



**SISTEM INFORMASI PENJUALAN PRODUK UNGGULAN
KABUPATEN LUMAJANG DENGAN FITUR PEMILIHAN PRODUK
UNGGULAN MENGGUNAKAN ALGORITMA *WEIGHTED PRODUCT*
(WP)**

SKRIPSI

Oleh :

Kiki Amarita

112410101053

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2018



**SISTEM INFORMASI PENJUALAN PRODUK UNGGULAN
KABUPATEN LUMAJANG DENGAN FITUR PEMILIHAN PRODUK
UNGGULAN MENGGUNAKAN ALGORITMA *WEIGHTED PRODUCT*
(WP)**

SKRIPSI

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

Oleh :

Kiki Amarita

112410101053

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS ILMU KOMPUTER

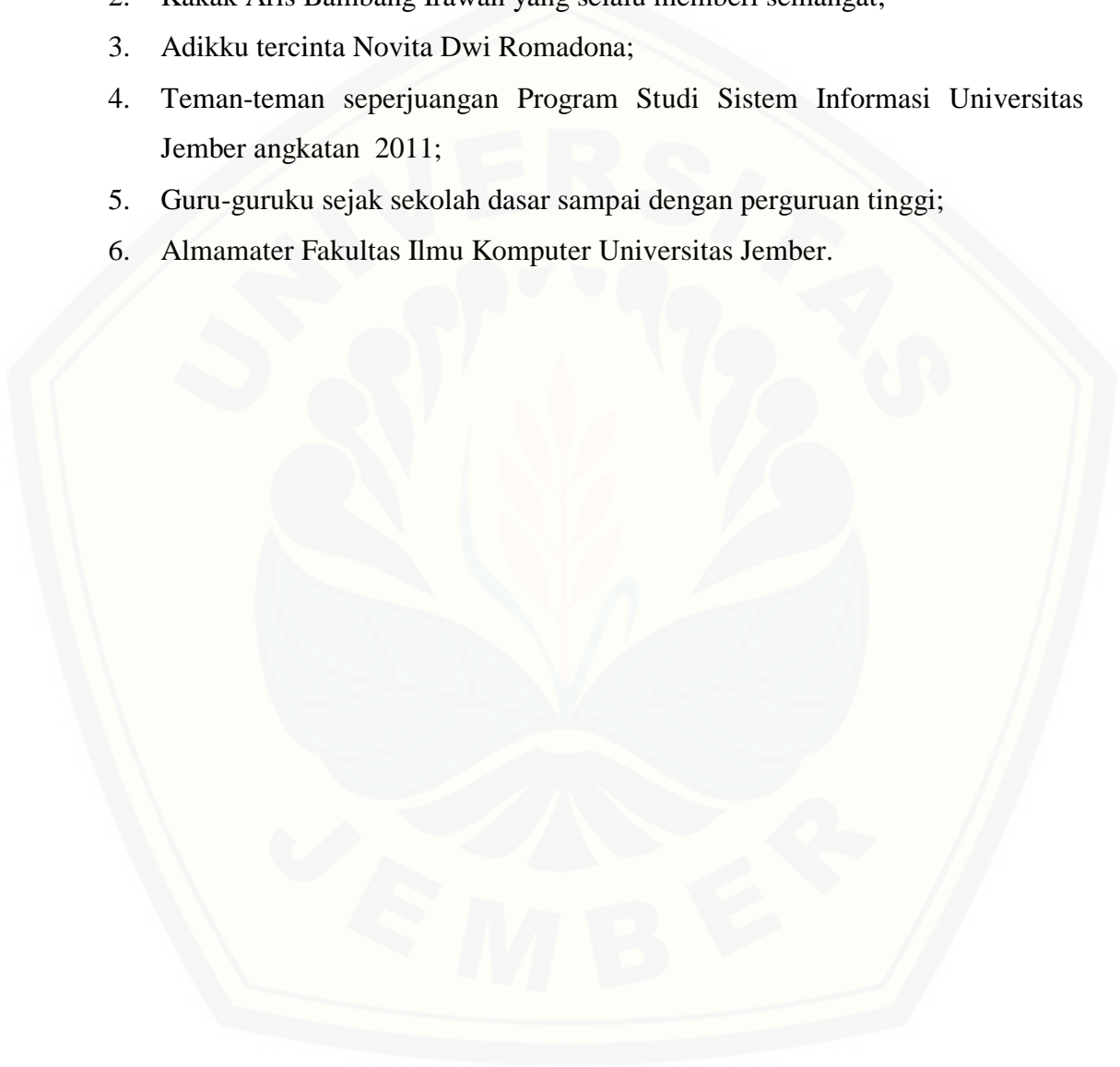
UNIVERSITAS JEMBER

2018

PERSEMBAHAN

Skripsi ini saya persembahkan untuk :

1. Ibu Suwarsini dan Ayah Juki yang tercinta;
2. Kakak Aris Bambang Irawan yang selalu memberi semangat;
3. Adikku tercinta Novita Dwi Romadona;
4. Teman-teman seperjuangan Program Studi Sistem Informasi Universitas Jember angkatan 2011;
5. Guru-guruku sejak sekolah dasar sampai dengan perguruan tinggi;
6. Almamater Fakultas Ilmu Komputer Universitas Jember.



MOTTO

“Lebih baik terlambat daripada tidak sama sekali. Jalan masih panjang, terus berjuang, pantang menyerah, perbaiki semua kekacauan”.



PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Kiki Amarita

NIM : 112410101053

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Sistem Informasi Penjualan Produk Unggulan Kabupaten Lumajang dengan Fitur Pemilihan Produk Unggulan Menggunakan Algoritma *Weighted Product* (WP), adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya, 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, 30 November 2018

Yang menyatakan,

Kiki Amarita

NIM 112410101053

SKRIPSI

**SISTEM INFORMASI PENJUALAN PRODUK UNGGULAN
KABUPATEN LUMAJANG DENGAN FITUR PEMILIHAN PRODUK
UNGGULAN MENGGUNAKAN ALGORITMA *WEIGHTD PRODUCT*
(WP)**

Oleh :

Kiki Amarita

112410101053

Pembimbing:

Pembimbing Utama : Drs. Antonius Cahya Prihandoko M.App.Sc.,Ph.D

NIP 19690928 199302 1 001

Pembimbing : Muhamad Arief Hidayat, S.Kom.,M.Kom.

Pendamping

NIP 19810123 201012 1 003

PENGESAHAN PEMBIMBING

Skripsi berjudul Sistem Informasi Penjualan Produk Unggulan Kabupaten Lumajang dengan Fitur Pemilihan Produk Unggulan Menggunakan Algoritma *Weighted Product* (WP) telah diuji dan disahkan pada:

Hari, tanggal : Jumat, 30 November 2018

Tempat : Program Studi Sistem Informasi Fakultas Ilmu Komputer
Universitas Jember

Disetujui oleh:

Pembimbing Utama,

Pembimbing Pendamping,

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

NIP 196909281993021001

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

NIP 198101232010121003

PENGESAHAN

Skripsi berjudul Sistem Informasi Penjualan Produk Unggulan Kabupaten Lumajang dengan Fitur Pemilihan Produk Unggulan Menggunakan Algoritma *Weighted Product* (WP) telah diuji dan disahkan pada:

Hari, tanggal : Jumat, 30 November 2018

Tempat : Program Studi Sistem Informasi Fakultas Ilmu Komputer
Universitas Jember

Tim Penguji:

Penguji I,

Penguji II,

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

Fahrobby Adnan, S.Kom., M.MSI
NIP 198706192014041001

Mengesahkan:
Dekan,

Prof. Dr. Saiful Bukhori, ST., M.Kom.
NIP 196811131994121001

RINGKASAN

Sistem Informasi Penjualan Produk Unggulan Kabupaten Lumajang dengan Fitur Pemilihan Produk Unggulan Menggunakan Algoritma *Weighted Product* (WP); Kiki Amarita, 112410101053; 2018: 167 Halaman; Fakultas Ilmu Komputer Universitas Jember.

Kabupaten Lumajang memiliki banyak produk makanan baik itu yang diolah maupun tanpa melakukan proses pengolahan. Karena banyaknya produk yang dihasilkan, untuk menentukan produk yang diprioritaskan sebagai produk unggulan atau produk khas di Kabupaten Lumajang masih memerlukan perhitungan yang tepat.

Permasalahan tersebut diperlukan sistem informasi penjualan produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan menggunakan metode *Weighted Product* (WP). Sistem dibangun untuk mempermudah pedagang dan pembeli dalam penjualan dan pembelian makanan ringan unggulan serta dalam menghitung produk yang unggulan. Metode WP merupakan metode yang menggunakan perkalian guna menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan dahulu dengan bobot atribut yang bersangkutan, menentukan kriteria yang akan digunakan, kemudian memberi nilai dari setiap kriteria tersebut. Sistem informasi pemilihan produk unggulan dibangun dengan menerapkan SDLC model *waterfall*.

Algoritma *Weighted Product* ini lebih efisien karena waktu yang dibutuhkan dalam perhitungan lebih singkat. Metode ini dipilih karena dapat menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perangkungan yang akan menghasilkan pilihan produk makanan ringan unggulan yang sesuai dengan kriteria. Algoritma *Weighting Product* merupakan algoritma yang dapat digunakan untuk menganalisis atau menghitung produk unggulan menggunakan kriteria yang sesuai untuk menentukan produk unggulan. Kriteria yang digunakan yaitu omset, ketersediaan pasar, daya saing, spesifikasi kekhasan, dan ketersediaan bahan baku.

Fitur pemilihan produk unggulan dapat diakses oleh admin dengan menginputkan data produk dan menginputkan nilai kriteria dengan memilih pilihan subkriteria yang ada. Apabila pada kriteria diberikan subkriteria paling tinggi oleh admin maka perangkingan produk akan muncul pada urutan teratas. Sedangkan bila kriteria diberikan subkriteria dengan nilai terendah maka perangkingan produk tersebut berada diurutan paling bawah. Kemudian bila nilai subkriteria diubah, maka urutan dan hasil perhitungan WP juga berubah.



PRAKATA

Puji syukur kehadiran Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi dengan judul “Sistem Informasi Penjualan Produk Unggulan Kabupaten Lumajang dengan Fitur Pemilihan Produk Unggulan Menggunakan Algoritma *Weighted Product* (WP). 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. Drs. Antonius Cahya P, M.App., Sc Ph.D selaku Dosen Pembimbing Utama dan Muhammad Arif Hidayat S.Kom, M.Kom selaku Dosen Pembimbing Pendamping yang telah meluangkan waktu, pikiran dan perhatian dalam penulisan skripsi;
3. Nelly Oktavia Adiwijaya, S,Si., MT., selaku Dosen Pembimbing Akademik yang telah membimbing selama penulis menjadi mahasiswa;
4. Seluruh Bapak dan Ibu dosen beserta staf karyawan di Program Studi Sistem Informasi Universitas Jember;
5. Kakak Aris Bambang Irawan yang setia menemani dan memotivasi hingga skripsi ini dapat terselesaikan.
6. Teman-teman Angkatan 2011 (NEFOTION);

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

Jember, 30 November 2018

Penulis

DAFTAR ISI

PERSEMBAHAN	ii
MOTTO	iii
PERNYATAAN	iv
SKRIPSI	v
PENGESAHAN PEMBIMBING	vi
PENGESAHAN	vii
RINGKASAN	viii
PRAKATA	x
DAFTAR ISI	xi
DAFTAR TABEL	xvi
DAFTAR GAMBAR	xvii
DAFTAR LAMPIRAN	xviii
BAB 1. PENDAHULUAN	1
1.1 LATAR BELAKANG	1
1.2 PERUMUSAN MASALAH.....	2
1.3 TUJUAN.....	2
1.4 MANFAAT	3
1.5 SISTEMATIKA PENULISAN	3
BAB 2. TINJAUAN PUSTAKA	5
2.1 Penelitian Terdahulu.....	5
2.2 Metode Weighted Product (WP)	6
2.3 Produk Unggulan.....	9
2.4 Penggunaan Weighted Product (WP) dalam Menganalisis Produk Unggulan	11
2.5 Sistem Informasi Penjualan	13
BAB 3. METODOLOGI PENELITIAN	16
3.1 Jenis Penelitian	16
3.2 Metode Penelitian	16
3.3 Langkah Perancangan Sistem.....	17
3.3.1 Analisa Kebutuhan	18
3.3.2 Desain.....	19

3.3.3	Implementasi	20
3.3.4	Pengujian.....	20
3.3.5	<i>Maintenance</i>	20
3.4	Gambaran Sistem.....	21
3.5	Pengujian Algoritma Weighting Product	23
BAB 4. ANALISIS DAN PERANCANGAN SISTEM.....		29
4.1	Statement Of Purpose.....	29
4.2	Analisa Kebutuhan	29
4.2.1	Kebutuhan Fungsional	29
4.2.2	Kebutuhan non fungsional	30
4.3	Desain Sistem.....	31
4.3.1	<i>Business Process</i>	31
4.3.2	<i>Use Case Diagram</i>	32
4.3.3	Definisi Aktor.....	32
4.3.4	Definisi <i>Usecase</i>	33
4.3.5	<i>Usecase Skenario</i>	34
4.3.6	<i>Activity Diagram</i>	39
4.3.7	<i>Sequence Diagram</i>	43
4.3.8	<i>Class Diagram</i>	47
4.3.9	<i>Entity Relationship Diagram (ERD)</i>	49
4.4	Penulisan Kode Program.....	49
4.5	Pengujian Sistem.....	50
4.5.1	<i>White Box Testing</i>	50
4.5.2	<i>Black Box Testing</i>	52
BAB 5. HASIL DAN PEMBAHASAN.....		56
5.1	Sistem Informasi Penjualan Produk Unggulan Kabupaten Lumajang dengan Fitur Pemilihan Produk Unggulan	56
5.2	Implementasi Sistem	57
5.3	Hasil Implementasi WP pada Sistem	60
5.3.1	Kriteria dan Bobot Kriteria	61
5.3.2	Perhitungan Bobot Secara Manual.....	63
5.4	Hasil Pengujian Sistem	71

5.5	Hasil pengujian Algoritma Weighted Product	72
5.6	Pembahasan.....	74
BAB 6. PENUTUP		78
6.1	Kesimpulan	78
6.2	Saran.....	79
DAFTAR PUSTAKA		80
LAMPIRAN.....		82
Lampiran A. Skenario Diagram		82
A.1	Skenario Login.....	82
A.2	Skenario Melihat Data Pelanggan.....	83
A.3	Skenario Mengolah Data Kriteria dan Subkriteria	84
A.4	Skenario Melihat Hasil Analisis WP	86
A.5	Skenario Mengolah Data Produk.....	87
A.6	Skenario Mengolah Data Penjualan.....	89
A.7	Skenario Mengolah Data Ongkos Kirim	90
A.8	Skenario Mengolah Profil.....	91
A.9	Skenario Mengolah Keranjang	92
A.10	Skenario Mengolah History Pembelian	93
A.11	Skenario Pemesanan	94
A.12	Skenario Registrasi	95
A.13	Skenario Logout.....	96
Lampiran B. Lampiran Activity Diagram		96
B.1	Activity Diagram Login.....	96
B.2	Activity Diagram Melihat Data Pelanggan.....	97
B.3	Activity Diagram Mengolah Data Kriteria dan Subkriteria.....	97
B.4	Activity Diagram Hasil analisis WP	98
B.5	Activity Diagram Mengolah Data Produk.....	98
B.6	Activity Diagram Mengolah Data Penjualan.....	99
B.7	Activity Diagram Mengolah Data Ongkos Kirim.....	99
B.8	Activity Diagram Mengolah Profil.....	100
B.9	Activity Diagram Mengolah Keranjang	100
B.10	Activity Diagram Mengolah History Pembelian	100

B.11	Activity Diagram Pemesanan	101
B.12	Activity Diagram Registrasi	101
B.13	Activity Diagram Logout	101
Lampiran C. Lampiran Sequence Diagram		102
C.1	Sequence Diagram Login.....	102
C.2	Sequence Diagram Melihat Data Pelanggan	102
C.3	Sequence Diagram Mengolah Data Kriteria dan Subkriteria	103
C.4	Sequence Diagram Melihat Hasil Analisis WP	103
C.5	Sequence Diagram Mengolah Data Produk.....	104
C.6	Sequence Diagram Mengolah Data Penjualan.....	105
C.7	Sequence Diagram Mengolah Data Ongkos Kirim	105
C.8	Sequence Diagram Mengolah Profil.....	106
C.9	Sequence Diagram Mengolah Keranjang	106
C.10	Sequence Diagram Mengolah History Pembelian	106
C.11	Sequence Diagram Pemesanan	107
C.12	Sequence Diagram Registrasi	107
C.13	Sequence Diagram Logout.....	107
Lampiran D. Lampiran Pengujian White Box		108
D.1	<i>Function</i> account_login().....	108
D.2	<i>Function</i> account_logout().....	110
D.3	<i>Function</i> batal().....	111
D.4	<i>Function</i> cart()	111
D.5	<i>Function</i> buy()	113
D.6	<i>Function</i> customer_form().....	117
D.7	<i>Function</i> delete_inventory()	120
D.8	<i>Function</i> delete_item().....	120
D.9	<i>Function</i> detail_item()	122
D.10	<i>Function</i> detail_sale_form() pada dashboard	124
D.11	<i>Function</i> detail_sale_form() pada sale.....	126
D.12	<i>Function</i> get_kabupaten()	128
D.13	<i>Function</i> get_kecamatan()	129
D.14	<i>Function</i> get_kelurahan().....	130

D.15	<i>Function</i> inventory_form()	131
D.16	<i>Function</i> kriteria().....	133
D.17	<i>Function</i> pembelian().....	134
D.18	<i>Function</i> perhitungan_wp()	136
D.19	<i>Function</i> registrasi().....	137
D.20	<i>Function</i> reset_login().....	139
D.21	<i>Function</i> save_customer().....	140
D.22	<i>Function</i> save_inventory()	143
D.23	<i>Function</i> save_subkriteria()	146
D.24	<i>Function</i> simpan_wilayah()	147
D.25	<i>Function</i> subkriteria_delete().....	148
D.26	<i>Function</i> subkriteria_form()	149
D.27	<i>Function</i> update_bayar() pada dashboard.....	151
D.28	<i>Function</i> update_bayar() pada sale	153
D.29	<i>Function</i> update_riview()	157
D.30	<i>Function</i> wilayah().....	158
Lampiran E. Lampiran Implementasi Sistem		160
E.1	Halaman Data Kriteria	160
E.2	Halaman Perhitungan WP.....	161
E.3	Halaman Data Produk.....	161
E.4	Halaman Data Pelanggan.....	162
E.5	Halaman Data Ongkir	162
E.6	Halaman Data Penjualan.....	162
E.7	Halaman Profil.....	163
E.8	Halaman Keranjang	163
E.9	Halaman History Pembelian	163
Lampiran F. Lampiran Kode Program		164

DAFTAR TABEL

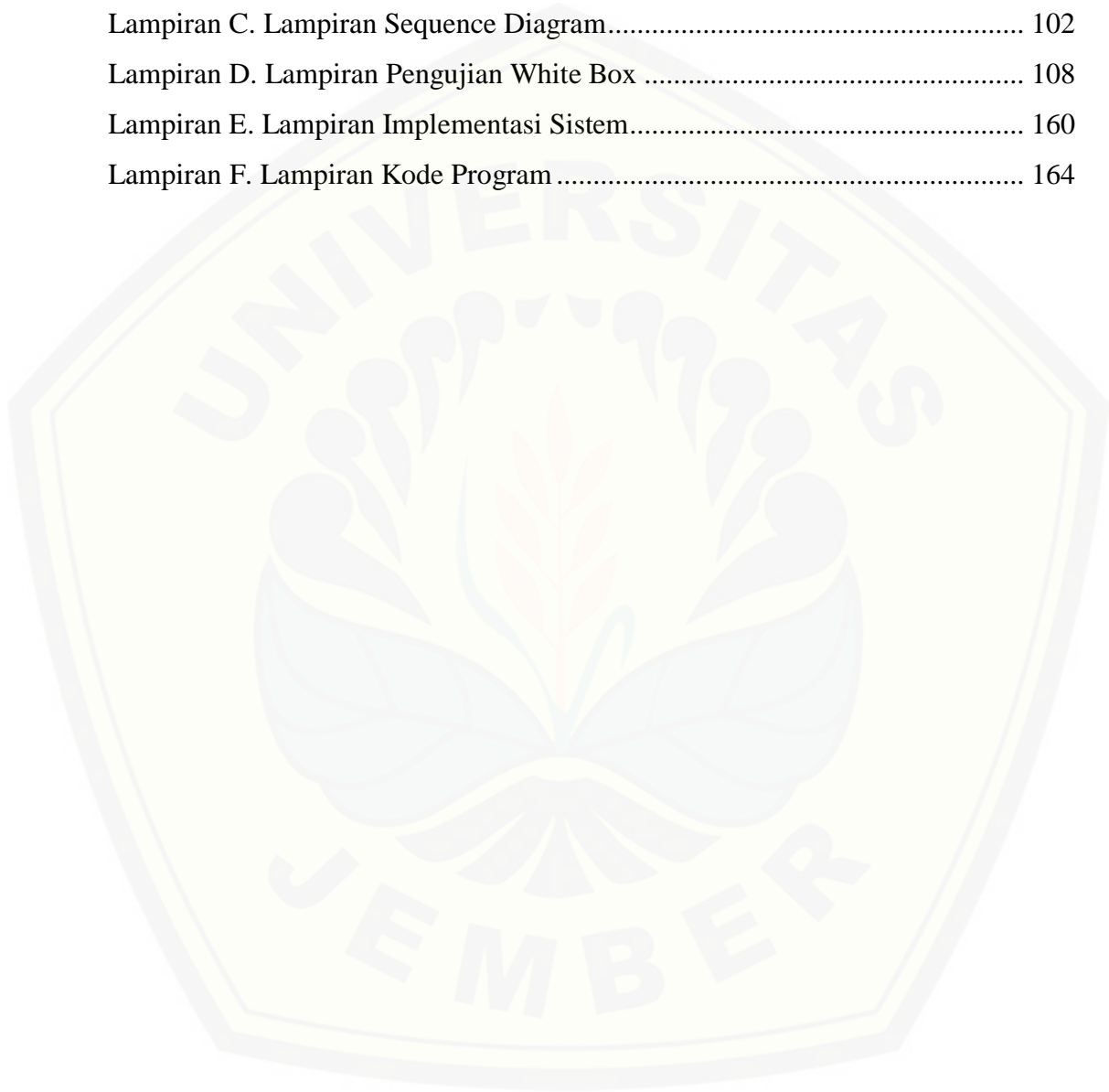
Tabel 2. 1 Contoh Tabel Kriteria dan Penilaiannya.....	6
Tabel 2. 2 Tabel Pembobotan Berdasarkan Tingkat Kepentingan.....	7
Tabel 2. 3 Contoh Tabel Kriteria dan Bobot Kriteria	11
Tabel 2. 4 Contoh Tabel Alternatif Produk dan Nilai Kriteria	12
Tabel 3.1 Kriteria dan Bobot Masing-Masing Kriteria.....	23
Tabel 3.2 Tabel Pembobotan Berdasarkan Tingkat Kepentingan.....	23
Tabel 3.3 Tabel Kriteria dan Bobot Awal serta Perbaikan Bobot.....	24
Tabel 3.4 Tabel Subkriteria dan Nilai Subkriteria	25
Tabel 3.5 Tabel Produk dan Nilai Kriteria.....	26
Tabel 3.6 Tabel Hasil Perhitungan Vektor V	27
Tabel 4. 1 Kebutuhan fungsional	30
Tabel 4. 2 Kebutuhan nonfungsional	30
Tabel 4. 3 Definisi Aktor	32
Tabel 4. 4 Definisi <i>usecase</i>	33
Tabel 4. 5 Usecase Skenario	34
Tabel 4. 6 Tabel Pengujian <i>Black Box</i>	52
Tabel 5. 1 Kriteria dan Bobot Kriteria	61
Tabel 5. 2 Range dan Nilai Subkriteria.....	62
Tabel 5. 3 Nilai Inputan Admin	64
Tabel 5. 4 Tabel Nilai Vektor S yang sudah dirangking.....	66
Tabel 5. 5 Tabel Nilai Vektor V yang sudah dirangking	68
Tabel 5. 6 Tabel Hasil Perhitungan WP Setelah Dirubah.....	69
Tabel 5. 7 Tabel Hasil Perangkingan	72

DAFTAR GAMBAR

Gambar 3. 1 Diagram Alir Tahapan Penelitian.....	16
Gambar 3. 2 Tahapan Model Waterfall.....	18
Gambar 3. 3 Gambaran Umum Sistem	22
Gambar 4. 1 Gambaran Komponen Sistem.....	31
Gambar 4. 2 <i>Usecase diagram system</i>	32
Gambar 4. 3 <i>Class Diagram</i> Sistem.....	48
Gambar 4. 4 <i>Entity Relathionship Diagram (ERD)</i>	49
Gambar 4. 5 Diagram alir <i>function inventory()</i>	51
Gambar 5. 1 Tampilan Halaman Utama Admin	57
Gambar 5. 2 Tampilan Halaman Utama Pembeli	57
Gambar 5. 3 Tampilan Form Tambah Data Produk.....	60
Gambar 5. 4 Tampilan Hasil Perhitungan WP.....	61
Gambar 5. 5 Gambar tampilan produk setelah dilakukan perhitungan dengan metode Weighted Product.....	76
Gambar 5. 6 Gambar tampilan Produk Unggulan Setelah Perubahan	77

DAFTAR LAMPIRAN

Lampiran A. Skenario Diagram	82
Lampiran B. Lampiran Activity Diagram	96
Lampiran C. Lampiran Sequence Diagram.....	102
Lampiran D. Lampiran Pengujian White Box	108
Lampiran E. Lampiran Implementasi Sistem.....	160
Lampiran F. Lampiran Kode Program	164



BAB 1. PENDAHULUAN

Bab ini merupakan bab pembuka pada buku skripsi ini. Pada bab ini akan membahas tentang latar belakang, perumusan masalah, tujuan, batasan masalah dan sistematika penulisan.

1.1 LATAR BELAKANG

Kabupaten Lumajang merupakan salah satu daerah yang berada di wilayah bagian selatan provinsi Jawa Timur. Kabupaten Lumajang terdiri dari 21 Kecamatan, 195 desa dan 7 kelurahan dengan batas-batas wilayah yaitu sebelah utara Kabupaten Probolinggo, sebelah timur Kabupaten Jember, sebelah selatan Samudera Indonesia, dan sebelah barat Kabupaten Malang. Kabupaten Lumajang terletak pada 112°53'-113°23' Bujur Timur dan 7°54'-8°23' Lintang Selatan. Lumajang merupakan kota yang dibidang subur, karena diapit oleh tiga gunung berapi yaitu Gunung Semeru, gunung Bromo, dan gunung Lamongan. Kota yang memiliki luas wilayah 1.790,90 km² ini memiliki iklim tropis dan sebagian besar berada pada dataran tinggi sehingga sangat efektif untuk mengembangkan pertanian dan perkebunan di Jawa Timur (Firdausy, 2012).

Kabupaten Lumajang saat ini memiliki banyak produk makanan baik itu yang diolah maupun tanpa melakukan proses pengolahan. Lahannya yang subur dan masih banyak lahan-lahan yang dapat dimanfaatkan untuk menanam bahan baku produk. Produk makanan yang diolah seperti kripik pisang, kripik singkong, kacang otok, dan lain-lain.

Banyaknya produk yang dihasilkan memerlukan perhitungan yang tepat untuk menentukan produk unggulan di Kabupaten Lumajang. Perhitungan produk makanan ringan unggulan saat ini dilakukan dengan melihat produk yang merupakan inovasi warga, diproduksi dalam kabupaten dan jarang ditemui atau bahkan tidak ada didaerah lain. Apabila individu atau kelompok memiliki inovasi baru terhadap suatu produk dan diproduksi sendiri oleh warga Kabupaten Lumajang serta tidak semua kabupaten atau kota memiliki produk yang sama, maka produk tersebut dapat dikatakan sebagai produk unggulan.

Berdasarkan pembahasan diatas, diperlukan sebuah sistem informasi untuk menganalisis produk mana yang unggul. Sistem informasi ini juga untuk memudahkan dalam membeli tanpa harus langsung datang ke Luamajang, namun pembeli dapat memesan produk yang ingin dibeli.

Untuk menganalisis produk unggulan digunakan algoritma *Weighting Product* karena algoritma ini dapat mencari alternatif yang paling optimal dari sejumlah alternatif optimal dengan kriteria tertentu. Dengan menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan. Dengan adanya sistem ini, diharapkan suatu daerah dapat menentukan prioritas produk unggulannya dan membantu mengambil keputusan serta meningkatkan penjualan produk unggulan.

1.2 PERUMUSAN MASALAH

1.2.1. Identifikasi Masalah

Berdasarkan uraian yang telah disampaikan dalam latar belakang mendefinisikan beberapa permasalahan yang harus diselesaikan dalam penulisan ini yaitu :

1. Bagaimana menerapkan algoritma *Weighted Product* (WP) dalam menentukan produk unggulan Kabupaten Lumajang?
2. Bagaimana membangun sistem informasi penjualan dan promosi produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan menggunakan algoritma *Weighted Product* (WP)?

1.2.2 Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut :

1. Dalam pembuatan tugas akhir ini, penelitian dilakukan di toko pusat oleh-oleh (Jati Arum).
2. Produk yang dibahas adalah produk makanan ringan.

1.3 TUJUAN

Tujuan yang ingin dicapai dalam penulisan ini adalah :

1. Menerapkan algoritma *Weighted Product* (WP) dalam menentukan produk unggulan Kabupaten Lumajang.

2. Membangun sistem informasi penjualan dan promosi produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan menggunakan algoritma *Weighted Product* (WP).

1.4 MANFAAT

1. Bagi objek penelitian.

Hasil penelitian diharapkan dapat dijadikan standart pemilihan produk unggulan makanan ringan.

2. Bagi penulis
 - a. Mengetahui kriteria-kriteria yang dapat digunakan sebagai standart pemilihan produk unggulan.
 - b. Mengetahui cara kerja algoritma *Weighted Product* dalam menghasilkan perangkungan yang dapat digunakan sebagai rekomendasi pemilihan produk unggulan.
3. Bagi perkembangan ilmu pengetahuan

Penelitian ini dapat digunakan sebagai bahan referensi dalam menambah pengetahuan maupun pengembangan lebih lanjut bagi peneliti lain.

1.5 SISTEMATIKA PENULISAN

Sistematika dan keruntutan penulisan tugas akhir ini terdiri dari enam bab dengan urutan sebagai berikut:

- a. BAB 1. PENDAHULUAN

Bab ini menguraikan latar belakang, rumusan masalah, tujuan dan sistematika penulisan.

- b. BAB 2. TINJAUAN PUSTAKA

Bab ini membahas mengenai tinjauan pustaka, definisi, pengertian, penjelasan teori dan metode yang berhubungan dengan permasalahan yang akan dibahas.

- c. BAB 3. METODE PENELITIAN

Bab ini menjelaskan tentang penelitian yang akan dilakukan dan tahapan yang harus dilalui. Metode penelitian terdiri dari tiga tahapan yaitu pengumpulan data, analisis dan perancangan sistem.

d. **BAB 4. ANALISIS DAN PERANCANGAN SISTEM**

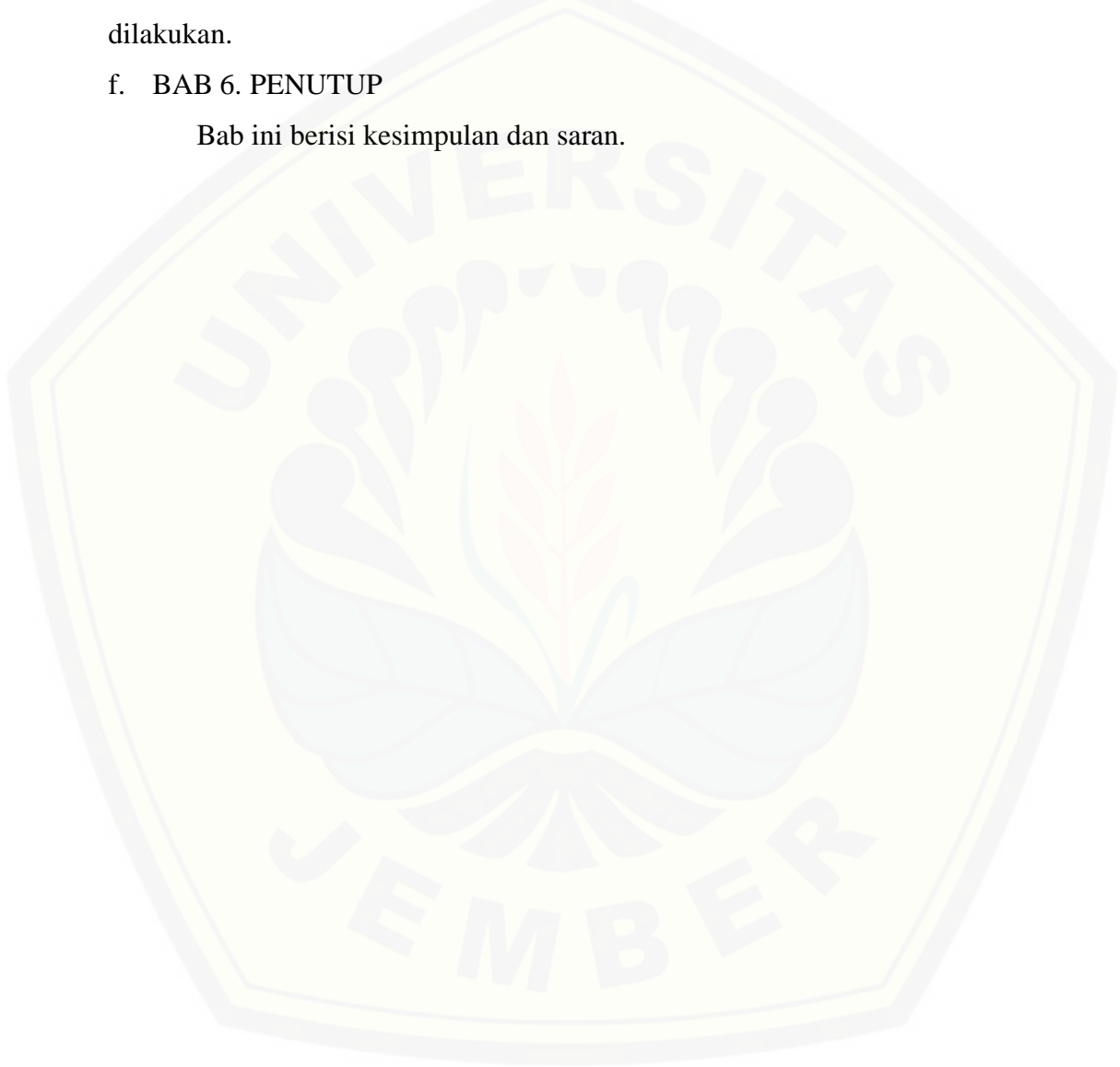
Bab ini berisi penerapan metodologi penelitian yang terdiri dari pengumpulan data, analisis dan perancangan sistem.

e. **BAB 5. HASIL DAN PEMBAHASAN**

Bab ini berisi tentang hasil dan pembahasan penelitian yang telah dilakukan.

f. **BAB 6. PENUTUP**

Bab ini berisi kesimpulan dan saran.



BAB 2. TINJAUAN PUSTAKA

Pada bagian ini menguraikan tentang teori-teori dan pustaka yang digunakan dalam penelitian. Teori yang dibahas adalah teori tentang penelitian terdahulu yang diambil dari buku literatur dan jurnal. Berikut merupakan teori-teori yang digunakan dalam penelitian :

2.1 Penelitian Terdahulu

Metode *Weighted Product (WP)* pernah diterapkan dalam sistem pendukung keputusan yang berjudul Sistem Pendukung Keputusan Penentuan Bonus Karyawan Menggunakan Metode *Weighted Product (WP)*. Penelitian ini menjelaskan tentang perbandingan karyawan terbaik yang pantas mendapatkan bonus sebagai imbalan dari kinerjanya menggunakan metode *Weighted Product*. Metode *Weighted Product* merupakan bagian dari konsep *Multi-Attribut Decision Making (MADM)* dimana diperlukan normalisasi pada perhitungannya. Kriteria yang digunakan adalah absen, prestasi, perilaku, pengalaman, disiplin, wawasan, sosialisasi tim adapun kriteria tersebut sudah di tentukan oleh perusahaan. Dari hasil penelitian menunjukkan pemanfaatan *Weighted Product* sebagai model sistem pendukung keputusan penentuan bonus karyawan di PT. Gunung Sari Medan dapat membantu manager dalam menentukan karyawan yang berhak di rekomendasikan mendapat bonus proses pembobotan multikriteria dan seleksi dengan lebih cepat, cermat dan lebih efektif (Jaya, 2013).

Metode *Weighted Product (WP)* juga pernah digunakan untuk menentukan prioritas produk unggulan yang berjudul Sistem Pendukung Keputusan Penentuan Prioritas Produk Unggulan Daerah Menggunakan Metode *Weighted Product (WP)* (Alfita, 2010). Penelitian ini digunakan untuk menentukan prioritas produk unggulan suatu daerah dengan menggunakan tujuh kriteria antara lain omset, tenaga kerja, target pasar, asal bahan baku, teknologi, spesifikasi kekhasan, dan kuantitas bahan baku. Hasil dari penelitian tersebut diperoleh aplikasi yang dapat membantu untuk pengambilan keputusan suatu instansi menentukan prioritas

produk unggulan sesuai dengan kebutuhan dan kemampuan daerah untuk meminimalisir kerugian dari segi biaya dan waktu.

Dua penelitian di atas dapat menjadi referensi pada penelitian ini, namun agar penggunaan metode *Weighted Product* (WP) bisa lebih tepat untuk kasus yang diteliti maka perlu memilih jenis kriteria yang digunakan agar proses penentuan prioritas produk unggulan lebih spesifik. Berbeda dengan penelitian yang dilakukan oleh Alfita, penelitian ini memiliki 5 kriteria yaitu omset, ketersediaan pasar, daya saing, spesifikasi kekhasan, dan ketersediaan bahan baku. Penelitian ini memperoleh hasil analisis dalam bentuk tabel yang disertai hasil perhitungan metode *Weighted Product* (WP) yang akan direkomendasikan sesuai dengan urutan hasil analisis terbesar, maka produk unggulan yang dijual dengan hasil analisis terbesar akan berada di urutan pertama pada tampilan awal.

2.2 Metode *Weighted Product* (WP)

Metode WP merupakan metode yang menggunakan perkalian guna menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan dahulu dengan bobot atribut yang bersangkutan, menentukan kriteria yang akan digunakan, kemudian memberi nilai dari setiap kriteria tersebut (Rani, 2014). Tabel 2.1 menjelaskan contoh dari beberapa kriteria yang digunakan berdasarkan nilai dari setiap kriteria.

Tabel 2. 1 Contoh Tabel Kriteria dan Penilaiannya

No	Kriteria	Nilai	Bobot
1	A_1	W_{o1}	W_1
2	A_2	W_{o2}	W_2
3	A_3	W_{o3}	W_3
4	A_4	W_{o4}	W_4

dilanjut

Lanjutan:

No	Kriteria	Nilai	Bobot
5	A ₅	W _{o5}	W ₅
6	A _n	W _{on}	W _n

(Sumber : Riza Alfita, 2010)

Pada tabel 1 dijelaskan bahwa kriteria yang digunakan diasumsikan dengan kriteria A₁ sampai A_n. Kemudian rating setiap kiteria diasumsikan dengan nilai W_{o1} sampai W_{on} dan bobot dari masing-masing kriteria diasumsikan dengan W₁ sampai W_n. Menurut Kusumadewi dkk (2006:75) bobot yang diberikan pada setiap nilai kriteria ditentukan berdasarkan tingkat kepentingan setiap kriteria. Tingkat kepentingan yang diberikan kepada setiap kriteria dapat dilihat pada Tabel 2.2.

Tabel 2. 2 Tabel Pembobotan Berdasarkan Tingkat Kepentingan

No	Tingkat Kepentingan	Bobot
1	Rendah	1
3	Cukup	3
4	Tinggi	5

(Sumber: Kusumadewi dkk, 2006)

Dalam proses perhitungan metode *Weighted Product* (WP) memiliki tiga tahapan, yaitu:

1. Menghitung nilai perbaikan bobot menggunakan rumus:

$$W_j = W_o / \sum W_o \dots\dots\dots (2.1)$$

Keterangan :

W_j = Nilai perbaikan bobot user

W_o = Nilai Bobot dari user

∑W_o = Jumlah nilai bobot dari user

(Sumber: Kusumadewi dkk, 2006)

2. Menghitung nilai preferensi untuk alternatif An dengan vektor S menggunakan rumus:

$$S_i = \prod_{j=1}^n X_{ij}^{w_j} \dots\dots\dots (2.2)$$

Keterangan :

S_i = Nilai prefensi alternatif An (Vektor S)

W_j = Nilai perbaikan bobot user (Positif jika benefit/ Negatif jika cost)

X_{ij} = Nilai rating kinerja atribut ke-i terhadap subkriteria ke-j

(Sumber: Kusumadewi dkk, 2006)

3. Menghitung nilai preferensi relatif dari setiap alternatif dengan menggunakan rumus:

$$V_i = \frac{\prod_{j=1}^n X_{ij}^{w_j}}{\prod_{j=1}^n (X_j^*)} \dots\dots\dots (2.3)$$

Keterangan :

V_i = Nilai preferensi relatif dari setiap alternatif (Vektor V)

W_j = Nilai perbaikan bobot *user* (Positif jika benefit/ Negatif jika *cost*)

X_{ij} = Nilai rating kinerja atribut ke-i terhadap subkriteria ke-j

X_j^* = Jumlah nilai preferensi alternatif (Vektor S)

(Sumber: Kusumadewi dkk, 2006)

Hasil dari nilai vektor V merupakan acuan yang digunakan untuk perankingan dari alternatif yang dihitung. Nilai alternatif terbaik yang dipilih merupakan nilai vektor V terbesar dari setiap alternatif. Tahapan metode *Weighted Product* (WP), tahap awal yaitu menentukan nilai *benefit* dan *cost* semua kriteria. Kemudian menghitung perbaikan bobot kriteria, dimana apabila dijumlahkan harus sama dengan 1. Tahap selanjutnya, yaitu memangkatkan semua nilai atribut alternatif dengan nilai perbaikan bobot kriteria yang saling berhubungan. Pada tahap tersebut menghasilkan nilai vektor S. Seluruh nilai vektor S semua alternatif dijumlahkan. Lalu membagi nilai vektor setiap alternatif dengan hasil penjumlahan vektor S semua lternatif. Setelah itu kan mendapatkan nilai vektor V setiap lternatif. Tahap terakhir yaitu meranking vektor V dari semua alternatif.

2.3 Produk Unggulan

Sistem Produk Unggulan Daerah (PUD) adalah produk unggulan yang memiliki ciri khas dan keunikan yang tidak dimiliki daerah lain serta berdaya saing handal dan dapat memberikan peluang atau kesempatan kerja kepada masyarakat lokal. Produk unggulan daerah juga berorientasi ramah lingkungan dan berorientasi pada pasar baik lokal maupun nasional dan regional. Pengembangan produk unggulan dan pemberdayaan sebagai potensi ekonomi daerah pada era otonomi adalah suatu pekerjaan yang tidak mudah dilaksanakan, hal tersebut disebabkan karena pengembangan PUD terkait erat dengan kemauan politik atau kebijakan dari Pemerintah Daerah. Peranan pemerintah daerah sangat diperlukan dan sangat penting dalam pengembangan dan pemberdayaan produk unggulan daerah sebagai salah satu tonggak dari pada ekonomi daerah. Oleh karena itu, produk unggulan daerah terkait beberapa *stakeholders* yang saling berperan sesuai dengan kewenangannya masing-masing. *Stakeholders* dimaksud adalah pemilik bahan baku dan pengolah/penghasil bahan baku, pengguna atau konsumen, fasilitator atau pemerintah dan lembaga sosial masyarakat. *Stakeholders* tersebut saling terkait dan menunjang satu sama lain sehingga peranan koordinasi dalam pencapaian tujuan menjadi unsur utama dalam pengembangan PUD. Koordinasi ini menjadi instrumen penting dalam pengembangan produk unggulan daerah. Produk unggulan merupakan suatu strategi pembangunan yang tidak mudah didikte oleh daerah atau negara lain. Produk unggulan daerah tidaklah harus berupa hasil industri yang berteknologi canggih atau dengan investasi tinggi tetapi produk unggulan bisa dengan produk lokal (Alfita, 2010).

Menurut Nusantoro (2011), produk Unggulan Daerah (PUD) merupakan suatu barang atau jasa yang dimiliki dan dikuasai oleh suatu daerah, yang mempunyai nilai ekonomis dan daya saing tinggi serta menyerap tenaga kerja dalam jumlah besar, yang diproduksi berdasarkan pertimbangan kelayakan teknis (bahan baku dan pasar), talenta masyarakat dan kelembagaan (penguasaan teknologi, kemampuan sumberdaya manusia, dukungan infrastruktur, dan kondisi sosial budaya setempat) yang berkembang di lokasi tertentu. Pengembangan

ekonomi lokal merupakan proses membangun dialog dan kemitraan aksi para pihak yang meliputi pemerintah daerah, para pengusaha, dan organisasi-organisasi masyarakat lokal. Pilar-pilar pokok strateginya adalah meningkatkan daya tarik, daya tahan, dan daya saing ekonomi lokal. Tujuan utamanya adalah untuk menciptakan pertumbuhan yang tinggi dan pembangunan berkelanjutan yang bermanfaat bagi semua pihak di daerah dalam rangka meningkatkan kesempatan kerja baru, peningkatan dan pengurangan kemiskinan secara signifikan.

Kriteria produk unggulan menurut Unkris Satya Wacana salatiga, adalah komoditi yang memenuhi persyaratan kecukupan sumberdaya lokal, keterkaitan komoditas, posisi bersaing dan potensi bersaing. Dari kriteria ini memunculkan pengelompokan komoditas sebagai berikut (Muhammad dkk, 2009):

1. Komoditas potensial, yaitu komoditas daerah yang memiliki potensi untuk berkembang karena keunggulan komparatif. Keunggulan komparatif terjadi misalnya karena kecukupan ketersediaan sumberdaya, seperti bahan baku lokal, keterampilan sumberdaya lokal, teknologi produksi lokal serta sarana dan prasarana lokal lainnya.
2. Komoditas andalan, yaitu komoditas potensial yang dipandang dapat mempersandingkan dengan produk sejenis di daerah lain, karena disamping memiliki keunggulan komparatif juga memiliki efisiensi usaha yang tinggi. Efisiensi usaha itu tercermin dari efisiensi produksi, produktivitas pekerja, profitabilitas dan lain-lain.
3. Komoditas unggulan, yaitu komoditas yang memiliki keunggulan kompetitif, karena telah memenangkan persaingan dengan produk sejenis di daerah lain. Keunggulan kompetitif demikian dapat terjadi karena efisiensi produksinya yang tinggi akibat posisi tawarnya yang tinggi baik terhadap pemasok, pembeli, serta daya saingnya yang tinggi terhadap pesaing, pendatang baru maupun barang substitusi.

2.4 Penggunaan *Weighted Product* (WP) dalam Menganalisis Produk Unggulan

Untuk menetapkan produk unggulan daerah ada kriteria tertentu, sistem penskala-an terhadap variabel kriteria unggulan. Sistem penskala-an tiap variabel ini didasarkan pada nilai interval masing-masing kelompok (sub sektor) dengan kisaran nilai dari 1 sampai 6. Sementara untuk data yang bukan berupa angka, penskala-an dilakukan dengan sistem strata (Alfita 2010). Masing-masing kriteria (variabel) memiliki bobot yang berbeda-beda disesuaikan dengan tingkat sumbangan kriteria terhadap produk unggulan. Sebagai contoh dalam penentuan prioritas produk unggulan antara produk A, produk B dan produk C dengan kriteria yang memiliki nilai dan bobot kriteria seperti pada tabel berikut:

Tabel 2. 3 Contoh Tabel Kriteria dan Bobot Kriteria

No	Nama Kriteria	Kriteria	Nilai	Bobot
1	Omset	A ₁	6	0,2222
2	Tenaga Kerja	A ₂	5	0,1851
3	Target Pasar	A ₃	4	0,1481
4	Asal Bahan Baku	A ₄	4	0,1481
5	Teknologi	A ₅	3	0,1111
6	Spesifikasi Kekhasan	A ₆	3	0,1111
7	Kuantitas Bahan Baku	A ₇	2	0,0740

(Sumber: Riza Alfita, 2010)

$$\begin{aligned} \sum w &= 0,2222 + 0,1851 + 0,1481 + 0,1481 + 0,1111 + 0,1111 + 0,0740 \\ &= 1 \end{aligned}$$

Karena total dari seluruh bobot kriteria sama dengan 1, maka tidak perlu dilakukan perbaikan bobot lagi. Sehingga dapat langsung dihitung nilai vektor S dan vektor V setelah masing-masing alternatif diberi nilai sesuai dengan masing-masing kriteria (Alfita, 2010) seperti pada tabel 2.4.

Tabel 2. 4 Contoh Tabel Alternatif Produk dan Nilai Kriteria

Alternatif Produk	Kriteria						
	A ₁	A ₂	A ₃	A ₄	A ₅	A ₆	A ₇
Produk A	5	3	4	3	1	5	1
Produk B	5	1	3	4	2	4	3
Produk C	4	5	4	1	4	1	5

Perhitungan Vektor S:

a. Perhitungan Vektor S Produk A

$$\begin{aligned}
 S_1 &= \prod_{j=1}^n X_{ij}^{w_j} \\
 &= (5^{0,2222}) \cdot (3^{-0,1851}) \cdot (4^{0,1481}) \cdot (3^{0,1481}) \cdot (1^{-0,1111}) \cdot (5^{0,1111}) \cdot (1^{0,0740}) \\
 &= 2,0153
 \end{aligned}$$

b. Perhitungan Vektor S Produk B

$$\begin{aligned}
 S_2 &= \prod_{j=1}^n X_{ij}^{w_j} \\
 &= (5^{0,2222}) \cdot (1^{-0,1851}) \cdot (3^{0,1481}) \cdot (4^{0,1481}) \cdot (2^{-0,1111}) \cdot (4^{0,1111}) \cdot (3^{0,0740}) \\
 &= 2,4197
 \end{aligned}$$

c. Perhitungan Vektor S Produk C

$$\begin{aligned}
 S_3 &= \prod_{j=1}^n X_{ij}^{w_j} \\
 &= (4^{0,2222}) \cdot (5^{-0,1851}) \cdot (4^{0,1481}) \cdot (1^{0,1481}) \cdot (4^{-0,1111}) \cdot (1^{0,1111}) \cdot (5^{0,0740}) \\
 &= 1,1970
 \end{aligned}$$

Perhitungan Vektor V:

a. Perhitungan Vektor V Produk A

$$V_A = \frac{\prod_{j=1}^n X_{ij}^{w_j}}{\prod_{j=1}^n (X_j^*)}$$

$$= \frac{2,0153}{2,0153+2,4197+1,1970}$$

$$= 0,3578$$

b. Perhitungan Vektor V Produk B

$$V_B = \frac{\prod_{j=1}^n X_{ij}^{w_j}}{\prod_{j=1}^n (X_j^*)}$$

$$= \frac{2,4197}{2,0153+2,4197+1,1970}$$

$$= 0,4296$$

c. Perhitungan Vektor V Produk C

$$V_C = \frac{\prod_{j=1}^n X_{ij}^{w_j}}{\prod_{j=1}^n (X_j^*)}$$

$$= \frac{1,1970}{2,0153+2,4197+1,1970}$$

$$= 0,21253$$

Berdasarkan perhitungan vektor diatas, didapatkan nilai perhitungan vektor S dan vektor V dimana produk yang mempunyai nilai tertinggi termasuk dalam Produk Unggulan Daerah (Produk B) dan yang lainnya disebut produk potensial.

2.5 Sistem Informasi Penjualan

Sistem informasi yaitu suatu sistem yang menyediakan informasi untuk manajemen dalam mengambil keputusan dan juga untuk menjalankan operasional perusahaan, di mana sistem tersebut merupakan kombinasi dari orang-orang, teknologi informasi dan prosedur-prosedur yang terorganisasi. Biasanya suatu perusahaan atau badan usaha menyediakan semacam informasi yang berguna bagi manajemen. Sebagai contoh: Perusahaan toko buku mempunyai sistem informasi yang menyediakan informasi penjualan buku-buku setiap harinya, serta stock buku-buku yang tersedia, dengan informasi tersebut, seorang manajer bisa membuat keputusan, stock buku apa yang harus segera mereka sediakan untuk toko buku mereka, manajer juga bisa tahu buku apa yang paling laris dibeli

konsumen, sehingga mereka bisa memutuskan buku tersebut jumlah stocknya lebih banyak dari buku lainnya (Dimas, 2015).

Sedangkan Abdillah (2004) berpendapat bahwa secara sederhana, Sistem Informasi merupakan kumpulan komponen yang saling berhubungan untuk mengolah input (data) menjadi output (informasi) sehingga dapat memenuhi kebutuhan pemakai.

Penjualan merupakan aktivitas utama perusahaan. Pendapatan perusahaan sangat ditentukan oleh besar kecilnya penjualan. Kegiatan penjualan itu sendiri berhubungan erat dengan kegiatan *marketing* atau pemasaran, dimana penjualan merupakan bagian dari pemasaran. Bagi perusahaan distributor, kegiatan penjualan menjadi tugas para *salesman* (Bunafit, 2005).

Menurut Suryana (2003), penjualan adalah menyajikan barang agar konsumen menjadi tertarik dan melakukan pembelian. Penjualan dapat dilakukan dengan cara langsung mendatangi konsumen, menunggu kedatangan konsumen dan melayani konsumen.

Menurut Al-Bahra (2005), sistem informasi penjualan adalah suatu sistem informasi yang mengorganisasikan serangkaian prosedur dan metode yang dirancang untuk menghasilkan, menganalisa, menyebarkan dan memperoleh informasi guna mendukung pengambilan keputusan mengenai penjualan.

Sistem informasi penjualan produk unggulan Kabupaten Lumajang merupakan sistem informasi yang digunakan untuk menjual produk-produk unggulan Kabupaten Lumajang. Sistem ini dapat diakses oleh pembeli dan admin (pedagang). Pembeli dapat memesan produk dan melihat rekomendasi tentang produk unggulannya. Sedangkan admin mengoperasikan atau mengolah semua data yang dibutuhkan dalam sistem ini.

Sebelum produk unggulan dijual, terlebih dahulu dianalisis dengan menggunakan algoritma *Weighted Product* dengan 5 kriteria sebagai acuan untuk menghitung analisisnya. Kriteria-kriteri tersebut antara lain omset, ketersediaan pasar, daya saing, spesifikasi kekhasan dan ketersediaan bahan baku yang diberi bobot dan nilai untuk masing-masing kriteria. Sehingga dapat dihitung untuk

mendapatkan produk paling unggul dari sejumlah produk dengan menghitung vektor sesuai dengan metode WP dan dirangkai.

Hasil analisis produk tersebut akan ditampilkan dalam sistem untuk membantu menentukan produk mana yang unggulan dan dijual secara online. Sehingga pembeli juga dapat mengetahui produk mana yang unggul dan dapat memilih produk yang akan dibeli. Pedagang juga dapat mengetahui produk mana yang unggul sehingga dapat meningkatkan produksi atau pemesanan produk unggul tersebut.



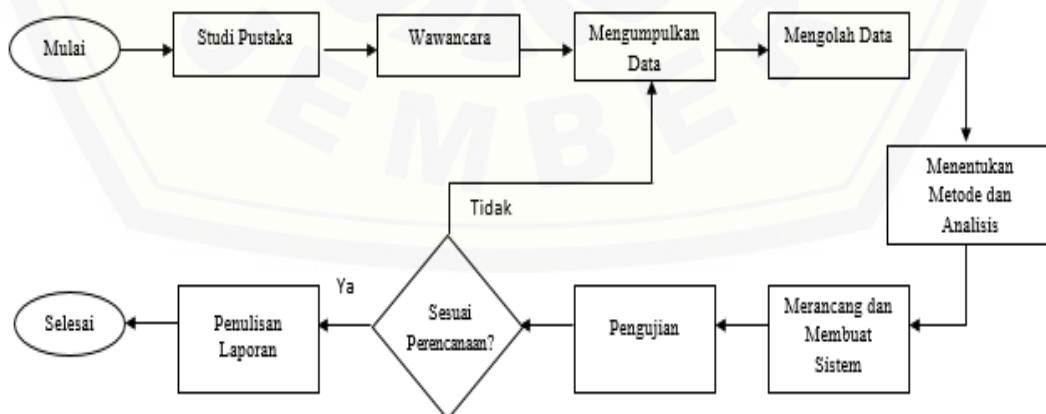
BAB 3. METODOLOGI PENELITIAN

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

3.1 Jenis Penelitian

Penelitian ini termasuk jenis penelitian pengembangan karena bertujuan untuk merancang dan membangun sistem penjualan dan promosi produk unggulan dengan fitur pemilihan produk unggulan. Sistem ini menjelaskan mengenai desain untuk membangun sebuah sistem. Pengembangan sistem dalam penelitian ini menggunakan model *waterfall*. Penjelasan mengenai *waterfall* dapat dilihat pada poin 3.3, sedangkan pada tahap pengembangan sistem menjelaskan mengenai tahapan desain untuk membangun sistem pada penelitian ini. Pembuatan sistem informasi penjualan dan promosi produk unggulan dengan fitur pemilihan produk unggulan ini mengikuti tahapan *software development life cycle (SDLC) waterfall*. Karena sistem ini tergolong sistem berskala kecil. Tahapan SDLC dengan metode *waterfall* meliputi tahapan analisis, desain, implementasi, pengujian, dan pemeliharaan.

3.2 Metode Penelitian



Gambar 3. 1 Diagram Alir Tahapan Penelitian

Gambar 3.1 merupakan gambaran dalam tahap penelitian yang dilakukan dengan tahap diantaranya pengumpulan data, tahap analisis, tahap perancangan dan implementasi sistem, tahap pengujian sistem, dan tahap penulisan laporan. Pada tahapan awal untuk memulai penelitian, peneliti melakukan studi pustaka yang terkait dan sesuai dengan masalah yang ada.

Tahap pengumpulan data dalam penelitian ini sebagai berikut:

1. Studi Literatur

Studi literatur dilakukan dengan mencari teori dari penelitian yang terdahulu, kemudian menyusun dasar-dasar teori yang sesuai dengan penelitian. Sumber yang digunakan dalam studi literatur ini yaitu jurnal, karya ilmiah, artikel, situs web yang terkait, *Weighted Product* dan *Decision Support System*.

2. Wawancara

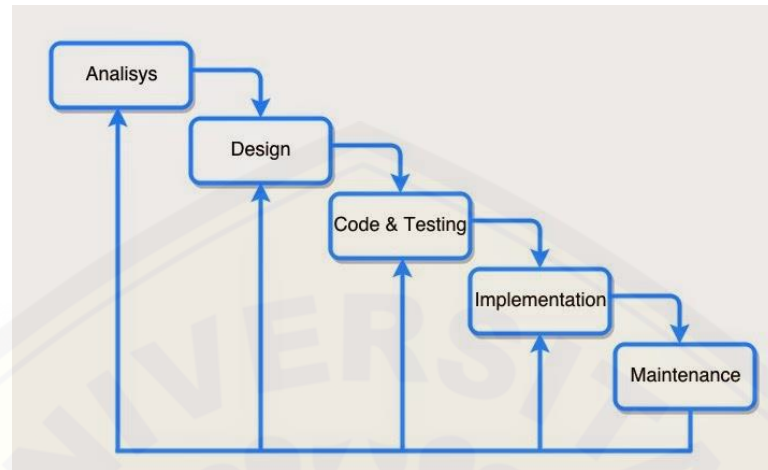
Salah satu cara untuk mengumpulkan data adalah melakukan wawancara dengan menanyakan langsung kepada narasumber. Pada penelitian ini penulis melakukan wawancara dengan beberapa pihak terkait untuk memperoleh data yang dibutuhkan.

Tahapan kedua setelah melakukan studi pustaka yaitu dengan melakukan wawancara kepada pihak-pihak terkait dan sekaligus untuk mendapatkan data-data yang diperlukan untuk membuat sebuah aplikasi. Setelah data diperoleh, tahapan selanjutnya yaitu menentukan metode dan menganalisis data yang telah didapatkan untuk merancang dan membuat sebuah aplikasi. Testing atau pengujian dilakukan setelah perancangan dan pembuatan aplikasi, jika aplikasi belum sesuai dengan rencana yang diharapkan maka kembali ke tahapan mengumpulkan data sedangkan jika hasil pengujian telah sesuai dengan apa yang direncanakan maka akan berlanjut ke tahap penulisan laporan.

3.3 Langkah Perancangan Sistem

Pembuatan sistem informasi pada penelitian ini mengadopsi metode *waterfall*. Menurut Pressman (2012), Model *waterfall* mengusulkan sebuah pendekatan perkembangan perangkat lunak yang sistematis dan sekuensial yang dimulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, pengujian

dan penerapan serta pemeliharaan. Alur perancangan model *waterfall* dapat di lihat pada gambar 3.2



Gambar 3. 2 Tahapan Model Waterfall

(Pressman, 2012)

Secara garis besar metode *waterfall* mempunyai langkah-langkah sebagai berikut : Analisis, Desain, Pengujian dan Penerapan serta Pemeliharaan.

3.3.1 Analisa Kebutuhan

Analisis kebutuhan merupakan langkah awal dari tahapan pengembangan perangkat lunak model *waterfall*. Proses yang dilakukan dalam tahap ini adalah mengumpulkan semua data yang dibutuhkan untuk membuat atau mengembangkan aplikasi. Data tersebut nantinya akan digunakan untuk menentukan kebutuhan fungsional dan nonfungsional dari aplikasi yang akan dibuat. Data-data yang dibutuhkan adalah data produk makanan yang nantinya digunakan dalam sistem informasi penjualan dan promosi produk unggulan dengan fitur pemilihan produk unggulan.

Dilakukan tahap analisis kebutuhan dengan menggunakan metode observasi dan wawancara kepada pihak terkait yang menjual produk makanan ringan di Lumajang. Hasil dari observasi tersebut adalah informasi mengenai pemberian nilai bobot pada masing-masing kriteria produk untuk menjual dan mempromosikan produk unggulan serta memilih produk yang termasuk produk

unggulan. Dengan menggunakan metode WP diharapkan bisa memberikan informasi lebih mengenai produk unggulan berdasarkan kriteria yang ada.

3.3.2 Desain

Desain Sistem Tahap selanjutnya setelah anilisa kebutuhan selesai adalah tahap desain sistem. Tahap ini akan mengimplementasikan data yang didapat kedalam sistem dengan gambaran-gambaran diagram. Pada desain sistem ini menggunakan bahasa *Unified Modeling Language* (UML) dan menggunakan konsep *Object-Oriented Programming* (OOP). Model desain UML yang digunakan sebagai berikut :

a. *Usecase Diagram*

Usecase diagram adalah gambaran fitur dari sistem yang dijalankan oleh aktor. Pada diagram ini dapat dilihat juga hak akses dari aktor.

b. *Usecase Skenario*

Usecase skenario adalah deskripsi tentang alur sistem dan user berdasarkan fitur yang berada didalam usecase diagram.

c. *Activity Diagram*

Activity diagram adalah gambaran tentang aktivitas sistem dan user dalam penggunaan aplikasi dari awal hingga akhir fitur. Menggambarkan implentasi algoritma dalam aplikasi.

d. *Sequence Diagram*

Sequance diagram adalah diagram yang menggambarkan interaksi antara objek satu dengan yang lain di dalam sistem yang dibangun pada urutan waktu. Diagram juga menggambarkan interaksi antara aktor, fitur, serta data yang berjalan. Sehingga, dengan squence diagram, aliran logika dalam sebuah sistem dapat dimodelkan secara visual dalam bentuk diagram.

e. *Class Diagram*

Class Diagram merupakan model statis yang menggambarkan struktur dan deskripsi *class* serta hubungannya antara *class*. *Class Diagram* terdiri dari nama kelas, atribut dan operasi atau *method*.

f. Entity Relation Diagram

Entity relation diagram (ERD) adalah diagram yang menggambarkan relasi objek-objek dasar data dalam sebuah basis data.

3.3.3 Implementasi

Pada tahap ini desain yang telah dibuat akan diimplementasikan kedalam sistem yang akan dibangun. Penulisan kode program dilakukan menggunakan bahasa pemrograman *Page Hyper Text Pre-Processor* (PHP), *Hyper Text Markup Language* (HTML). DBMS (*Database Management System*) yang digunakan adalah *MySQL* dengan tool XAMPP.

3.3.4 Pengujian

Tahap ini adalah tahap untuk menguji kode program yang telah dibuat. Tujuan pengujian adalah menilai apakah sistem yang dirancang telah sesuai dengan apa yang diharapkan, serta untuk mengevaluasi keunggulan dan kelemahan terhadap kualitas sistem. Teknik untuk pengujian sistem menggunakan

1. *White box testing* merupakan pengujian pada modul pengkodean program yang dilakukan oleh peneliti. Pengujian ini dilakukan dengan menghitung *independent path* dengan menggunakan *cyclomatic complexity*. Untuk menghitung *cyclomatic complexity* digunakan rumus:

$$V(G) = E - N + 2 \quad \dots\dots\dots (3.1)$$

Keterangan : E = jumlah *edge* grafik alir

N = jumlah *node* grafik alir

2. *Black box Testing* adalah metode pengujian perangkat lunak yang memeriksa fungsionalitas dari aplikasi yang berkaitan dengan struktur internal atau kerja (Agissa,2015).

3.3.5 Maintenance

Tahap ini dilakukan untuk mengoreksi beberapa kesalahan yang tidak ditemukan pada tahap sebelumnya. Perangkat lunak yang sudah disampaikan pada pelanggan sangat mungkin akan mengalami perubahan. Perubahan tersebut dapat terjadi seiring pengerjaan program dengan penyesuaian kinerja sistem. Penyempurnaan *bug* (error) akan membantu penggunaan perangkat lunak.

3.4 Gambaran Sistem

Sistem informasi penjualan produk unggulan merupakan suatu sistem yang berbasis web yang dapat digunakan untuk membantu penjualan produk yang ada di Kabupaten Lumajang yang disertai dengan fitur untuk menentukan produk unggulan. Dalam sistem ini terdapat dua aktor yaitu admin (pedagang) dan pembeli. Admin berwenang untuk mengolah data produk, data kriteria, melihat data pelanggan, mengolah data penjualan, mengolah data ongkir dan melihat hasil perhitungan WP. Sedangkan pembeli berwenang untuk melihat produk-produk yang ditawarkan, melakukan pemesanan, mengolah keranjang pembelian dan melihat history pembelian serta dapat melakukan registrasi bagi calon pembeli.

Data yang diinputkan admin yaitu data produk yang meliputi nama produk, berat produk, stok, harga, catatan, dan foto produk. Selain itu, admin juga menginputkan nilai untuk masing-masing kriteria yang telah ditentukan dengan bobot yang juga telah ditentukan. Kemudian produk tersebut dianalisis menggunakan algoritma *Weighted Product* sehingga didapat produk unggulan dari seluruh produk yang telah diinputkan. Produk yang telah dianalisis selanjutnya akan direkomendasikan sebagai produk unggulan ataupun sebagai produk yang potensial di Kabupaten Lumajang yang dapat dilihat oleh pembeli dan admin itu sendiri. Bila ada pemesanan, pedagang akan dapat melihat informasi dalam data penjualan sehingga produk yang dipesan dapat langsung dikirimkan dan didapatkan output dari produk unggulan dan transaksi penjualannya.

Gambar 3.3 menunjukkan gambaran umum sistem yang memiliki dua aktor yaitu admin dan pembeli. Bagian admin memiliki fitur utama yaitu menu login, menu data kriteria yang berfungsi untuk mengolah data kriteria dan subkriteria, menu perhitungan WP yang menampilkan hasil analisa perhitungan metode WP, menu data produk yang berfungsi untuk menambah, mengubah, menghapus dan mencari data produk, menu data pelanggan yang menampilkan daftar data pembeli yang sudah terdaftar, menu data ongkir yang menampilkan data ongkos kirim dan juga berfungsi untuk mengubah data ongkos kirim, dan menu data penjualan yang menampilkan daftar penjualan yang dapat diolah seperti mengubah data tersebut yang awalnya dengan status order menjadi status

pembayaran sudah diterima, ditolak atau dibatalkan. Pembeli memiliki fitur utama yaitu menu login, menu registrasi yang berfungsi untuk mendaftarkan akun yang akan digunakan untuk masuk ke sistem, menu profil yang berfungsi menampilkan detail data diri atau profil pembeli yang dapat diubah datanya, menu keranjang yang berfungsi menampilkan data produk yang dipesan dan menginputkan bukti pembayaran sesuai dengan harga yang harus dibayar, menu history pembelian yang berfungsi menampilkan daftar produk yang dibeli dengan menampilkan status produk tersebut masih diorder, sudah diterima pembayaran, ditolak atau dibatalkan dan dapat menginputkan review dari produk.



Gambar 3. 3 Gambaran Umum Sistem

3.5 Pengujian Algoritma *Weighting Product*

Untuk menggunakan dan menerapkan algoritma *Weighting Product* pada suatu sistem diperlukan pengujian terlebih dahulu. Langkah pertama dalam pengujian algoritma WP yaitu dengan menentukan kriteria dan bobot masing-masing kriteria yang digunakan seperti pada tabel 3.1. Bobot kriteria tersebut ditentukan berdasarkan tingkat kepentingan pada masing-masing kriteria yang terdapat pada tabel 3.2.

Tabel 3.1 Kriteria dan Bobot Masing-Masing Kriteria

No	Nama Kriteria	Bobot
1	Omset	5
2	Ketersediaan Pasar	3
3	Daya Saing	5
4	Spesifikasi Kekhasan	3
5	Ketersediaan Bahan Baku	5

Tabel 3.2 Tabel Pembobotan Berdasarkan Tingkat Kepentingan

No	Tingkat Kepentingan	Bobot
1	Rendah	1
3	Cukup	3
4	Tinggi	5

Langkah kedua yaitu menjumlahkan semua nilai bobot kriteria, apabila jumlah bobot lebih dari 1 maka diperlukan normalisasi atau perbaikan agar jumlahnya sama dengan 1. Perhitungan normalisasi menggunakan rumus:

$$W_j = W_o / \sum W_o \dots\dots\dots (2.1)$$

Keterangan :

W_j = Nilai perbaikan atau normalisasi bobot

W_o = Nilai bobot awal

$\sum W_o$ = Jumlah nilai bobot awal

$$\begin{aligned} \sum W_o &= W_{o1} + W_{o2} + W_{o3} + W_{o4} + W_o \\ &= 5 + 3 + 5 + 3 + 5 \\ &= 21 \end{aligned}$$

Perbaikan nilai bobot kriteria 1:

$$W_1 = 5 / 21 = 0,2381$$

Perbaikan nilai bobot kriteria 2:

$$W_2 = 3 / 21 = 0,1429$$

Perbaikan nilai bobot kriteria 3:

$$W_3 = 5 / 21 = 0,2381$$

Perbaikan nilai bobot kriteria 4:

$$W_4 = 3 / 21 = 0,1429$$

Perbaikan nilai bobot kriteria 5:

$$W_5 = 5 / 21 = 0,2381$$

Dari perhitungan diatas diperoleh hasil normalisasi atau perbaikan bobot kriteria seperti tabel 3.3.

Tabel 3.3 Tabel Kriteria dan Bobot Awal serta Perbaikan Bobot

No	Nama Kriteria	Bobot	Perbaikan Bobot
1	Omset	5	0.2381
2	Ketersediaan Pasar	3	0.1429
3	Daya Saing	5	0.2381
4	Spesifikasi Kekhasan	3	0.1429
5	Ketersediaan Bahan Baku	5	0.2381

Langkah ketiga yaitu menentukan subkriteria dan nilai dari masing-masing subkriteria karena subkriteria bukan berupa angka. Subkriteria yang digunakan harus sesuai dengan kriteria dan nilai dari subkriteria juga berdasarkan tingkat kepentingan pembobotan pada tabel 3.2. Tabel subkriteria dan nilai subkriteria dapat dilihat pada tabel 3.4.

Tabel 3.4 Tabel Subkriteria dan Nilai Subkriteria

Omset		Ketersediaan Pasar		Daya Saing	
Range	Nilai	Range	Nilai	Range	Nilai
< 5.000.000/bulan	1	produk dijual di < 20 toko	1	Rendah	1
5.000.000 - 10.000.000/bulan	3	produk dijual di 21-40 toko	3	Cukup	3
> 10.000.000/bulan	5	produk dijual di > 40 toko	5	Tinggi	5
Spesifikasi Kekhasan		Ketersediaan Bahan Baku			
Range	Nilai	Range	Nilai		
produk yang sama ada di >10 kabupaten	1	<6 bulan/tahun	1		
produk yang sama ada di 5 - 10 kabupaten	3	6 - 10 bulan/tahun	3		
produk yang sama ada di <5 kabupaten	5	>10 bulan/tahun	5		

Langkah keempat yaitu menentukan data produk yang akan dihitung dan memberi nilai pada setiap kriteria. Setelah itu dihitung nilai vektor S, yaitu dengan mengalikan semua nilai alternatif yang terlebih dahulu dipangkatkan dengan bobot masing-masing kriteria. Hasil dari pemberian nilai dan perhitungan vektor S pada masing-masing produk terdapat pada tabel 3.5.

Tabel 3.5 Tabel Produk dan Nilai Kriteria

No	Produk	A1	A2	A3	A4	A5	Vektor S
1	Sale Oven Pisang Mas Kirana	3	3	3	3	5	3,389
2	Keripik Jahe Vilanda	3	3	3	1	3	2,564
3	Keripik Bonggol Pisang	5	3	3	5	5	4,116
4	Sale Pisang Mas Kirana	5	3	3	3	5	3,827
5	Keripik Ketela Ungu	3	3	3	1	5	2,896
6	Rengginang Ebi	1	3	5	3	5	2,946
7	Marning Jagung	3	3	5	3	5	3,827
8	Keripik Ceker	3	1	5	3	5	3,271
9	Carangmas Pisang	3	3	5	5	5	4,116
10	Keripik Cempolong	3	3	5	5	5	4,116
11	Keripik Pisang Asin	5	3	5	5	5	4,649
12	Keripik Pisang Balado	5	3	5	5	5	4,649
13	Keripik Pisang Coklat	5	3	5	5	5	4,649
14	Original Jamodin	3	3	3	3	5	3,389
15	Kacang Otok	3	3	3	3	5	3,389
16	Keripik Usus	3	3	3	1	5	2,896
17	Madumongso Pisang	3	3	5	5	5	4,116
18	Keripik Salak Pondoh	3	3	5	5	5	4,116
19	Keripik Pisang Agung Oven	3	3	5	5	5	4,116
20	Stik Pisang	3	3	5	5	5	4,116
21	Tape Pisang	5	3	5	5	5	4,649
22	Sale Pisang Mas Kirana Goreng Tepung	3	3	5	3	5	3,827
23	Keripik Pisang Buah	3	3	5	3	5	3,827

Keterangan:

A1 = Omset;

A3 = Daya Saing;

A5 = Ketersediaan

A2 = Ketersediaan Pasar;

A4 = Spesifikasi Kekhasan;

Bahan Baku.

Setelah diperoleh hasil perhitungan nilai vektor S, langkah kelima yaitu menghitung nilai vektor V yaitu dengan membagikan nilai vektor S produk dengan jumlah seluruh vektor S. Hasil perhitungan vektor V terdapat pada tabel 3.6.

Tabel 3.6 Tabel Hasil Perhitungan Vektor V

No	Produk	Vektor S	Vektor V
1	Sale Oven Pisang Mas Kirana	3,389	0,039
2	Keripik Jahe Vilanda	2,564	0,029
3	Keripik Bonggol Pisang	4,116	0,047
4	Sale Pisang Mas Kirana	3,827	0,044
5	Keripik Ketela Ungu	2,896	0,033
6	Rengginang Ebi	2,946	0,033
7	Marning Jagung	3,827	0,044
8	Keripik Ceker	3,271	0,037
9	Carangmas Pisang	4,116	0,047
10	Keripik Cempolong	4,116	0,047
11	Keripik Pisang Asin	4,649	0,053
12	Keripik Pisang Balado	4,649	0,053
13	Keripik Pisang Coklat	4,649	0,053
14	Original Jamodin	3,389	0,039
15	Kacang Otok	3,389	0,039
16	Keripik Usus	2,896	0,033
17	Madumongso Pisang	4,116	0,047
18	Keripik Salak Pondoh	4,116	0,047
19	Keripik Pisang Agung Oven	4,116	0,047
20	Stik Pisang	4,116	0,047
21	Tape Pisang	4,649	0,053
22	Sale Pisang Mas Kirana Goreng Tepung	3,827	0,044
23	Keripik Pisang Buah	3,827	0,044

Ketika nilai vektor V diperoleh, maka nilai tersebut yang menjadi acuan untuk menentukan apakah produk termasuk produk unggulan atau bukan. Untuk mengetahui produk yang unggulan, menggunakan langkah terakhir yaitu dengan mengurutkan atau meranking hasil dari nilai vektor V dari yang terbesar ke yang terkecil.



BAB 4. ANALISIS DAN PERANCANGAN SISTEM

Bab ini akan membahas tentang analisis dan perancangan untuk membuat Sistem Pendukung Keputusan Menggunakan Algoritma Weighted Product (WP). Tahapan - tahapan perancangan dilaksanakan berdasarkan metode *waterfall*.

4.1 *Statement Of Purpose*

Sistem informasi penjualan dan promosi produk unggulan dengan fitur pemilihan produk unggulan, merupakan sistem yang digunakan untuk membantu masyarakat khususnya bagi para pedagang dan pembeli yang ingin memasarkan dan membeli produk unggulan Kabupaten Lumajang. Tujuan dikembangkan sistem ini yaitu untuk memudahkan pedagang dan pembeli dalam memasarkan dan membeli tanpa harus datang ke Kabupaten Lumajang. Sistem ini dapat memberikan informasi produk olahan makanan ringan yang menjadi unggulan di Kabupaten Lumajang.

4.2 Analisa Kebutuhan

Kebutuhan sistem adalah kemampuan yang harus dimiliki oleh sistem untuk memenuhi apa yang diinginkan oleh pengguna. Untuk mempermudah menganalisis sebuah sistem dibutuhkan dua jenis kebutuhan. Kebutuhan fungsional dan kebutuhan nonfungsional. Kebutuhan fungsional adalah kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Sedangkan kebutuhan nonfungsional adalah kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem.

4.2.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan jenis kebutuhan yang berisikan proses apa saja yang diberikan oleh perangkat lunak yang akan dibangun. Sistem informasi penjualan produk unggulan Kabupaten Lumajang yang dirancang ini memiliki kebutuhan fungsional yang dapat dilihat pada Tabel 4.1.

Tabel 4. 1 Kebutuhan fungsional

SRS_ID	Identifikasi
SRS_01	Sistem dapat menampilkan data pelanggan
SRS_02	Sistem dapat mengolah data produk
SRS_03	Sistem dapat mengolah data kriteria dan subkriteria
SRS_04	Sistem dapat menampilkan hasil analisa produk unggulan
SRS_05	Sistem ini dapat mengolah data penjualan
SRS_06	Sistem ini dapat mengolah data ongkos kirim
SRS_07	Sistem ini dapat mengolah profil pengguna
SRS_08	Sistem ini dapat mengolah data keranjang
SRS_09	Sistem ini dapat mengolah data history pembelian

4.2.2 Kebutuhan non fungsional

Kebutuhan non-fungsional merupakan fungsi layanan pada sistem yang tidak secara langsung terkait pada fungsi sistem. Sistem penjualan produk unggulan Kabupaten Lumajang memiliki kebutuhan non fungsional yang dapat dilihat pada Tabel 4.2.

Tabel 4. 2 Kebutuhan nonfungsional

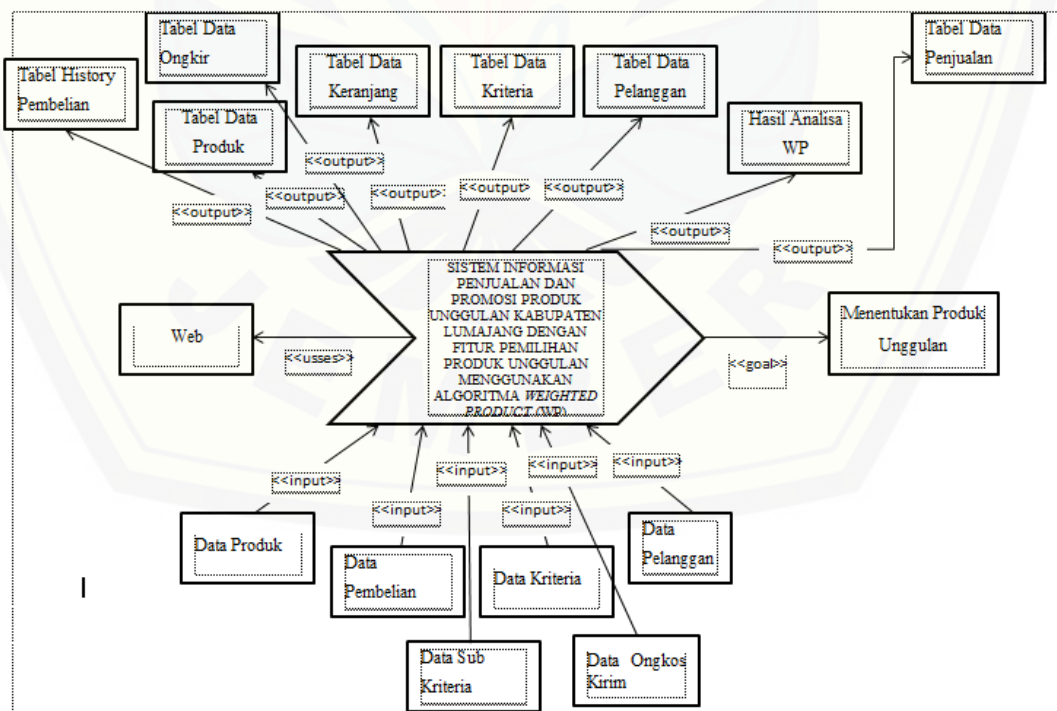
SRS_ID	Identifikasi
SRS_10	Sistem dapat diakses secara bersamaan oleh user yang berbeda
SRS_11	Sistem dapat berjalan pada platform atau sistem operasi apa saja yang mendukung aplikasi berbasis web
SRS_12	Sistem menggunakan enkripsi password pengguna untuk keamanan saat autentifikasi login
SRS_13	Tampilan sistem mudah di mengerti pengguna agar pengguna tidak kesulitan dalam mengoperasikannya (<i>user friendly</i>)

4.3 Desain Sistem

Desain sistem adalah tahapan untuk memodelkan sistem informasi penjualan produk unggulan Kabupaten Lumajang. Dokumen desain sistem yang akan dibuat meliputi *Bussines Process*, *Usecase Diagram*, *Usecase Skenario*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram* dan *Entity Relationship Diagrams (ERD)*.

4.3.1 Business Process

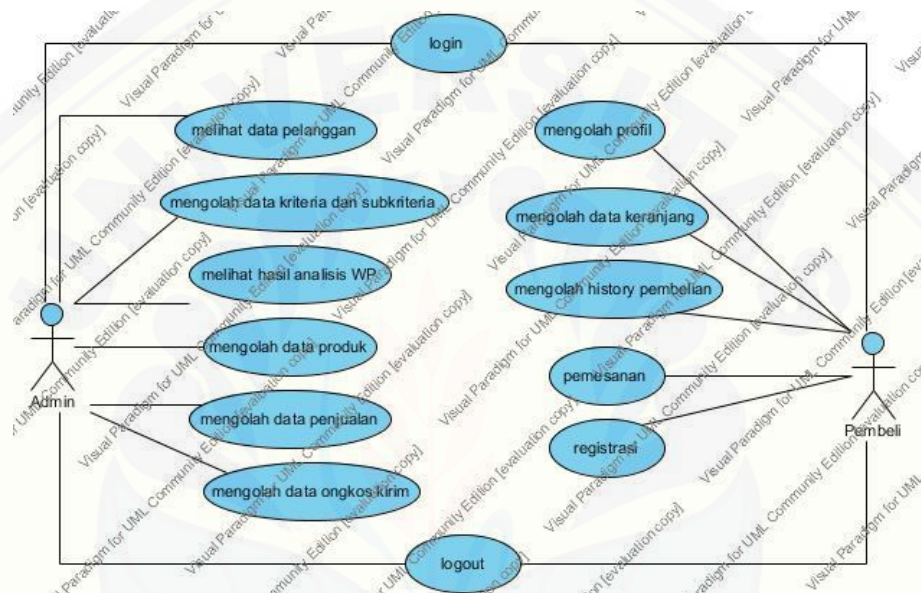
Business Process merupakan sekumpulan proses yang dilakukan untuk mencapai hasil yang diinginkan. *Business Process* dari sistem informasi penjualan produk unggulan Kabupaten Lumajang dapat dilihat pada gambar 4.1. Terdapat beberapa komponen untuk mencapai tujuan tersebut yaitu, data kriteria, data sub kriteria, data ongkos kirim, data pelanggan, data produk yang menjadi masukan (*input*) dan keluaran (*output*) yang dihasilkan tabel data history pembelian, tabel data ongkir, tabel data keranjang, tabel data produk, tabel data pelanggan, tabel data kriteria, tabel data penjualan, hasil analisa WP beserta tujuan (*goal*) adalah menentukan produk unggulan.



Gambar 4. 1 Gambaran Komponen Sistem

4.3.2 Use Case Diagram

Use Case Diagram adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. Usecase Diagram berfungsi untuk menggambarkan fitur apa saja yang akan dijalankan pada software yang akan dibuat. Usecase Diagram sistem informasi penjualan dan promosi produk unggulan Kabupaten Lumajang dapat dilihat pada gambar 4.2



Gambar 4. 2 Usecase diagram system

4.3.3 Definisi Aktor

Tabel 4. 3 Definisi Aktor

No.	Aktor	Deskripsi
1	Admin	Aktor yang memiliki hak akses penuh terhadap sistem. Aktor ini bisa menggunakan semua fitur yang ada pada sistem pengelolaan data. Pengelolaan tersebut dapat berupa tambah, edit, dan hapus data. Admin juga dapat melihat data pelanggan dan analisis WP untuk mendapatkan produk unggulan

dilanjutkan

Lanjutan:

No.	Aktor	Deskripsi
2	Pembeli	Aktor yang memiliki akses terbatas pada sistem. Aktor ini dapat mengelola data profil, melihat histori pembelian, melakukan pemesanan serta keranjang.

Definisi aktor pada tabel 4.3 diatas menjelaskan tentang aktor yang berinteraksi dengan sistem dan menjelaskan hak akses fitur yang dapat digunakan oleh aktor tersebut.

4.3.4 Definisi *Usecase*

Definisi *usecase* merupakan penjelasan dari setiap *usecase diagram* yang merupakan fitur-fitur dari sistem. Penjelasan definisi *usecase* dari sistem informasi penjualan produk unggulan dapat dilihat pada tabel 4.4.

Tabel 4. 4 Definisi *usecase*

No.	Use Case	Deskripsi
1	Login	<i>Usecase</i> yang menggambarkan proses login
2	Melihat Data Pelanggan	<i>Usecase</i> yang menampilkan data pelanggan
3	Mengolah Data Kriteria dan Subkriteria	<i>Usecase</i> yang menggambarkan proses pengelolaan data kriteria dan subkriteria
4	Melihat Hasil Analisis WP	<i>Usecase</i> yang menampilkan data hasil analisis WP
5	Mengolah Data Produk	<i>Usecase</i> yang menggambarkan proses pengelolaan data produk
6	Mengolah Data Penjualan	<i>Usecase</i> yang menggambarkan proses pengolahan data penjualan
7	Mengolah Data Ongkos Kirim	<i>Usecase</i> yang menggambarkan proses mengolah data ongkos kirim

dilanjutkan

Lanjutan:

No.	Use Case	Deskripsi
8	Mengolah Profil	<i>Usecase</i> yang menggambarkan proses pengelolaan data profil
9	Mengolah Data Keranjang	<i>Usecase</i> yang menggambarkan proses pengelolaan data keranjang
10	Mengolah History Pembelian	<i>Usecase</i> yang menggambarkan proses pengelolaan history pembelian
11	Pemesanan	<i>Usecase</i> yang menggambarkan proses pemesanan
12	Registrasi	<i>Usecase</i> yang menggambarkan proses registrasi
13	Loguot	<i>Usacase</i> yang menggambarkan proses logout

4.3.5 *Usecase Skenario*

Use case skenario adalah dokumentasi terhadap kebutuhan fungsional sistem. *Usecase skenario* sistem informasi penjualan produk unggulan adalah sebagai berikut:

Tabel 4. 5 Usecase Skenario

ID	05
Nama	Mengolah data produk
Aktor	Admin
Prekondisi	Menu data produk sudah ditampilkan
Post kondisi	Data produk sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu produk	

dilanjutkan

Lanjutan:

Skenario Normal	
	2. Menampilkan tabel data produk
3. Klik tambah produk	
	4. Menampilkan form tambah data produk
5. Mengisi form tambah data produk	
6. Klik simpan	
	7. Menyimpan data produk didatabase
	8. Menampilkan tabel data produk yang berhasil di tambah
Ketika klik tombol home	
9. Klik home	
	10. Menampilkan tabel data produk
Ketika ingin mengubah data produk	
11. Klik ubah	
	12. Menampilkan form ubah data produk
13. Mengubah data produk	
14. Klik simpan	
	15. Menyimpan perubahan data produk didatabase
	16. Menampilkan table data produk yang sudah berhasil di ubah
Ketika klik tombol home	
17. Klik home	
	18. Menampilkan tabel data produk

dilanjutkan

Lanjutan:

Skenario Normal	
Ketika ingin menghapus data produk	
19. Klik hapus	
	20. Menampilkan message dialog konfirmasi yakin ingin menghapus produk?
21. Klik Ok	
	22. Menghapus data di database
	23. Menampilkan data produk yang sudah terhapus
24. Klik Cancel	
	25. Menampilkan tabel data produk
Ketika mencari data produk	
26. Ketik data produk yang dicari	
27. Klik cari	
	28. Mengecek data produk didatabase
	29. Menampilkan data produk yang dicari
Skenario Alternatif	
Ketika data tidak lengkap	
5a. Mengisi form tambah data produk	
6a. Klik simpan	
	8a. Menampilkan peringatan harus mengisi data yang kosong.

4.3.5.1 Skenario Login

Login menjelaskan bahwa admin dan pembeli harus login terlebih dahulu kedalam sistem dengan menginputkan username dan password kemudian sistem akan membedakan halaman utama yang akan ditampilkan karena halaman utama admin dan halaman utama pembeli berbeda. Seperti pada lampiran A.1, terjadi skenario login untuk admin dan pembeli karena admin dan pembeli diwajibkan melakukan login agar dapat masuk dan mengakses sistem.

4.3.5.2 Skenario Melihat Data Pelanggan

Skenario ini memnjelaskan bahwa data pelanggan hanya dapat dilihat oleh admin tanpa bisa mengubah data tersebut. Admin dapat melakukan pencarian data pribadi dari pembeli dan melihat detail dari data pelanggan. Penjelasan digambarkan melalui skenario diagram pada lampiran A.2.

4.3.5.3 Skenario Mengolah Data Kriteria dan Subkriteria

Mengolah data kriteria dan subkriteria pada skenario ini menjelaskan bahwa yang dapat mengakses dan mengolah data (tambah, edit dan delete) kriteria dan subkriteria adalah admin. Namun data yang dapat diolah adalah data subkriteria karena data kriteria hanya dapat diinputkan melalui database. Penjelasan skenario ini digambarkan melalui skenario diagram pada lampiran A.3.

4.3.5.4 Skenario Melihat Hasil Analisis WP

Skenario ini menjelaskan bahwa hasil analisis metode WP (*Weighted Product*) ditampilkan dalam bentuk tabel dan hanya dapat dilihat oleh admin. Penjelasan ini digambarkan melalui skenario diagram seperti pada lampiran A.4.

4.3.5.5 Skenario Mengolah Data Produk

Mengolah data produk pada skenario ini menjelaskan bahwa user (admin) dapat melihat dan mengolah data produk dengan menambah data produk baru, mengubah data produk yang ada dan menghapus data produk tersebut. User juga dapat mencari data produk yang diinginkan. Penjelasan tersebut digambarkan melalui skenario diagram mengolah data produk pada lampiran A.5.

4.3.5.6 Skenario Mengolah Data Penjualan

Mengolah data penjualan pada skenario ini menjelaskan bahwa user yang dapat mengolah data penjualan adalah admin. User dapat melihat dan

menginputkan persetujuan pemesanan produk dan menentukan apakah pesanan pembeli akan disetujui atau ditolak. User juga dapat mencari data penjualan yang diinginkan. Penjelasan tersebut digambarkan melalui skenario diagram mengolah data penjualan pada lampiran A.6.

4.3.5.7 Skenario Mengolah Data Ongkos Kirim

Mengolah data ongkos kirim menjelaskan bahwa user yang dapat mengakses adalah admin. User dapat melihat dan mengolah data ongkos kirim dengan mengubah jumlah ongkos kirim yang sudah ada. Penjelasan tersebut digambarkan melalui skenario diagram mengolah data ongkos kirim pada lampiran A.7.

4.3.5.8 Skenario Mengolah Profil

Mengolah profil menjelaskan bahwa user yang dapat mengakses adalah pembeli. Disini user dapat mengubah data profilnya sesuai dengan keinginan dan alamat yang digunakan sesuai dengan alamat tujuan untuk pengiriman produk setelah memesan produk dan untuk menentukan ongkos kirim yang dibutuhkan. Penjelasan tersebut digambarkan melalui skenario diagram mengolah profil pada lampiran A.8.

4.3.5.9 Skenario Mengolah Keranjang

Mengolah keranjang menjelaskan bahwa user yang dapat mengakses keranjang (data produk yang dipesan) adalah user. User dapat mengetahui jumlah yang harus dibayar untuk pemesanan produk dan mengolah data keranjang dengan menginputkan bukti pembayaran produk yang dipesan. Penjelasan tersebut digambarkan melalui skenario diagram mengolah keranjang pada lampiran A.9.

4.3.5.10 Skenario Mengolah History Pembelian

History pembelian merupakan data produk yang sudah dibeli maupun masih dipesan. History pembelian dapat diolah oleh user yaitu pembeli dengan melihat detail pada pembelian dan menambah review bila diperlukan. User juga dapat melakukan pencarian produk yang telah dibeli. Penjelasan tersebut digambarkan melalui skenario diagram mengolah history pembelian pada lampiran A.10.

4.3.5.11 Skenario Registrasi

Registrasi merupakan pendaftaran akun untuk dapat mengakses atau masuk pada sistem yang dapat dilakukan oleh pembeli. User (pembeli) melakukan registrasi dengan menginputkan data pribadi serta username dan password yang digunakan untuk login. Penjelasan tersebut digambarkan melalui skenario diagram registrasi pada lampiran A.11.

4.3.5.12 Skenario Pemesanan

Skenario pemesanan menjelaskan alur pemesanan produk yang diinginkan yang dapat dilakukan oleh user (pembeli). User melakukan pemesanan dengan memilih produk yang diinginkan dan menginputkan jumlah produk yang dipesan. Penjelasan tersebut digambarkan melalui skenario diagram pemesanan pada lampiran A.12.

4.3.5.13 Skenario Logout

Skenario logout menjelaskan proses logout admin dan pembeli, jika sudah selesai menggunakan sistem dianjurkan logout. Sistem menyediakan menu logout agar admin dan pembeli dapat keluar dari sistem. Penjelasan tersebut digambarkan melalui skenario diagram logout pada lampiran A.13.

4.3.6 Activity Diagram

Activity diagram menggambarkan aliran aktivitas dalam sistem informasi penjualan dan promosi produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan yang akan dibangun. *Activity diagram* pada sistem informasi penjualan dan promosi produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan terdapat dalam penjelasan sebagai berikut:

4.3.6.1 Activity Diagram Login

Activity Diagram Login seperti pada lampiran B.1 menjelaskan bahwa terjadi aktivitas yang dilakukan user dan sistem, pada proses ini aktor yang dapat melakukan login adalah admin dan pembeli. User login dengan menginputkan username dan password. Apabila username dan password benar maka akan

muncul notifikasi kalau login sukses, sedangkan bila username dan password belum terdaftar atau salah maka akan muncul notifikasi login failed.

4.3.6.2 *Activity Diagram* Melihat Data Pelanggan

Activity Diagram Melihat Data Pelanggan seperti pada lampiran B.2 menjelaskan bahwa terjadi aktivitas admin dan sistem. Dalam proses aktivitas ini admin melihat data pelanggan atau pembeli dan dapat mencari data pelanggan yang diinginkan. Cara mencari data pelanggan yaitu dengan menginputkan nama atau data lain dari pelanggan dan mengklik tombol cari sehingga ditampilkan data pelanggan atau pembeli yang dicari.

4.3.6.3 *Activity Diagram* Mengolah Data Kriteria dan Subkriteria

Activity Diagram Mengolah Data Kriteria dan Subkriteria seperti pada lampiran B.3 menjelaskan bahwa terjadi aktivitas admin melihat data kriteria dan subkriteria, dalam *activity* juga menjelaskan aktivitas admin menginput data (tambah, edit dan delete). Data yang dapat diinput yaitu data subkriteria saja karena data kriteria sudah ditentukan oleh sistem. Klik tombol tambah untuk menambahkan data subkriteria baru, tombol edit untuk mengubah data subkriteria yang sudah ada, dan tombol delete untuk menghapus data subkriteria yang tidak dibutuhkan.

4.3.6.4 *Activity Diagram* Melihat Hasil Analisis WP

Activity Diagram Melihat Hasil Analisis WP seperti pada lampiran B.4 menjelaskan bahwa terjadi aktivitas admin dan sistem. Menjelaskan juga aktivitas proses admin dalam melihat hasil analisa WP. Proses tersebut yaitu admin mengklik menu perhitungan WP dan sistem akan menampilkan data kriteria dan hasil perhitungan WP.

4.3.6.5 *Activity Diagram* Mengolah Data Produk

Activity Diagram Mengolah Data Produk seperti pada lampiran B.5 menjelaskan bahwa terjadi aktivitas admin menginputkan data produk (tambah, ubah dan hapus). Dalam *activity* ini juga menjelaskan proses aktivitas admin melihat data produk. Untuk menambah data produk, admin mengklik tombol tambah dan sistem menampilkan form tambah data produk sehingga admin dapat menginputkan data produk dan kriteria produk. Ketika ingin mengubah data

produk, maka admin mengklik tombol ubah dan sistem menampilkan form ubah data produk, setelah data diubah admin mengklik tombol simpan. Sedangkan untuk menghapus data produk yaitu dengan mengklik tombol hapus dan sistem akan menampilkan *message dialog* konfirmasi yakin ingin menghapus produk, jika iya maka admin mengklik tombol ok dan data produk terhapus, jika tidak maka admin mengklik cancel.

4.3.6.6 *Activity Diagram* Mengolah Data Penjualan

Activity Diagram Mengolah Data Penjualan seperti pada lampiran B.6 menjelaskan bahwa terjadi proses aktivitas admin melihat data penjualan. *Activity* ini juga menjelaskan proses aktivitas admin mengubah data penjualan. Admin mengklik menu data penjualan dan sistem akan menampilkan tabel data penjualan atau pesanan. Admin dapat mencari data penjualan yaitu dengan menginputkan nama produk atau data yang bersangkutan dan mengklik tombol cari dan sistem menampilkan data penjualan yang dicari. Untuk melihat detail atau mengubah data penjualan, admin mengklik tombol detail dan sistem menampilkan form data penjualan yang dapat diubah admin seperti status pemesanan dan nomor resi pengiriman, kemudian mengklik tombol update.

4.3.6.7 *Activity Diagram* Mengolah Data Ongkos Kirim

Activity Diagram Mengolah Data Ongkos Kirim seperti pada lampiran B.7 menjelaskan bahwa terjadi proses aktivitas admin melihat data ongkos kirim dan menjelaskan juga tentang proses aktivitas admin dalam mengubah data ongkos kirim. Proses untuk melihat data ongkos kirim yaitu dengan mengklik menu data ongkir dan sistem menampilkan tabel data ongkos kirim. Sedangkan proses untuk mengubah data ongkos kirim yaitu admin mengklik tombol ubah maka sistem menampilkan form data ongkos kirim dan yang dapat diubah hanya ongkos kirim saja karena data daerah atau wilayah sudah ditentukan sistem. Kemudian admin mengklik tombol simpan untuk menyimpan perubahan atau mengklik tombol batal untuk kembali ke tabel data ongkos kirim.

4.3.6.8 *Activity Diagram* Mengolah Profil

Activity Diagram Mengolah Profil seperti pada lampiran B.8 menjelaskan terjadinya aktivitas antara pembeli dan sistem. *Activity* ini juga menjelaskan

proses aktivitas pembeli dalam mennginput profil dan mengubah data profil tersebut. Pembeli mengklik menu profil untuk menampilkan form data profilyang dapat diubah dengan langsung mengubah data profil. Jika ingin menyimpan perubahan data profil, maka pembeli mengklik tombol simpan.

4.3.6.9 *Activity Diagram* Mengolah Data Keranjang

Activity Diagram Mengolah Data Keranjang seperti pada lampiran B.9 menjelaskan bahwa terjadi proses aktivitas pembeli dalam melihat data keranjang dan menginputkan serta mengubah data. Pembeli mengklik menu keranjang untuk menampilkan data keranjang atau data produk yang sudah dipesan. Kemudian pembeli mengisi form yang kosong seperti bukti bayar dan mengklik tombol update untuk menyimpan perubahan data keranjang.

4.3.6.10 *Activity Diagram* Mengolah History Pembelian

Activity Diagram Mengolah History Pembelian seperti pada lampiran B.10 menjelaskan bahwa terjadi aktivitas proses pembeli melihat history pembelian dan mengubah atau mereview data. Pembeli mengklik menu history pembelian dan sistem menampilkan tabel data history pembelian. Untuk mencari data history pembelian, pembeli mengisi form pencarian dengan menginputkan nama produk atau data yang bersangkutan dengan history pembelian kemudian mengklik tombol cari dan sistem akan menampilkan data history pembelian yang dicari. Sedangkan untuk melihat detail history pembelian, pembeli mengklik tombol detail dan mengisi riview untuk pembeli, lalu mengklik tombol riview bila ingin meriview history pembelian.

4.3.6.11 *Activity Diagram* Pemesanan

Activity Diagram Pemesanan seperti pada lampiran B.11 menjelaskan bahwa terjadi aktivitas pembeli melakukan pemesanan produk dan menginputkan data serta melihat jumlah harga yang harus dibayar. Untuk melakukan pesanan, pembeli membuka halaman utama dan mengklik tombol beli pada bawah gambar produk dan sistem menampilkan detail produk dan form jumlah produk yang dipesan, pembeli menginputkan jumlah produk yang dipesan dan mengklik beli maka sistem menampilkan form pemesanan. Pembeli mengisi form pesanan seperti bukti oembayaran bila sudah mengirim uang ke admin dan mengklik

update untuk menyimpan perubahan form pemesanan. Apabila pembeli belum mengirim uang ke admin, maka bukti bayar dapat dikosongi terlebih dahulu.

4.3.6.12 *Activity Diagram* Registrasi

Activity Diagram Registrasi seperti pada lampiran B.12 menjelaskan bahwa terjadi aktivitas proses pembeli mendaftarkan akun yang digunakan untuk login atau masuk ke dalam sistem. Pembeli mengklik menu registrasi dan sistem akan menampilkan form registrasi. Form registrasi harus diisi dan tidak boleh ada yang kosong. Pembeli menginputkan data nama, username, password dan nomor telepon dan mengklik tombol daftar, bila data terisi semua maka data tersebut akan terdaftar dalam sistem dan sistem menampilkan halaman utama, sedangkan bila ada data yang kosong maka pembeli harus melengkapi data terlebih dahulu.

4.3.6.13 *Activity Diagram* Logout

Activity Diagram Logout seperti pada lampiran B.13 menjelaskan terjadinya aktivitas antara admin dan pembeli dengan sistem. *Activity* ini juga menjelaskan proses aktivitas logout atau keluar dari sistem. Admin dan pembeli mengklik menu logout untuk dapat keluar dari sistem dan sistem akan menampilkan halaman utama user yang terdapat menu login untuk masuk kembali ke sistem.

4.3.7 *Sequence Diagram*

Sequence diagram merupakan penggambaran interaksi dari masing-masing komponen pada satu fungsi. Interaksi tersebut dilakukan oleh user pada sistem. Didalam sistem sendiri juga terjadi interaksi yaitu anatara view, controller dan model. Pada proses perancangan ini setiap fitur akan digambarkan kedalam *Sequence diagram*.

4.3.7.1 *Sequence Diagram* Login

Sequence Diagram Login menggambarkan antara aktor yaitu admin dan pembeli dengan sistem yang berupa interaksi dengan *view*, *controller*, dan *model*. Aktor memilih menu login dan yang berjalan yaitu *function* `log_in()` untuk menampilkan halaman login. Kemudian aktor mengisi username dan password dan klik tombol login dan yang berjalan adalah *function* `account_login()` untuk

masuk ke halaman utama dan menampilkan notifikasi apakah login sukses atau gagal. Klik tombol ok untuk masuk ke halaman awal atau ke halaman login apabila login gagal. Apabila lupa dengan username atau password, dapat klik tombol lupa password dan yang berjalan adalah *function* lupa_pass() untuk menampilkan form lupa password. Kemudian mengisi username dan klik tombol reset pasword dan yang berjalan adalah *function* reset_login() untuk mereset password yang dipakai. Interaksi yang terjadi digambarkan dengan jelas pada lampiran C.1.

4.3.7.2 *Sequence Diagram* Melihat Data Pelanggan

Sequence Diagram Melihat Data Pelanggan menggambarkan aktor yaitu admin dan sistem yang berupa *view*, *controller*, dan *model*. Admin memilih menu data pelanggan dan yang berjalan yaitu *function* index() untuk menampilkan daftar data pelanggan. Kemudian admin mengklik tombol detail dan yang berjalan adalah *function* customer_form() untuk menampilkan detail data pelanggan. Interaksi yang terjadi digambarkan pada lampiran C.2.

4.3.7.3 *Sequence Diagram* Mengolah Data Kriteria dan Subkriteria

Sequence Diagram Mengolah Data Kriteria dan Subkriteria menggambarkan interaksi antara aktor yaitu admin dengan sistem yang berupa *view*, *controller*, dan *model*. Admin memilih menu data kriteria dan yang berjalan adalah *function* kriteria() untuk menampilkan data kriteria dan subkriteria. Untuk menambahkan subkriteria, klik tombol tambah dan yang berjalan adalah *function* subkriteria_form() untuk menampilkan form kriteria yang kemudian admin mengisi form kriteria dan klik tombol simpan dan *function* save_subkriteria berjalan untuk menyimpan data subkriteria. Untuk mengubah data subkriteria yang berjalan juga *function* subkriteria_form(), sedangkan saat menghapus data subkriteria yang berjalan adalah *function* subkriteria_delete(). Interaksi yang terjadi digambarkan pada lampiran C.3.

4.3.7.4 *Sequence Diagram* Melihat Hasil Analisis WP

Sequence Diagram Melihat Hasil Analisa WP menggambarkan interaksi antara aktor yaitu admin dengan sistem yang berupa *view*, *controller*, dan *model*. Admin memilih menu perhitungan WP dan yang berjalan adalah *function*

perhitungan_wp() untuk menampilkan tabel data kriteria dan normalisasi bobot tiap kriteria dan hasil perhitungan WP pada masing-masing produk. Interaksi yang terjadi digambarkan pada lampiran C.4.

4.3.7.5 *Sequence Diagram* Mengolah Data Produk

Sequence Diagram Mengolah Data Produk menggambarkan interaksi aktor yaitu admin dengan sistem yang berupa *view*, *controller*, dan *model*. Admin memilih menu data produk dan sistem akan menjalankan *function* inventory() untuk menampilkan tabel data produk. Kemudian untuk menambah data produk baru, admin mengklik tombol tambah dan sistem menjalankan *function* inventory_form() untuk menampilkan form data produk. Untuk menyimpan data produk yang berjalan adalah *function* save_inventory() dan saat menghapus data produk, maka yang berjalan adalah *function* delete_inventory(). Interaksi yang terjadi digambarkan pada lampiran C.5.

4.3.7.6 *Sequence Diagram* Mengolah Data Penjualan

Sequence Diagram Mengolah Data Penjualan menggambarkan interaksi aktor yaitu admin dengan sistem yang berupa *view*, *controller*, dan *model*. Admin memilih menu data penjualan dan sistem akan menjalankan *function* index() untuk menampilkan tabel data penjualan. Kemudian admin mengklik tombol detail data penjualan dan sistem menjalankan *function* detail_sale_form() untuk menampilkan detail dari data penjualan produk, saat mengubah detail penjualan dan mengklik tombol update yang dijalankan adalah *function* update_bayar(). Interaksi yang terjadi digambarkan pada lampiran C.6.

4.3.7.7 *Sequence Diagram* Mengolah Data Ongkos Kirim

Sequence Diagram Mengolah Data Ongkos Kirim menggambarkan interaksi aktor yaitu admin dengan sistem yang berupa *view*, *controller*, dan *model*. Admin memilih menu data ongkir maka yang berjalan adalah *function* wilayah() untuk menampilkan tabel data ongkos kirim dan admin mengubah data ongkos kirim dengan mengklik tombol ubah. Saat menyimpan perubahan data ongkos kirim dengan mengklik tombol simpan maka yang berjalan adalah *function* simpan_wilayan(). Interaksi yang terjadi digambarkan pada lampiran C.7.

4.3.7.8 *Sequence Diagram* Mengolah Profil

Sequence Diagram Mengolah Profil menggambarkan interaksi aktor yaitu pembeli dan sistem yang berupa *view*, *controller*, dan *model*. Pembeli memilih menu profil dan sistem menjalankan *function* `index()` untuk menampilkan form data profil yang dapat diubah dan menyimpannya dengan mengklik tombol simpan, kemudian sistem menjalankan *function* `customer_form()` untuk menyimpan dan menampilkan data profil yang telah disimpan. Interaksi yang terjadi digambarkan pada lampiran C.8.

4.3.7.9 *Sequence Diagram* Mengolah Data Keranjang

Sequence Diagram Mengolah Data Keranjang menggambarkan interaksi aktor yaitu pembeli dengan sistem yang berupa *view*, *controller*, dan *model*. Pembeli memilih menu keranjang dan sistem akan menjalankan *function* `cart()` untuk menampilkan data keranjang atau daftar produk yang telah dipesan. Kemudian admin mengubah bukti bayar dan catatan dan mengklik tombol update maka sistem akan menjalankan *function* `update_bayar()` untuk menyimpan dan menampilkan data keranjang yang sudah diubah. Interaksi yang terjadi digambarkan pada lampiran C.9.

4.3.7.10 *Sequence Diagram* Mengolah History Pembelian

Sequence Diagram Mengolah History Pembelian menggambarkan interaksi aktor yaitu pembeli dengan sistem yang berupa *view*, *controller*, dan *model*. Pembeli memilih menu history pembelian dan yang berjalan adalah *function* `pembelian()` untuk menampilkan data produk yang telah dibeli atau masih dipesan. Kemudian pembeli mengklik tombol detail dan yang berjalan adalah *function* `detai_sale_form()` untuk menampilkan detail history pembelian dan pembeli mengubah review produk serta mengklik tombol update, maka sistem akan menjalankan *function* `update_riview()` untuk menyimpan dan menampilkan detail history pembelian yang diubah. Interaksi yang terjadi digambarkan pada lampiran C.10.

4.3.7.11 *Sequence Diagram* Pemesanan

Sequence Diagram Pemesanan menggambarkan interaksi aktor yaitu pembeli dengan sistem yang berupa *view*, *controller*, dan *model*. Pembeli

mengklik tombol beli pada produk dan sistem menjalankan *function* `detail_item()` untuk menampilkan detail produk serta mengubah jumlah produk yang dipesan dan mengisi catatan pemesanan, kemudian mengklik tombol beli maka sistem menjalankan *function* `buy()` untuk menampilkan data pemesanan. Pembeli menginputkan bukti bayar dan mengklik tombol update dan yang dijalankan adalah *function* `update_bayar()` untuk menampilkan perubahan data pemesanan. Interaksi yang terjadi digambarkan pada lampiran C.11.

4.3.7.12 *Sequence Diagram* Registrasi

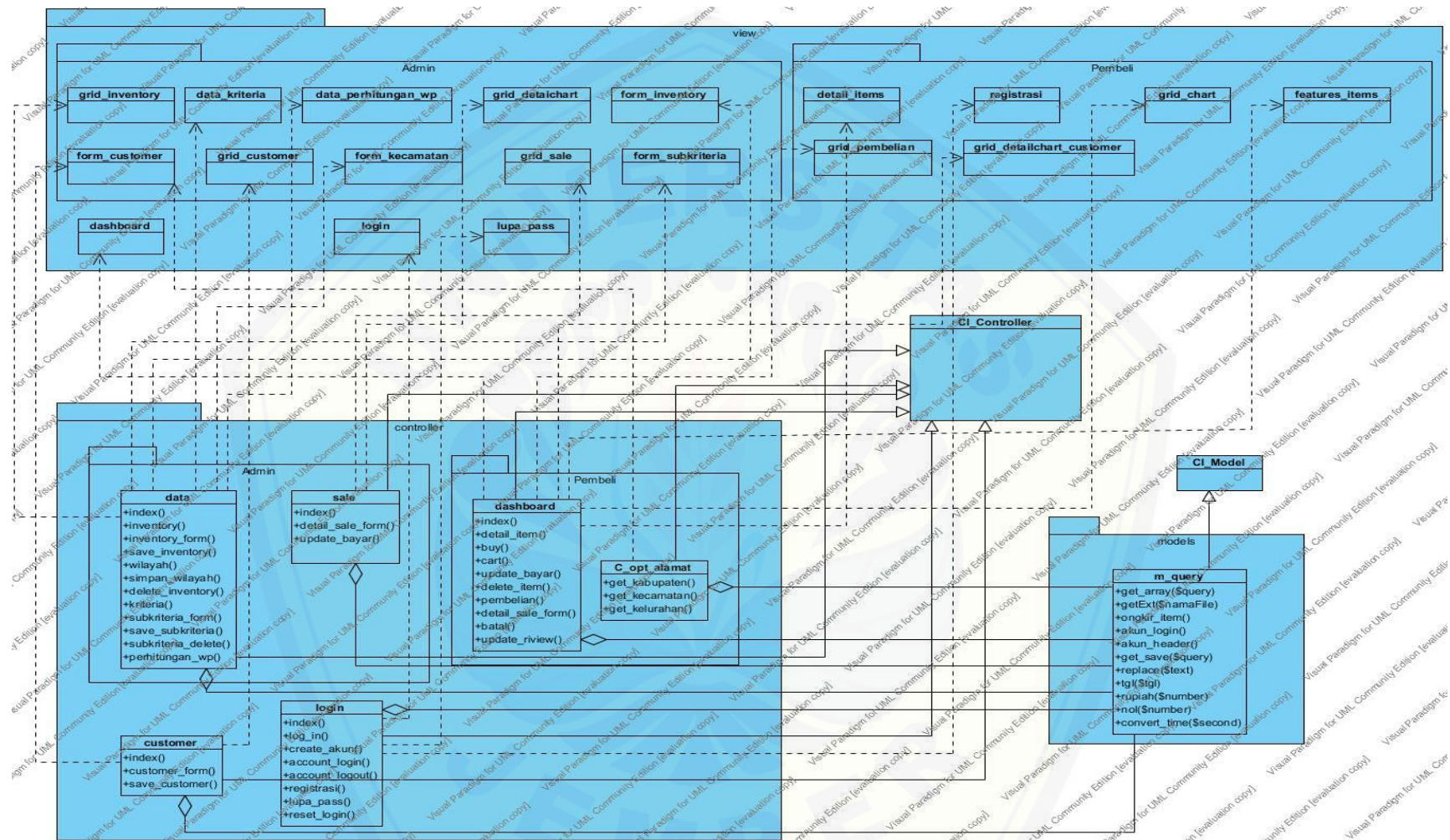
Sequence Diagram Registrasi menggambarkan interaksi aktor yaitu pembeli dengan sistem yang berupa *view*, *controller*, dan *model*. Pembeli memilih menu registrasi dan yang dijalankan adalah *function* `registrasi()` untuk menampilkan form registrasi. Kemudian pembeli mengisi data registrasi dan mengklik tombol daftar dan yang dijalankan adalah *function* `registrasi()`, apabila data lengkap maka registrasi berhasil tetapi bila data tidak lengkap maka ada notifikasi data tidak boleh kosong. Interaksi yang terjadi ditampilkan pada lampiran C.12.

4.3.7.13 *Sequence Diagram* Logout

Sequence Diagram Logout menggambarkan interaksi aktor yaitu admin dan pembeli dengan sistem yang berupa *view*, *controller*, dan *model*. Admin dan pembeli memilih menu logout dan yang berjalan adalah *function* `account_logout()` untuk keluar dari sistem dan menampilkan halaman utama user. Interaksi yang terjadi digambarkan pada lampiran C.13.

4.3.8 *Class Diagram*

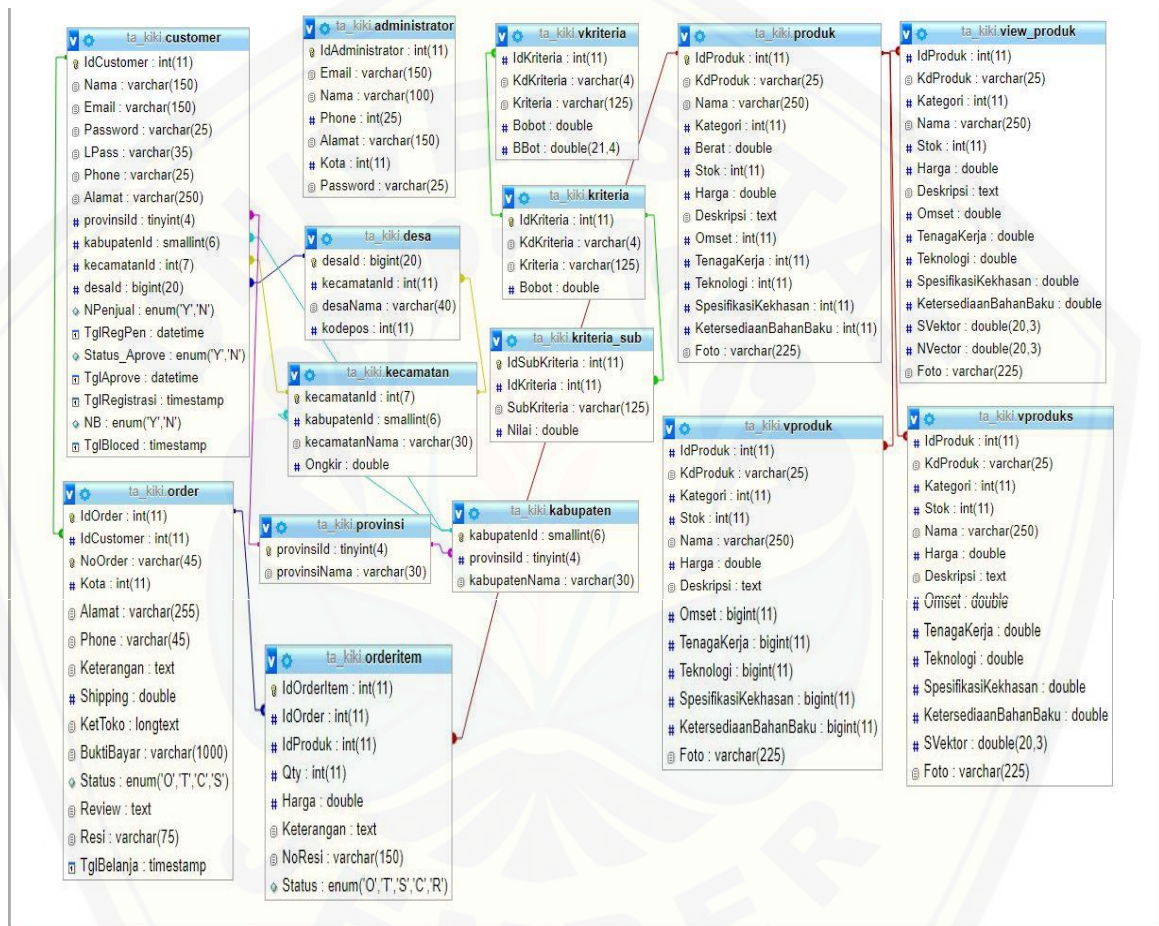
Class Diagram menggambarkan struktur dan penjelasan class, paket dan objek serta hubungan satu sama lain seperti pewarisan, asosiasi, dan sebagainya. Selain itu *class diagram* juga menjelaskan hubungan antarclass dalam sebuah sistem yang sedang dirancang sehingga bagaimana caranya setiap class saling berkolaborasi untuk mencapai sebuah tujuan. Class diagram sistem dapat dilihat pada gambar 4.3.



Gambar 4. 3 Class Diagram Sistem

4.3.9 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) pada sistem informasi penjualan produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan ini menjelaskan hubungan antardata dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antarrelasi. ERD aplikasi ditunjukkan pada gambar 4.4.



Gambar 4. 4 Entity Relationship Diagram (ERD)

4.4 Penulisan Kode Program

Setelah melalui tahapan desain sistem yang digambarkan melalui *usecase diagram*, *usecase skenario*, *activity diagram*, *sequence diagram*, *class diagram*, dan *entity relationship diagram* adalah tahapan penulisan kode program atau sering disebut dengan istilah *coding*. Dalam tahapan ini dilakukan dengan

menggunakan bahasa pemrograman php, html dan css dengan *framework Code Igniter (CI)* dan *database* yang digunakan adalah *mysql*.

Implementasi metode *Weighted Product (WP)* dalam sistem informasi penjualan produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan ini dibangun dengan menggunakan *framework Code Igniter (CI)* terdiri dari *model, controller* dan *view*. Penulisan kode program ditulis berdasarkan fungsi-fungsi yang ada pada sistem. Detail penulisan kode program dapat dilihat pada lampiran F.

4.5 Pengujian Sistem

Pada penelitian ini penulis menggunakan dua metode pengujian sistem yaitu *white box testing* dan *black box testing*.

4.5.1 White Box Testing

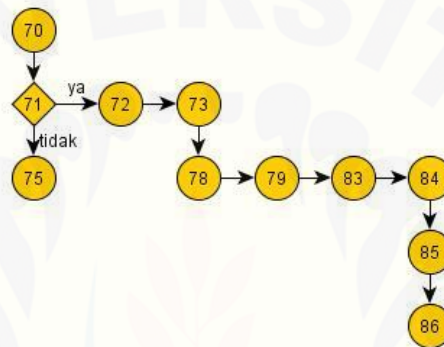
Pengujian *white box* yang dilakukan pada penelitian ini dimulai dari pembuatan diagram alir dari program yang diujikan. Dalam pengujian *white box* terdapat beberapa tahapan yaitu *cyclomatic complexity, listing program, penentuan jalur independen, dan test case*. Pada tahap pengujian ini, fungsi yang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

Pengujian Listing Program *function inventory()*

```
70 public function inventory(){
71     if($this->session->userdata("Level")==="Customer"){
72         $idUser = $this->session->userdata("IdUser");
73         $WHere = "AND p.IdCustomer='".$idUser.'";";
74     }else{
75         $WHere = "";
76     }
77
78     $SrcInventory = $this->m_query->replace($this->input-
        >post('SrcInventory'));
79     $data['qproduk'] = $this->m_query->get_array("SELECT p.IdProduk,
        p.KdProduk, p.Berat, p>Nama,
```

```

80     p.Stok, p.Harga
81     FROM produk p
82     WHERE (p>Nama like '%".$SrcInventory."%')->result();
83     $data['header']    = "";
84     $data['page']      = "dashboard/dashboard";
85     $data['content']= "data/grid_inventory";
86     $this->load->view('themes',$data);
87 }
    
```



Gambar 4. 5 Diagram alir *function* inventory()

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

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 11 - 11 + 2 \\
 &= 2
 \end{aligned}$$

- c. Basis Set

Jalur 1 : 1 – 2 – 3 – 4 – 6 – 7 – 8 – 9 – 10 – 11

Jalur 2 : 1 – 2 – 5 – 6

- d. Test Case

Test Case	Memanggil data inventory
Target yang diharapkan	Menampilkan data inventory atau data produk
Hasil Pengujian	Benar

Path / Jalur	1 – 2 – 3 – 4 – 6 – 7 – 8 – 9 – 10 – 11
---------------------	---

4.5.2 Black Box Testing

Pengujian *Black box* merupakan pengujian yang dilakukan oleh user dan aplikasi yang dibangun pada penelitian ini akan diuji dengan mengujikan langsung *running* aplikasi dan melakukan kegiatan pengujian dengan menganalisis proses *input* dan *output* yang dihasilkan aplikasi. Dokumentasi hasil pengujian sistem dapat dilihat pada lampiran pengujian.

Tabel 4. 6 Tabel Pengujian *Black Box*

No	Menu	Fungsi	Kasus	Hasil	Ket
1	Data Kriteria	Melihat Data Kriteria dan Subkriteria	Admin mengklik menu data kriteria	Menampilkan tabel data kriteria dan subkriteria	OK
		Menambah Data Subkriteria	Admin mengklik tombol tambah	Menampilkan form tambah data subkriteria	OK
		Mengubah Data Subkriteria	Admin mengklik tombol edit	Menampilkan form ubah data subkriteria	OK
		Menghapus Data Subkriteria	Admin mengklik tombol delete	Menampilkan notifikasi penghapusan data	OK

dilanjutkan

Lanjutan:

No	Menu	Fungsi	Kasus	Hasil	Ket
2	Perhitungan WP	Melihat Hasil Perhitungan WP	Admin mengklik menu perhitungan wp	Menampilkan tabel hasil perhitungan WP	OK
3	Data Pelanggan	Melihat Data Pelanggan	Admin mengklik menu data pelanggan	Menampilkan tabel data pelanggan	OK
4	Data Produk	Melihat Data Produk	Admin mengklik menu data produk	Menampilkan tabel data produk	OK
		Menambah Data Produk	Admin mengklik tombol tambah	Menampilkan form tambah data produk	OK
		Mengubah Data Produk	Admin mengklik tombol ubah	Menampilkan form ubah data produk	OK
		Menghapus Data Produk	Admin mengklik tombol hapus	Menampilkan notifikasi penghapusan data	OK

dilanjutkan

Lanjutan:

No	Menu	Fungsi	Kasus	Hasil	Ket
5	Data Ongkir	Melihat Data Ongkir	Admin mengklik menu data ongkir	Menampilkan tabel data ongkir	OK
		Mengubah Data Ongkir	Admin mengklik tombol ubah	Menampilkan form ubah data ongkir	OK
6	Data Penjualan	Melihat Data Penjualan	Admin mengklik menu data penjualan	Menampilkan tabel data penjualan	OK
		Melihat Detail Data Penjualan	Admin mengklik tombol detail	Menampilkan detail data penjualan	OK
7	Profil	Melihat Profil	Pembeli mengklik menu profil	Menampilkan form profil pembeli	OK
		Mengubah Profil	Pembeli mengubah profil dan mengklik tombol simpan	Menampilkan profil yang telah diubah	OK

dilanjutkan

Lanjutan:

No	Menu	Fungsi	Kasus	Hasil	Ket
8	Keranjang	Melihat Keranjang Pembelian	Pembeli mengklik menu keranjang	Menampilkan daftar keranjang pembelian	OK
9	History Pembelian	Melihat History Pembelian	Pembeli mengklik menu history pembelian	Menampilkan tabel history pembelian	OK
		Melihat Detail History Pembelian	Pembeli mengklik tombol detail	Menampilkan detail history pembelian	OK

BAB 6. PENUTUP

6.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan dari tahap analisis, perancangan sistem, pengkodean, dan pengujian sistem, maka dapat ditarik beberapa kesimpulan yaitu:

1. Penerapan metode *Weighted Product* (WP) untuk memperoleh alternatif pemilihan produk unggulan sebagai alat bantu untuk mengambil keputusan pemilihan produk unggul yaitu berdasarkan input nilai kriteria dan bobot yang telah ditentukan oleh admin. Nilai kriteria dan bobot tersebut akan dihitung vektor S dan vektor V yang kemudian dirangking dengan *descending* (urutan dari nilai yang terbesar) untuk mendapatkan alternatif produk unggulan. Semakin besar nilai bobot yang diinputkan oleh admin pada suatu kriteria, maka kriteria tersebut akan lebih diutamakan dalam melakukan pemilihan produk unggulan. Sistem informasi pemilihan produk unggulan menggunakan metode *Weighted Product* dibangun untuk memudahkan admin dan pembeli untuk melakukan jual beli produk dan menentukan produk unggulan. Dalam membangun sistem informasi ini dibutuhkan data produk dan data kriteria yang digunakan untuk menentukan produk unggulan.
2. Sistem informasi penjualan produk unggulan Kabupaten Lumajang dengan fitur pemilihan produk unggulan menggunakan metode *Weighted Product* dibangun dengan model *waterfall*. Dalam tahap menganalisa kebutuhan, diperoleh 9 kebutuhan fungsional yaitu sistem dapat menampilkan data pelanggan, mengolah data produk, mengolah data kriteria dan subkriteria, menampilkan hasil analisa produk unggulan, mengolah data penjualan, mengolah data ongkos kirim, mengolah profil, mengolah data keranjang, dan mengolah history pembelian, serta 4 kebutuhan nonfungsional. Untuk tahap desain sistem yang dibuat meliputi *bussines process* yang memerlukan 6 data inputan, menghasilkan 8 tabel data output, sistem berbasis web, dan tujuannya yaitu menentukan produk unggulan; *usecase diagram* yang terdapat

13 *usecase*; 13 *usecase scenario*; 13 *activity diagram*; 13 *sequence diagram*; *class diagram*; dan *entity relationship diagram* (ERD) dengan 11 tabel dan 4 tabel view pada database. Tahap penulisan kode program menggunakan bahasa pemrogramanphp, html dan css dengan *framework Code Igniter* (CI) dan database mysql. Serta tahap pengujian sistem dengan *whitebox testing* dan *blackbox testing*.

3. Dari penelitian yang dilakukan, diperoleh hasil perhitungan produk unggulan yang dapat digunakan sebagai acuan untuk pengambilan keputusan produk mana yang akan diutamakan penjualannya.

6.2 Saran

Beberapa hal yang dapat disarankan untuk pengembangan lebih lanjut adalah sebagai berikut:

1. Menerapkan sistem informasi penjualan dan promosi produk unggulan ini pada kabupaten lain sehingga kabupaten lain dapat menentukan produk unggulan serta mempermudah dalam penjualan dan promosi produk unggulannya.
2. Mengembangkan aplikasi ini dengan menggunakan metode lain untuk dapat membandingkan hasil perhitungan dengan menggunakan algoritma *Weighted Product*.

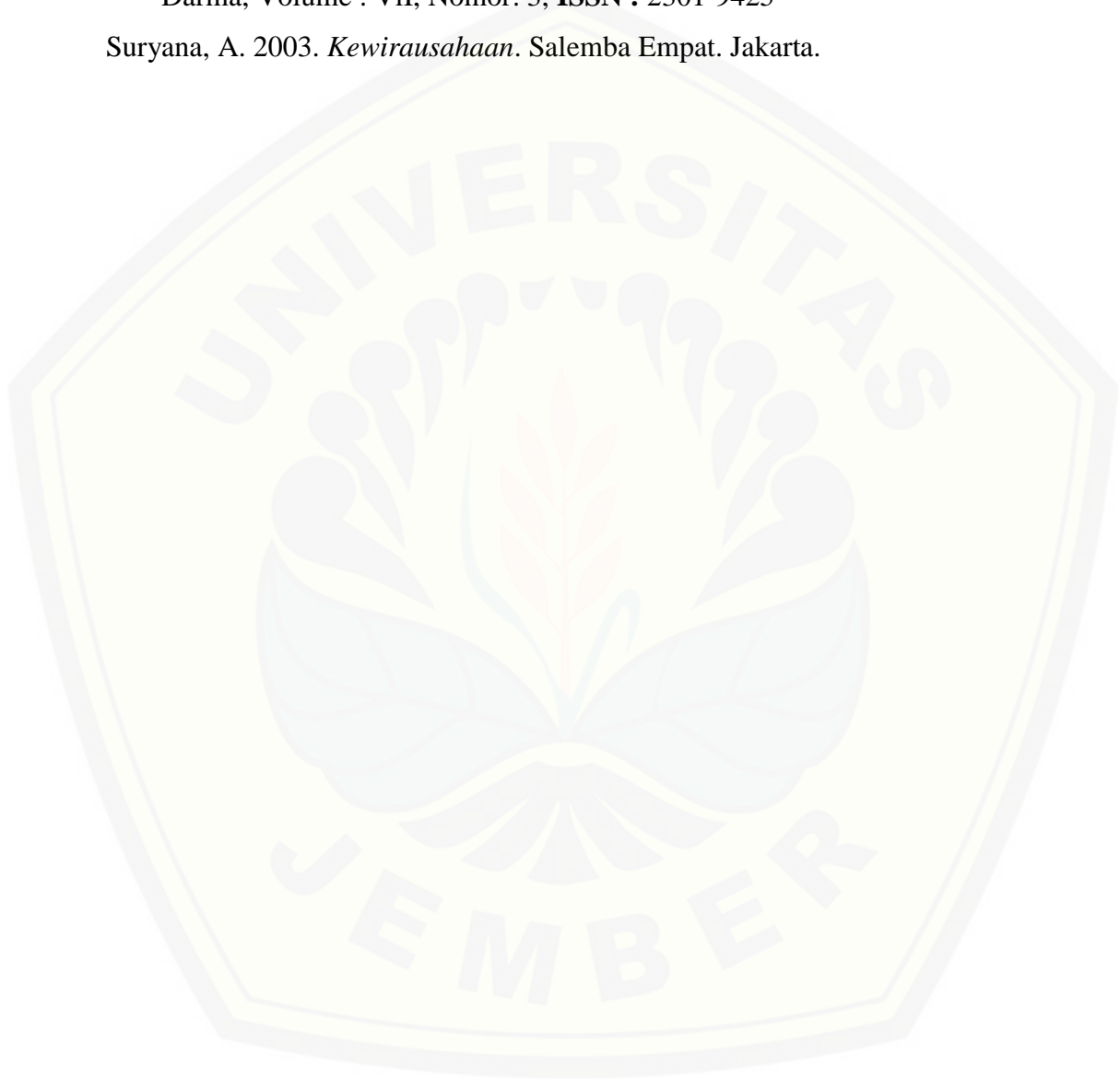
DAFTAR PUSTAKA

- Abdillah, Leon, Andretti. 2004. *Sistem Informasi Inventaris Barang*. Jurnal Ilmiah MATRIK, 6(3):133-152.
- Agissa, W. 2013. *With Box and Black Box Testing*. <http://bangwildan.web.id/>. Diakses 18 Maret 2016.
- Al-Bahra, B. L. 2005. *Analisis dan Desain Sistem Informasi dan Penjualan*. Yogyakarta: Graha Ilmu.
- Alfita, Riza. 2010 . *Perancangan Sistem Pendukung Keputusan Penentuan Prioritas Produk Unggulan Daerah Menggunakan Metode Weighted Product (WP)*. Fakultas Teknik Universitas Trunojoyo Madura
- Bunafit, N. 2005. *Perancangan Web dengan Fireworks dan Dreamweaver MX*. Bandung: Gava Media.
- Dimas. 2015. *Pengertian SI (Sistem Informasi)*. <https://www.kompasiana.com/dimasosd/55291077f17e6126268b48b6/pengertian-si-sistem-informasi>. Diakses 29 April 2016.
- Firdausy, Vionita. 2012. *Letak Geografis Lumajang*. <http://viogeo.blogspot.com/2012/03/letak-geografis-lumajang.html/>. Diakses 18 Maret 2016.
- Jaya, P. 2013. *Sistem Pendukung Keputusan Penentuan Bonus Karyawan Menggunakan Metode Weighted Product (WP)*. *Pelita Informatika Budi Darma* Vol V, No.2, 90-95, ISSN :2301-9425.
- Kusumadewi, Hartati, Harjoko and Wardoyo. (2006). *Fuzzy Multi- Attribute Decision Making (FUZZY MADM)*. Yogyakarta : Graha Ilmu.
- Muhammad, Siregar, Effendi, Fahlevi. 2009. *Identifikasi Potensi Ekonomi Masyarakat Kabupaten Tapin*.
- Nofriansyah, D. 2014. *Konsep Datamining VS Sistem Penunjang Keputusan*. Yogyakarta: Deepublish.
- Nusantoro, Jawoto. 2011. *Model Pengembangan Produk Unggulan Daerah Melalui Pendekatan Klaster Di Provinsi Lampung*. Universitas Muhammadiyah Metro.

Pressman, R. S. 2012. *Rekayasa Perangkat Lunak Pendekatan Praktisi*. Yogyakarta: Andi.

Rani, Sasika. 2014. *Sistem Pendukung Keputusan Pemilihan Sepeda Motor Berbasis Web dengan Metode Weighted Product*. Pelita Informatika Budi Darma, Volume : VII, Nomor: 3, **ISSN** : 2301-9425

Suryana, A. 2003. *Kewirausahaan*. Salemba Empat. Jakarta.



LAMPIRAN

Lampiran A. Skenario Diagram

A.1 Skenario Login

ID	01
Nama	Login
Aktor	Admin, pembeli
Prekondisi	Menu Login sudah ditampilkan
Post kondisi	Data Login sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Membuka aplikasi	
	2. Menampilkan halaman utama user
3. Klik menu Login	
	4. Menampilkan form login
5. Mengisi username dan password	
6. Klik login	
	7. Mengecek database
	8. Menampilkan peringatan login berhasil
9. Klik Ok	
	10. Menampilkan halaman utama
11. Klik lupa password	
	12. Menampilkan form lupa password
13. Mengisi username yang telah terdaftar	
14. Klik reset password	

Dilanjutkan

Lanjutan

Skenario Normal	
	15. Mereset password di database
	16. Menampilkan form login
Skenario Alternatif	
Ketika salah menginputkan username dan password	
5. Mengisi username dan password	
6. Klik login	
	7. Mengecek database
	8. Menampilkan peringatan login gagal.
9. Klik OK	
	10. Menampilkan form login

A.2 Skenario Melihat Data Pelanggan

ID	02
Nama	Melihat data pelanggan
Aktor	Admin
Prekondisi	Menu data pelanggan sudah ditampilkan
Post kondisi	Data pelanggan sudah ditampilkan
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu data pelanggan	
	2. Menampilkan tabel data pelanggan
3. Mengisi nama user di text field pencarian	

Dilanjutkan

Lanjutan

Skenario Normal	
4. Klik cari	
	5. Mengecek data pelanggan didatabase
	6. Menampilkan data akun user yang dicari
Ketika ingin melihat detail data pelanggan	
7. Klik detail	
	8. Menampilkan detail data pelanggan
9. Klik home	
	10. Menampilkan tabel data pelanggan

A.3 Skenario Mengolah Data Kriteria dan Subkriteria

ID	03
Nama	Mengolah data kriteria dan subkriteria
Aktor	Admin
Prekondisi	Menu data kriteria sudah ditampilkan
Post kondisi	Data sub kriteria sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu data kriteria	
	2. Menampilkan tabel data kriteria dan sub kriteria

Dilanjutkan

Lanjutan

Skenario Normal	
3. Klik tambah	
	4. Menampilkan form sub kriteria
5. Mengisi form	
6. Klik simpan	
	7. Menyimpan data kriteria didatabase
	8. Menampilkan data sub kriteria yang telah ditambah
9. Klik home	
	10. Menampilkan tabel data kriteria dan sub kriteria
Ketika ingin mengedit data sub kriteria	
11. Klik edit	
	12. Menampilkan form sub kriteria
13. Mengisi form sub kriteria	
14. Klik simpan	
	15. Menyimpan perubahan data sub kriteria didatabase
	16. Menampilkan data kriteria dan sub kriteria yang telah disimpan
17. Klik home	
	18. Menampilkan data kriteria dan sub kriteria
Ketika ingin menghapus data sub kriteria	
19. Klik delete	

Dilanjutkan

Lanjutan

Skenario Normal	
	20. Menampilkan message dialog yakin akan menghapus data?
21. Klik Ok	
	22. Menghapus data sub kriteria didatabase
	23. Menampilkan data sub kriteria yang telah dihapus
24. Klik Cancel	
	25. Menampilkan data kriteria dan sub kriteria

A.4 Skenario Melihat Hasil Analisis WP

ID	04
Nama	Perhitungan WP
Aktor	Admin
Prekondisi	Menu data akun user sudah ditampilkan
Post kondisi	Data perhitungan WP sudah ditampilkan
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu perhitungan WP	
	2. Menampilkan tabel data perhitungan WP

A.5 Skenario Mengolah Data Produk

ID	05
Nama	Mengolah data produk
Aktor	Admin
Prekondisi	Menu data produk sudah ditampilkan
Post kondisi	Data produk sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu produk	
	2. Menampilkan tabel data produk
3. Klik tambah produk	
	4. Menampilkan form tambah data produk
5. Mengisi form tambah data produk	
6. Klik simpan	
	7. Menyimpan data produk didatabase
	8. Menampilkan tabel data produk yang berhasil di tambah
Ketika klik tombol home	
9. Klik home	
	10. Menampilkan tabel data produk
Ketika ingin mengubah data produk	
11. Klik ubah	
	12. Menampilkan form ubah data produk

Dilanjutkan

Lanjutan

Skenario Normal	
13. Mengubah data produk	
14. Klik simpan	
	15. Menyimpan perubahan data produk didatabase
	16. Menampilkan table data produk yang sudah berhasil di ubah
Ketika klik tombol home	
17. Klik home	
	18. Menampilkan tabel data produk
Ketika ingin menghapus data produk	
19. Klik hapus	
	20. Menampilkan message dialog konfirmasi yakin ingin menghapus produk?
21. Klik Ok	
	22. Menghapus data di database
	23. Menampilkan data produk yang sudah terhapus
24. Klik Cancel	
	25. Menampilkan tabel data produk
Ketika mencari data produk	
26. Ketik data produk yang dicari	
27. Klik cari	
	28. Mengecek data produk didatabase
	29. Menampilkan data produk yang dicari

Dilanjutkan

Lanjutan

Skenario Alternatif	
Ketika data tidak lengkap	
5a. Mengisi form tambah data produk	
6a. Klik simpan	
	8a. Menampilkan peringatan harus mengisi data yang kosong.

A.6 Skenario Mengolah Data Penjualan

ID	06
Nama	Mengolah data penjualan
Aktor	Admin
Prekondisi	Menu data pesanan sudah ditampilkan
Post kondisi	Data penjualan sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu data penjualan	
	2. Menampilkan data penjualan
3. Menginputkan data penjualan di text field	
4. Klik cari	
	5. Mengecek data penjualan di database
	6. Menampilkan data penjualan yang dicari
7. Klik detail	

Dilanjutkan

Lanjutan

Skenario Normal	
	8. Menampilkan form keranjang belanja
9. Mengisi form	
10. Klik update	
	11. Menyimpan perubahan data penjualan didatabase
	12. Menampilkan data penjualan yang telah diupdate

A.7 Skenario Mengolah Data Ongkos Kirim

ID	07
Nama	Mengolah data ongkos kirim
Aktor	Admin
Prekondisi	Menu data ongkir sudah ditampilkan
Post kondisi	Data ongkir sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu data ongkir	
	2. Menampilkan tabel data ongkir
3. Klik ubah	
	4. Menampilkan form data ongkir
5. Mengubah form data ongkir	
6. Klik simpan	
	7. Menyimpan perubahan data ongkir didatabase

Dilanjutkan

Lanjutan

Skenario Normal	
	8. Menampilkan data ongkir yang telah diubah
9. Klik batal	
	10. Menampilkan tabel data ongkir

A.8 Skenario Mengolah Profil

ID	08
Nama	Mengolah data profil
Aktor	Pembeli
Prekondisi	Menu data profil sudah ditampilkan
Post kondisi	Data profil sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu profil	
	2. Menampilkan form data profil
3. Mengisi form data profil	
4. Klik simpan	
	5. Menyimpan data profil didatabase
	6. Menampilkan form data profil yang sudah tersimpan di database

A.9 Skenario Mengolah Keranjang

ID	09
Nama	Mengolah keranjang
Aktor	Pembeli
Prekondisi	Menu keranjang sudah ditampilkan
Post kondisi	Data keranjang sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu keranjang	
	2. Menampilkan tabel data dan form keranjang belanja
3. Mengisi form	
4. Klik update	
	5. Menyimpan perubahan data keranjang didatabase
	6. Menampilkan data keranjang yang diupdate

A.10 Skenario Mengolah History Pembelian

ID	10
Nama	Mengolah history pembelian
Aktor	Pembeli
Prekondisi	Menu history pembelian sudah ditampilkan
Post kondisi	Review data pembelian sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu history pembelian	
	2. Menampilkan data history pembelian
3. Mengisi nama user di text field pencarian	
4. Klik cari	
	5. Mengecek data pembelian didatabase
	6. Menampilkan data pembelian yang dicari
Ketika ingin melihat detail history pembelian	
7. Klik detail	
	8. Menampilkan detail history pembelian
9. Mengisi review produk	
10. Klik review	
	11. Menyimpan review produk didatabase

A.11 Skenario Pemesanan

ID	11
Nama	Pemesanan
Aktor	Pembeli
Prekondisi	Data produk dan tombol beli sudah ditampilkan
Post kondisi	Data pemesanan sudah disimpan didatabase
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik beli di gambar produk	
	2. Menampilkan form jumlah produk yang dibeli dan catatan pembelian
3. Mengisi form	
4. Klik beli	
	5. Menampilkan form pemesanan
6. Mengisi form pemesanan	
7. Klik Update	
	8. Menyimpan data pemesanan didatabase

A.12 Skenario Registrasi

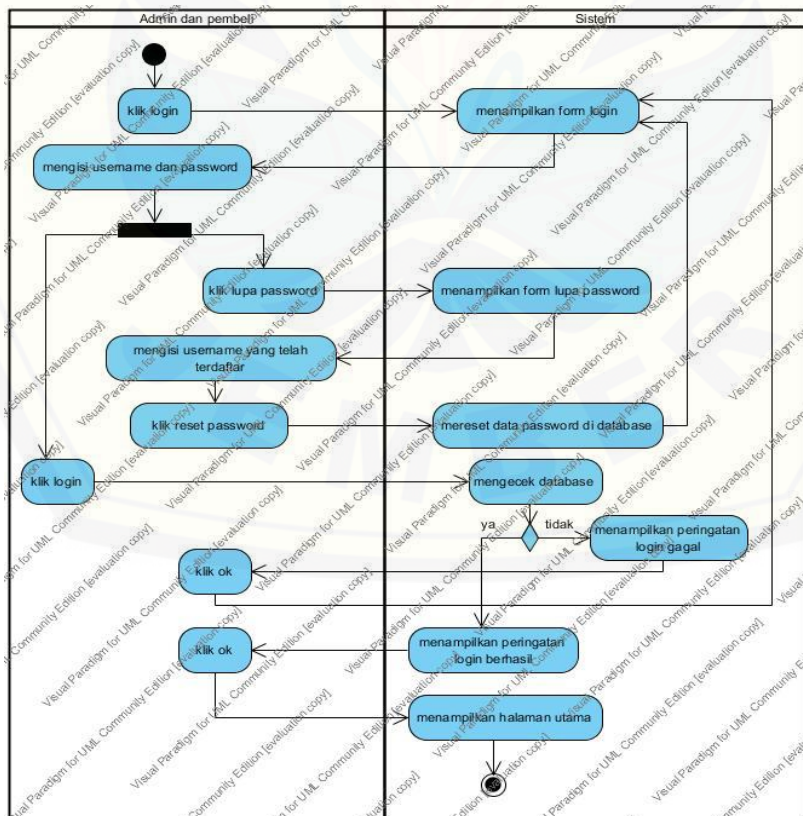
ID	12
Nama	Registrasi
Aktor	Pembeli
Prekondisi	Menu Registrasi sudah ditampilkan
Post kondisi	Data Registrasi sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Membuka aplikasi	
	2. Menampilkan halaman utama user
3. Klik menu Registrasi	
	4. Menampilkan form registrasi
5. Mengisi form registrasi	
6. Klik daftar	
	7. Menyimpan data registrasi didatabase
	8. Menampilkan halaman utama
Skenario Alternatif	
Ketika data tidak lengkap	
5. Mengisi form registrasi	
6. Klik daftar	
	7. Menampilkan peringatan harus mengisi data yang kosong.

A.13 Skenario Logout

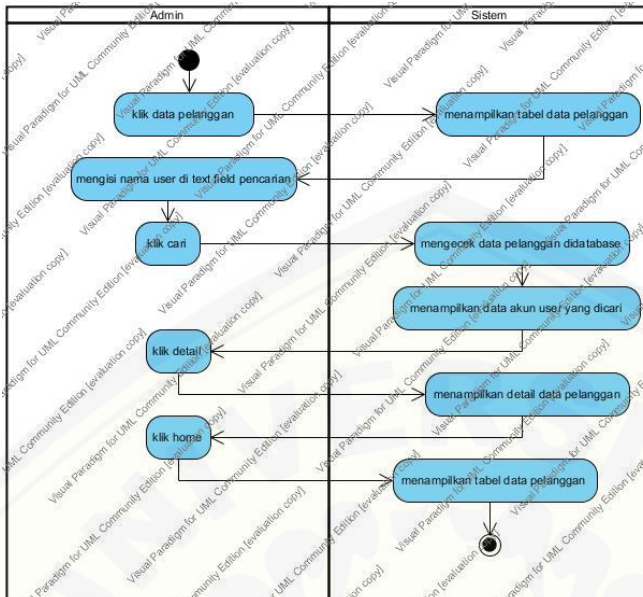
ID	13
Nama	Logout
Aktor	Admin, pembeli
Prekondisi	Menu data Logout sudah ditampilkan
Post kondisi	Data Logout sudah tersimpan di database
SKENARIO	
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Klik menu logout	2. Menampilkan halaman utama user

Lampiran B. Lampiran Activity Diagram

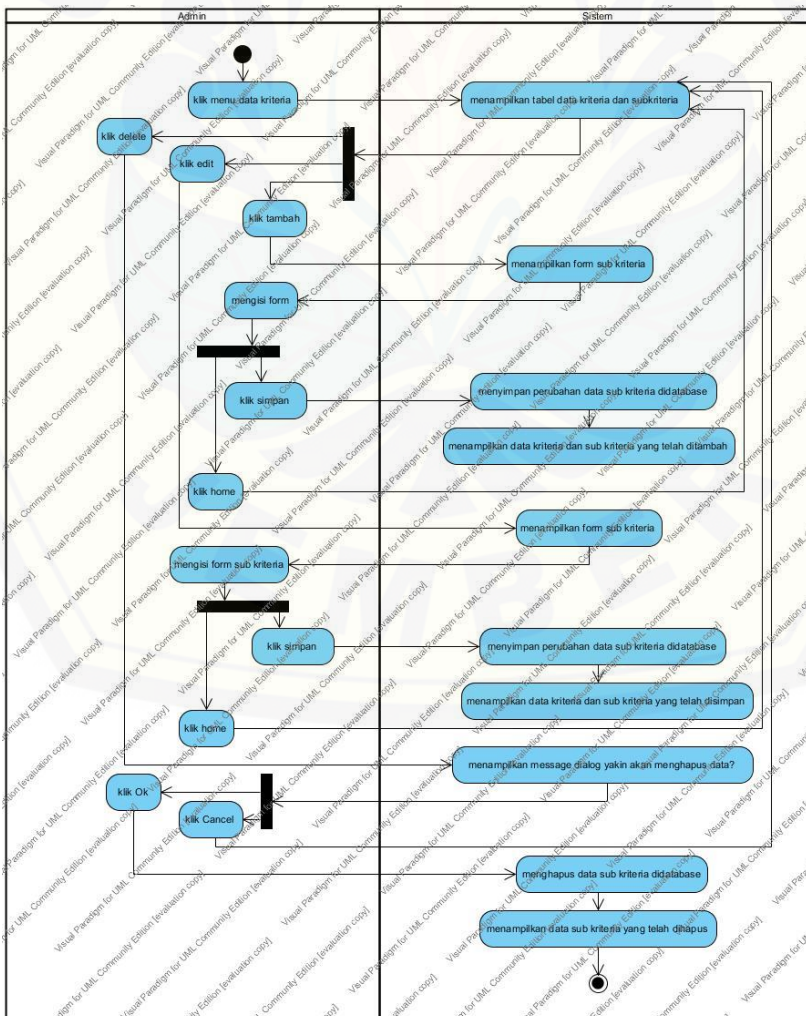
B.1 Activity Diagram Login



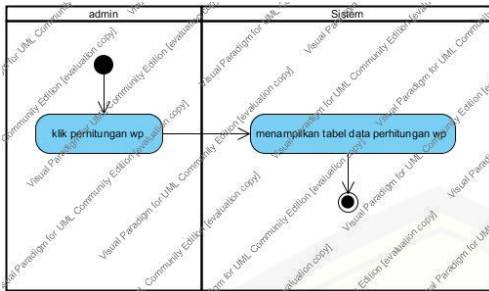
B.2 Activity Diagram Melihat Data Pelanggan



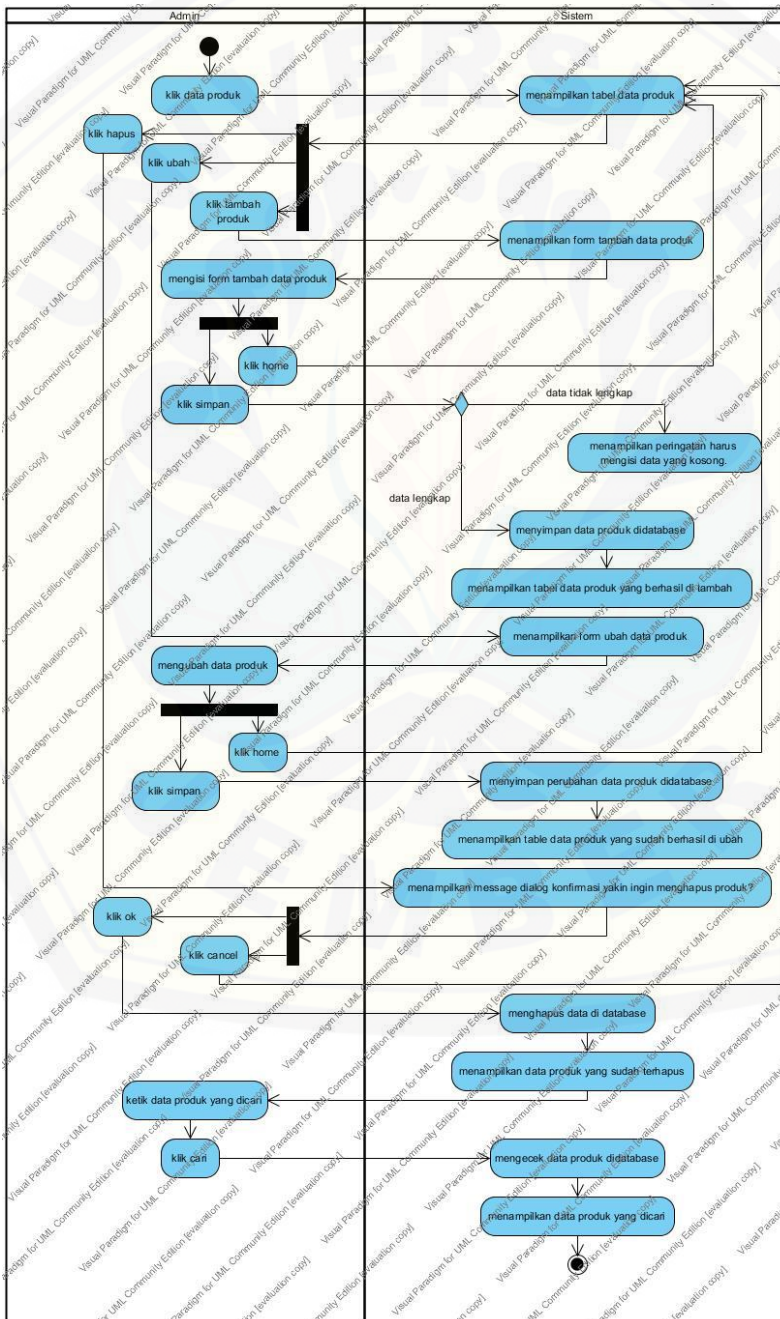
B.3 Activity Diagram Mengolah Data Kriteria dan Subkriteria



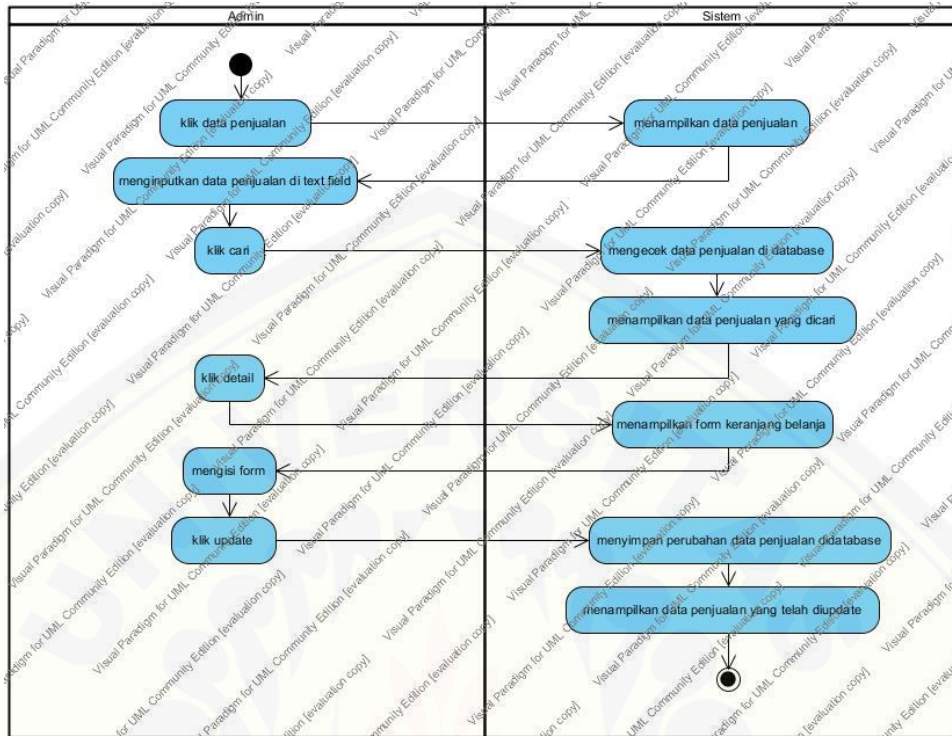
B.4 Activity Diagram Hasil analisis WP



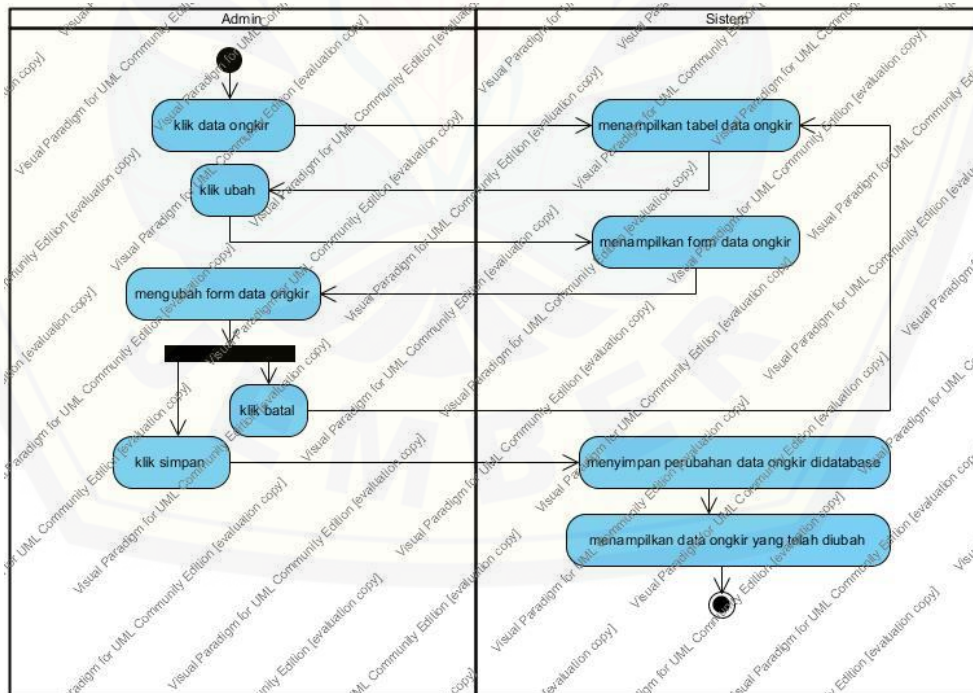
B.5 Activity Diagram Mengolah Data Produk



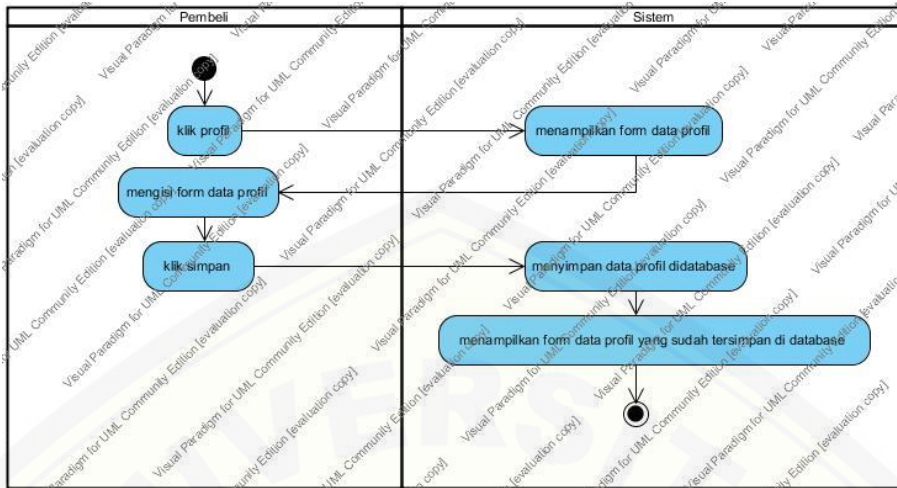
B.6 Activity Diagram Mengolah Data Penjualan



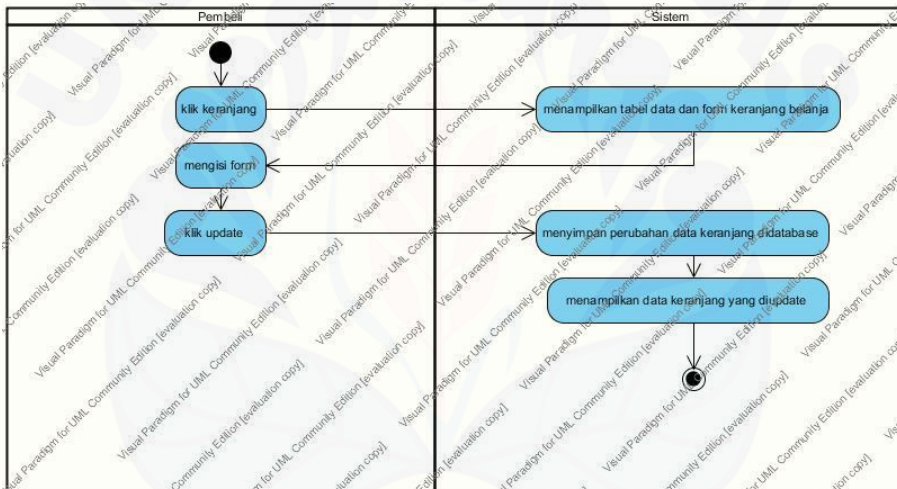
B.7 Activity Diagram Mengolah Data Ongkos Kirim



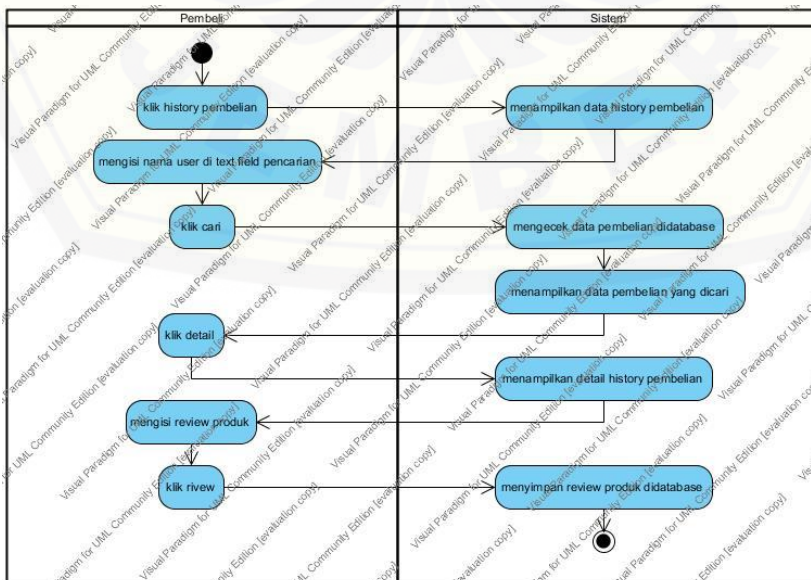
B.8 Activity Diagram Mengolah Profil



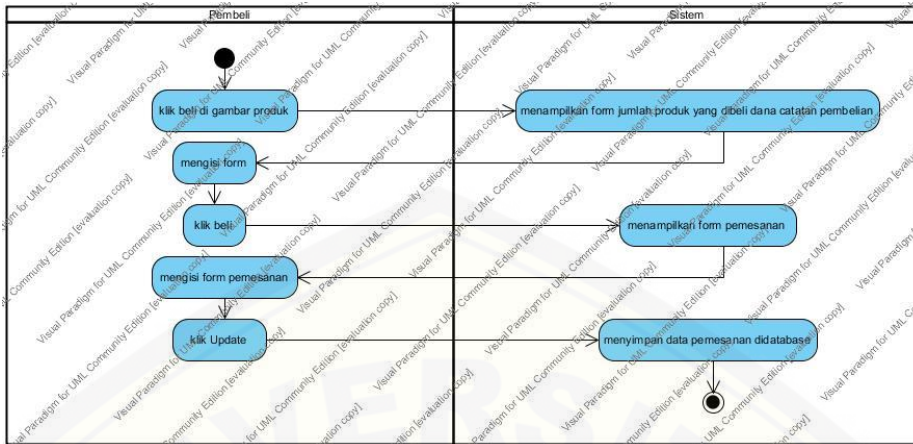
B.9 Activity Diagram Mengolah Keranjang



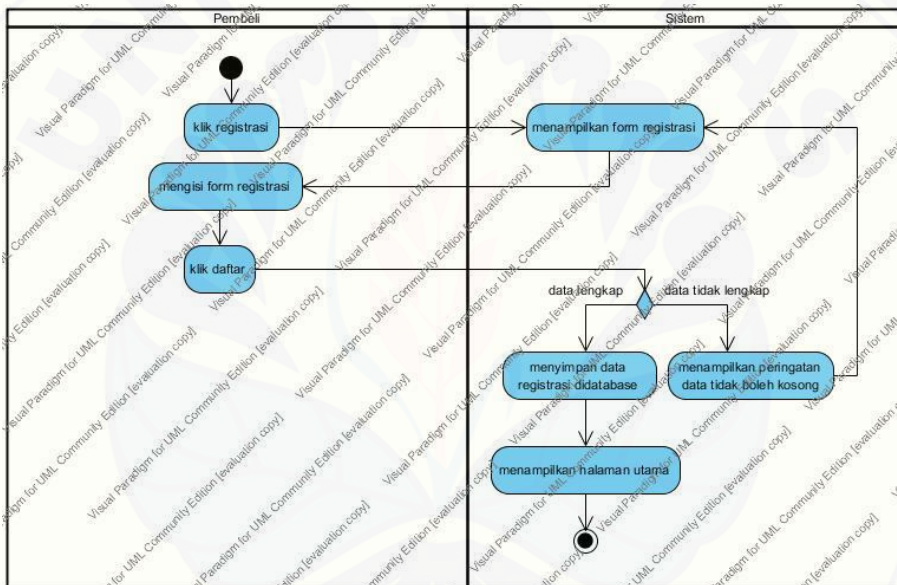
B.10 Activity Diagram Mengolah History Pembelian



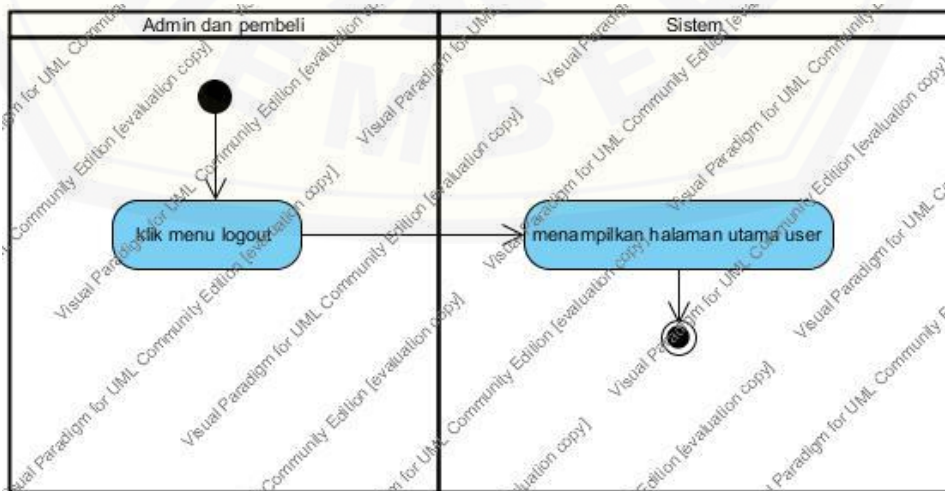
B.11 Activity Diagram Pemesanan



B.12 Activity Diagram Registrasi

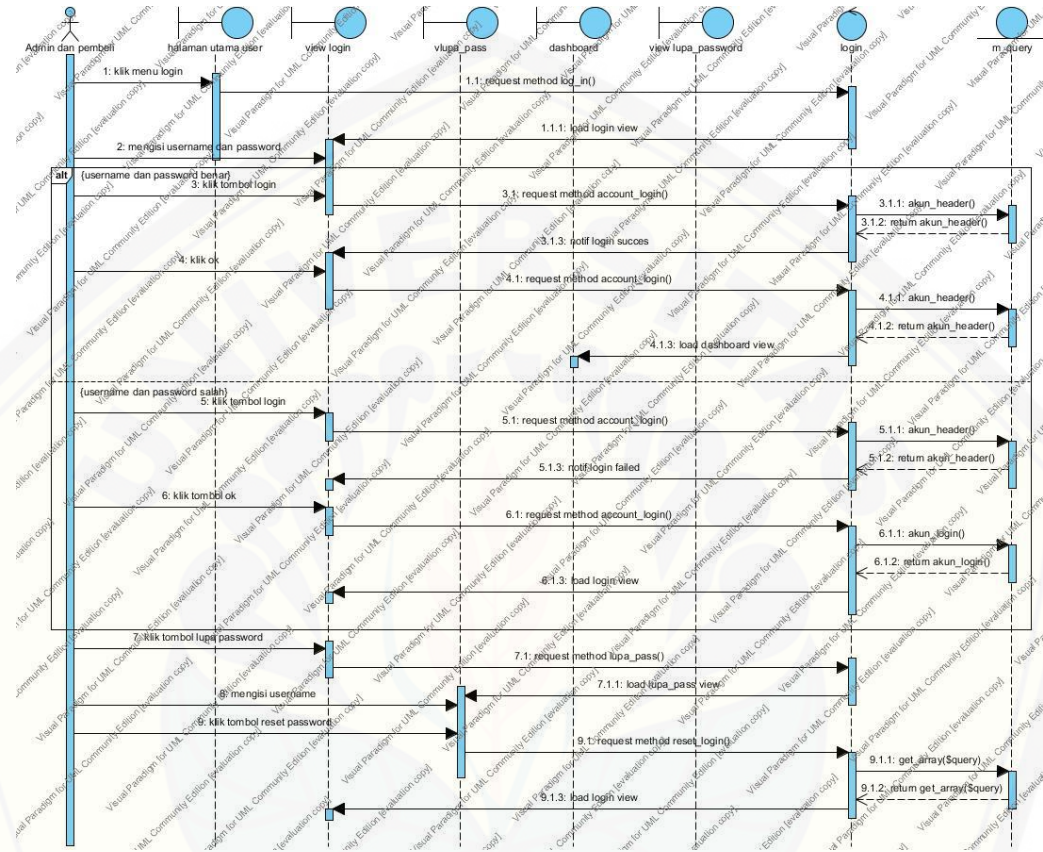


B.13 Activity Diagram Logout

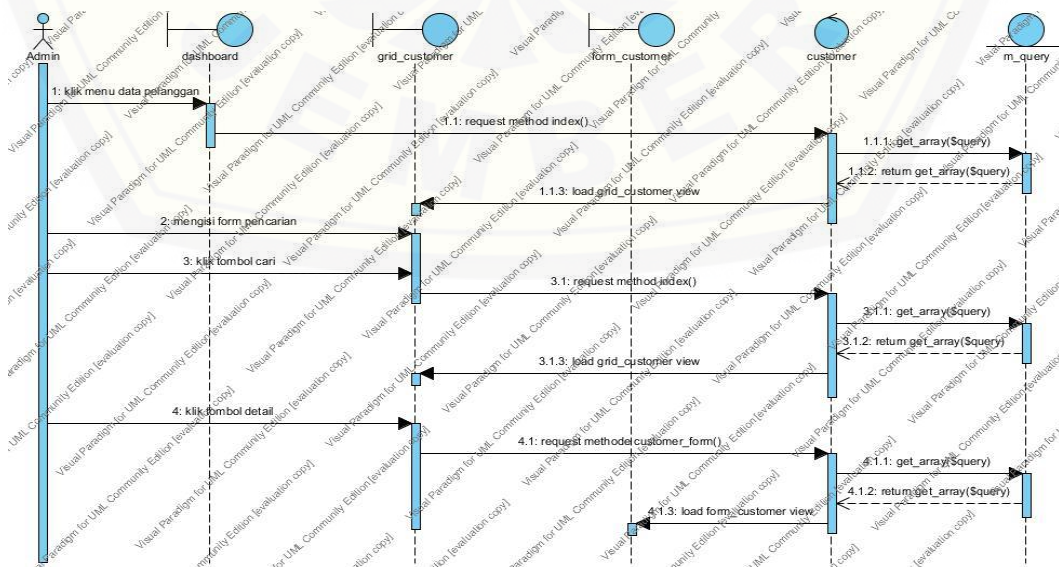


Lampiran C. Lampiran Sequence Diagram

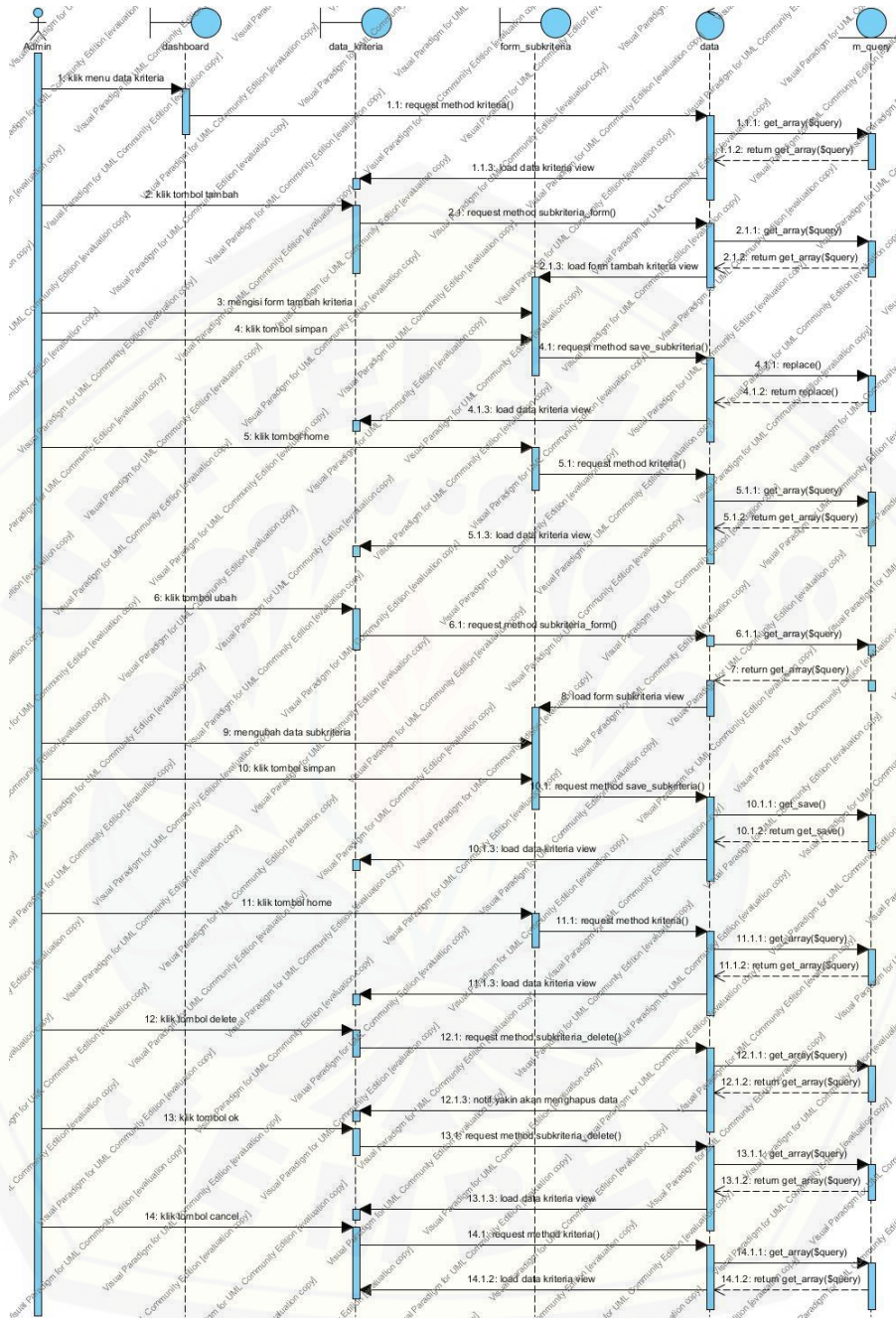
C.1 Sequence Diagram Login



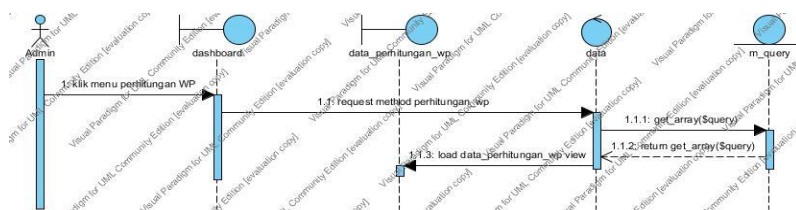
C.2 Sequence Diagram Melihat Data Pelanggan



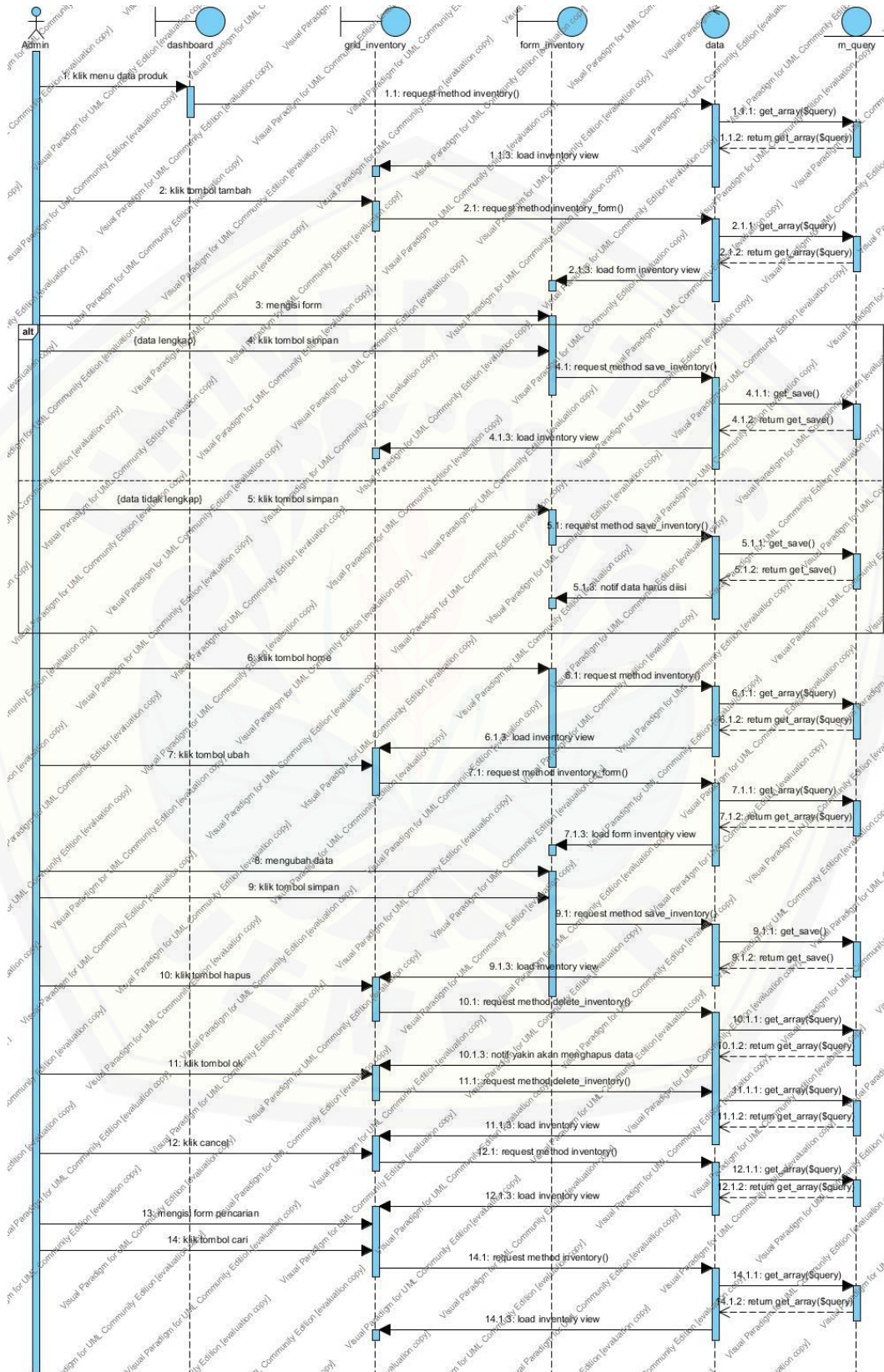
C.3 Sequence Diagram Mengolah Data Kriteria dan Subkriteria



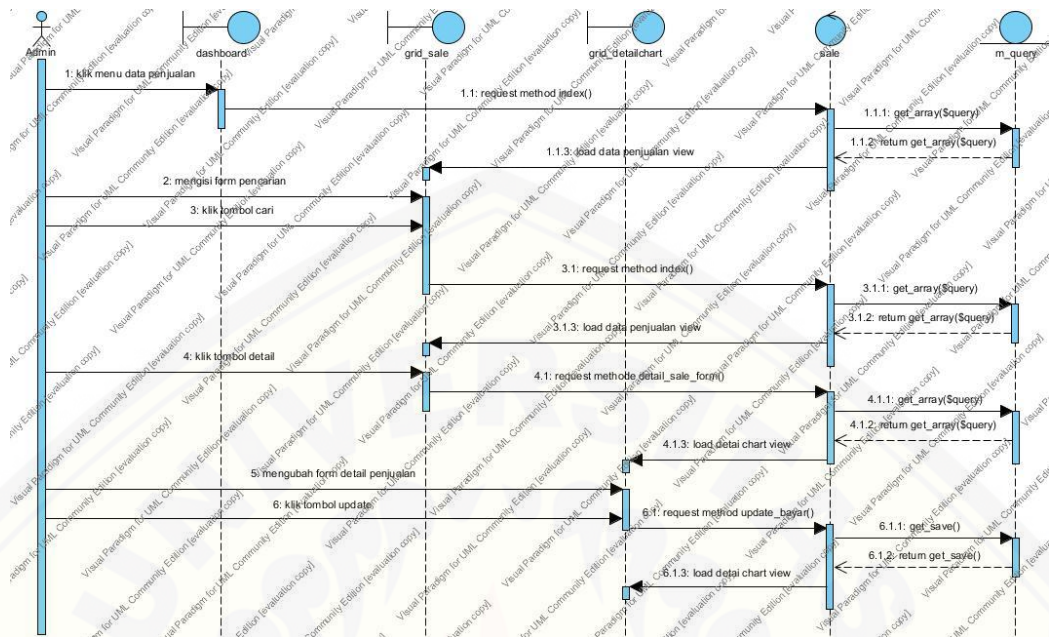
C.4 Sequence Diagram Melihat Hasil Analisis WP



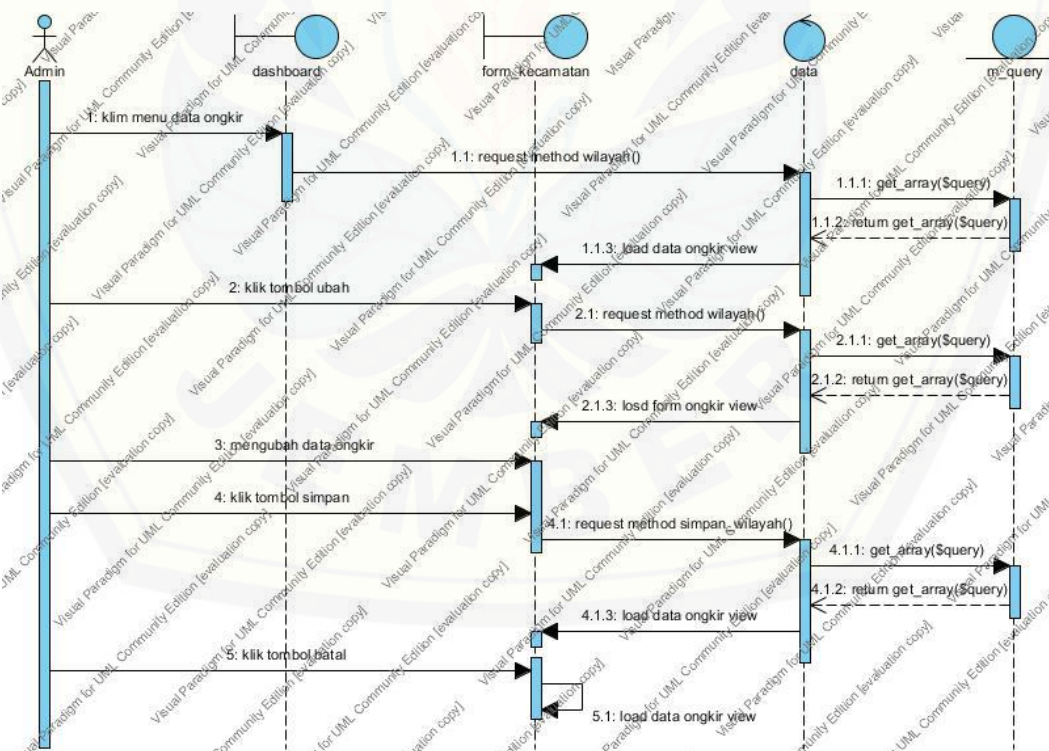
C.5 Sequence Diagram Mengolah Data Produk



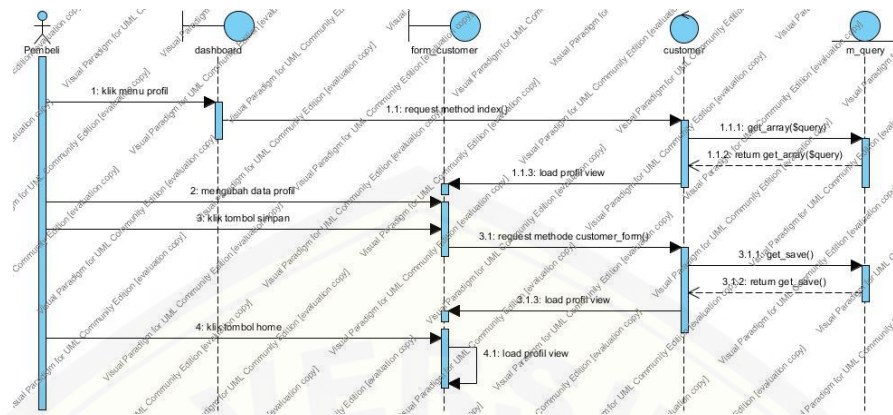
C.6 Sequence Diagram Mengolah Data Penjualan



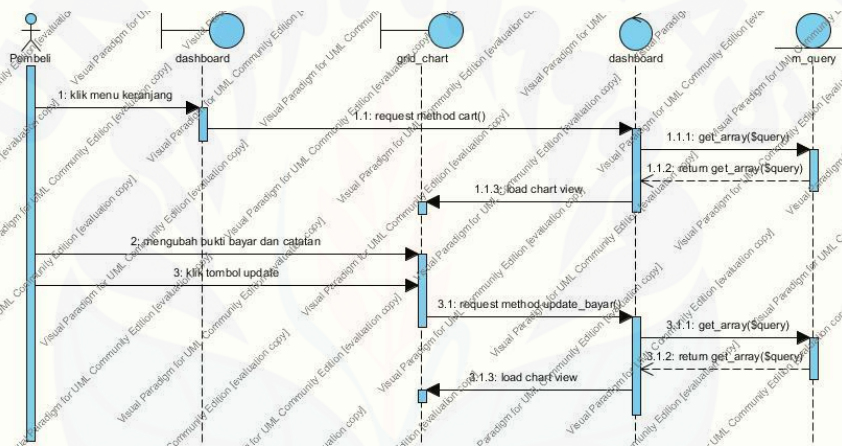
C.7 Sequence Diagram Mengolah Data Ongkos Kirim



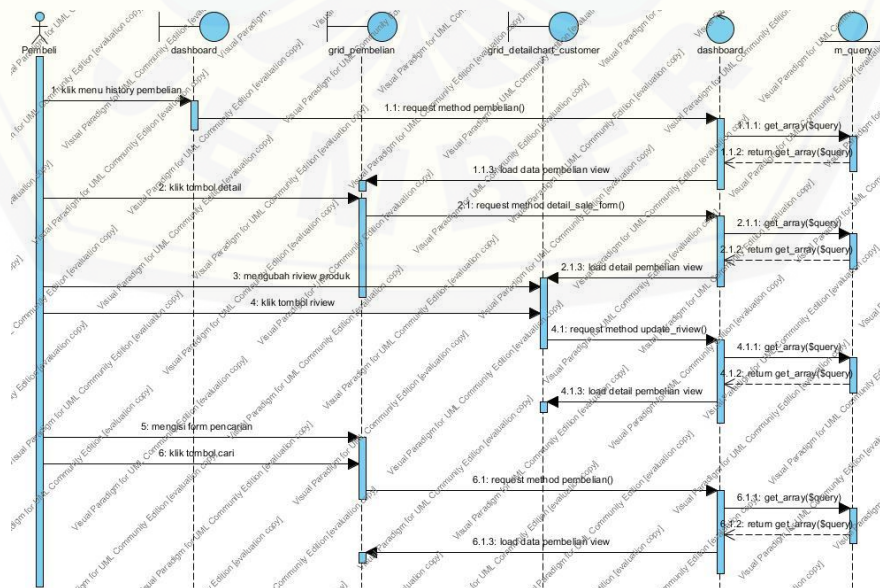
C.8 Sequence Diagram Mengolah Profil



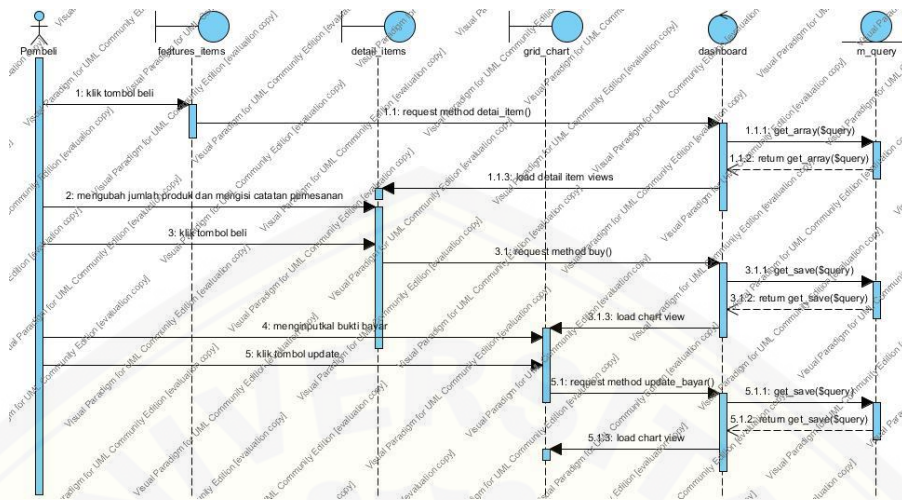
C.9 Sequence Diagram Mengolah Keranjang



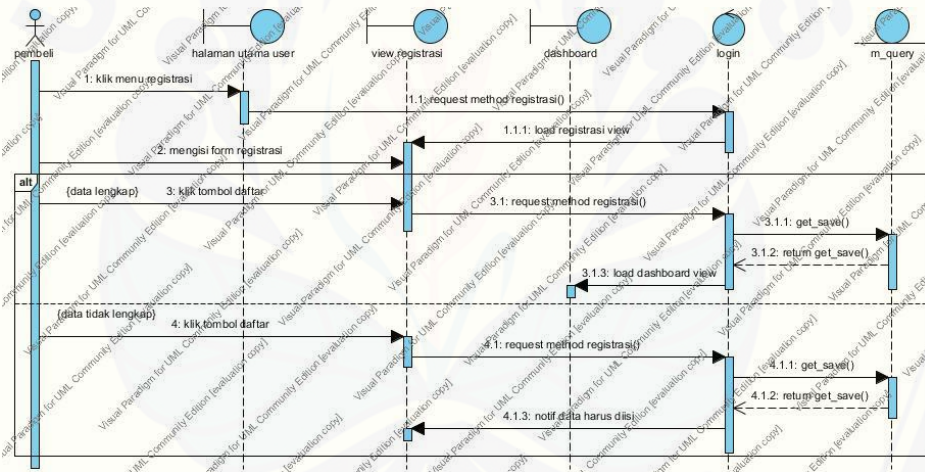
C.10 Sequence Diagram Mengolah History Pembelian



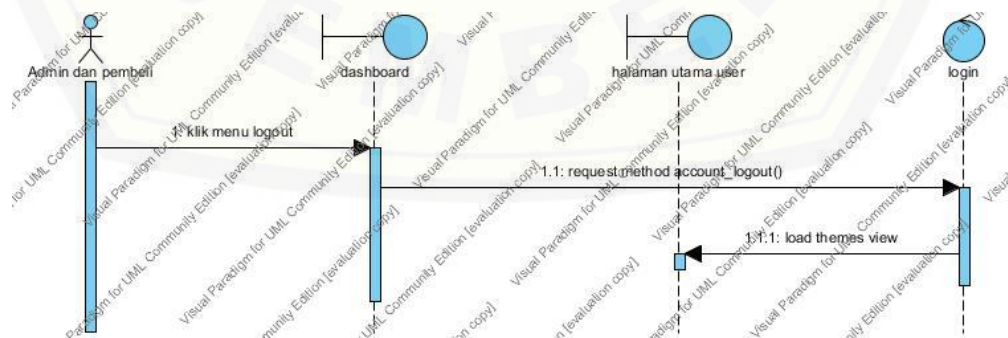
C.11 Sequence Diagram Pemesanan



C.12 Sequence Diagram Registrasi



C.13 Sequence Diagram Logout



Lampiran D. Lampiran Pengujian White BoxD.1 *Function* account_login()

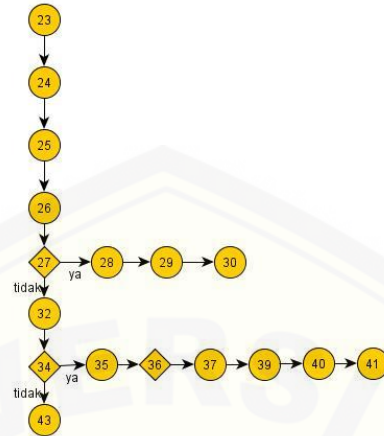
a. Listing Program

```

23 public function account_login(){
24     $Email          = $this->m_query->replace($this->input-
        >post("Email"));
25     $Password      = $this->m_query->replace($this->input->post("Password"));
26     $account       = $this->m_query->get_array("SELECT * FROM
        `administrator` WHERE Email='".$Email.'" and
        Password=right(password('$Password'),12)");
27     if($account->num_rows()>=1){
28         $this->session->set_userdata("IdUser",$account->row()-
            >IdAdministrator);
29         $this->session->set_userdata("Level",'Admin');
30         echo "<script type=\"text/javascript\">alert('Login Success');
            window.location.href=\".site_url()\";.</script>";
31     }else{
32         $account2 = $this->m_query->get_array("SELECT * FROM `customer`
            WHERE Email='".$Email.'" and
            Password=right(password('$Password'),12)
33                                     or
            LPass=right(password('$Password'),12)");
34         if($account2->num_rows()>=1){
35             $LPass = $account2->row()->LPass;
36             if($LPass!=""){
37                 $this->db->query("UPDATE customer set Password='".$LPass."',
                    LPass=NULL WHERE IdCustomer='".$account2->row()-
                    >IdCustomer.'");
38             }
39             $this->session->set_userdata("IdUser",$account2->row()-
                >IdCustomer);
40             $this->session->set_userdata("Level",'Customer');
41             echo "<script type=\"text/javascript\">alert('Login Success');
                window.location.href=\".site_url()\";.</script>";
42         }else{
43             echo "<script type=\"text/javascript\">alert('Login Failed');
                window.location.href=\".site_url('login')\";.</script>";
44         }
45     }
46 }
47 }

```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 16 - 17 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 9 – 10 – 11 – 12 – 13 – 14 – 15 - 16

Jalur 3 : 1 – 2 – 3 – 4 – 5 – 9 – 10 – 11 – 17

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Login Admin	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8	[√] berhasil [] gagal
2	Login Pembeli	1 – 2 – 3 – 4 – 5 – 9 – 10 – 11 – 12 – 13 – 14 – 15 - 16	1 – 2 – 3 – 4 – 5 – 9 – 10 – 11 – 12 – 13 –	[√] berhasil [] gagal

			14 - 15 - 16	
--	--	--	--------------	--

D.2 *Function* account_logout()

a. Listing Program

```

49 public function account_logout(){
50     $this->session->unset_userdata("IdUser");
51     $this->session->unset_userdata("Level");
52     $this->session->sess_destroy();
53     redirect("");
54 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 4 - 5 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 - 2 - 3 - 4 - 5

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Logout Admin dan Pembeli	1 - 2 - 3 - 4 - 5	1 - 2 - 3 - 4 - 5	[√] berhasil [] gagal

D.3 *Function* batal()

a. Listing Program

```

315 public function batal(){
316     $this->session->unset_userdata("Id_ProdukTadi");
317     $this->session->unset_userdata("QtyTadi");
318     $this->session->unset_userdata("KeteranganTadi");
319 }
    
```

b. Diagram Alir



c. Cyclomatic Complexcity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 3 - 4 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Membatalkan Pemesanan Produk	1 – 2 – 3 – 4	1 – 2 – 3 – 4	[√] berhasil [] gagal

D.4 *Function* cart()

a. Listing Program

```

144 public function cart(){
145     $NoOrder = $this->session->userdata('NoOrder');
146     $order = "select
147     mor.IdOrderItem,
148     mor.IdOrder,
149     o.NoOrder,
150     o.Alamat,
    
```

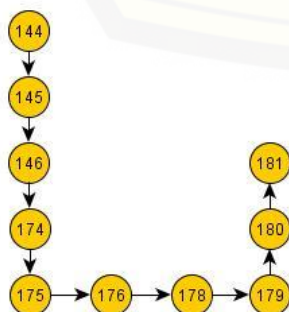


```

151     o.Kota,
152     o.Keterangan as KetOrder,
153     o.Phone,
154     o.BuktiBayar,
155     o.Shipping,
156     o.Resi,
157     o.KetToko,
158     p.KdProduk,
159     p>Nama as NamaProduk,
160     p.Stok,
161     o.Status,
162     mor.Qty,
163     mor.Harga,
164     mor.Keterangan,
165     p.Deskripsi as KetProduk,
166     p.Foto
167 from
168 `orderitem` mor
169 inner join
170 `order` o ON o.IdOrder = mor.IdOrder
171 inner join
172 produk p ON p.IdProduk = mor.IdProduk
173 WHERE o.IdCustomer="".$this->session->userdata('idUser')." and
        o.`Status`='O';
174 $data['order']           = $this->m_query->get_array("$order");
175 $data['qorder']         = $this->m_query->get_array("$order and
        mor.`Status`='O'")->result();
176 $data['order']           = "$order group by mor.IdOrder";
177
178 $data['header']         = "";
179 $data['page']           = "dashboard/dashboard";
180 $data['content']        = "data/grid_chart";
181 $this->load->view('themes',$data);
182 }

```

b. Diagram Alir



c. Cyclomatic Complexcity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 9 - 10 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Melihat Keranjang Belanja	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10	[√] berhasil [] gagal

D.5 Function buy()

a. Listing Program

```

63 public function buy(){
64     $this->session->unset_userdata('Alert_Lebih');
65     $this->session->unset_userdata('Kurang');
66     $IdProduk = $this->uri->segment(3);
67     $qty      = $this->input->post('qty');
68     $Keterangan = $this->m_query->replace($this->input->post('Keterangan'));
69     $Ongkir    = $this->m_query->ongkir_item();
70     $barang    = $this->m_query->get_array("select * from produk WHERE IdProduk='$IdProduk'");
71     if($qty<=0){
72         $this->session->set_userdata('Kurang','OK');
73         redirect("dashboard/detail_item/".$IdProduk);
74     }else{
75
76         if($this->session->userdata('IdUser')=="" || $this->session->userdata('Level')!="Customer"){
77             $this->session->set_userdata("Id_ProdukTadi", $IdProduk);
78             $this->session->set_userdata("QtyTadi", $qty);
79             $this->session->set_userdata("KeteranganTadi", $Keterangan);
80             redirect("login");
81         }else{

```

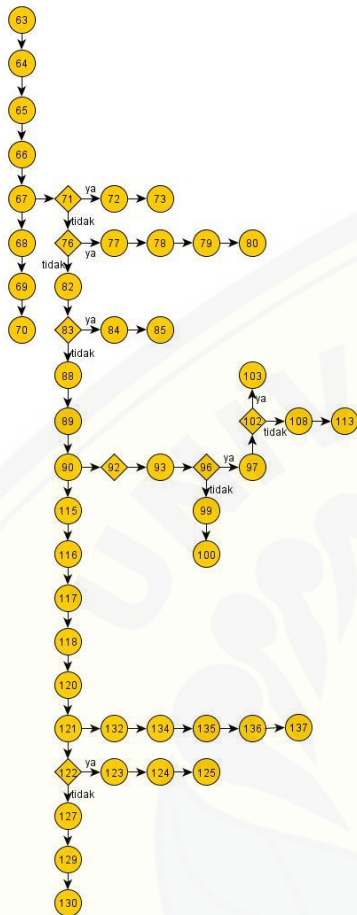
```

82     $Stok      = $barang->row()->Stok-$qty;
83     if($Stok<0){
84         $this->session->set_userdata('Alert_Lebih','OK');
85         redirect('dashboard/detail_item/'.$IdProduk);
86     }else{
87
88         $IdUser = $this->session->userdata('IdUser');
89         $order   = $this->m_query->get_array("select * from `order`
          order by IdOrder desc");
90         $blanja  = $this->m_query->get_array("select * from `order`
          WHERE IdCustomer='".$IdUser.'" and Status='O'");
91
92         if($blanja->num_rows()>0){
93             $this->session->set_userdata('NoOrder', $blanja->row()-
              >NoOrder);
94         }
95
96         if($order->num_rows()==0){
97             $NoOrder= "ORD".$this->m_query->nol(1);
98         }else{
99             $No = explode("ORD",$order->row()->NoOrder);
100            $NoOrder= "ORD".$this->m_query->nol($No[1]+1);
101        }
102        if($this->session->userdata('NoOrder')== ""){
103            $query   = "insert into `order` set
104            IdCustomer='".$IdUser."',
105            Shipping='".$Ongkir."',
106            NoOrder='$NoOrder'";
107        }else{
108            $query   = "update `order` set
109            Shipping='".$Ongkir."',
110            IdCustomer='".$IdUser.'"
111            where
112            NoOrder='".$this->session->userdata('NoOrder')."';
113            $NoOrder= $this->session->userdata('NoOrder');
114        }
115        $this->m_query->get_save("$query");
116        $this->session->set_userdata("NoOrder", $NoOrder);
117        $order   = $this->m_query->get_array("select * from `order`
          WHERE NoOrder='$NoOrder' and Status='O'");
118        $IdOrder= $order->row()->IdOrder;
119
120        $Harga   = $barang->row()->Harga;
121        $item= $this->m_query->get_array("select * from `orderitem`
          WHERE IdOrder='$IdOrder' and IdProduk='$IdProduk' and
          Status='O'");

```

```
122     if($item->num_rows()==0){
123         $Stok    = $barang->row()->Stok-$qty;
124         $qitem   = "insert into `orderitem` set IdOrder='$IdOrder',
                IdProduk='$IdProduk', Qty='$qty', Harga='$Harga',
                Keterangan='$Keterangan'";
125         $this->m_query->get_array("update produk set
                Stok='".($Stok)." WHERE IdProduk='$IdProduk'");
126     }else{
127         $qitem   = "update `orderitem` set IdOrder='$IdOrder',
                IdProduk='$IdProduk', Qty='$qty', Harga='$Harga',
                Keterangan='$Keterangan',
128         Status=".". $item->row()->Status." WHERE
                IdOrderItem=".". $item->row()->IdOrderItem."";
129         $Stok    = $barang->row()->Stok+$item->row()->Qty-$qty;
130         $this->m_query->get_array("update produk set
                Stok=".".($Stok)." WHERE IdProduk='$IdProduk'");
131     }
132     $this->m_query->get_save("$qitem");
133
134     $this->session->unset_userdata("Id_ProdukTadi");
135     $this->session->unset_userdata("QtyTadi");
136     $this->session->unset_userdata("KeteranganTadi");
137     redirect('dashboard/cart');
138
139     }
140 }
141 }
142 }
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 50 - 51 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 9 – 10 – 11

Jalur 3 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 13 – 14 – 15 – 16

Jalur 4 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 17 – 18 – 19 – 20

Jalur 5 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 17 – 18 – 21 – 22 – 23 – 24 – 25 – 26 – 27 – 30 – 31

Jalur 6 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 17 – 18 – 21 – 22 – 23 – 24 – 25 – 26 – 28 – 29

Jalur 7 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 17 – 18 – 21 – 22 – 23 – 24 – 25 – 26 – 27 – 30 – 32 – 33

Jalur 8 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 17 – 18 – 21 – 22 – 23 – 34 – 35 – 36 – 37 – 38 – 39 – 40 – 41 – 42 – 43

Jalur 9 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 17 – 18 – 21 – 22 – 23 – 34 – 35 – 36 – 37 – 38 – 39 – 40 – 44 – 45 – 46

Jalur 10 : 1 – 2 – 3 – 4 – 5 – 9 – 12 – 17 – 18 – 21 – 22 – 23 – 34 – 35 – 36 – 37 – 38 – 39 – 47 – 48 – 49 – 50 – 51

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Membeli atau memesan produk	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8	[√] berhasil [] gagal

D.6 Function customer_form()

a. Listing Program

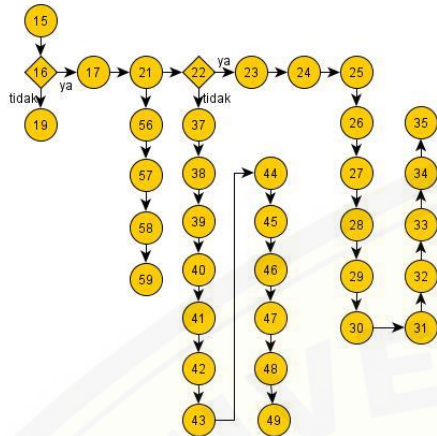
15	public function customer_form(){
16	if(\$this->session->userdata("Level")== 'Customer'){
17	\$IdCustomer = \$this->session->userdata("IdUser");
18	}else{
19	\$IdCustomer = \$this->uri->segment(3);
20	}
21	\$customer = \$this->m_query->get_save("select * from customer WHERE IdCustomer=\$IdCustomer");
22	if(\$customer->num_rows()>=1){
23	\$data['IdCustomer'] = \$customer->row()->IdCustomer;
24	\$data['Nama'] = \$customer->row()->Nama;
25	\$data['Email'] = \$customer->row()->Email;
26	\$data['Password'] = \$customer->row()->Password;

```

27 $data['Phone'] = $customer->row()->Phone;
28 $data['Alamat'] = $customer->row()->Alamat;
29 $data['provinsiId'] = $customer->row()->provinsiId;
30 $data['kabupatenId'] = $customer->row()->kabupatenId;
31 $data['kecamatanId'] = $customer->row()->kecamatanId;
32 $data['desaId'] = $customer->row()->desaId;
33 $data['TglRegistrasi'] = $this->m_query->tgl($customer->row()-
    >TglRegistrasi);
34 $data['NB'] = $customer->row()->NB;
35 $data['TglBloced'] = $customer->row()->TglBloced;
36 }else{
37 $data['IdCustomer'] = "";
38 $data['Nama'] = "";
39 $data['Email'] = "";
40 $data['Password'] = "";
41 $data['Phone'] = "";
42 $data['Alamat'] = "";
43 $data['provinsiId'] = "";
44 $data['kabupatenId'] = "";
45 $data['kecamatanId'] = "";
46 $data['desaId'] = "";
47 $data['TglRegistrasi'] = "";
48 $data['NB'] = "";
49 $data['TglBloced'] = "";
50 }
51
52 $data['provinsi'] = $this->m_query->get_save("SELECT * FROM
    `provinsi` ORDER BY `provinsiNama` ASC")->result();
53 $data['kabupaten'] = $this->m_query->get_save("SELECT * FROM
    `kabupaten` WHERE provinsiId='".$data['provinsiId']."' ORDER BY
    `kabupatenNama` ASC")->result();
54 $data['kecamatan'] = $this->m_query->get_save("SELECT * FROM
    `kecamatan` WHERE kabupatenId='".$data['kabupatenId']."' ORDER
    BY `kecamatanNama` ASC")->result();
55 $data['desa'] = $this->m_query->get_save("SELECT * FROM `desa`
    WHERE kecamatanId='".$data['kecamatanId']."' ORDER BY
    `desaNama` ASC")->result();
56 $data['header'] = "";
57 $data['page'] = "dashboard/dashboard";
58 $data['content'] = "data/form_customer";
59 $this->load->view('themes',$data);
60 }

```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 35 - 36 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15 – 16 – 17 – 18 – 19

Jalur 2 : 1 – 2 – 4

Jalur 3 : 1 – 2 – 3 – 5 – 6 – 20 – 21 – 22 – 23 – 24 – 25 – 26 – 27 – 28 – 29 – 30 – 31 – 32

Jalur 4 : 1 – 2 – 3 – 5 – 33 – 34 – 35 – 36

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Mengolah data profil	1 – 2 – 3 – 5 – 33 – 34 – 35 – 36	1 – 2 – 3 – 5 – 33 – 34 – 35 – 36	[√] berhasil [] gagal

D.7 *Function* delete_inventory()

a. Listing Program

```

249 public function delete_inventory(){
250     $IdProduk = $this->uri->segment(3);
251     $this->db->query("DELETE FROM `orderitem` WHERE
        `IdProduk`='".$IdProduk.'");
252     $this->db->query("DELETE FROM `produk` WHERE
        `IdProduk`='".$IdProduk.'");
253     redirect("data/inventory");
254 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 4 - 5 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menghapus data produk	1 – 2 – 3 – 4 – 5	1 – 2 – 3 – 4 – 5	[√] berhasil [] gagal

D.8 *Function* delete_item()

a. Listing Program

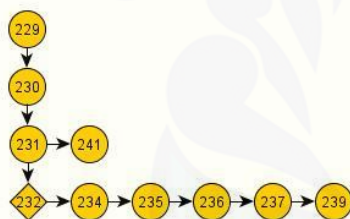
```

229 public function delete_item(){
    
```

```

230 $IdOrderItem = $this->uri->segment(3);
231 $item = $this->m_query->get_array("select * from `orderitem`
    WHERE IdOrderItem='$IdOrderItem'");
232 if($item->num_rows()>=1){
233
234     $IdProduk = $item->row()->IdProduk;
235     $barang = $this->m_query->get_array("select * from produk
    WHERE IdProduk='$IdProduk'");
236     $Stok = $barang->row()->Stok+$item->row()->Qty;
237     $this->m_query->get_array("update produk set Stok='".$($Stok)."'
    WHERE IdProduk='$IdProduk'");
238
239     $this->m_query->get_save("delete from `orderitem` where
    IdOrderItem='$IdOrderItem'");
240 }
241 redirect("dashboard/cart");
242 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 9 - 10 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

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

Jalur 2 : 1 - 2 - 3 - 10

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menghapus data penjualan	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10	[√] berhasil [] gagal

D.9 *Function* detail_item()

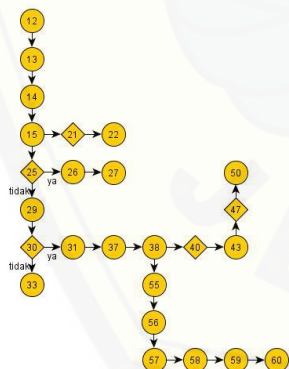
a. Listing Program

12	public function detail_item(){
13	\$this->session->unset_userdata('NoOrder');
14	\$IdProduk = \$this->uri->segment(3);
15	\$sqproduk = \$this->m_query->get_array("SELECT
16	mp.*, mkp.Kategori AS NKategori
17	FROM
18	`produk` mp
19	LEFT JOIN
20	kategoriproduk mkp ON mkp.IdKategori = mp.Kategori where
21	mp.IdProduk='".\$IdProduk.'");
22	if(\$sqproduk->num_rows()<=0){
23	redirect("");
24	}
25	if(\$this->session->userdata('Level')== "Admin"){
26	\$user = \$this->m_query->get_array("select * from administrator
27	WHERE IdAdministrator='".\$this->session->
28	userdata('IdUser').'");->row());
29	\$Nama = \$user->Nama;
30	}else{
31	\$user = \$this->m_query->get_array("select * from customer
32	WHERE IdCustomer='".\$this->session->userdata('IdUser').'");
33	if(\$user->num_rows()>0){
34	\$Nama = \$user->row()->Nama;
35	}else{
36	\$Nama = "";
37	}
38	}
39	\$data['NameLogin'] = \$Nama;
40	\$data['produk'] = \$sqproduk->row();

```

39
40     if($this->session->userdata('Alert_Lebih')== "OK"){
41         ?>
42         <script>
43             alert('Maaf Stok tidak mencukupi');
44         </script>
45         <?php
46     }
47     if($this->session->userdata('Kurang')== "OK"){
48         ?>
49         <script>
50             alert('Maaf pembelian tidak boleh kurang dari 1');
51         </script>
52         <?php
53     }
54
55     $data['header']          = "";
56     $data['page']           = "dashboard/dashboard";
57     $data['content']        = "dashboard/detail_items";
58     $this->load->view('themes',$data);
59     $this->session->unset_userdata('Alert_Lebih');
60     $this->session->unset_userdata('Kurang');
61 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 24 - 25 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6

Jalur 2 : 1 – 2 – 3 – 4 – 7 – 8 – 9

Jalur 3 : 1 – 2 – 3 – 4 – 7 – 10 – 11 – 12 – 14 – 15 – 16 – 17 – 18 – 19

Jalur 4 : 1 – 2 – 3 – 4 – 7 – 10 – 11 – 13

Jalur 5 : 1 – 2 – 3 – 4 – 7 – 10 – 11 – 12 – 14 – 15 – 20 – 21 – 22 – 23 – 24 – 25

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Melihat detail produk	1 – 2 – 3 – 4 – 5 – 6	1 – 2 – 3 – 4 – 5 – 6	[√] berhasil [] gagal

D.10 *Function* detail_sale_form() pada dashboard

a. Listing Program

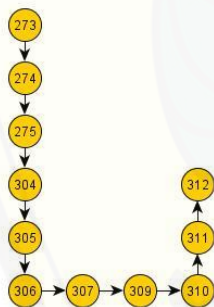
273	public function detail_sale_form(){
274	\$IdOrder = \$this->uri->segment(3);
275	\$order = "select
276	mor.IdOrderItem,
277	mor.IdOrder,
278	o.NoOrder,
279	o.Alamat,
280	o.Kota,
281	o.Keterangan as KetOrder,
282	o.Phone,
283	o.BuktiBayar,
284	o.Resi,
285	o.Shipping,
286	o.KetToko,
287	p.KdProduk,
288	p>Nama as NamaProduk,
289	p.Stok,
290	o.Status,
291	o.Review,
292	mor.Qty,

```

293     mor.Harga,
294     mor.Keterangan,
295     p.Deskripsi as KetProduk,
296     p.Foto
297 from
298     `orderitem` mor
299     inner join
300     `order` o ON o.IdOrder = mor.IdOrder
301     inner join
302     produk p ON p.IdProduk = mor.IdProduk
303     WHERE mor.IdOrder="".$IdOrder."";
304     $data['IdOrder']    = $IdOrder;
305     $data['order']      = $this->m_query->get_array("$order");
306     $data['qorder']    = $this->m_query->get_array("$order")-
307         >result();
308     $data['order']      = "$order group by mor.IdOrder";
309     $data['header']    = "";
310     $data['page']      = "dashboard/dashboard";
311     $data['content']   = "data/grid_detailchart_costomer";
312     $this->load->view('themes',$data);
313 }

```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 10 - 11 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

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

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Melihat detail history pembelian	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11	[√] berhasil [] gagal

D.11 *Function* detail_sale_form() pada sale

a. Listing Program

```

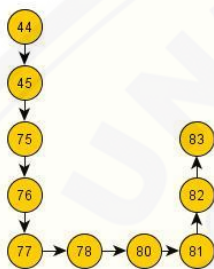
42 public function detail_sale_form(){
43     $IdOrder = $this->uri->segment(3);
44     $order = "select
45     mor.IdOrderItem,
46     mor.IdOrder,
47     o.NoOrder,
48     o.Alamat,
49     o.Kota,
50     o.Keterangan as KetOrder,
51     o.Phone,
52     o.BuktiBayar,
53     o.Shipping,
54     o.Resi,
55     o.Review,
56     o.KetToko,
57     p.KdProduk,
58     p>Nama as NamaProduk,
59     p.Stok,
60     o.Status,
61     mor.Qty,
62     mor.Harga,
63     mor.Keterangan,
64     p.Deskripsi as KetProduk,
65     p.Foto
66     from
67     orderitem mor
68     inner join
69     `order` o ON o.IdOrder = mor.IdOrder
70     inner join
71     produk p ON p.IdProduk = mor.IdProduk
72     WHERE mor.IdOrder='".$IdOrder.'";
73     $data['IdOrder'] = $IdOrder;
74     $data['order'] = $this->m_query->get_array("$order");
    
```



```

75 $data['qorder']      = $this->m_query->get_array("$order")-
    >result();
76 $data['order']      = "$order group by mor.IdOrder";
77
78 $data['header']     = "";
79 $data['page']       = "dashboard/dashboard";
80 $data['content']    = "data/grid_detailchart";
81 $this->load->view('themes',$data);
82 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 9 - 10 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Melihat detail data penjualan	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10	[√] berhasil [] gagal

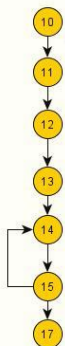
D.12 *Function* get_kabupaten()

a. Listing Program

```

10 public function get_kabupaten(){
11     $Id_Provinsi= $this->uri->segment(3);
12     $query = $this->m_query->get_array("SELECT * FROM `kabupaten`
        WHERE provinsiId='".$Id_Provinsi.'"")->result();
13     $data = "<option value=" selected=" disabled=">Pilih
        Kabupaten</option>";
14     foreach($query as $kkab => $vkab){
15         $data .= "<option value='".$vkab->kabupatenId.'">".$vkab-
            >kabupatenNama."</option>";
16     }
17     echo $data;
18 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 7 - 7 + 2 \\
 &= 2
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6 – 7

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan

1	Menampilkan Kabupaten	1 - 2 - 3 - 4 - 5 - 6 - 7	1 - 2 - 3 - 4 - 5 - 6 - 7	<input checked="" type="checkbox"/> berhasil <input type="checkbox"/> gagal
---	-----------------------	------------------------------	------------------------------	--

D.13 *Function* get_kecamatan()

a. Listing Program

```

20 public function get_kecamatan(){
21     $kabupatenId= $this->uri->segment(3);
22     $query = $this->m_query->get_array("SELECT * FROM `kecamatan`
        WHERE kabupatenId='".$.$kabupatenId.'"");->result();
23     $data = "<option value=" selected=" disabled=">Pilih
        Kecamatan</option>";
24     foreach($query as $kkab => $vkab){
25         $data .= "<option value=".$vkab->kecamatanId.">".$vkab-
            >kecamatanNama."</option>";
26     }
27     echo $data;
28 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 7 - 7 + 2 \\
 &= 2
 \end{aligned}$$

d. Jalur Independen

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

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan Kecamatan	1 – 2 – 3 – 4 – 5 – 6 – 7	1 – 2 – 3 – 4 – 5 – 6 – 7	[√] berhasil [] gagal

D.14 *Function get_kelurahan()*

a. Listing Program

```

30 public function get_kelurahan(){
31     $kecamatanId= $this->uri->segment(3);
32     $query = $this->m_query->get_array("SELECT * FROM `desa`
        WHERE kecamatanId='".$kecamatanId.'"")->result();
33     $data = "<option value=" selected=" disabled=">Pilih
        Kelurahan</option>";
34     foreach($query as $kkab => $vkab){
35         $data .= "<option value='".$vkab->desaId.'">".$vkab-
            >desaNama."</option>";
36     }
37     echo $data;
38 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 7 - 7 + 2 \\
 &= 2
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6 – 7

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan Kelurahan	1 – 2 – 3 – 4 – 5 – 6 – 7	1 – 2 – 3 – 4 – 5 – 6 – 7	[√] berhasil [] gagal

D.15 *Function* inventory_form()

a. Listing Program

```

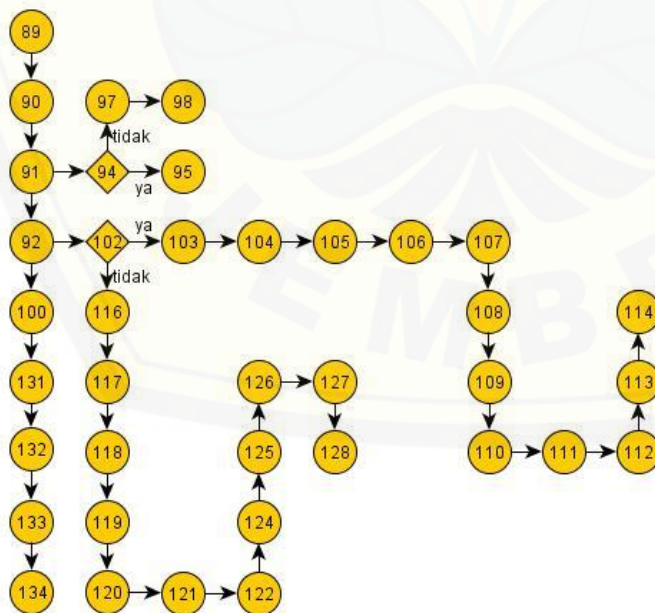
89 public function inventory_form(){
90     $IdProduk = $this->uri->segment(3);
91     $produk_new = $this->m_query->get_array("select * from produk
        ORDER BY IdProduk desc");
92     $produk = $this->m_query->get_array("select * from produk WHERE
        IdProduk='".$IdProduk.'");
93
94     if($produk_new->num_rows()<=0){
95         $No = 1;
96     }else{
97         $Kd = explode("B",$produk_new->row()->KdProduk);
98         $No = ($Kd[1]+1);
99     }
100     $KdProduk = "B".$this->m_query->nol($No);
101
102     if($produk->num_rows()>=1){
103         $data['IdProduk'] = $produk->row()->IdProduk;
104         $data['KdProduk'] = $produk->row()->KdProduk;
105         $data['Nama'] = $produk->row()->Nama;
106         $data['Stok'] = $produk->row()->Stok;
107         $data['Harga'] = $produk->row()->Harga;
108         $data['Berat'] = $produk->row()->Berat;
109         $data['Deskripsi'] = $produk->row()->Deskripsi;
110         $data['Omset'] = $produk->row()->Omset;
111         $data['TenagaKerja'] = $produk->row()->TenagaKerja;
112         $data['Teknologi'] = $produk->row()->Teknologi;
113         $data['SpesifikasiKekhasan'] = $produk->row()-
    
```



```

114     >SpesifikasiKekhasan;
        $data['KetersediaanBahanBaku'] = $produk->row()-
        >KetersediaanBahanBaku;
115     }else{
116         $data['IdProduk'] = "";
117         $data['KdProduk'] = $KdProduk;
118         $data['Nama'] = "";
119         $data['Stok'] = "";
120         $data['Harga'] = "";
121         $data['Deskripsi'] = "";
122         $data['Berat'] = "";
123
124         $data['Omset'] = "";
125         $data['TenagaKerja'] = "";
126         $data['Teknologi'] = "";
127         $data['SpesifikasiKekhasan'] = "";
128         $data['KetersediaanBahanBaku'] = "";
129     }
130
131     $data['header'] = "";
132     $data['page'] = "dashboard/dashboard";
133     $data['content'] = "data/form_inventory";
134     $this->load->view('themes',$data);
135 }
    
```

b. Diagram Alir



c. Cyclomatic Complexcity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 37 - 38 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 5 – 6

Jalur 2 : 1 – 2 – 3 – 5 – 7 – 8

Jalur 3 : 1 – 2 – 3 – 4 – 10 – 11 – 12 – 13 – 14 – 15 – 16 – 17 – 18 – 19 – 20 – 21 – 22

Jalur 4 : 1 – 2 – 3 – 4 – 10 – 23 – 24 – 25 – 26 – 27 – 28 – 29 – 30 – 31 – 32 – 33 – 34

Jalur 5 : 1 – 2 – 3 – 4 – 9 – 35 – 36 – 37 – 38

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan form data produk	1 – 2 – 3 – 4 – 9 – 35 – 36 – 37 – 38	1 – 2 – 3 – 4 – 9 – 35 – 36 – 37 – 38	[√] berhasil [] gagal

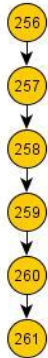
D.16 *Function* kriteria()

a. Listing Program

```

256 public function kriteria(){
257     $data['qkriteria'] = $this->db->query("SELECT * FROM vkriteria
        ORDER BY IdKriteria asc")->result();
258     $data['header'] = "";
259     $data['page'] = "dashboard/dashboard";
260     $data['content'] = "data/data_kriteria";
261     $this->load->view('themes',$data);
262 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 5 - 6 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan tabel data kriteria	1 – 2 – 3 – 4 – 5 – 6	1 – 2 – 3 – 4 – 5 – 6	[√] berhasil [] gagal

D.17 *Function* pembelian()

a. Listing Program

```

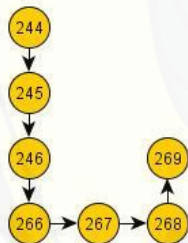
244 public function pembelian(){
245     $SrcSale = $this->m_query->replace($this->input-
        >post('SrcSale'));
246     $data['qsale'] = $this->m_query->get_array("select
247     mo . *, mc.Nama,
248     (select
249     sum(it.Qty * it.Harga)
250     from
    
```

```

251 `orderitem` it
252 where
253 it.IdOrder = mo.IdOrder) as Total_Belanja,
254 if(mo.Status='O','Order',if(mo.Status='T','Pembelian
    Ditolak',if(mo.Status='C','Batal','Pembayaran Telah Diterima')))) as
    SPembelian,
255 (select count(oi.IdOrderItem) from orderitem oi where
    oi.IdOrder=mo.IdOrder) as Kosong
256 from
257 `order` mo
258 left join
259 customer mc ON mc.IdCustomer = mo.IdCustomer
260 WHERE mo.IdCustomer="".$this->session->userdata('idUser').""
261 and (mo.NoOrder like "%".$SrcSale."%" or mo.NoOrder
262 like "%".$SrcSale."%" or mc>Nama like "%".$SrcSale."%")
263 having Total_Belanja is not null and Kosong is not null
264 ORDER BY mo.`TglBelanja` DESC
265 )->result();
266 $data['header'] = "";
267 $data['page'] = "dashboard/dashboard";
268 $data['content'] = "data/grid_pembelian";
269 $this->load->view('themes',$data);
270 }

```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 6 - 7 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6 – 7

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan tabel data history pembelian	1 – 2 – 3 – 4 – 5 – 6 – 7	1 – 2 – 3 – 4 – 5 – 6 – 7	[√] berhasil [] gagal

D.18 *Function* perhitungan_wp()

a. Listing Program

```

323 public function perhitungan_wp(){
324     $data['qkriteria'] = $this->db->query("SELECT * FROM vkriteria
        ORDER BY IdKriteria asc")->result();
325     $data['qproduk'] = $this->db->query("SELECT * FROM view_produk
        WHERE Stok>0 ORDER BY NVector desc")->result();
326     $data['header'] = "";
327     $data['page'] = "dashboard/dashboard";
328     $data['content'] = "data/data_perhitungan_wp";
329     $this->load->view('themes',$data);
330 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 6 - 7 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6 – 7

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan tabel data history pembelian	1 – 2 – 3 – 4 – 5 – 6 – 7	1 – 2 – 3 – 4 – 5 – 6 – 7	[√] berhasil [] gagal

D.19 *Function* registrasi()

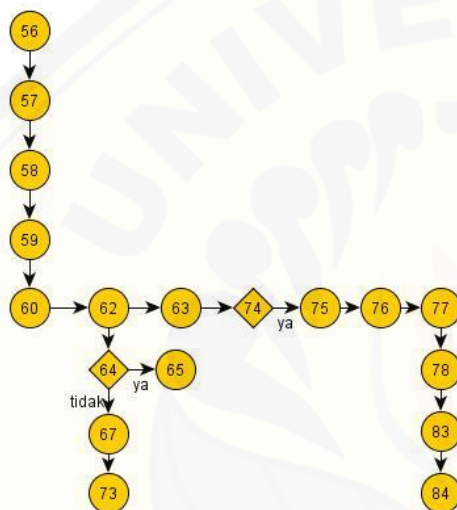
a. Listing Program

```

56 public function registrasi(){
57     $Name      = $this->m_query->replace($this->input->post("Name"));
58     $Email     = $this->m_query->replace($this->input->post("Email"));
59     $Password  = $this->m_query->replace($this->input->
        >post("Password"));
60     $Phone    = $this->m_query->replace($this->input->post("Phone"));
61
62     $admin = $this->m_query->get_array("SELECT * FROM
        `administrator` WHERE Email='".$Email.'");
63     $account = $this->m_query->get_array("SELECT * FROM `customer`
        WHERE Email='".$Email.'");
64     if($admin->num_rows()>=1 || $account->num_rows()>=1){
65         $data['ready'] = "Username sudah digunakan.";
66     }else{
67         $data['ready'] = "";
68         $this->m_query->get_save("insert into customer set
69         Nama='$Name',
70         Email='$Email',
71         Phone='$Phone',
72         Password=right(password('$Password'),12)");
73         $account= $this->m_query->get_array("SELECT * FROM
        `customer` WHERE Email='".$Email.'");
74         if($account){
75             $this->session->set_userdata("idUser",$account->row()-
                >IdCustomer);
76             $this->session->set_userdata("Level",'Customer');
77             $Pesan      = "Hai, ".$Name." akun berhasil didaftarkan di
    
```

	Penjualan Online. Username Login : ".\$Email." dan Password : ".\$Password;
78	redirect('customer/customer_form');
79	}
80	
81	}
82	
83	\$data['page'] = "login/login";
84	\$this->load->view('themes',\$data);
85	}

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 17 - 18 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

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

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

Jalur 3 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 12 - 13 - 14 - 15 - 16 - 17 - 18

e. Test Case

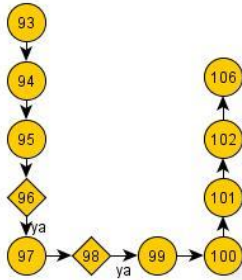
No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Registrasi pembeli	1 – 2 – 3 – 4 – 5 – 6 – 7 – 12 – 13 – 14 – 15 – 16 – 17 – 18	1 – 2 – 3 – 4 – 5 – 6 – 7 – 12 – 13 – 14 – 15 – 16 – 17 – 18	[√] berhasil [] gagal

D.20 *Function* reset_login()

a. Listing Program

93	public function reset_login(){
94	\$email = \$this->db->escape_str(\$this->input->post('Email'));
95	\$saccount = \$this->m_query->get_array("SELECT * FROM `customer` WHERE Email='".\$email."'");
96	if(\$saccount->num_rows()>0){
97	\$phone = \$saccount->row()->Phone;
98	if(\$phone!=""){
99	\$getPass = \$this->db->query("select right(Password(curtime()),5) as Pass")->row()->Pass;
100	\$this->db->query("UPDATE customer SET LPass=right(password("'" . \$getPass . "'),12) WHERE Email='".\$email."'");
101	\$message = "Hai, " . \$saccount->row()->Nama . " Password Login Online Shop Pondok Gunung sudah direset menjadi : " . \$getPass;
102	\$this->m_query->get_array("INSERT INTO outbox (DestinationNumber, TextDecoded, CreatorID) VALUES ('\$phone', '\$message', 'SMS GATEWAY')");
103	
104	}
105	}
106	redirect('login');
107	}

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 10 - 11 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

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

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan tabel data history pembelian	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11	[√] berhasil [] gagal

D.21 *Function save_customer()*

a. Listing Program

62	public function save_customer(){
63	\$IdCustomer = \$this->m_query->replace(\$this->input->post('IdCustomer'));
64	\$Nama = \$this->m_query->replace(\$this->input->post('Nama'));
65	>Email = \$this->m_query->replace(\$this->input->post('Email'));
66	>Password = \$this->m_query->replace(\$this->input->post('Password'));
67	\$Phone = \$this->m_query->replace(\$this->input->post('Phone'));
68	\$Alamat = \$this->m_query->replace(\$this->input->post('Alamat'));
69	\$provinsiId = \$this->m_query->replace(\$this->input->post('provinsiId'));

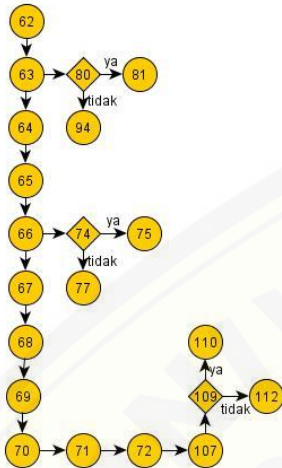
```

70 $kabupatenId= $this->m_query->replace($this->input-
    >post('kabupatenId'));
71 $kecamatanId= $this->m_query->replace($this->input-
    >post('kecamatanId'));
72 $desaId = $this->m_query->replace($this->input->post('desaId'));
73
74 if($Password==""){
75     $Password= "";
76 }else{
77     $Password= " , Password=right(password('$Password'),12)";
78 }
79
80 if($IdCustomer==""){
81     $query = "insert into customer set
82     Nama='$Nama',
83     Email='$Email',
84     $Password
85     Phone='$Phone',
86     Alamat='$Alamat',
87     provinsiId='$provinsiId',
88     kabupatenId='$kabupatenId',
89     kecamatanId='$kecamatanId',
90     desaId='$desaId'
91
92     ";
93 }else{
94     $query = "update customer set
95     Nama='$Nama',
96     Email='$Email',
97     $Password
98     Phone='$Phone',
99     Alamat='$Alamat',
100     provinsiId='$provinsiId',
101     kabupatenId='$kabupatenId',
102     kecamatanId='$kecamatanId',
103     desaId='$desaId'
104
105     WHERE IdCustomer='$IdCustomer'";
106 }
107 $this->m_query->get_save("$query");
108
109 if($this->session->userdata("Level")==='Customer'){
110     redirect("customer/customer_form");
111 }else{
112     redirect("customer");
113 }

```


114	}
115	

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 20 - 21 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 18 - 19 - 20

Jalur 2 : 1 - 2 - 3 - 4 - 5 - 12 - 13

Jalur 3 : 1 - 2 - 3 - 4 - 5 - 12 - 14

Jalur 4 : 1 - 2 - 15 - 16

Jalur 5 : 1 - 2 - 15 - 17

Jalur 6 : 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 18 - 19 - 21

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menyimpan profil	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 18 – 19 – 20	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 18 – 19 – 20	[√] berhasil [] gagal

D.22 *Function save_inventory()*

a. Listing Program

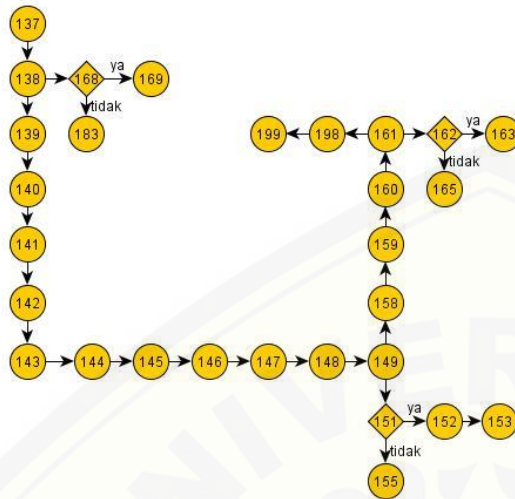
```

137 public function save_inventory(){
138     $IdProduk = $this->m_query->replace($this->input->post('IdProduk'));
139     $KdProduk = $this->m_query->replace($this->input-
        >post('KdProduk'));
140     $Nama      = $this->m_query->replace($this->input->post('Nama'));
141     $Berat    = $this->m_query->replace($this->input->post('Berat'));
142     $Stok     = $this->m_query->replace($this->input->post('Stok'));
143     $Harga   = $this->m_query->replace($this->input->post('Harga'));
144     $Deskripsi = $this->m_query->replace($this->input->post('Deskripsi'));
145     $Omset    = $this->m_query->replace($this->input->post('Kriteria1'));
146     $TenagaKerja = $this->m_query->replace($this->input-
        >post('Kriteria2'));
147     $Teknologi = $this->m_query->replace($this->input->post('Kriteria3'));
148     $SpesifikasiKekhasan = $this->m_query->replace($this->input-
        >post('Kriteria4'));
149     $KetersediaanBahanBaku = $this->m_query->replace($this->input-
        >post('Kriteria5'));
150
151     if($this->session->userdata("Level")==="Customer"){
152         $IdUser = $this->session->userdata("IdUser");
153         $IdCustomer = "IdCustomer=".$IdUser.", ";
154     }else{
155         $IdCustomer = "";
156     }
157
158     $file_upload= $_FILES["Foto"]["name"];
159     $file_tmp = $_FILES["Foto"]["tmp_name"];
160     $file_type = $_FILES["Foto"]["type"];
161     $tujuan    = "assets/foto/".$file_upload;
162     if(move_uploaded_file($file_tmp,$tujuan)){

```

```
163     $Foto = ", Foto=" . $this->m_query->replace($tujuan)."";
164 }else{
165     $Foto = "";
166 }
167
168 if($IdProduk==""){
169     $query = "insert into produk set
170     KdProduk='$KdProduk',
171     Nama='$Nama',
172     Berat='$Berat',
173     Stok='$Stok',
174     Harga='$Harga',
175     Omset='$Omset',
176     TenagaKerja='$TenagaKerja',
177     Teknologi='$Teknologi',
178     SpesifikasiKekhasan='$SpesifikasiKekhasan',
179     KetersediaanBahanBaku='$KetersediaanBahanBaku',
180     Deskripsi='$Deskripsi'
181     $Foto";
182 }else{
183     $query = "update produk set
184     KdProduk='$KdProduk',
185     Nama='$Nama',
186     Berat='$Berat',
187     Stok='$Stok',
188     Harga='$Harga',
189     Omset='$Omset',
190     TenagaKerja='$TenagaKerja',
191     Teknologi='$Teknologi',
192     SpesifikasiKekhasan='$SpesifikasiKekhasan',
193     KetersediaanBahanBaku='$KetersediaanBahanBaku',
194     Deskripsi='$Deskripsi'
195     $Foto
196     where IdProduk=" . $IdProduk . """;
197 }
198 $this->m_query->get_save("$query");
199 redirect("data/inventory");
200 }
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 28 - 29 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

- Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15 – 16
- Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 17
- Jalur 3 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 18 – 19 – 20 – 21 – 22 – 23
- Jalur 4 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 18 – 19 – 20 – 21 – 22 – 24
- Jalur 5 : 1 – 2 – 25 – 26
- Jalur 6 : 1 – 2 – 25 – 27
- Jalur 7 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 18 – 19 – 20 – 21 – 28 – 29

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menyimpan data produk	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 18 – 19 – 20 – 21 – 28 – 29	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 18 – 19 – 20 – 21 – 28 – 29	[√] berhasil [] gagal

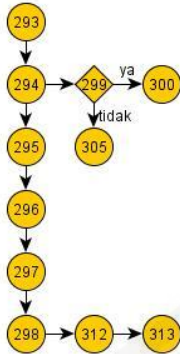
D.23 *Function save_subkriteria()*

a. Listing Program

```

293 public function save_subkriteria(){
294     $IdSubKriteria = $this->m_query->replace($this->input-
        >post('IdSubKriteria'));
295     $IdKriteria = $this->m_query->replace($this->input-
        >post('IdKriteria'));
296     $Kriteria = $this->m_query->replace($this->input->post('Kriteria'));
297     $SubKriteria = $this->m_query->replace($this->input-
        >post('SubKriteria'));
298     $Nilai = $this->m_query->replace($this->input->post('Nilai'));
299     if($IdSubKriteria==""){
300         $query = "INSERT INTO kriteria_sub SET
301             IdKriteria="."$IdKriteria.",
302             SubKriteria="."$SubKriteria.",
303             Nilai="."$Nilai.""";
304     }else{
305         $query = "UPDATE kriteria_sub SET
306             IdKriteria="."$IdKriteria.",
307             SubKriteria="."$SubKriteria.",
308             Nilai="."$Nilai."
309             WHERE IdSubKriteria="."$IdSubKriteria."
310             ";
311     }
312     $this->db->query($query);
313     redirect("data/kriteria");
314 }
    
```


b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 10 - 11 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6 – 10 – 11

Jalur 2 : 1 – 2 – 7 – 8

Jalur 3 : 1 – 2 – 7 – 9

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menyimpan data subkriteria	1 – 2 – 3 – 4 – 5 – 6 – 10 – 11	1 – 2 – 3 – 4 – 5 – 6 – 10 – 11	[√] berhasil [] gagal

D.24 *Function* simpan_wilayah()

a. Listing Program

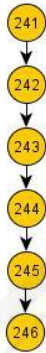
```

241 private function simpan_wilayah(){
242     $IdKecamatan = $this->input->post('IdKecamatan');
243     $Kecamatan = $this->input->post('Kecamatan');
244     $Ongkos = $this->input->post('Ongkos');
    
```

```

245     $this->m_query->get_array("update kecamatan set
        Ongkir=".$Ongkos." where kecamatanId=".$IdKecamatan."");
246     redirect("data/wilayah");
247 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 5 - 6 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5 – 6

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menyimpan data wilayah	1 – 2 – 3 – 4 – 5 – 6	1 – 2 – 3 – 4 – 5 – 6	[√] berhasil [] gagal

D.25 *Function* subkriteria_delete()

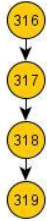
a. Listing Program

```

316 public function subkriteria_delete(){
317     $IdSubKriteria = $this->uri->segment(3);
318     $this->db->query("DELETE FROM kriteria_sub where
        IdSubKriteria=".$IdSubKriteria."");
    
```

```
319 | redirect("data/kriteria");
320 | }
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 3 - 4 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menghapus data subkriteria	1 – 2 – 3 – 4	1 – 2 – 3 – 4	[√] berhasil [] gagal

D.26 *Function* subkriteria_form()

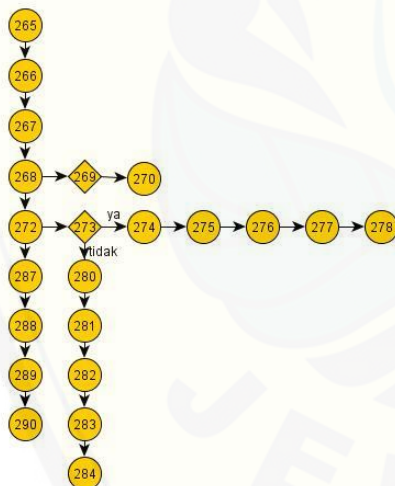
a. Listing Program

```
265 | public function subkriteria_form(){
266 |     $IdKriteria = $this->uri->segment(3);
267 |     $IdSubKriteria = $this->uri->segment(4);
268 |     $kriteria = $this->m_query->get_array("SELECT * FROM `kriteria`
      |         where IdKriteria='".$IdKriteria.'");
269 |     if($kriteria->num_rows()<=0){
270 |         redirect("data/kriteria");
271 |     }
272 |     $skriteria = $this->m_query->get_array("SELECT * FROM
      |         `kriteria_sub` where IdSubKriteria='".$IdSubKriteria.'");
```

```

273     if($skriteria->num_rows()<=0){
274         $data['IdSubKriteria'] = "";
275         $data['IdKriteria'] = $IdKriteria;
276         $data['Kriteria'] = $skriteria->row()->Kriteria;
277         $data['SubKriteria'] = "";
278         $data['Nilai'] = "";
279     }else{
280         $data['IdSubKriteria'] = $skriteria->row()->IdSubKriteria;
281         $data['IdKriteria'] = $skriteria->row()->IdKriteria;
282         $data['Kriteria'] = $skriteria->row()->Kriteria;
283         $data['SubKriteria'] = $skriteria->row()->SubKriteria;
284         $data['Nilai'] = $skriteria->row()->Nilai;
285     }
286
287     $data['header'] = "";
288     $data['page'] = "dashboard/dashboard";
289     $data['content'] = "data/form_subkriteria";
290     $this->load->view('themes',$data);
291 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 21 - 22 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 - 2 - 3 - 4 - 5 - 6

Jalur 2 : 1 - 2 - 3 - 4 - 7 - 8 - 9 - 10 - 11 - 12 - 13

Jalur 3 : 1 – 2 – 3 – 4 – 7 – 8 – 13 – 14 – 15 – 16 – 17

Jalur 4 : 1 – 2 – 3 – 4 – 18 – 19 – 20 – 21

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan tform subkriteria	1 – 2 – 3 – 4 – 7 – 8 – 9 – 10 – 11 – 12 – 13	1 – 2 – 3 – 4 – 7 – 8 – 9 – 10 – 11 – 12 – 13	[√] berhasil [] gagal

D.27 *Function* update_bayar() pada dashboard

a. Listing Program

```

184 public function update_bayar(){
185     $order = "select
186     mor.IdOrderItem,
187     mor.IdOrder,
188     o.NoOrder,
189     o.Alamat,
190     o.Kota,
191     o.Keterangan as KetOrder,
192     o.Phone,
193     mor.NoResi,
194     p.KdProduk,
195     p>Nama as NamaProduk,
196     p.Stok,
197     mor.Qty,
198     mor.Harga,
199     mor.Keterangan,
200     p.Deskripsi as KetProduk,
201     p.Foto
202     from
203     `orderitem` mor
204     inner join
205     `order` o ON o.IdOrder = mor.IdOrder
206     inner join
207     produk p ON p.IdProduk = mor.IdProduk
208     WHERE o.IdCustomer='".$this->session->userdata('idUser')." and
209     o.`Status`='O'";
209     if($this->m_query->get_array("$order")->num_rows()>=1){

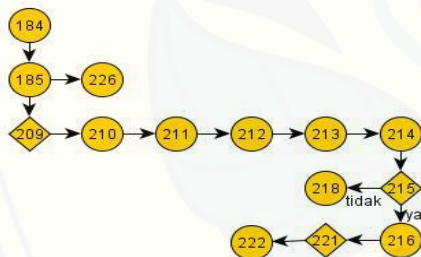
```



```

210 $Keterangan = $this->input->post('Keterangan');
211 $file_upload= $_FILES["BuktiBayar"]["name"];
212 $file_tmp = $_FILES["BuktiBayar"]["tmp_name"];
213 $file_type = $_FILES["BuktiBayar"]["type"];
214 $tujuan = "assets/buktibayar/".$file_upload;
215 if(move_uploaded_file($file_tmp,$tujuan)){
216 $BuktiBayar = "BuktiBayar=".$this->m_query-
    >replace($tujuan)."";
217 }else{
218 $BuktiBayar = "";
219 }
220
221 if($BuktiBayar!=""){
222 $this->m_query->get_array("update `order` set $BuktiBayar,
    Keterangan='$Keterangan'
223 WHERE IdOrder=".$this->m_query->get_array("$order")->row()-
    >IdOrder."");
224 }
225 }
226 redirect("dashboard/cart");
227 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 13 - 14 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 12 – 13

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 11

Jalur 3 : 1 – 2 – 14

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Mengupdate pembayaran customer	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 12 – 13	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 12 – 13	[√] berhasil [] gagal

D.28 *Function* update_bayar() pada sale

a. Listing Program

84	public function update_bayar(){
85	\$IdOrder = \$this->uri->segment(3);
86	\$order = \$this->m_query->get_array("select
867	mor.IdOrderItem,
88	mor.IdOrder,
89	o.NoOrder,
90	o.Alamat,
91	o.Kota,
92	o.Keterangan as KetOrder,
93	o.Phone,
94	o.BuktiBayar,
95	o.Shipping,
96	o.Resi,
97	o.KetToko,
98	p.IdProduk,
99	p.KdProduk,
100	p>Nama as NamaProduk,
101	p.Stok,
102	o.Status,
103	mor.Qty,
104	mor.Harga,
105	mor.Keterangan,
106	p.Deskripsi as KetProduk,
107	p.Foto,
108	c.Phone,
109	c>Nama
110	from
111	orderitem mor
112	inner join
113	`order` o ON o.IdOrder = mor.IdOrder
114	inner join

```

115     produk p ON p.IdProduk = mor.IdProduk
116     inner join
117     customer c ON c.IdCustomer = o.IdCustomer
118     WHERE mor.IdOrder="".$IdOrder.""->row());
119
120     $Shipping = $this->input->post('Shipping');
121     $Keterangan = $this->input->post('Keterangan');
122     $Status_Awal = $this->input->post('Status_Awal');
123     $Status_Pembelian = $this->input->post('Status_Pembelian');
124     $Resi_Awal = $this->input->post('Resi_Awal');
125     $Resi = $this->input->post('Resi');
126     $file_upload= $_FILES["BuktiBayar"]["name"];
127     $file_tmp = $_FILES["BuktiBayar"]["tmp_name"];
128     $file_type = $_FILES["BuktiBayar"]["type"];
129     $tujuan = "assets/buktibayar/".$file_upload;
130     if(move_uploaded_file($file_tmp,$tujuan)){
131         $BuktiBayar = "BuktiBayar="".$this->m_query-
            >replace($tujuan)."";
132     }else{
133         $BuktiBayar = "";
134     }
135
136     if($BuktiBayar!=""){
137         $this->m_query->get_array("update `order` set $BuktiBayar,
            KetToko='$Keterangan', Shipping='$Shipping',
138         Resi='$Resi', Status='$Status_Pembelian' WHERE
            IdOrder="".$IdOrder.""");
139     }else{
140         $this->m_query->get_array("update `order` set
            KetToko='$Keterangan', Shipping='$Shipping', Resi='$Resi',
            Status='$Status_Pembelian'
141         WHERE IdOrder="".$IdOrder.""");
142     }
143
144     switch($Status_Pembelian){
145     case "O":
146         $Status = "Order";
147         break;
148     case "C":
149         $Status = "diBatalkan";
150         break;
151     case "T":
152         $Status = "Pembayaran ditolak";
153         break;
154     case "S":
155         $Status = "Pembayaran Telah Diterima";

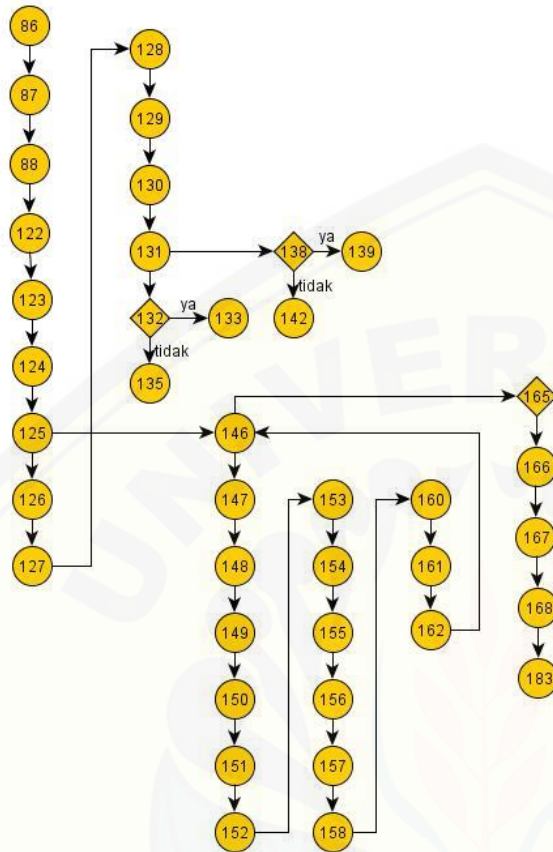
```

```

156     break;
157
158     default:
159         $Status="";
160         break;
161     }
162
163     if($Status_Pembelian=="C" || $Status_Pembelian=="T"){
164         $barang = $this->m_query->get_array("select * from produk
            WHERE IdProduk='".$order->IdProduk.'");
165         $Stok    = $barang->row()->Stok+$order->Qty;
166         $this->m_query->get_array("update produk set Stok='".$Stok.'"
            WHERE IdProduk='".$order->IdProduk.'");
167     }
168
169     $Pesan="";
170     $Nama    = $order->Nama;
171     if($Resi_Awal!=$Resi && $Status_Awal!=$Status_Pembelian){
172         $Pesan = "Hai, $Nama. Pembelian #".$order->NoOrder.", dengan
            status pembayaran : ".$Status." & No.Resi Pengiriman : ".$Resi;
173     }else if($Resi_Awal==$Resi && $Status_Awal!=$Status_Pembelian){
174         $Pesan = "Hai, $Nama. Pembelian #".$order->NoOrder.", dengan
            status pembayaran : ".$Status;
175     }else if($Resi_Awal!=$Resi && $Status_Awal==$Status_Pembelian){
176         $Pesan = "Hai, $Nama. Pembelian #".$order->NoOrder.", dikirim
            dengan No.Resi Pengiriman : ".$Resi;
177     }
178     $Phone    = $order->Phone;
179
180
181     redirect("sale/detail_sale_form/".$IdOrder);
182 }

```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned} \text{CC} &= \text{EDGE} - \text{NODE} + 2 \\ &= 40 - 40 + 2 \\ &= 2 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 16

Jalur 3 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 17 – 18

Jalur 4 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 17 – 19

Jalur 5 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 20 – 21 – 22 – 23 – 24 – 25 – 26 – 27 – 28 – 29
– 30 – 31 – 32 – 33 – 34 – 35

Jalur 6 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 20 – 36 – 37 – 38 – 39 – 40

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Mengupdate data pembayaran	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15 – 40	1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10 – 11 – 12 – 13 – 14 – 15 – 40	[√] berhasil [] gagal

D.29 *Function* update_riview()

a. Listing Program

```

321 public function update_review(){
322     $IdOrder = $this->input->post('IdOrder');
323     $Review = $this->db->escape_str($this->input->post('Review'));
324     $this->db->query("UPDATE `order` set Review='".$Review.'"
        WHERE IdOrder='".$IdOrder.'");
325     redirect("dashboard/detail_sale_form/".$IdOrder);
326 }
    
```

b. Diagram Alir



c. Cyclomatic Complexcity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 4 - 5 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur : 1 – 2 – 3 – 4 – 5

e. Test Case

No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Mengupdate riviw pembeli	1 – 2 – 3 – 4 – 5	1 – 2 – 3 – 4 – 5	[√] berhasil [] gagal

D.30 *Function* wilayah()

a. Listing Program

```

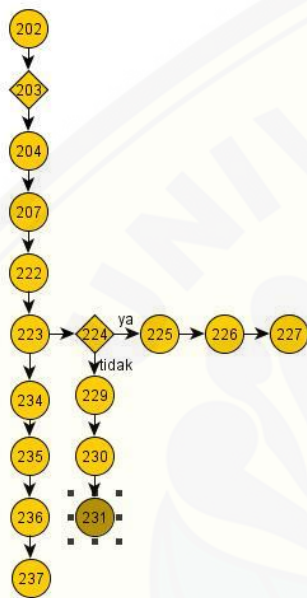
202 public function wilayah(){
203     if($this->input->post('submit')== "ok"){
204         $this->simpan_wilayah();
205     }
206
207     $data['qkecamatan'] = $this->m_query->get_array("select
208         pro.provinsiId,
209         pro.provinsiNama,
210         kab.kabupatenId,
211         kab.kabupatenNama,
212         kec.kecamatanId,
213         kec.kecamatanNama,
214         kec.Ongkir
215     from
216         kecamatan kec
217     inner join
218         kabupaten kab ON kab.kabupatenId = kec.kabupatenId
219     inner join
220         provinsi pro ON pro.provinsiId = kab.provinsiId")->result();
221
222     $IdKecamatan = $this->uri->segment(3);
223     $kec = $this->m_query->get_array("select * from kecamatan kec where
224         kec.kecamatanId='".$IdKecamatan.'"");
225     if($kec->num_rows()<=0){
226         $data['IdKecamatan'] = "";
227         $data['Kecamatan'] = "";
228         $data['Ongkos'] = "";
229     }else{
230         $data['IdKecamatan'] = $kec->row()->kecamatanId;
231         $data['Kecamatan'] = $kec->row()->kecamatanNama;
232         $data['Ongkos'] = $kec->row()->Ongkir;

```

```

232     }
233
234     $data['header']    = "";
235     $data['page']     = "dashboard/dashboard";
236     $data['content'] = "data/form_kecamatan";
237     $this->load->view('themes',$data);
238 }
    
```

b. Diagram Alir



c. Cyclomatic Complexity

$$\begin{aligned}
 CC &= \text{EDGE} - \text{NODE} + 2 \\
 &= 16 - 17 + 2 \\
 &= 1
 \end{aligned}$$

d. Jalur Independen

Jalur 1 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 6 – 7 – 11 – 12 – 13

Jalur 3 : 1 – 2 – 3 – 4 – 5 – 6 – 14 – 15 – 16 – 17

e. Test Case


No.	Pengujian	Jalur yang diharapkan	Jalur hasil pengamatan	Keterangan
1	Menampilkan alamat dan ongkos kirim sesuai alamat	1 – 2 – 3 – 4 – 5 – 6 – 14 – 15 – 16 – 17	1 – 2 – 3 – 4 – 5 – 6 – 14 – 15 – 16 – 17	[√] berhasil [] gagal

Lampiran E. Lampiran Implementasi Sistem

E.1 Halaman Data Kriteria

No	Kriteria	Bobot	Menu
1	A1. Omset	0.2273	Tambah
Sub.Kriteria			
	< 5.000.000/bulan	1	Edit Delete
	5.000.000 - 10.000.000/bulan	3	Edit Delete
	> 10.000.000/bulan	5	Edit Delete
2	A2. Ketersediaan Pasar	0.1364	Tambah
Sub.Kriteria			
	produk dijual di < 20 toko	1	Edit Delete
	produk dijual di 21-40 toko	3	Edit Delete
	produk dijual di > 40 toko	5	Edit Delete
3	A3. Daya Saing	0.2273	Tambah
Sub.Kriteria			
	Rendah	1	Edit Delete
	Cukup	3	Edit Delete
	Tinggi	5	Edit Delete
4	A4. Spesifikasi Kekhasan	0.1818	Tambah
Sub.Kriteria			
	produk yang sama ada di >10 kabupaten	1	Edit Delete
	produk yang sama ada di 5 - 10 kabupaten	3	Edit Delete
	produk yang sama ada di <5 kabupaten	5	Edit Delete
5	A5. Ketersediaan Bahan Baku	0.2273	Tambah
Sub.Kriteria			
	<6 bulan/tahun	1	Edit Delete
	6 - 10 bulan/tahun	3	Edit Delete
	>10 bulan/tahun	5	Edit Delete

E.2 Halaman Perhitungan WP


 +62 857-4663-3740
 
 eproduk-unggulan@gmail.com

[Home](#)
[Hallo admin, Logout](#)

MENU ADMIN

DATA MENU +



DATA PENJUALAN

HALAMAN PERHITUNGAN WP

No	Kriteria	Bobot	W_j
1	A1, Omset	5	0.2273
2	A2, Ketersediaan Pasar	3	0.1364
3	A3, Daya Saling	5	0.2273
4	A4, Spesifikasi Kekhasan	4	0.1818
5	A5, Ketersediaan Bahan Baku	5	0.2273

No	Produk	Bobot	S_j	V_j
1	B016-Keripik Pisang Asin	5, 8, 15, 20, 25	12.696	0.047
2	B017-Keripik Pisang Balado	5, 8, 15, 20, 25	12.696	0.047
3	B018-Keripik Pisang Coklat	5, 8, 15, 20, 25	12.696	0.047
4	B026-Tape Pisang	5, 8, 15, 20, 25	12.696	0.047
5	B007-Keripik Kulit Pisang	5, 8, 13, 20, 25	12.290	0.045
6	B008-Keripik Bonggol Pisang	5, 8, 13, 20, 25	12.290	0.045
7	B009-Sale Pisang Mas Kirana	5, 8, 13, 18, 25	12.056	0.044

E.3 Halaman Data Produk


 +62 857-4663-3740
 
 eproduk-unggulan@gmail.com

[Home](#)
[Hallo admin, Logout](#)

MENU ADMIN

DATA MENU +

DATA PENJUALAN

DATA PRODUK

Cari Produk

No	ID Produk	Nama	Stok	Berat	Harga	Menu
1	B006	Keripik Jahe Vilanda	100	0.25	15000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
2	B007	Keripik Kulit Pisang	150	0.25	12000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
3	B008	Keripik Bonggol Pisang	94	0.25	15000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
4	B010	Keripik Ketela Ungu	150	0.35	16000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
5	B005	Sale Oven Pisang Mas Kirana	45	0.25	15000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
6	B009	Sale Pisang Mas Kirana	99	0.25	12000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
7	B011	Rengginang Ebi	100	0.08	12500	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
8	B012	Maming Jagung	150	0.1	13000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
9	B013	Keripik Ceker	100	0.05	8500	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
10	B014	Carangmas Pisang	100	0.175	15000	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>

E.4 Halaman Data Pelanggan

MENU ADMIN

- DATA MENU +
- DATA PENJUALAN

DATA PELANGGAN

Cari Pelanggan: Q Cari

No	Nama	Email	Telepon	Menu
1	Tasya kusuma	tasya@yahoo.com	081290217456	Detail
2	Aji Seto	Ajseto@gmail	081232534534	Detail
3	mimin	mimin@gmail.com		Detail
4	coba	coba@gmail.com	081	Detail
5	beli	beli@gmail.com	123	Detail
6	qta	qta@gmail.com	55778954456	Detail
7	kiki	asss@gmail.com	123	Detail
8	abc	abc@gmail.com	08937867452	Detail
9	aris	aris@gmail.com	085765476897	Detail
10	yurike	yuri@gmail.com	086780118371	Detail

E.5 Halaman Data Ongkir

MENU ADMIN

- DATA MENU +
- DATA PENJUALAN

DATA ONGKIR

Cari: Q Cari

No	Provinsi	Kabupaten	Kecamatan	Ongkos Kirim	Menu
1	Aceh	Kab. Simeulue	Teupah Selatan	73.000	Ubah
2	Aceh	Kab. Simeulue	Simeulue Timur	73.000	Ubah
3	Aceh	Kab. Simeulue	Teupah Barat	73.000	Ubah
4	Aceh	Kab. Simeulue	Teupah Tengah	73.000	Ubah
5	Aceh	Kab. Simeulue	Simeulue Tengah	73.000	Ubah
6	Aceh	Kab. Simeulue	Teluk Dalam	73.000	Ubah
7	Aceh	Kab. Simeulue	Simeulue Cut	73.000	Ubah
8	Aceh	Kab. Simeulue	Salang	73.000	Ubah
9	Aceh	Kab. Simeulue	Simeulue Barat	73.000	Ubah
10	Aceh	Kab. Simeulue	Alafan	73.000	Ubah

E.6 Halaman Data Penjualan

MENU ADMIN

- DATA MENU +
- DATA PENJUALAN

DAFTAR PENJUALAN

Cari: Q Cari

No	ID Order	Pelanggan	Jumlah Belanja	Status	Bukti Bayar	Menu
1	ORD015	qta	87000	Order	NO	Detail
2	ORD013	beli	10000	Pembelian Ditolak	OK	Detail

E.7 Halaman Profil

+62 857-4663-3740 eprodukt-unggulan@gmail.com
Home Halo qta, Logout

MENU PELANGGAN

PROFIL

KERANJANG

HISTORY PEMBELIAN

PELANGGAN

Catatan:

1. Data harus diisi dengan benar dan tidak boleh kosong
2. Alamat yang diinputkan harus sesuai dg alamat tujuan pengiriman

Nama lengkap

Username

Password

Telepon

Alamat

Provinsi

Kabupaten

Kecamatan

Desa/Keurahan

Tgl.Registrasi 17-07-2018 22:30:54

E.8 Halaman Keranjang

+62 857-4663-3740 eprodukt-unggulan@gmail.com
Home Halo qta, Logout

MENU PELANGGAN

PROFIL

KERANJANG

HISTORY PEMBELIAN

Home Keranjang Belanja

Nama Produk	Harga	Jumlah	Total
<div style="font-size: small;">Sale Pisang Mas Kirana ID Produk: B009</div>	Rp.12.000,-	1	Rp.12.000,-
<div style="font-size: small;">Keripik Bonggol Pisang ID Produk: B008</div>	Rp.15.000,-	5	Rp.75.000,-

Bukti Pembayaran:

Catatan:

*No Rekening : 800-00-3426512-5(Mandiri) a/n PondokGunung

Total Belanja	Rp.87.000,-
Ongkos Kirim	Rp.22.700,-
Total	Rp.109.700,-

E.9 Halaman History Pembelian

+62 857-4663-3740 eprodukt-unggulan@gmail.com
Home Halo qta, Logout

MENU PELANGGAN

PROFIL

KERANJANG

HISTORY PEMBELIAN

DAFTAR PEMBELIAN

Cari:

No	ID Order	Tanggal	Jumlah Belanja	Status	No Resi	Menu
1	ORD015	2018-07-19 11:10:30	87000	Order		<input type="button" value="Detail"/>

Lampiran F. Lampiran Kode Program

Kode Program Pemilihan Produk Unggulan

```

<div class="col-sm-9 padding-right">
<div class="features_items"><!--features_items-->
<h2 class="title text-center">Halaman Perhitungan WP</h2>
<div class="col-sm-12">
<table class="table table-bordered table-responsive" style="width: auto;"
  align="center">
<thead>
<tr>
<th>No</th>
<th>Kriteria</th>
<th>Bobot</th>
<th>W<sub>j</sub></th>
</tr>
</thead>
<tbody>
<?php
$No=0;
foreach($qkriteria as $key => $vk){
$No++;
?>
<tr>
<td class="text-right"><?php echo $No; ?></td>

```

```
<td><?php echo $vk->KdKriteria.", ".$vk->Kriteria; ?></td>
<td><?php echo $vk->Bobot; ?></td>
<td><?php echo $vk->BBot; ?></td>
</tr>
<?php
}
?>
</tbody>
</table>

<table class="table table-bordered table-responsive" style="width: auto;"
  align="center">
<thead>
<tr>
<th>No</th>
<th>Produk</th>
<th>Bobot</th>
<th>S<sub>j</sub></th>
<th>V<sub>j</sub></th>
</tr>
</thead>
<tbody>
<?php
$No=0;
foreach($qproduk as $key => $vk){
```

```

$No++;
?>
<tr>
<td class="text-right"><?php echo $No; ?></td>
<td><?php echo $vk->KdProduk."-".$vk->Nama; ?></td>
<td><?php echo $vk->Omset.", ".$vk->TenagaKerja.", ".$vk->Teknologi.",
    ".$vk->SpesifikasiKekhasan.", ".$vk->KetersediaanBahanBaku; ?></td>
<td><?php echo $vk->SVektor; ?></td>
<td><?php echo $vk->NVector; ?></td>
</tr>
<?php
}
?>
</tbody>
</table>
</div>
</div><!--features_items-->
</div>

public function perhitungan_wp(){
    $data['qkriteria']    = $this->db->query("SELECT * FROM vkriteria
        ORDER BY IdKriteria asc")->result();
    $data['qproduk']     = $this->db->query("SELECT * FROM view_produk
        WHERE Stok>0 ORDER BY NVector desc")->result();
}

```



```
$data['header'] = "";  
$data['page'] = "dashboard/dashboard";  
$data['content'] = "data/data_perhitungan_wp";  
$this->load->view('themes',$data);  
}
```

