses');

    if($this->input->post('username')){
        $data['id'] = $id;
        $data['username'] = $this->input->post('username');
        $data['password'] = $this->input->post('password');
        $this->model->updateprofil($data);

        redirect(base_url('clogin/logout?pesan=berhasilupdate'));
    }
    if($level == 2){
        $this->load->view('headerpegawai',$bantu);
        $this->load->view('profil',$bantu);
        $this->load->view('footer');
    }else{
        $this->load->view('header',$bantu);
        $this->load->view('profil',$bantu);
        $this->load->view('footer');
    }
}
}

```

Kode Program *Model* Mengubah Data Profil

```

public function updateprofil($input) {

```

```

$this->db->where('id',$input['id']);
$query = $this->db->
>update('user',array('username'=>$input['username'],
    'password'=>$input['password']));
return $query;
}

```

E6. Kode Program *Logout*Kode Program *Controller Logout*

```

public function logout(){
    if($_GET['pesan']=="gagal"){
        $this->session->sess_destroy();
        redirect(base_url()."?pesan=error");
    }else if($_GET['pesan']=="gagallogin"){
        redirect(base_url()."?pesan=belum");
    }else if($_GET['pesan']=="berhasilupdate"){
        $this->session->sess_destroy();
        redirect(base_url()."?pesan=berhasilupdate");
    }
    $this->session->sess_destroy();
    redirect(base_url());
}

```

Kode Program *Model Logout*

```

public function md_getdata($id) {
    if($id == null){
        redirect(site_url('/clogin/logout') . "?pesan=gagallogin");
    }else{
        $query = $this->db->query("SELECT * FROM user where id='$id'");
        return $query->row_array();
    }
}

```

LAMPIRAN F. Black Box

Tabel F. Pengujian Balck Box

No.	Menu	Fungsi	Aksi	Hasil	Ket
1.	Mengelola Data Penjualan (Manager dan Pegawai)	Halaman yang menampilkan tabel data produk, tambah data produk, edit data produk dan status data produk	Memilih tombol tambah data dan klik <i>save</i>	Menampilkan <i>from</i> tambah data	√
			Memilih tombol tambah data dan klik <i>save</i>	Menampilkan tabel data produk, tambah data produk, edit data produk, status data produk dan <i>alert</i> data berhasil di tambah	√
			Memilih tombol tambah data dan klik <i>close</i>	Menampilkan tabel data produk, tambah data produk, edit data produk dan status data produk	√
			Memilih <i>icon edit</i> data dan klik <i>save</i>	Menampilkan tabel data produk, tambah data produk, edit data produk, status data produk dan <i>alert</i> data berhasil dirubah	√

Memilih <i>icon edit</i> data dan klik <i>close</i>	Menampilkan tabel data produk, tambah data produk, edit data produk dan status data produk	√
---	--	---

Memilih <i>icon delete</i> data dan klik <i>ok</i>	Menampilkan tabel data produk, tambah data produk, edit data produk, status data produk dan <i>alert</i> data berhasil dihapus	√
--	--	---

Memilih <i>icon delete</i> data dan klik <i>close</i>	Menampilkan tabel data produk, tambah data produk, edit data produk dan status data produk	√
---	--	---

2. Mengelola Data Prediksi (Manager)	Menampilkan <i>dropbox</i> jenis produk dan tabel peramalan	Memilih <i>dropbox</i> jenis produk	Menampilkan <i>dropbox</i> jenis produk dan tabel peramalan serta hasil perhitungan peramalan sesuai jenis produk yang di pilih	√
--------------------------------------	---	-------------------------------------	---	---

3.	Mengelola Data Profil (Manager dan Pegawai)	Menampilkan halaman dan form profil <i>username</i> dan <i>password</i> yang telah terisi sesuai <i>login</i> pengguna	Memilih tombol simpan dan yang sesuai	Menampilkan halaman <i>login</i> serta <i>alert username</i> dan <i>password</i> sudah di <i>update</i> , silakan <i>login</i> kembali	√
4.	<i>Logout</i> (Manager dan Pegawai)	Digunakan untuk keluar dari sistem	klik logout pada pojok kiri bawah sistem	Melakukan proses keluar dari sistem dan menampilkan halaman <i>login</i>	√