



**TUGAS AKHIR - KI141502**

**RANCANG BANGUN APLIKASI SISTEM INFORMASI  
MANAJEMEN RANTAI PASOK DISTRIBUSI DAGING  
SAPI NASIONAL**

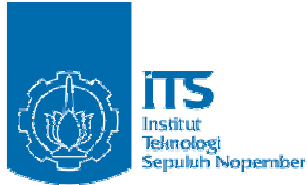
**LIDRA TRIFIDYA  
NRP 5112100152**

**Dosen Pembimbing I  
Sarwosri, S.Kom., M.T.**

**Dosen Pembimbing II  
Erma Suryani, ST., M.T., Ph.D.**

**JURUSAN TEKNIK INFORMATIKA  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2016**

*[Halaman ini sengaja dikosongkan]*



**UNDERGRADUATE THESES - KI141502**

**APPLICATION DESIGN AND IMPLEMENTATION  
OF SUPPLY CHAIN MANAGEMENT  
INFORMATION SYSTEM FOR NATIONAL BEEF  
DISTRIBUTION**

**LIDRA TRIFIDYA  
NRP 5112100152**

**Supervisor I  
Sarwosri, S.Kom., M.T.**

**Supervisor II  
Erma Suryani, ST., M.T., Ph.D.**

**DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA 2016**

*[Halaman ini sengaja dikosongkan]*

## LEMBAR PENGESAHAN

### RANCANG BANGUN APLIKASI SISTEM INFORMASI MANAJEMEN RANTAI PASOK DISTRIBUSI DAGING SAPI NASIONAL

#### TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Bidang Studi Rekayasa Perangkat Lunak  
Program Studi S-1 Jurusan Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember

Oleh  
**LIDRA TRIFIDYA**  
**NRP : 5112 100 152**

Disetujui oleh Dosen Pembimbing Tugas Akhir:

1. Sarwosri, S.Kom., M.Eng., Ph.D. (Pembimbing 1)  
NIP: 19760809 20011201001
2. Erma Suryani, S.T., M.Eng., Ph.D. (Pembimbing 2)  
NIP: 19700427 2005012001



**SURABAYA**  
**Juli, 2016**

*[Halaman ini sengaja dikosongkan]*

# **RANCANG BANGUN APLIKASI SISTEM INFORMASI MANAJEMEN RANTAI PASOK DISTRIBUSI DAGING SAPI NASIONAL**

**Nama Mahasiswa : LIDRA TRIFIDYA**  
**NRP : 5112100152**  
**Jurusan : Teknik Informatika FTIF-ITS**  
**Dosen Pembimbing 1 : Sarwosri, S.Kom., M.T.**  
**Dosen Pembimbing 2 : Erma Suryani, S.T., M.T., Ph.D.**

## *Abstrak*

*Manajemen distribusi merupakan salah satu aspek penting dalam memastikan kelancaran distribusi produk dari produsen ke konsumen dengan cara yang efisien. Berdasarkan pendekatan Supply Chain Management, lemahnya sistem distribusi merupakan salah satu penyebab terjadinya disparitas harga serta kelangkaan stok pada komoditas bahan pokok terutama daging sapi. Dengan langkanya stok di berbagai daerah dapat menyebabkan peningkatan harga sehingga terjadi perbedaan harga yang tinggi antar wilayah. Kelancaran distribusi akan menjamin ketersediaan stok sehingga akan mampu memperbaiki permasalahan yang ada. Melalui pendekatan Supply Chain Management, distribusi daging sapi dapat lebih transparan dan dapat terlihat ketersediaan stock di berbagai wilayah.*

*Pada tugas akhir ini, dibangun sistem informasi distribusi daging sapi dengan pendekatan Supply Chain Management. Aplikasi ini dibuat dengan menggunakan metode aplikasi berorientasi obyek dengan PHP serta menggunakan struktur basis data relational MySQL. Dalam melakukan perancangan sistem, terlebih dahulu dilakukan analisis sistem dengan mewawancarai stakeholder yang berkaitan dengan distribusi*

*daging sapi. Hasil pengujian sistem informasi ini menunjukkan bahwa kebutuhan pengguna telah terpenuhi dan mampu membantu proses distribusi daging sapi.*

***Kata kunci: Supply Chain Management, Distribusi, Komoditas Daging Sapi***



# **APPLICATION DESIGN AND IMPLEMENTATION OF SUPPLY CHAIN MANAGEMENT INFORMATION SYSTEM FOR NATIONAL BEEF DISTRIBUTION**

**Student's Name** : LIDRA TRIFIDYA  
**Student's ID** : 5112100152  
**Department** : Teknik Informatika FTIF-ITS  
**First Advisor** : Sarwosri, S.Kom., M.T.  
**Second Advisor** : Erma Suryani, S.T., M.T., Ph.D.

## ***Abstract***

*Distribution management is one of the important aspect in ensuring the smooth distribution of product from producer to consumer efficiently. Based on supply chain management approach, the weakness of distribution system is one of the causes of the disparity in price and scarcity of stock on staple commodity, especially beef. The scarcity of stocks in various regions can lead to price increases and become the cause of high price differences between regions. The smooth distribution can be a guarantee for the availability of stock and it will be fixed the problem. Through the supply chain management approach, the distribution of beef can be more transparent and stakeholder can see the availability of stocks in various regions.*

*In this final project, information system of beef distribution with supply chain management approach is constructed. This application is built using an object-oriented methods with PHP and MySQL relational database. In designing this system, system analysis is conducted by interviewing stakeholder related with beef distribution. This system information test result shows that the user needs are met and able to help in the distribution process of beef.*

***Keywords: Supply Chain management, Distribution, Beef Commodities***

*[Halaman ini sengaja dikosongkan]*

## KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil'alamin, segala puji bagi Allah SWT, yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **“RANCANG BANGUN APLIKASI SISTEM INFORMASI MANAJEMEN RANTAI PASOK DISTRIBUSI DAGING SAPI NASIONAL”**.

Selesaiannya tugas akhir ini tidak lepas dari bantuan dan dukungan beberapa pihak, sehingga pada kesempatan ini penulis ingin mengucapkan syukur serta terima kasih yang sebesar-besarnya kepada:

1. Kepada kedua orangtua, mama dan papa yang telah memberikan kasih sayang, semangat dan motivasi kepada penulis sehingga dapat menyelesaikan Tugas Akhir ini tepat waktu.
2. Kakak dan Abang yang selalu mendukung dan memberikan semangat dalam mengerjakan Tugas Akhir ini.
3. Ibu Sarwosri, S.KOM., MT. selaku pembimbing I yang selalu memberikan semangat, motivasi serta membimbing penulis selama pengerjaan tugas akhir.
4. Ibu Erma Suryani, S.T., M.T., Ph.D. selaku pembimbing II yang juga telah membantu, membimbing serta memberikan masukan-masukan selama pengerjaan tugas akhir ini.
5. Bapak dan ibu dosen Jurusan Teknik Informatika ITS yang telah mengajarkan dan memberikan banyak ilmu yang berguna kepada penulis.
6. Bapak dan ibu karyawan Jurusan Teknik Informatika ITS yang telah memberikan bantuan kepada penulis selama masa perkuliahan.
7. Teman-teman “Rantau” yang selalu menjadi tempat senang dan sedih selama 4 tahun masa perkuliahan.

8. Teman-teman Teknik Informatika ITS angkatan 2012, yang telah menemani dan berjuang bersama dari awal masa perkuliahan hingga lulus.
9. Teman-teman dari UKM PLH SIKLUS ITS yang telah memberikan ilmu-ilmu baru kepada penulis dan selalu memberikan semangat kepada penulis selama pengerjaan Tugas Akhir ini.
10. Serta semua pihak yang tidak disebutkan yang telah membantu penulis dalam menyelesaikan tugas akhir ini.

Penulis berharap Tugas Akhir ini dapat memberikan manfaat bagi semua pihak terutama bagi penulis sendiri. Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kata sempurna dan masih memiliki banyak kekurangan, sehingga kritik dan saran yang membangun sangat penulis harapkan dari pembaca untuk perbaikan selanjutnya.

Surabaya, Juni 2016

# DAFTAR ISI

LEMBAR PENGESAHAN.....	v
<i>Abstrak</i> .....	vii
<i>Abstract</i> .....	ix
KATA PENGANTAR.....	xi
DAFTAR ISI .....	xiii
DAFTAR GAMBAR .....	xvii
DAFTAR TABEL .....	xxi
DAFTAR KODE SUMBER.....	xxiii
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah .....	3
1.4 Tujuan.....	3
1.5 Manfaat.....	4
1.6 Metodologi.....	4
1.7 Sistematika Penulisan Laporan Tugas Akhir .....	5
BAB II TINJAUAN PUSTAKA.....	7
2.1 Sistem Informasi Manajemen Rantai Pasok.....	7
2.2 Komoditas Daging Sapi .....	7
2.3 Manajemen Rantai Pasok.....	9
2.4 <i>E-Supply Chain Management</i> .....	12
2.5 Kerangka Kerja CodeIgniter .....	13
2.6 MVC (Model View Controller) .....	13
2.7 PHP: <i>HyperText Preprocessor</i> .....	14
2.8 MySQL .....	14
2.9 Web Server .....	14
BAB III ANALISIS DAN PERANCANGAN SISTEM .....	17
3.1 Analisis .....	17
3.1.1 Analisis Sistem Yang Sedang Berjalan .....	17
3.1.2 Analisis Proses Bisnis .....	18
3.1.3 Deskripsi Umum Sistem.....	19
3.1.4 Spesifikasi Kebutuhan Perangkat Lunak.....	20
3.1.4.1 Kebutuhan Fungsional .....	21
3.1.4.2 Kebutuhan Non-Fungsional.....	22
3.1.4.3 Aktor.....	22
3.1.4.4 Kasus Penggunaan Sistem .....	24

3.1.4.5	Kasus Penggunaan Melakukan Registrasi User .....	25
3.1.4.6	Kasus Penggunaan melakukan Approval User .....	26
3.1.4.7	Kasus Penggunaan Menambah Data Admin .....	28
3.1.4.8	Kasus Penggunaan Melihat Detail Data Admin .....	30
3.1.4.9	Kasus Penggunaan Mengubah Data Admin .....	32
3.1.4.10	Kasus Penggunaan Menghapus Data Admin .....	33
3.1.4.11	Kasus Penggunaan Melihat History Proses Transaksi .	35
3.1.4.12	Kasus Penggunaan Melakukan Pemesanan Produk .....	37
3.1.4.13	Kasus Penggunaan Upload Berkas Dokumen Transaksi .....	38
3.1.4.14	Kasus Penggunaan Menerima Proses Permintaan Order .....	39
3.1.4.15	Kasus Penggunaan Melakukan Permintaan Order .....	42
3.1.4.16	Kasus Penggunaan Menerima Pemesanan Produk .....	43
3.1.4.17	Kasus Penggunaan Menambah Jumlah Stok Tersedia .	45
3.1.4.18	Kasus Penggunaan Mengubah Jumlah Stock Tersedia	47
3.1.4.19	Kasus Penggunaan Melihat Jumlah Stock Tersedia .....	49
3.1.4.20	Kasus Penggunaan Melihat Profile User .....	50
3.1.4.21	Kasus Penggunaan Mengubah Data User .....	51
3.2	Perancangan .....	53
3.2.1	Perancangan Arsitektur Sistem .....	54
3.2.2	Perancangan Diagram Kelas .....	55
3.2.3	Perancangan Basis Data .....	55
3.2.3.1	Rancangan Antarmuka Halaman Registrasi User .....	59
3.2.3.2	Rancangan Antarmuka Halaman Utama .....	59
3.2.3.3	Rancangan Antarmuka Halaman Approval User .....	62
3.2.3.4	Rancangan Antarmuka Halaman Mengelola Data Admin .....	64
3.2.3.5	Rancangan Antarmuka Halaman Melakukan Pemesanan Produk .....	64
3.2.3.6	Rancangan Antarmuka Halaman Menerima Pemesanan Produk .....	66
3.2.3.7	Rancangan Antarmuka Halaman Upload Berkas Dokumen Transaksi .....	69
3.2.3.8	Rancangan Antarmuka Halaman Menambah Jumlah Stock Tersedia .....	69
3.2.3.9	Rancangan Antarmuka Halaman Update Jumlah Stock Tersedia .....	70

3.2.3.10	Rancangan Antarmuka Halaman Melihat Jumlah Stock Tersedia .....	71
3.2.3.11	Rancangan Antarmuka Halaman Mengelola Data User.....	71
3.2.3.12	Rancangan Antarmuka History Transaksi .....	71
BAB IV IMPLEMENTASI .....		75
4.1	Lingkungan Implementasi.....	75
4.2	Implementasi <i>Model-View-Controller</i> .....	75
4.2.1	Implementasi View.....	76
4.2.2	Implementasi Kelas <i>Model</i> .....	81
4.2.2.1	Fungsi <i>isDuplicate</i> .....	81
4.2.2.2	Fungsi Login User .....	81
4.2.2.3	Fungsi <i>Get List Admin</i> .....	82
4.2.2.4	Fungsi Get Transaksi List .....	83
4.2.2.5	Fungsi <i>Update</i> Pembayaran .....	83
4.2.2.6	Fungsi <i>Insert to Transaction</i> .....	84
4.2.2.7	Fungsi Transaksi .....	84
4.2.2.8	Fungsi <i>Add Stock</i> .....	85
4.2.2.9	Fungsi <i>Manage User</i> .....	86
4.2.3	Implementasi Kelas <i>Controller</i> .....	87
4.2.3.1	Fungsi Registrasi .....	87
4.2.3.2	Fungsi <i>getPrice</i> .....	89
4.2.3.3	Class Login .....	89
4.2.3.4	Fungsi <i>insertToTransaction</i> .....	89
4.2.3.5	Fungsi <i>do Upload</i> .....	91
4.2.3.6	Class Add Stock.....	92
4.3	Implementasi Antarmuka Pengguna .....	93
4.3.1	Halaman Login.....	93
4.3.2	Halaman Registrasi User .....	94
4.3.3	Halaman Approve User .....	94
4.3.4	Halaman Mengelola Data Admin .....	94
4.3.5	Halaman Transaksi .....	97
4.3.6	Halaman Terima Transaksi.....	100
4.3.7	Halaman Upload Berkas Bukti Pembayaran .....	102
4.3.8	Halaman Mengelola Stock Barang .....	103
4.3.9	Halaman History Transaksi .....	104
4.3.10	Halaman Grafik Produksi Daging Sapi .....	104
BAB V PENGUJIAN DAN EVALUASI .....		107
5.1	Lingkungan Pengujian .....	107

5.2	Skenario Pengujian.....	107
5.2.1	Kasus Pengujian Melakukan Registrasi User .....	108
5.2.2	Kasus Pengujian Approval Data User.....	109
5.2.3	Kasus Pengujian Melihat History Proses Transaksi .....	111
5.2.4	Kasus Pengujian Mengelola Data Admin .....	112
5.2.5	Kasus Pengujian Melakukan Pemesanan Produk .....	114
5.2.6	Kasus Pengujian Menerima Pemesanan Produk.....	117
5.2.7	Kasus Pengujian Upload Berkas Bukti Pembayaran .....	119
5.2.8	Kasus Pengujian Menambah Jumlah Stock Barang.....	121
5.3	Evaluasi Pengujian .....	124
BAB VI KESIMPULAN DAN SARAN .....		127
6.1	Kesimpulan .....	127
6.2	Saran.....	128
DAFTAR PUSTAKA .....		129
LAMPIRAN .....		131
BIODATA PENULIS .....		135



## DAFTAR GAMBAR

Gambar 2.1 Data Produksi Daging Sapi.....	8
Gambar 2.2 Data Kebutuhan Daging Sapi .....	9
Gambar 2.3 Ilustrasi Manajemen Rantai pasok .....	10
Gambar 2.4 Arsitektur SCM Distribusi Daging Sapi .....	11
Gambar 2.5 Model View Controller.....	13
Gambar 2.6 Arsitektur Web Server .....	15
Gambar 3.1 Alur Distribusi Daging Sapi .....	19
Gambar 3.2 Alur Kerja Sistem Distribusi Daging Sapi.....	21
Gambar 3.3 Diagram Kasus Penggunaan .....	24
Gambar 3.4 Diagram Aktivitas Melakukan Registrasi Akun .....	26
Gambar 3.5 Diagram Aktivitas Melakukan Approval User .....	28
Gambar 3.6 Diagram Aktivitas Menambah Data Admin .....	30
Gambar 3.7 Diagram Aktivitas Melihat Detail Data Admin .....	31
Gambar 3.8 Diagram Aktivitas Mengubah Data Admin .....	33
Gambar 3.9 Diagram Aktivitas Menghapus Data Admin.....	35
Gambar 3.10 Diagram Aktivitas Melihat History Transaksi .....	36
Gambar 3.11 Diagram Aktivitas Melakukan Pemesanan Produk ...	38
Gambar 3.12 Diagram Aktivitas Upload Berkas Dokumen Transaksi .....	40
Gambar 3.13 Diagram Aktivitas Menerima Proses Permintaan Order .....	41
Gambar 3.14 Diagram Aktivitas Melakukan Permintaan Order .....	43
Gambar 3.15 Diagram Aktivitas Menerima Pemesanan Produk .....	45
Gambar 3.16 Diagram Aktivitas Menambah Jumlah Stock Tersedia .....	47
Gambar 3.17 Diagram Aktivitas Megnubah Jumlah Stock Tersedia	49
Gambar 3.18 Diagram Aktivitas Melihat Jumlah Stock Tersedia ...	50
Gambar 3.19 Diagram Aktivitas Melihat Profile User .....	51
Gambar 3.20 Diagram Aktivitas Mengubah Data User.....	53
Gambar 3.21 Ilustrasi Arsitektur Model Sistem Infomasi Distribusi Daging Sapi.....	54
Gambar 3.22 Perancangan Diagram Kelas.....	56
Gambar 3.23 Rancangan Basis Data SQL.....	57
Gambar 3.24 Rancangan Antarmuka Registrasi User .....	59
Gambar 3.25 Rancangan Antarmuka Halaman Utama Admin.....	60
Gambar 3.26 Rancangan Antarmuka Halaman Utama Supplier .....	61

Gambar 3.27 Rancangan Antarmuka Halaman Utama Distributor..	61
Gambar 3.28 Rancangan Antarmuka Halaman Customer .....	62
Gambar 3.29 Rancangan Antarmuka Halaman User .....	62
Gambar 3.30 Rancangan Antarmuka Halaman Approval User .....	63
Gambar 3.31 Rancangan Antarmuka Halaman Lihat Detail User ...	64
Gambar 3.32 Rancangan Antarmuka Mengelola Data Admin .....	64
Gambar 3.33 Rancangan Antarmuka Menambah Data Admin.....	65
Gambar 3.34 Rancangan Antarmuka Melakukan Pemesanan Produk .....	65
Gambar 3.35 Rancangan Antarmuka Transaksi Baru.....	66
Gambar 3.36 Rancangan Antarmuka Detail Transaksi.....	67
Gambar 3.37 Rancangan Antarmuka Halaman Terima Order.....	67
Gambar 3.38 Rancangan Antarmuka Halaman Detail Order.....	68
Gambar 3.39 Rancangan Halaman Invoice.....	68
Gambar 3.40 Rancangan Antarmuka Halaman Upload Berkas.....	69
Gambar 3.41 Rancangan Antarmuka Halaman Mengelola Jumlah Stock .....	70
Gambar 3.42 Rancangan Amntarmuka Halaman Tamabah Stock Barang Tersedia .....	70
Gambar 3.43 Rancangan Antarmuka Halaman Update Jumlah Stock Tersedia .....	71
Gambar 3.44 Rancangan Antarmuka Halaman Melihat Jumlah Stock Tersedia .....	72
Gambar 3.45 Rancangan Antarmuka Halaman Mengelola Data User .....	72
Gambar 3.46 Rancangan Antarmuka Halaman History Transaksi ..	73
Gambar 4.1 Implementasi Antarmuka Halaman Login .....	94
Gambar 4.2 Implementasi Antarmuka Halaman Registrasi.....	95
Gambar 4.3 Implementasi Antarmuka Halaman <i>Approve User</i> .....	96
Gambar 4.4 Implementasi Antarmuka Halaman Lihat Detail User .	96
Gambar 4.5 Implementasi Antarmuka Halaman List User .....	97
Gambar 4.6 Implementasi Antarmuka Halaman Mengelola Data Admin .....	97
Gambar 4.7 Implementasi Antarmuka Halaman Menambah Data Admin .....	98
Gambar 4.8 Implementasi Antarmuka Halaman Melihat Data Admin .....	98

Gambar 4.9 Implementasi Antarmuka Halaman Mengubah Data Admin.....	99
Gambar 4.10 Implementasi Antarmuka Halaman Transaksi Baru ..	99
Gambar 4.11 Implementasi Antarmuka Halaman Detail Transaksi .....	100
Gambar 4.12 Implementasi Antarmuka Halaman Persetujuan Transaksi .....	101
Gambar 4.13 Implementasi Antarmuka Halaman Konfirmasi Pengiriman .....	101
Gambar 4.14 Implementasi Halaman Cetak Invoice .....	102
Gambar 4.15 Implementasi Antarmuka Halaman Upload Berkas.	102
Gambar 4.16 Implementasi Antarmuka Halaman Mengelola Data Stock.....	103
Gambar 4.17 Implementasi Antarmuka Halaman Menambah Data Stock.....	103
Gambar 4.18 Implementasi Antarmuka Halaman Update Stock...	104
Gambar 4.19 Implementasi Antarmuka Halaman Melihat Stock Tersedia .....	104
Gambar 4.20 Impleemntasi Antarmuka Halaman History Transaksi .....	105
Gambar 4.21 Implementasi Antarmuka Halaman Grafik Produksi Daging Sapi.....	105
Gambar 5.1 Tampilan Pengujian Pengisian Data Registrasi User.	109
Gambar 5.2 Tampilan Hasil Pengujian Registrasi User .....	109
Gambar 5.3 Tampilan Persetujuan Pengguna.....	110
Gambar 5.4 Tampilan Persetujuan Pengguna.....	111
Gambar 5.5 Tampilan Pengujian History Transaksi.....	112
Gambar 5.6 Pengisian Data Uji Penambahan Admin.....	113
Gambar 5.7 Tampilan Notifikasi Keberhasilan Penambahan Data	113
Gambar 5.8 Tampilan Hasil Pengujian Penambahan Admin .....	114
Gambar 5.9 Pengisian Data Uji Pemesanan Produk.....	115
Gambar 5.10 Pengujian Data Uji Pemesanan Produk .....	116
Gambar 5.11 Tampilan Notifikasi Keberhasilan Transaksi.....	116
Gambar 5.12 Tampilan Hasil Pengujian Pemesanan Produk .....	117
Gambar 5.13 Tampilan List Permintaan Produk .....	118
Gambar 5.14 Tampilan Halaman Persetujuan Transaksi.....	119
Gambar 5.15 Tampilan Notifikasi Keberhasilan Persetujuan Transaksi .....	119

Gambar 5.16 Pengujian Data Upload Berkas Pembayaran.....	120
Gambar 5.17 Tampilan Notifikasi Keberhasilan Upload Berkas...	121
Gambar 5.18 Tampilan Hasil Upload Bukti Pembayaran.....	121
Gambar 5.19 Pengujian Data Menambah Jumlah Stock Barang ...	122
Gambar 5.20 Tampilan Notifikasi Keberhasilan Proses .....	122
Gambar 5.21 Tampilan Hasil Uji Penambahan Stock Barang .....	123
Gambar 5.22 Pengujian Data Pembaharuan Stock .....	124
Gambar 5.23 Tampilan Notifikasi Keberhasilan Pembaharuan Stock .....	124
Gambar A.1 Evaluasi Pengujian Fungsionalitas Sebagai Customer .....	131
Gambar A.2 Evaluasi Pengujian Fungsionalitas Sebagai Distributor .....	131
Gambar A.3 Evaluasi Pengujian Fungsionalitas Sebagai Supplier	132
Gambar A.4 Evaluasi Pengujian Fungsionalitas Sebagai Administrator .....	132
Gambar A.5 Lembar <i>Feedback</i> Dari Sisi Administrator .....	132
Gambar A.6 Lembar <i>Feedback</i> Dari Sisi Supplier .....	133
Gambar A.7 Lembar <i>Feedback</i> Dari Sisi Distributor .....	133
Gambar A.8 Lembar <i>Feedback</i> Dari Sisi Customer Yaitu Pihak Superindo.....	134

## DAFTAR TABEL

Tabel 3.1 Analisis Kebutuhan Distribusi Daging Sapi.....	18
Tabel 3.2 Kebutuhan Fungsional Perangkat Lunak.....	21
Tabel 3.3 Kebutuhan Non-Fungsional.....	22
Tabel 3.4 Kasus Penggunaan Melakukan Registrasi User.....	25
Tabel 3.5 Kasus Penggunaan Melakukan Approval User .....	27
Tabel 3.6 Kasus Penggunaan Menambah Data Admin .....	29
Tabel 3.7 Kasus Penggunaan Melihat Detail Data Admin .....	31
Tabel 3.8 Kasus Penggunaan Mengubah Data Admin .....	32
Tabel 3.9 Kasus Penggunaan Menghapus Data Admin.....	34
Tabel 3.10 Kasus Penggunaan Melihat History Transaksi .....	35
Tabel 3.11 Kasus Penggunaan Melakukan Pemesanan Produk.....	37
Tabel 3.12 Kasus Penggunaan Upload Berkas Dokumen Transaksi .....	39
Tabel 3.13 Kasus Penggunaan Menerima Proses Permintaan Order.....	40
Tabel 3.14 Kasus Penggunaan Melakukan Permintaan Order.....	42
Tabel 3.15 Kasus Penggunaan Menerima Pemesanan Produk .....	44
Tabel 3.16 Kasus Penggunaan Menambah Jumlah Stok Tersedia ..	45
Tabel 3.17 Kasus Penggunaan Mengubah Jumlah Stock Tersedia..	48
Tabel 3.18 Kasus Penggunaan Melihat Jumlah Stock Tersedia .....	49
Tabel 3.19 Kasus Penggunaan Melihat Profile User .....	51
Tabel 3.20 Kasus Penggunaan Mengubah Data User.....	52
Tabel 4.1 Implementasi View.....	76
Tabel 5.1 Spesifikasi Sistem Pengujian.....	107
Tabel 5.2 Kasus Uji Melakukan Registrasi User.....	108
Tabel 5.3 Kasus Uji Approval User.....	109
Tabel 5.4 Kasus Uji Melihat History Proses Transaksi .....	111
Tabel 5.5 Kasus Uji Penambahan Admin Baru .....	112
Tabel 5.6 Kasus Uji Melakukan Pemesanan Produk.....	114
Tabel 5.7 Kasus Uji Menerima Pemesanan Produk .....	117
Tabel 5.8 Kasus Uji Melakukan Upload Berkas Bukti Pembayaran .....	119
Tabel 5.9 Kasus Uji Menambah Stock barang .....	121
Tabel 5.10 Kasus Uji Memperbaharui Stock Barang .....	123
Tabel 5.11 Evaluasi Pengujian oleh Mbak Qiqi Sebagai Customer .....	125

Tabel 5.12 Evaluasi Pengujian oleh Bu Erma Sebagai Administrator .....	125
Tabel 5.13 Evaluasi Pengujian oleh Bu Erma Sebagai Supplier....	125
Tabel 5.14 Evaluasi Pengujian oleh Bu Erma Sebagai Distributor	125

## DAFTAR KODE SUMBER

Kode Sumber 4. 1 Kode Sumber View Grafik .....	81
Kode Sumber 4. 2 Fungsi <i>isDuplicate</i> .....	82
Kode Sumber 4. 3 Fungsi <i>Login User</i> .....	82
Kode Sumber 4. 4 Fungsi <i>Get List Admin</i> .....	83
Kode Sumber 4. 5 Fungsi <i>Get List Transaksi</i> .....	83
Kode Sumber 4. 6 <i>Update</i> Pembayaran .....	84
Kode Sumber 4. 7 Fungsi <i>Insert to Transaction</i> .....	84
Kode Sumber 4. 8 Fungsi Transaksi .....	85
Kode Sumber 4. 9 Fungsi <i>Add Stock</i> .....	86
Kode Sumber 4. 10 Fungsi <i>Manage User</i> .....	87
Kode Sumber 4. 11 Fungsi Registrasi .....	88
Kode Sumber 4. 12 Fungsi <i>getPrice</i> .....	89
Kode Sumber 4. 13 Class Login .....	90
Kode Sumber 4. 14 Fungsi <i>insertToTransaction</i> .....	91
Kode Sumber 4. 15 Kode Sumber <i>do Upload</i> .....	92
Kode Sumber 4. 16 Class <i>Add Stock</i> .....	93

*[Halaman ini sengaja dikosongkan]*



# **BAB I**

## **PENDAHULUAN**

Bab ini membahas latar belakang, permasalahan, tujuan diambilnya topik tugas akhir ini. Serta manfaat yang didapatkan dari pembuatan sistem informasi yang dibangun.

### **1.1 Latar Belakang**

Daging sapi merupakan salah satu komoditas pada subsektor peternakan yang berperan penting dalam perekonomian Indonesia. Berdasarkan data dari BPS serta Direktorat Jenderal Peternakan dan Kesehatan Hewan Tahun 2014 diketahui bahwa dari tahun ke tahun jumlah produksi daging sapi masih belum bisa mencukupi jumlah kebutuhan konsumsi penduduk di Indonesia [1]. Peningkatan jumlah penduduk yang semakin tinggi berbanding lurus dengan tingginya permintaan terhadap daging sapi sehingga menyebabkan ketidak-seimbangan antar jumlah permintaan dan jumlah produksi daging sapi tersebut. Dengan ketidak-seimbangan ini menyebabkan Indonesia saat ini melakukan import daging sapi sekitar 30-35% berupa sapi bakalan dan daging beku [1].

Selain diakibatkan oleh tingginya peningkatan jumlah penduduk, kurangnya stock juga dapat diakibatkan oleh kelemahan manajemen distribusi. Manajemen distribusi merupakan salah satu aspek penting dalam menentukan ketersediaan barang diberbagai daerah. Sistem distribusi yang lemah dapat menyebabkan terjadinya kelangkaan stock serta terjadinya disparitas harga diberbagai wilayah. Dengan langkanya stock daging sapi akan memicu naiknya harga daging tersebut, kenaikan harga akan menyebabkan disparitas harga yang semakin lebar. Sehingga dengan lancarnya distribusi akan menjamin ketepatan stock serta mampu mengendalikan disparitas harga [2]. Terjadinya perbedaan serta fluktuasi harga juga disebabkan oleh panjangnya rantai pasok dan transportasi sapi hidup dari sentra produksi ke konsumen. Permasalahan yg dihadapi juga terlihat dari aspek produksi-

distribusi-konsumsi yang masih terpisah dan belum terintegrasi. Para peternak pada tingkat produksi masih terpisah dengan pelaku distribusi yang dilakukan oleh pihak lain. Hal ini menyebabkan ketidakefisienan rantai pasok dan juga membuka peluang pihak tertentu dalam mengambil keuntungan secara tidak proposional [3].

Salah satu upaya penyelesaian dari permasalahan distribusi daging sapi ini yaitu dengan menggunakan metode *Supply Chain Management* (Manajemen Rantai Pasok). Manajemen Rantai Pasok merupakan suatu proses pengelolaan seluruh aktivitas proses dari pembelian material, perencanaan produksi, proses transformasi dari produk masih dalam proses hingga menjadi produk jadi serta penyampaian produk jadi ke konsumen akhir melalui sistem distribusi, dengan kata lain kegiatan dalam rantai pasok merupakan proses penyampaian produk yang awalnya masih berupa sapi potong hidup menjadi daging sapi yang siap dipasarkan [3]. Kegiatan rantai pasok daging sapi perlu dilakukan untuk memenuhi permintaan konsumen yang tersebar di beberapa wilayah yang berbeda-beda di Indonesia dan juga jumlah permintaan konsumen yang tidak pasti.

Sistem informasi manajemen rantai pasok merupakan penggabungan penggunaan teknologi internet untuk meningkatkan kegiatan rantai pasok dan manajemen rantai pasok. Adanya penggunaan teknologi internet dalam penerapan SCM berbasis web dapat mengintegrasikan antara supplier, distributor dan konsumen akhir secara real-time dan online sehingga memudahkan antar pihak dan dapat menghasilkan koordinasi yang baik. Dengan adanya sistem informasi ini alur informasi mengenai ketersediaan jumlah barang diberbagai wilayah akan lebih transparan. Seluruh proses transaksi yang terjadi akan dicatat dan di update terus-menerus sehingga jumlah ketersediaan barang akan diketahui. Dengan adanya data yang ter-record dan terus diupdate, maka akan memudahkan dalam mengembangkan analisis *supply* dan *demand forecasting* sehingga dapat memperidiksi jumlah stock secara akurat. Dengan adanya sistem informasi ini diharapkan distribusi

daging sapi di berbagai wilayah lebih transparan dan terlihat stok yang tersedia sehingga bisa menghindari terjadinya fluktuasi harga daging sapi.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dibahas, berikut rumusan masalah yang diangkat untuk tugas akhir ini:

1. Bagaimana mendapatkan proses bisnis *supply chain* distribusi daging sapi nasional?
2. Bagaimana merancang dan membangun aplikasi Sistem Informasi Manajemen Rantai Pasok Distribusi Daging Sapi Nasional?
3. Bagaimana membangun sistem informasi distribusi daging sapi berdasarkan jumlah stok barang yang tersedia serta mengetahui status transaksi berbasis web?

## 1.3 Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan masalah, yaitu:

1. Data penelitian untuk distribusi daging sapi dari beberapa wilayah diambil dari penelitian sebelumnya [4].
2. Rancang bangun sistem ini menggunakan bahasa pemrograman PHP dengan *framework* CodeIgniter
3. Sistem informasi ini menggunakan basis data MySQL.

## 1.4 Tujuan

Tujuan dari penyusunan tugas akhir ini yaitu:

1. Mengetahui alur proses bisnis *supply chain* distribusi daging sapi dari supplier sampai ke konsumen akhir.
2. Merancang dan membangun aplikasi sistem informasi manajemen rantai pasok distribusi daging sapi nasional.
3. Membangun sistem informasi manajemen rantai pasok distribusi daging sapi berdasarkan jumlah stok barang yang tersedia serta mengetahui status transaksi.

## 1.5 Manfaat

Dengan dibuatnya tugas akhir ini, maka diharapkan dengan adanya aplikasi sistem informasi manajemen rantai pasok ini dapat mengintegrasikan setiap rantai pasok yang terlibat pada distribusi daging sapi dan dapat mengoptimalkan distribusi daging sapi ke berbagai wilayah.

## 1.6 Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

### 1. Penyusunan proposal tugas akhir.

Tahap awal dalam pembuatan tugas akhir ini yaitu penyusunan proposal tugas akhir. Proposal tugas akhir berisi pendahuluan yang terdiri atas hal yang menjadi latar belakang diajukannya usulan tugas akhir, rumusan masalah yang diangkat, batasan masalah untuk tugas akhir, tujuan dari pembuatan tugas akhir, dan manfaat dari hasil pembuatan tugas akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan tugas akhir.

### 2. Studi literatur

Pada tahap studi literatur ini akan dipelajari sejumlah referensi yang diperlukan dalam pembuatan aplikasi yaitu mengenai PHP, CodeIgniter, MySQL dan SMS Gateway. Selain itu juga mempelajari proses bisnis dari distribusi daging sapi. Informasi dan literatur didapatkan dari literatur buku, informasi dari website, serta sumber-sumber informasi yang berhubungan.

### 3. Wawancara

Tahap ini yaitu melakukan wawancara ke berbagai narasumber terkait yang berhubungan dengan distribusi daging sapi in sendiri. Wawancara dilakukan untuk mengetahui proses bisnis dari distribusi daging sapi dan menganalisa kebutuhan sistem.

#### 4. Analisis dan desain perangkat lunak

Tahap ini meliputi perumusan kebutuhan fungsional, kasus penggunaan, diagram aktivitas, rancangan antarmuka pengguna untuk aplikasi sistem informasi distribusi daging sapi dan pembuatan rancangan basis data.

#### 5. Implementasi perangkat lunak

Implementasi merupakan tahap membangun rancangan program yang telah dibuat. Pada tahapan ini merealisasikan rancangan yang terdapat pada tahapan sebelumnya, sehingga menjadi sebuah program yang sesuai dengan apa yang telah direncanakan. Tahapan implementasi menggunakan Bahasa pemrograman PHP dengan kerangka kerja CodeIgniter dan memanfaatkan metode *Model-View-Controller* (MVC).

#### 6. Pengujian dan evaluasi

Pada tahap ini dilakukan pengujian fungsionalitas dengan menggunakan metode *blackbox testing*, dimana metode ini dilakukan untuk memastikan kesesuaian hasil keluaran sistem. Pengujian fungsionalitas dilakukan dengan menyiapkan sejumlah skenario sebagai tolok ukur keberhasilan pengujian.

#### 7. Penyusunan buku tugas akhir.

Pada tahapan ini disusun buku yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

### 1.7 Sistematika Penulisan Laporan Tugas Akhir

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut ini.

**Bab I    Pendahuluan**

Bab yang berisi mengenai latar belakang, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penulisan juga merupakan bagian dari bab ini.

**Bab II    Tinjauan Pustaka**

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

**Bab III   Perancangan Perangkat Lunak**

Bab ini berisi tentang desain sistem, rancangan basis data, diagram kasus penggunaan, diagram aktivitas dan rancangan antarmuka pengguna.

**Bab IV    Implementasi**

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa *code* yang digunakan untuk proses implementasi.

**Bab V     Uji Coba dan Evaluasi**

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

**Bab VI    Kesimpulan dan Saran**

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

## **BAB II**

### **TINJAUAN PUSTAKA**

Bab ini berisi penjelasan teori-teori yang berkaitan dengan rancang bangun aplikasi sistem informasi manajemen rantai pasok distribusi daging sapi nasional. Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap program yang dibuat dan berguna sebagai penunjang dalam pengembangan perangkat lunak.

#### **2.1 Sistem Informasi Manajemen Rantai Pasok**

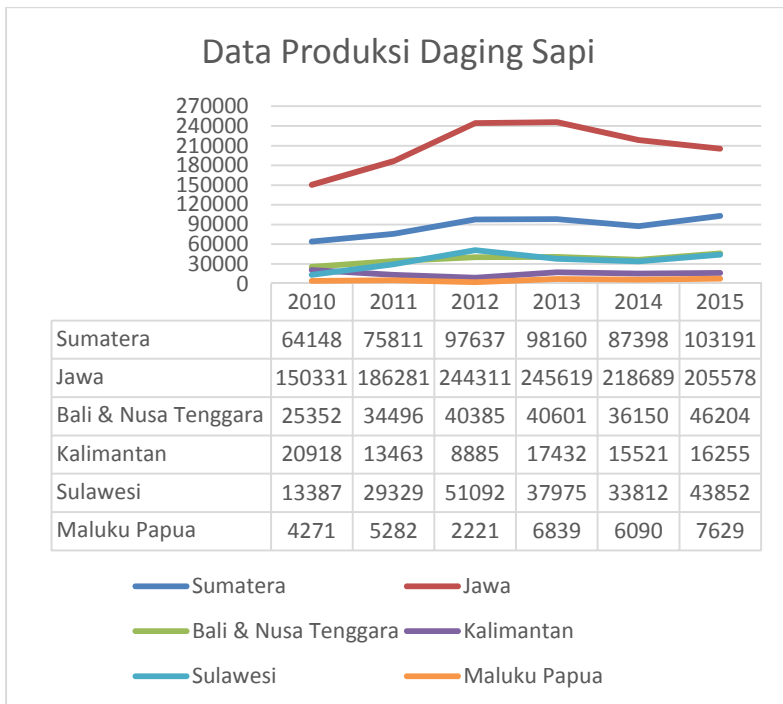
Pada pembuatan sistem informasi berbasis manajemen rantai pasok ini, penulis mendapatkan sumber studi literatur dari penelitian Didiek Sri Wiyono yang berjudul “Analisis dan Desain Aplikasi *Web* dan *Mobile Supply Chain Management* untuk Mendukung Distribusi Komoditas Padi Paska-Panen” [5]. Pada penelitian sebelumnya disebutkan permasalahan yang terjadi pada distribusi komoditas padi yaitu tingkat ketersediaan komoditas dan aksesibilitas informasi yang masih rendah. Adanya ketidakpastian informasi mengenai ketersediaan stok [5].

#### **2.2 Komoditas Daging Sapi**

Komoditas daging sapi merupakan salah satu komoditas prioritas dalam pembangunan nasional dalam upaya mewujudkan ketahanan pangan asal hewani. Komoditas daging sapi termasuk kedalam Rencana Strategis Kementerian Pertanian 2015-2019 dalam upaya mencapai target sukses pembangunan pertanian yaitu pencapaian swasembada padi, jagung dan kedelai serta peningkatan produksi gula dan daging [6]. Sapi merupakan salah satu komoditas pada subsektor peternakan. Sapi terbagi menjadi dua jenis yaitu sapi potong yang menghasilkan daging dan sapi perah yang memproduksi susu. Komoditas daging sapi merupakan salah satu komoditas prioritas dalam program pembangunan nasional dalam upaya mewujudkan ketahanan pangan asal hewani.

Produksi daging tidak hanya didasarkan pada jumlah permintaan, tetapi juga melihat dari pengaruh harga, ketersediaan pakan hewan, dan juga keterkaitan antara daging sapi dan produksi susu.

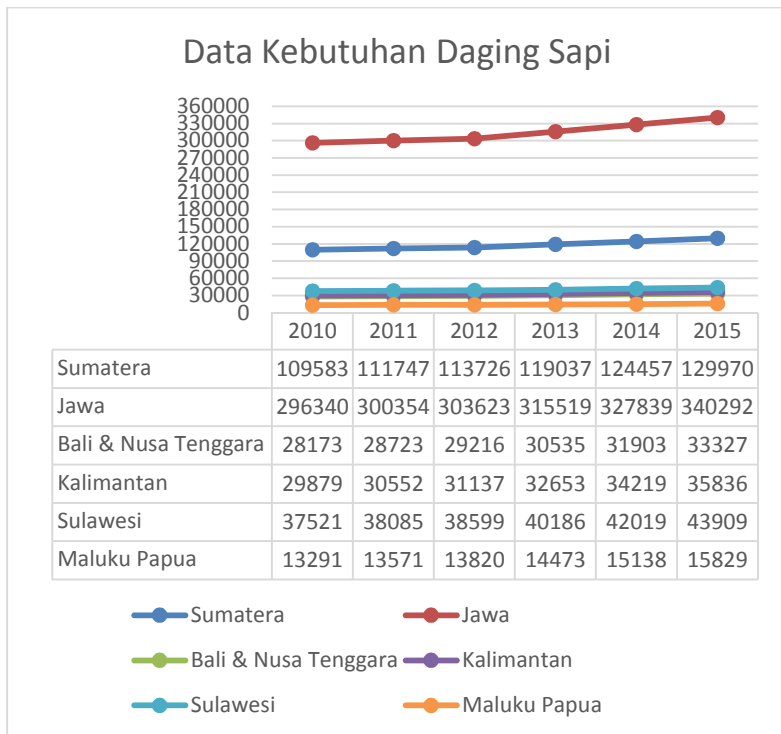
Kebutuhan daging sapi di Indonesia saat ini dipenuhi dari tiga sumber, yaitu peternakan rakyat (ternak lokal), industri peternakan rakyat (hasil penggemukan sapi ex-import), dan import daging dari luar negeri. Setiap tahun, Indonesia membutuhkan tambahan pasokan daging impor yang berasal dari 450.000 ekor sapi. Indonesia saat ini masih mengalami kekurangan pasokan sapi karena pertambahan populasi tidak seimbang dengan kebutuhan nasional [7].



**Gambar 2. 1 Data Produksi Daging Sapi**



Jika dilihat perbandingan antara jumlah produksi daging sapi dari tahun 2010 hingga 2015 (seperti terlihat pada Gambar 2. 1) dengan jumlah kebutuhan konsumsi di beberapa pulau di Indonesia (Gambar 2. 2), produksi daging sapi masih belum mencukupi permintaan konsumsi daging sapi untuk penduduk Indonesia. Peternakan domestik masih belum bisa memenuhi permintaan daging dalam negeri.



**Gambar 2. 2 Data Kebutuhan Daging Sapi**

### 2.3 Manajemen Rantai Pasok

Manajemen Rantai Pasok merupakan proses pengelolaan seluruh aktivitas perencanaan dan pengelolaan barang dan jasa dari

pembelian material, perencanaan produksi, proses transformasi dari produk masih dalam proses hingga menjadi produk jadi, dan penyimpanan baik bahan baku maupun produk jadi hingga penyampaian produk jadi ke kustomer akhir melalui sistem distribusi [3].



**Gambar 2. 3 Ilustrasi Manajemen Rantai pasok**

Terdapat 3 aliran proses yang terjadi dalam *supply chain* yaitu:

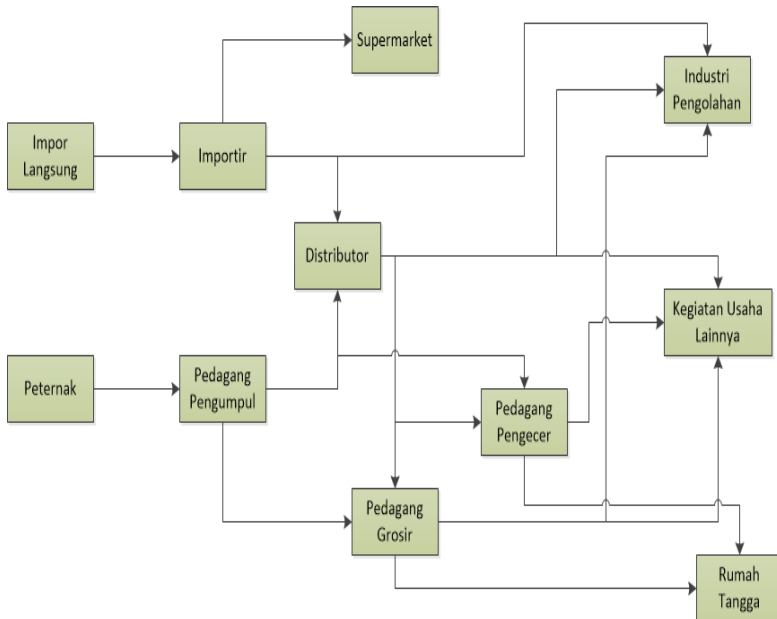
1. Aliran Material, yaitu aliran produk yang mengalir dari upstream ke downstream atau dari pemasok hingga ke pelanggan, contohnya bahan baku yang dikirim ke pabrik
2. Aliran Keuangan yang mengalir dari hilir ke hulu berupa invoice, term pembayaran, dan lain sebagainya.
3. Aliran Informasi terbentuk dari aliran material dan finansial. Aliran ini mengalirkan informasi dari supplier ke produsen berupa jumlah bahan baku yang dimiliki, status pengiriman dari bahan baku yang dipesan produsen, dll.

Manajemen Rantai Pasok memiliki 2 proses, yang pertama proses inbound terdiri dari penerimaan, penyimpanan, dan distribusi bahan-bahan masukan untuk menghasilkan produk dan jasa, kedua adalah proses outbound merupakan aktivitas-aktivitas yang melibatkan distribusi produk yang sudah jadi ke pelanggan. Menurut Min, H et al. (2002) *Supply Chain* merupakan sistem yang terintegrasi yang mana menyingkronkan serangkaian proses bisnis yang saling terkait, seperti [8]:

1. Memperoleh *raw material*
2. Mengubah bahan baku yang diperoleh sehingga menjadi produk jadi.

3. Menambahkan nilai produk jadi.
4. Mendistribusikan produk jadi ke retailer atau customer.
5. Memfasilitasi pertukaran informasi antara pelaku bisnis (supplier, manufacturer, distributor, retailer, dan customer).

Pada *Supply Chain Management* terdapat beberapa komponen yang terlibat dalam rantai pasok dapat dilihat pada Gambar 2. 4.



**Gambar 2. 4 Arsitektur SCM Distribusi Daging Sapi**

Menurut Suryati Herman (2004) SCM bertujuan untuk meningkatkan efektifitas dan efisiensi dari biaya, waktu, transaksi, dan mendapatkan kualitas yang lebih terjamin bagi barang atau jasa yang mengalir disepanjang rantai pasokan.

Tujuan manajemen rantai pasok sendiri selain untuk memenuhi permintaan konsumen juga untuk menguntungkan pihak-pihak *supply chain* yang terkait. Perencanaan rantai pasok juga dilakukan untuk mengintegrasikan aspek-aspek produksi-

distribusi-konsumen yang selama ini masih terpisah. Seperti peternak ditingkat produksi terpisah dengan pelaku distribusi yang dilakukan oleh pihak lain. Hubungan antar pihak tidak terdapat kerja sama jangka panjang yang memberikan manfaat antar pihak. Hal ini membuat rantai pasok menjadi tidak efisien.

#### **2.4 *E-Supply Chain Management***

*Electronic Supply Chain Management* merupakan suatu konsep manajemen yang memanfaatkan teknologi informasi untuk mengintegrasikan seluruh pelaku bisnis yang terlibat pada rantai pasok. Dengan adanya *electronic supply chain management* menjadikan hubungan antara supplier, distributor dan customer lebih mudah dalam berbagi informasi. Dalam melakukan perancangan *e-Supply Chain Management* ada beberapa modul-modul yang harus diperhatikan, yaitu [7]:

Beberapa hal yang harus diperhatikan dalam perancangan *e-Supply Chain Management* [9], yaitu:

##### **1. *Customer and Service Management***

Customer relationship menyediakan cara bagaimana hubungan dengan customer dibangun. Customer service memungkinkan pelanggan untuk melakukan pemesanan secara online, menyediakan sumber-sumber untuk informasi customer seperti ketersediaan produk, tanggal pengiriman dan status pemesanan [8].

##### **2. *Manufacturing and Supply Chain Planning***

Sebelum menggunakan SCM, proses manufacture produksi barang dilakukan dengan cara peramalan historical. Dengan SCM sekarang ini, memproduksi barang berdasarkan kebutuhan customer sehingga pengadaan barang menjadi lebih efektif dan efisien [8].

##### **3. *Supplier Relationship Management***

SRM merupakan proses sebuah perusahaan berhubungan dengan suppliernya.

##### **4. *Logistics Resource Management***

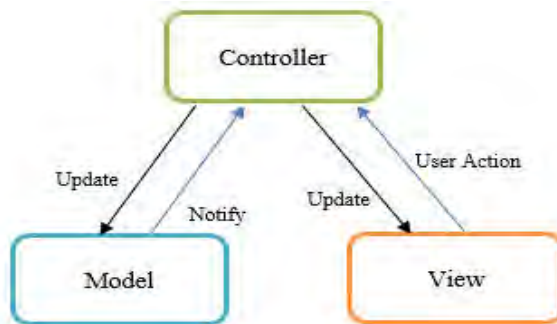
Dengan adanya e-SCM memudahkan perpindahan barang dan jasa diantara manufactur, distributor dan supplier ke konsumen akhir.

## 2.5 Kerangka Kerja CodeIgniter

CodeIgniter merupakan aplikasi *open source* yang berupa *frameeork* PHP dengan pola pengembangan MVC (Model, View, Controller) untuk membangun website dinamis. CodeIgniter memudahkan *developer* untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal. PHP digunakan untuk *developer* yang ingin membangun aplikasi web dengan fitur yang lengkap [10].

## 2.6 MVC (Model View Controller)

Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web. Terdapat 3 jenis komponen yang membangun suatu MVC pattern dalam suatu aplikasi, yaitu [10]:



**Gambar 2. 5 Model View Controller**

### 1. Model

Model merupakan bagian yang berhubungan dengan database untuk memanipulasi data (insert, update, delete), menangani

validasi dari bagian controller, namun tidak dapat berhubungan langsung dengan bagian view.

## 2. View

View merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data kepada user.

## 3. Controller

Controller merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

### 2.7 PHP: *HyperText Preprocessor*

PHP: *HyperText Preprocessor* merupakan bahasa skrip yang dapat disisipkan kedalam HTML. PHP biasa digunakan bagi web developer untuk pembuatan web dinamis. PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP. Pembatas PHP yang paling umum adalah “<?php” untuk membukan dan ?> untuk menutup kode PHP. Tujuan dari pembatas ini yaitu untuk memisahkan kode PHP dari kode diluar PHP seperti javascript dan HTML [11].

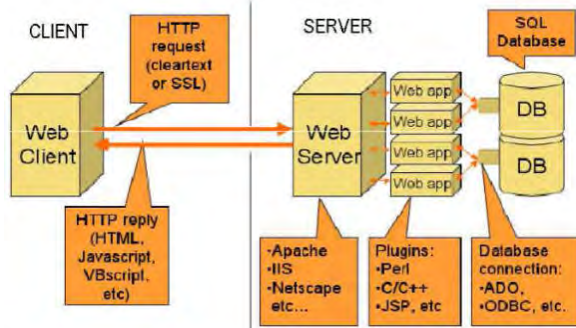
### 2.8 MySQL

MySQL merupakan sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS yang menggunakan struktur database relational. DBMS adalah perangkat lunak yang digunakan untuk mengelola, membuat, menghapus, memasukkan data, dan melakukan manajemen database lainnya [12].

### 2.9 Web Server

Web server merupakan suatu perangkat lunak yang menyediakan layanan akses kepada pengguna melalui protokol

komunikasi HTTP atau HTTPS atas berkas-berkas yang terdapat pada suatu situs web dalam layanan ke pengguna dengan menggunakan aplikasi tertentu. Web server yang paling banyak digunakan oleh web developer yaitu Apache Web Server [13].



**Gambar 2. 6** Arsitektur Web Server

*[Halaman ini sengaja dikosongkan]*



## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

Pada bab ini akan membahas analisis hasil wawancara proses bisnis sistem informasi manajemen rantai pasok distribusi daging sapi pada rumah potong hewan dan supermarket serta perancangan sistem dari hasil analisis kebutuhan. Analisis hasil wawancara akan menghasilkan suatu proses bisnis yang bersifat umum sesuai dengan pengerjaan tugas akhir. Perancangan sistem akan mencantumkan analisis kebutuhan yang diperlukan di perangkat lunak setelah mendapatkan proses bisnis. Perancangan sistem ini akan direpresentasikan dengan menggunakan diagram *Unified Modelling Language* (selanjutnya disebut UML).

#### **3.1 Analisis**

Tahap analisis dibagi menjadi beberapa bagian, antara lain analisis proses bisnis, deskripsi umum sistem, serta spesifikasi kebutuhan perangkat lunak yang terdiri dari kasus penggunaan sistem dan kebutuhan perangkat lunak.

##### **3.1.1 Analisis Sistem Yang Sedang Berjalan**

Berdasarkan hasil survey dan wawancara yang telah dilakukan pada 2 pihak yaitu supermarket Superindo dan Rumah Potong Hewan (RPH), didapat beberapa informasi mengenai proses bisnis yang sedang berjalan saat ini mengenai distribusi daging sapi.

Dari hasil wawancara dengan pihak RPH diketahui bahwa pihak RPH hanya bertindak sebagai fasilitator pemotongan Daging Sapi yang datang dari Jagal. Dari wawancara yang telah dilakukan didapat aliran informasi sebagai berikut:

- Aliran informasi antara pedagang sapi hidup dengan jagal:  
Aliran informasi antara pedagang sapi hidup dengan jagal terkait dengan jumlah permintaan terhadap sapi potong hidup.
- Aliran informasi penjagal dengan pedangan pengecer:

Aliran informasi antara penjalag dan pedagang pengecer adalah terkait dengan jumlah daging sapi yang akan diminta oleh pedagang pengecer. Informasi yang mengalir dari pedagang pengecer yang melakukan sistem pembayaran diakhir berkaitan dengan jumlah daging sapi yang terjual setiap harinya. Aliran informasi yang mengalir dari jagal ke pedagang pengecer berkaitan dengan jumlah stok daging sapi dan harga daging sapi ditingkat pedagang pengecer.

Sedangkan dari hasil wawancara dengan pihak SuperIndo didapatkan kekurangan dari sistem yang sudah digunakan saat ini, yaitu sistem yang sudah ada tidak dapat melihat jumlah stock barang yang dimiliki oleh distributornya. Sehingga hanya bisa diketahui ketika barang yang telah dipesan datang tidak sesuai dengan jumlah permintaan.

### 3.1.2 Analisis Proses Bisnis

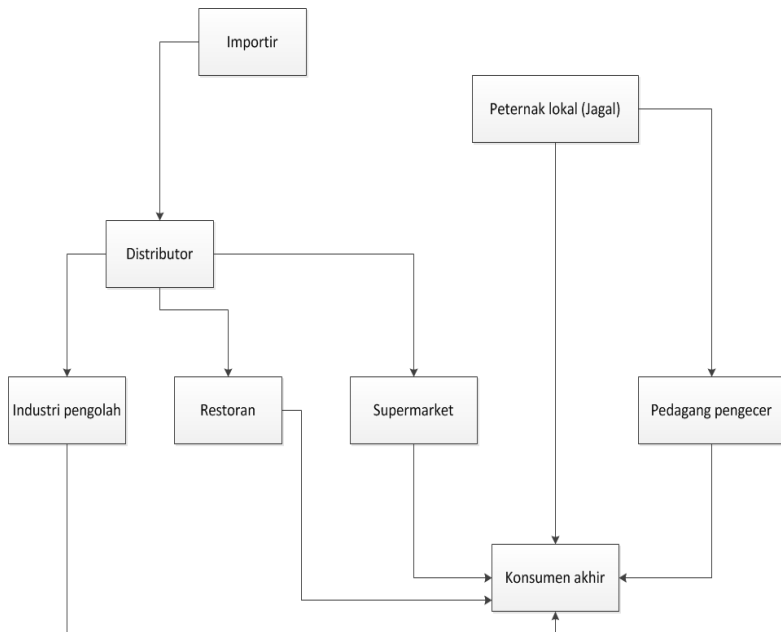
Berdasarkan hasil wawancara dengan beberapa pihak, diketahui bahwa sistem transaksi yang digunakan saat ini masih memiliki beberapa kekurangan. Dengan menggunakan metode Manajemen Rantai Pasok dibuat Analisis kebutuhan untuk proses bisnis sistem informasi distribusi daging sapi yaitu seperti terlihat pada Tabel 3. 1.

**Tabel 3. 1 Analisis Kebutuhan Distribusi Daging Sapi**

No	Analisis Kebutuhan Sistem Distribusi
1	Terdapat proses untuk melihat stok barang
2	Terdapat proses memasukkan jumlah stok barang
3	Adanya proses permintaan produk
4	Adanya proses penerimaan order produk
5	Terdapat proses penerimaan balasan tentang informasi pemenuhan permintaan produk
6	Terdapat informasi pengiriman transaksi yang disetujui
7	Adanya pembuatan faktur bukti transaksi

### 3.1.3 Deskripsi Umum Sistem

Sistem informasi manajemen rantai pasok ini dibuat dengan tujuan untuk memudahkan pihak-pihak yang terkait dengan distribusi daging sapi untuk melakukan transaksi pemesanan berdasarkan jumlah stok barang tersedia serta jumlah permintaan barang. Sistem informasi ini bisa digunakan oleh orang-orang yang sudah ter-registrasi pada sistem. Sistem informasi ini bisa memberikan informasi jumlah pasokan yang tersedia sehingga memudahkan customer untuk melakukan pemesanan. Untuk alur distribusi daging sapi bisa dilihat pada ilustrasi diagram pada Gambar 3. 1 berikut.



**Gambar 3. 1 Alur Distribusi Daging Sapi**

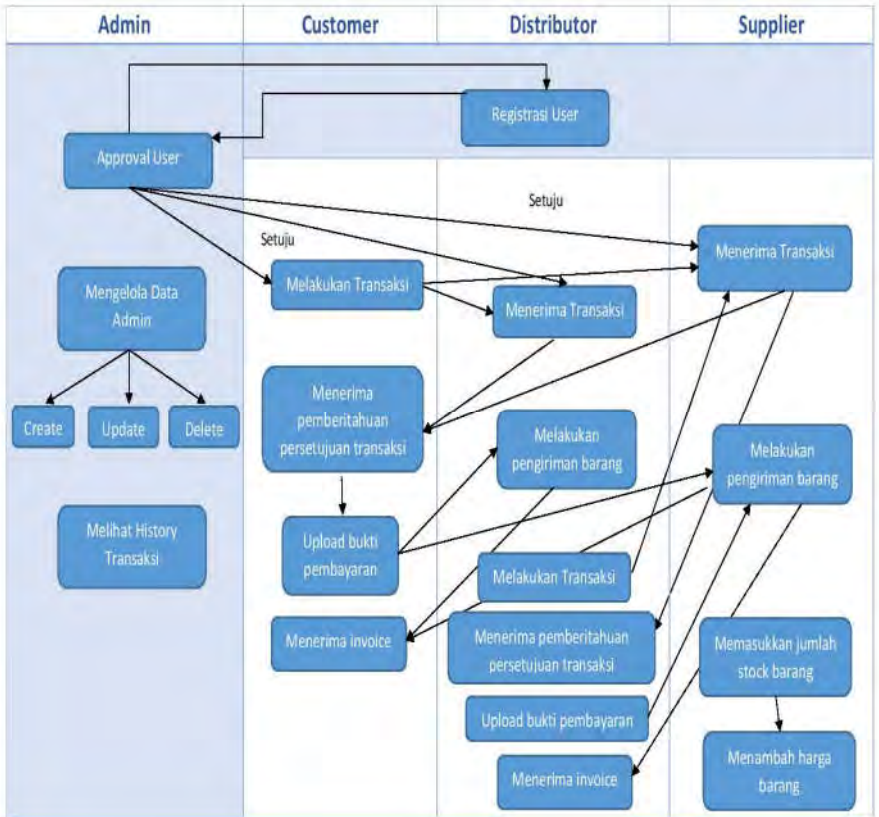
Berdasarkan Gambar 3. 1 terdapat 2 jenis alur distribusi yaitu distribusi pada daging sapi import dan distribusi pada daging

sapi lokal. Alur distribusi daging sapi pada umumnya berdasarkan ilustrasi pada Gambar 3. 1 menghubungkan antara customer, distributor dan supplier secara langsung. Proses yang terjadi saat melakukan transaksi pada sistem informasi manajemen rantai pasok ini yaitu, customer melakukan pemesanan daging sapi pada distributor atau langsung ke supplier berdasarkan stok daging sapi yang dimiliki oleh distributor atau supplier yang bersangkutan. Ketika customer telah selesai melakukan transaksi maka data transaksi tersebut akan masuk ke sistem distributor atau supplier terkait. Distributor atau supplier terkait bisa melakukan persetujuan atau menolak pemesanan produk dari customer. Proses selanjutnya yaitu ketika distributor atau supplier telah menyetujui permintaan maka customer akan mendapatkan pemberitahuan melalui sms apakah transaksi disetujui atau ditolak. Jika transaksi dari customer disetujui maka proses selanjutnya yaitu customer melakukan pembayaran dan meng-upload bukti pembayaran pada sistem. Bukti upload otomatis akan masuk pada sistem distributor atau supplier terkait. Ketika bukti telah diterima oleh distributor atau supplier, maka supplier akan melakukan pengiriman barang serta mencetak bukti faktur pembelian untuk customer. Proses selanjutnya customer akan mendapatkan pemberitahuan bahwa order telah dikirim melalui sms. Alur distribusi melalui sistem informasi yang dibangun dapat dilihat pada Gambar 3. 2.

### **3.1.4 Spesifikasi Kebutuhan Perangkat Lunak**

Sesuai dengan uraian mengenai cakupan perangkat lunak yang dibangun, dibutuhkan adanya spesifikasi perangkat lunak agar dapat memberikan solusi dari permasalahan yang diberikan dan dapat bekerja dengan baik dalam mengakomodasi kebutuhan. Diharapkan dengan adanya spesifikasi ini dapat menyesuaikan kebutuhan-kebutuhan pengguna berdasarkan wawancara yang telah dilakukan. Spesifikasi kebutuhan perangkat lunak adalah penjelasan mengenai kebutuhan sistem yang diinginkan pelanggan atau klien dalam bentuk tulisan. Spesifikasi kebutuhan perangkat

lunak pada tugas akhir ini terdiri dari kebutuhan fungsional yang dapat dilihat pada Gambar 3. 2.



Gambar 3. 2 Alur Kerja Sistem Distribusi Daging Sapi

### 3.1.4.1 Kebutuhan Fungsional

Kebutuhan fungsional dari sistem informasi manajemen rantai pasok ini dapat dilihat pada Tabel 3. 2.

Tabel 3. 2 Kebutuhan Fungsional Perangkat Lunak

Kode Kebutuhan	Deskripsi Kebutuhan
----------------	---------------------

FR-001	User dapat melakukan proses pemesanan produk
FR-002	User dapat menerima permintaan pasokan
FR-003	User dapat menerima balasan persetujuan proses transaksi
FR-004	User dapat melakukan proses inventori produk
FR-005	User dapat melihat jumlah ketersediaan stok barang dari supplier dan distributor
FR-006	User dapat memasukkan daftar harga produk
FR-007	Dapat melakukan persetujuan data user
FR-008	Dapat melakukan registrasi ke sistem
FR-009	Dapat menampilkan history transaksi yang telah dilakukan

### 3.1.4.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional sistem distribusi daging sapi bisa dilihat pada Tabel 3. 3.

**Tabel 3. 3 Kebutuhan Non-Fungsional**

<b>Kode Kebutuhan</b>	<b>Deskripsi Kebutuhan</b>
NFR-001	Menggunakan koneksi internet
NFR-002	Memiliki username dan password
NFR-003	Memiliki <i>error handling</i>
NFR-004	Tampilan aplikasi <i>user friendly</i>
NFR-005	Menggunakan Bahasa Indonesia

### 3.1.4.3 Aktor

Pengertian aktor adalah pihak-pihak, baik manusia maupun sistem/perangkat lunak lain yang terlibat dan berinteraksi langsung dengan sistem. Dalam sistem untuk tugas akhir ini

memiliki 4 aktor secara umum, yaitu admin, supplier, distributor, dan customer. Penjelasan lebih rinci mengenai aktor dari sistem ini sebagai berikut:

- Admin

Admin berperan sebagai pengelola data user (customer, distributor, dan supplier). Pada sistem ini admin yang akan mengelola dan memantau keseluruhan proses transaksi yang terjadi. Admin akan menyetujui atau menolak user-user yang telah ter-registrasi ke sistem. Selain itu, admin juga bisa melihat history transaksi yang telah dilakukan antar user. Admin pada sistem ini merupakan pihak pemerintah (Dinas Peternakan) yang akan memiliki otoritas pada sistem.

- Customer

Customer disini dibagi menjadi 2 bagian, yaitu pedagang pengecer yg termasuk ke kategori masyarakat tradisional serta supermarket, restaurant, dan industri pengolah daging yang termasuk ke importir. Customer hanya bisa melakukan pemesanan product daging sapi, melihat jumlah stok daging sapi yang tersedia dari setiap supplier maupun distributor. Customer bisa melakukan pemesanan baik melalui distributor maupun langsung ke supplier.

- Distributor

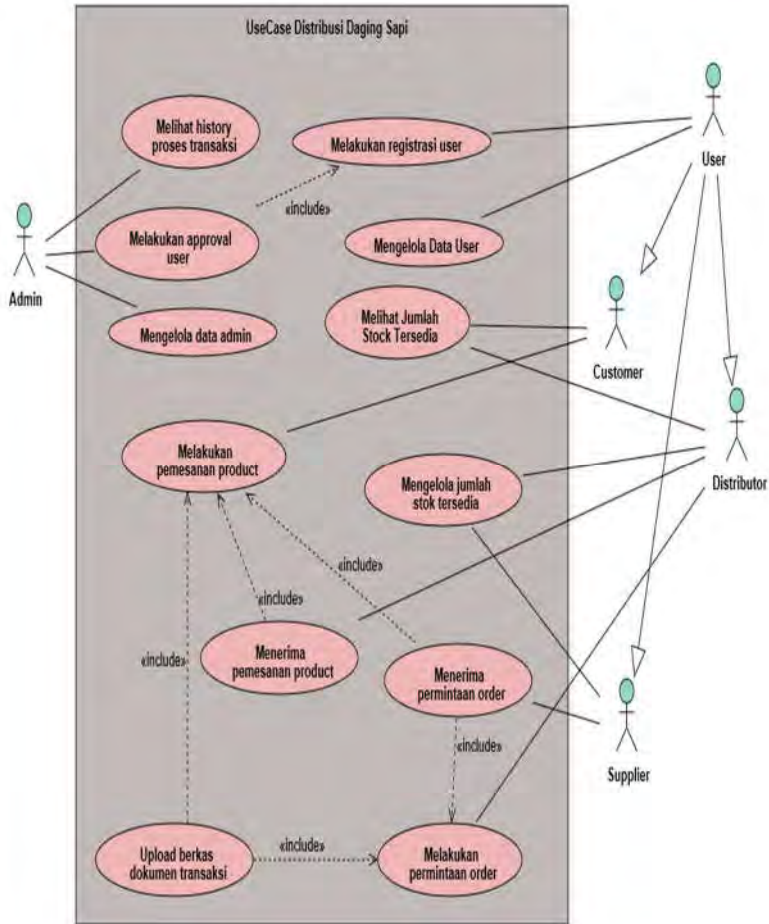
Distributor berasal dari perusahaan distributor daging sapi. Peran distributor yaitu bisa melakukan pemesanan barang ke supplier dan juga menerima pemesanan dari customer. Distributor bisa menampilkan jumlah stok produk yang tersedia.

- Supplier

Supplier terdiri dari 2 bagian, yaitu jagal dan pihak importir. Jagal nantinya akan berhubungan dengan pedagang pengecer sedangkan pihak importir akan berhubungan dengan distributor serta customer dari supermarket, restaurant dan industri pengolah. Supplier bisa menerima pemesanan barang dan juga menampilkan jumlah stok daging sapi yang tersedia.

### 3.1.4.4 Kasus Penggunaan Sistem

Bagian ini menjelaskan secara rinci kasus penggunaan yang terdapat pada perangkat lunak. Selain itu, terdapat juga spesifikasi kasus penggunaan, diagram aktivitas dan diagram urutan untuk tiap-tiap kasus penggunaan. Kasus penggunaan dapat dilihat pada Gambar 3. 3.



Gambar 3. 3 Diagram Kasus Penggunaan



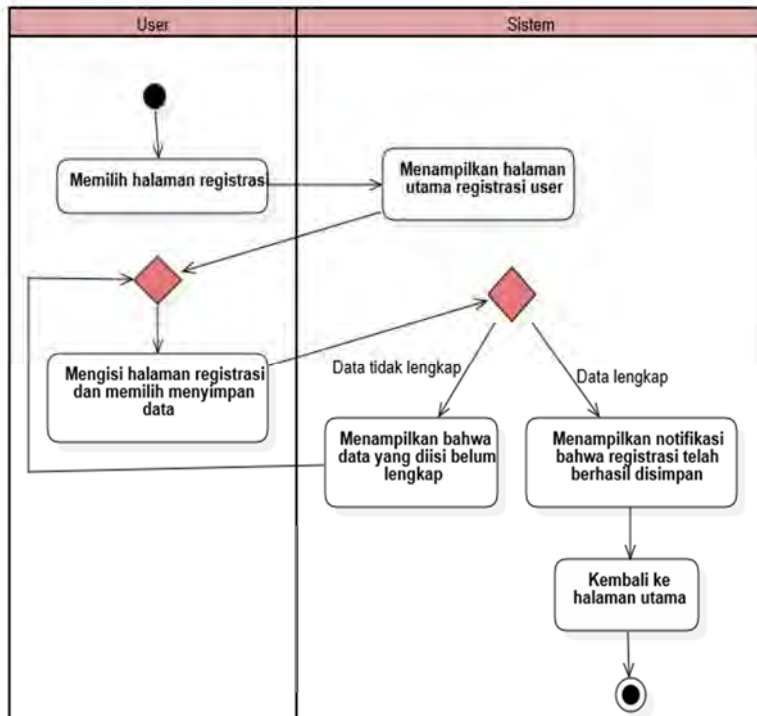
### 3.1.4.5 Kasus Penggunaan Melakukan Registrasi User

Pada kasus penggunaan ini, aktor melakukan registrasi sebelum bisa masuk ke aplikasi. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 4 dan diagram aktivitas terdapat pada Gambar 3. 4.

**Tabel 3. 4 Kasus Penggunaan Melakukan Registrasi User**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Melakukan Registrasi User
<b>Nomor</b>	UC-001
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana user melakukan registrasi akun.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Supplier, Distributor, Customer
<b>Kondisi Awal</b>	User belum memiliki akun dan melakukan registrasi
<b>Kondisi Akhir</b>	User telah ter-registrasi ke sistem dan bisa melakukan login
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. User (Customer, distributor dan supplier) memilih halaman registrasi.</li> <li>2. Sistem menampilkan halaman utama registrasi.</li> <li>3. User mengisi halaman registrasi dan memilih menyimpan data. [Alternative A3]</li> <li>4. Sistem menampilkan notifikasi bahwa registrasi telah berhasil disimpan dan kembali ke halaman utama.</li> </ol>
<b>Alur Alternative</b>	A.3 Data yang dimasukkan belum lengkap

1. Sistem menampilkan notifikasi bahwa data yang diisi belum lengkap.
2. User mengisi registrasi yang belum diisi.
3. Sistem menampilkan notifikasi bahwa data registrasi telah berhasil disimpan dan kembali ke halaman utama.



Gambar 3. 4 Diagram Aktivitas Melakukan Registrasi Akun

### 3.1.4.6 Kasus Penggunaan melakukan Approval User

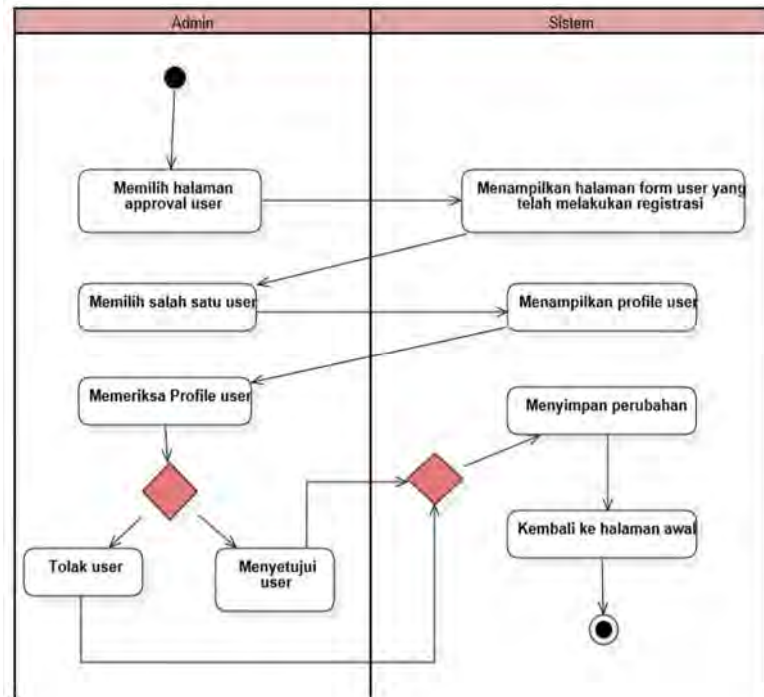
Pada kasus penggunaan ini, aktor sebagai admin melakukan approval kepada user yang telah melakukan registrasi.

Persetujuan dari admin menentukan apakah user yang telah terregistrasi bisa melakukan transaksi atau tidak melalui sistem. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3.5 dan diagram aktivitas terdapat pada Gambar 3. 5.

**Tabel 3.5 Kasus Penggunaan Melakukan Approval User**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Melakukan Approval User
<b>Nomor</b>	UC-002
<b>Deskripsi</b>	Kasus penggunaan ini digunakan untuk menyetujui akun dari user-user yang telah melakukan registrasi. User yang telah registrasi hanya bisa melihat halaman home dan halaman profile saja. Jika admin telah menyetujui akun mereka maka user bisa melakukan transaksi jual beli.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Admin
<b>Kondisi Awal</b>	Akun user yang telah registrasi belum disetujui
<b>Kondisi Akhir</b>	Akun user disetujui atau ditolak oleh admin
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Admin memilih halaman approval user pada bagian home admin</li> <li>2. Sistem menampilkan halaman user-user yang telah registrasi</li> <li>3. Admin memilih salah satu nama user yang akan di approve</li> <li>4. Sistem menampilkan profile user</li> <li>5. Admin memeriksa profil user</li> <li>6. Admin menyetujui user</li> </ol> <p>[Alternative A6]</p>

	7. Sistem menyimpan perubahan persetujuan dan kembali ke halaman approval.
<b>Alur Alternative</b>	A.6. Admin tolak user 1. Admin memilih tolak user. 2. Sistem menyimpan perubahan persetujuan.



Gambar 3. 5 Diagram Aktivitas Melakukan Approval User

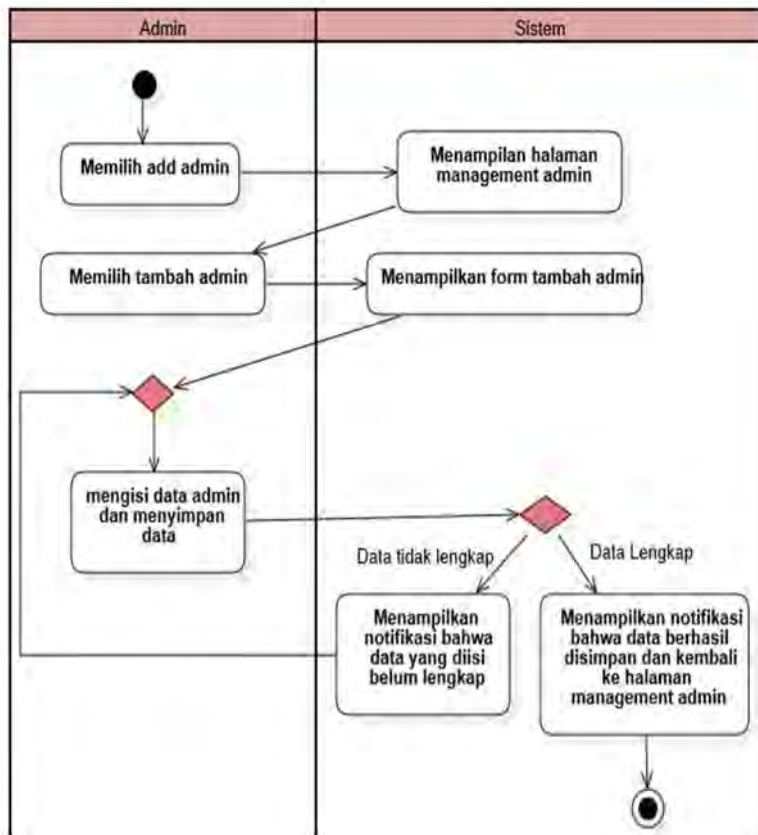
### 3.1.4.7 Kasus Penggunaan Menambah Data Admin

Pada kasus penggunaan ini, admin bisa menambah data admin baru. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 6 dan diagram aktivitas terdapat pada Gambar 3. 6.

Tabel 3. 6 Kasus Penggunaan Menambah Data Admin

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Menambah Data Admin
<b>Nomor</b>	UC-003
<b>Deskripsi</b>	Use-case ini mendeskripsikan user menambah data admin baru.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Admin
<b>Kondisi Awal</b>	Data admin baru belum tersedia
<b>Kondisi Akhir</b>	Data admin baru tersimpan
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Admin memilih bagian add admin.</li> <li>2. Sistem menampilkan halaman management admin.</li> <li>3. Admin memilih tambah admin.</li> <li>4. Sistem menampilkan halaman tambah admin.</li> <li>5. Admin mengisi data-data yang diperlukan dan menyimpan data. [Alternative A5]</li> <li>6. Sistem menampilkan notifikasi bahwa data berhasil disimpan dan kembali ke halaman management admin.</li> </ol>
<b>Alur Alternatif</b>	<p>A.5 Data yang dimasukkan belum lengkap</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan notifikasi bahwa data yang diisi belum lengkap.</li> <li>2. Admin mengisi data registrasi yang belum diisi.</li> <li>3. Sistem menampilkan notifikasi bahwa registrasi telah berhasil</li> </ol>

disimpan dan kembali ke halaman management admin.



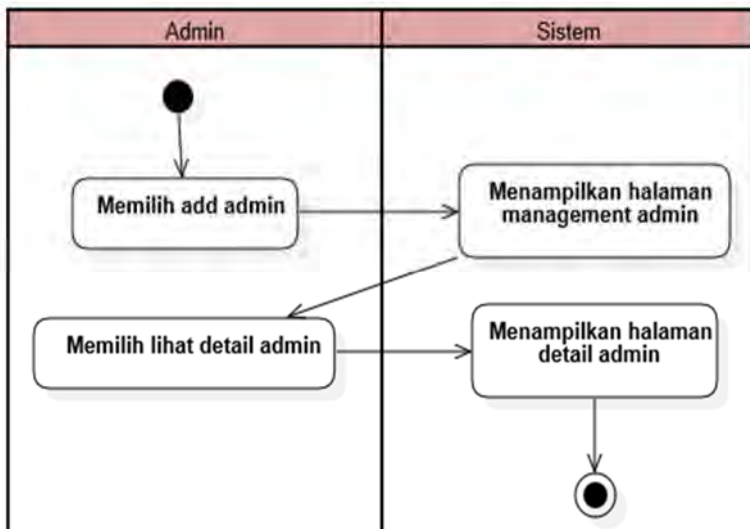
**Gambar 3. 6 Diagram Aktivitas Menambah Data Admin**

### 3.1.4.8 Kasus Penggunaan Melihat Detail Data Admin

Pada kasus penggunaan ini, admin bisa melihat data admin itu sendiri dan juga data admin lainnya. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 7 dan diagram aktivitas terdapat pada Gambar 3. 7.

Tabel 3. 7 Kasus Penggunaan Melihat Detail Data Admin

Komponen	Deskripsi
<b>Nama</b>	Melihat Detail Data Admin
<b>Nomor</b>	UC-004
<b>Deskripsi</b>	Use-case ini mendeskripsikan user melihat data admin
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Admin
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Data admin tersedia
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Admin memilih add admin.</li> <li>2. Sistem menampilkan halaman management admin.</li> <li>3. Admin memilih lihat detail admin.</li> <li>4. Sistem menampilkan halaman detail admin.</li> </ol>



Gambar 3. 7 Diagram Aktivitas Melihat Detail Data Admin

### 3.1.4.9 Kasus Penggunaan Mengubah Data Admin

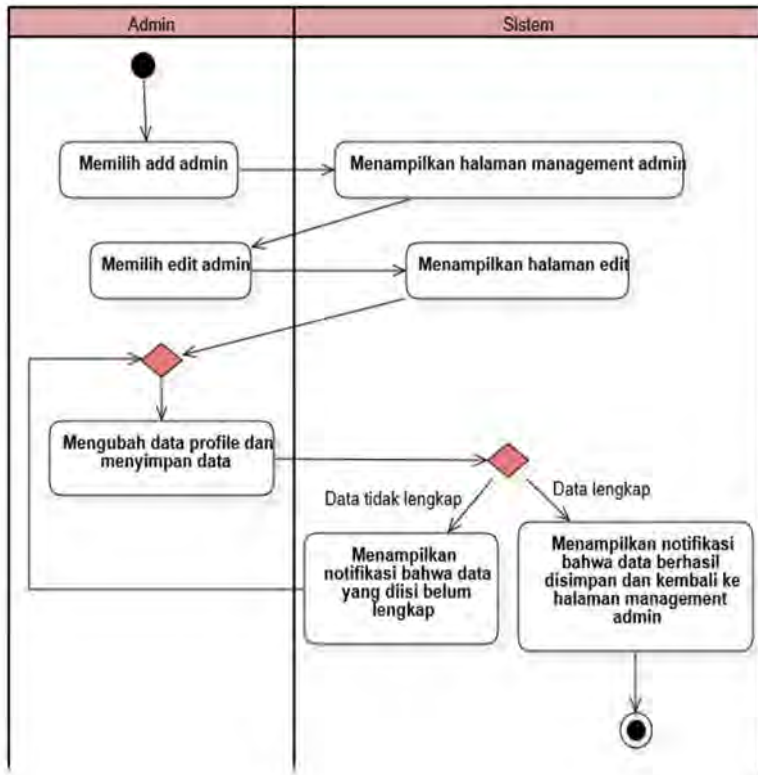
Pada kasus penggunaan ini, admin bisa mengubah data admin itu sendiri. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 8 dan diagram aktivitas terdapat pada Gambar 3. 8.

**Tabel 3. 8 Kasus Penggunaan Mengubah Data Admin**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Mengubah Data Admin
<b>Nomor</b>	UC-005
<b>Deskripsi</b>	Use-case ini mendeskripsikan user mengubah data admin.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Admin
<b>Kondisi Awal</b>	Data admin yang lama
<b>Kondisi Akhir</b>	Data admin yang baru tersimpan
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Admin memilih add admin.</li> <li>2. Sistem menampilkan halaman management admin.</li> <li>3. Admin memilih edit admin.</li> <li>4. Sistem menampilkan halaman edit.</li> <li>5. Admin mengubah data profile dan menyimpan data. [Alternative A5]</li> <li>6. Sistem menampilkan notifikasi bahwa data berhasil disimpan dan kembali ke halaman management admin.</li> </ol>
<b>Alur Alterntive</b>	<p>A.5 Data yang dimasukkan belum lengkap</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan notifikasi bahwa data yang diisi belum lengkap.</li> </ol>



- |  |  |
|--|--|
|  | <ol style="list-style-type: none"> <li>2. Admin mengisi form registrasi yang belum diisi.</li> <li>3. Sistem menampilkan notifikasi bahwa registrasi telah berhasil disimpan dan kembali ke halaman management admin.</li> </ol> |
|--|--|



Gambar 3. 8 Diagram Aktivitas Mengubah Data Admin

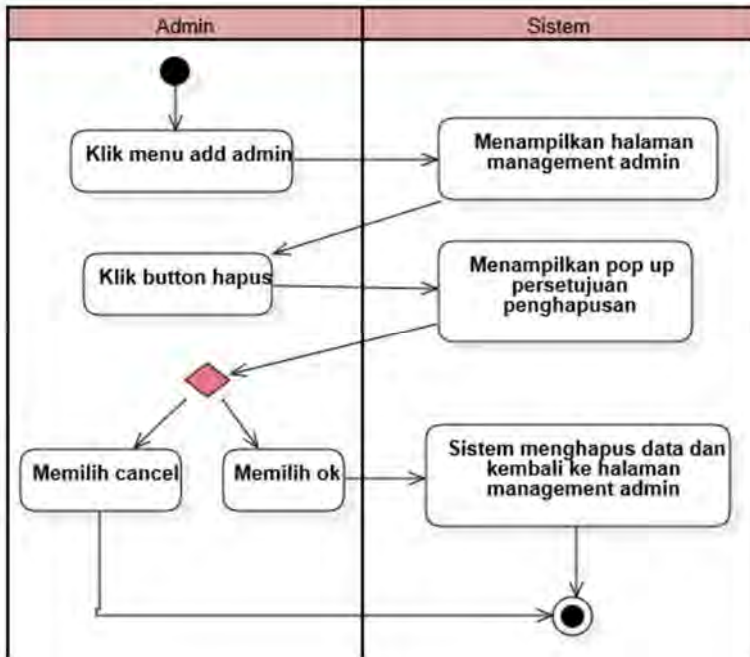
### 3.1.4.10 Kasus Penggunaan Menghapus Data Admin

Pada kasus penggunaan ini, admin bisa menghapus data admin itu sendiri dan juga data admin lainnya. Rincian kasus

penggunaan tersebut dapat dilihat pada Tabel 3. 9 dan diagram aktivitas terdapat pada Gambar 3. 9.

**Tabel 3. 9 Kasus Penggunaan Menghapus Data Admin**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Menghapus Data Admin
<b>Nomor</b>	UC-006
<b>Deskripsi</b>	Use-case ini mendeskripsikan user menghapus data admin.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Admin
<b>Kondisi Awal</b>	Data admin masih tersedia
<b>Kondisi Akhir</b>	Data admin terhapus
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Admin memilih add admin.</li> <li>2. Sistem menampilkan halaman management admin.</li> <li>3. Admin memilih hapus admin.</li> <li>4. Sistem menampilkan popup persetujuan penghapusan data.</li> <li>5. Admin memilih setuju.</li> </ol> <p>[Alternative A5]</p> <ol style="list-style-type: none"> <li>6. Sistem menghapus data dan kembali ke halaman management admin.</li> </ol>
<b>Alur Alterntive</b>	<p>A.5 Admin memilih cancel</p> <ol style="list-style-type: none"> <li>1. Sistem kembali ke halaman management admin.</li> </ol>



Gambar 3. 9 Diagram Aktivitas Menghapus Data Admin

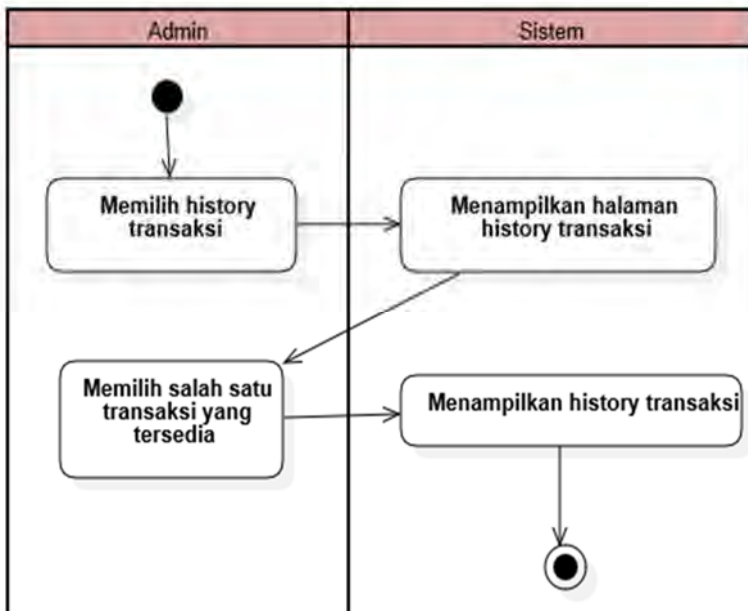
### 3.1.4.11 Kasus Penggunaan Melihat History Proses Transaksi

Pada kasus penggunaan ini, aktor melihat dan memantau transaksi yang telah dilakukan oleh user. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 10 dan diagram aktivitas terdapat pada Gambar 3. 10.

Tabel 3. 10 Kasus Penggunaan Melihat History Transaksi

Komponen	Deskripsi
<b>Nama</b>	Melihat History Transaksi
<b>Nomor</b>	UC-007
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana Admin melihat proses transaksi yang

	telah dilakukan antara Customer, Distributor dan Supplier.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Admin
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Admin melihat history transaksi yang telah dilakukan
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Admin memilih history transaksi.</li> <li>2. Sistem menampilkan halaman history transaksi.</li> <li>3. Admin memilih salah satu transaksi yang tersedia</li> <li>4. Sistem menampilkan history transaksi</li> </ol>



Gambar 3. 10 Diagram Aktivitas Melihat History Transaksi

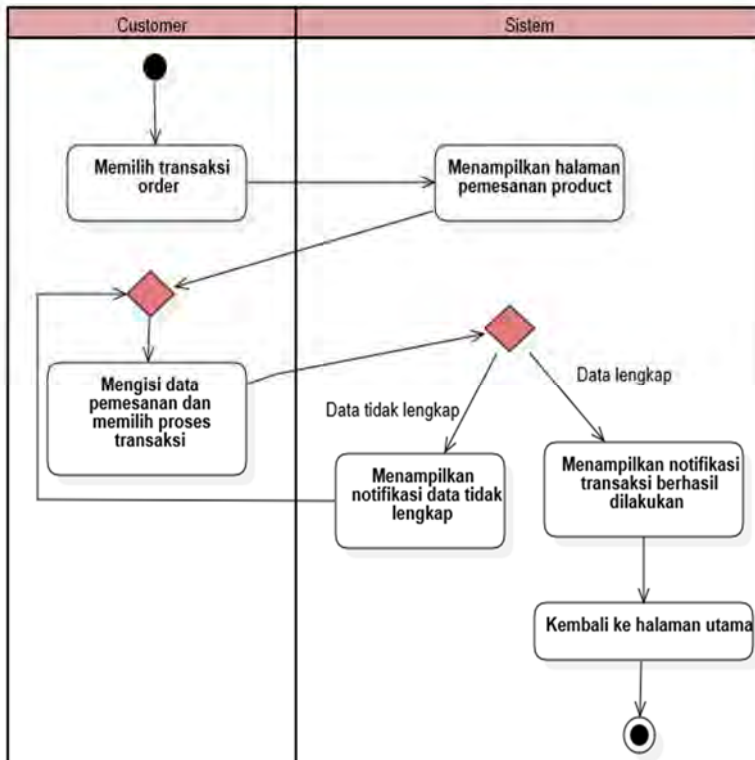
### 3.1.4.12 Kasus Penggunaan Melakukan Pemesanan Produk

Pada kasus penggunaan ini, aktor telah bisa melakukan transaksi yaitu pemesanan produk. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 11 dan diagram aktivitas pada Gambar 3. 11.

**Tabel 3. 11 Kasus Penggunaan Melakukan Pemesanan Produk**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Melakukan Pemesanan Produk
<b>Nomor</b>	UC-008
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana Customer melakukan transaksi pemesanan product baik ke Distributor maupun langsung transaksi ke Supplier.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Customer
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Aktor melakukan transaksi pemesanan produk
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Customer memilih transaksi pemesanan order.</li> <li>2. Sistem menampilkan halaman pemesanan product.</li> <li>3. Customer mengisi data pemesanan dan memilih poses transaksi. [Alternative A3]</li> <li>4. Sistem menampilkan notifikasi bahwa transaksi berhasil dilakukan dan kembali ke halaman transaksi.</li> </ol>
<b>Alur Alternative</b>	<p>A.3. Isian form tidak lengkap</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan notifikasi bahwa isian form tidak lengkap.</li> </ol>

	<p>2. Customer kembali mengisi data pemesanan dan memilih proses transaksi.</p> <p>3. Sistem menampilkan notifikasi bahwa transaksi berhasil dilakukan dan kembali ke halaman utama.</p>
--	--



Gambar 3. 11 Diagram Aktivitas Melakukan Pemesanan Produk

### 3.1.4.13 Kasus Penggunaan Upload Berkas Dokumen Transaksi

Pada kasus penggunaan ini, aktor melakukan upload berkas ketika transaksi yang dilakukan disetujui. Rincian kasus

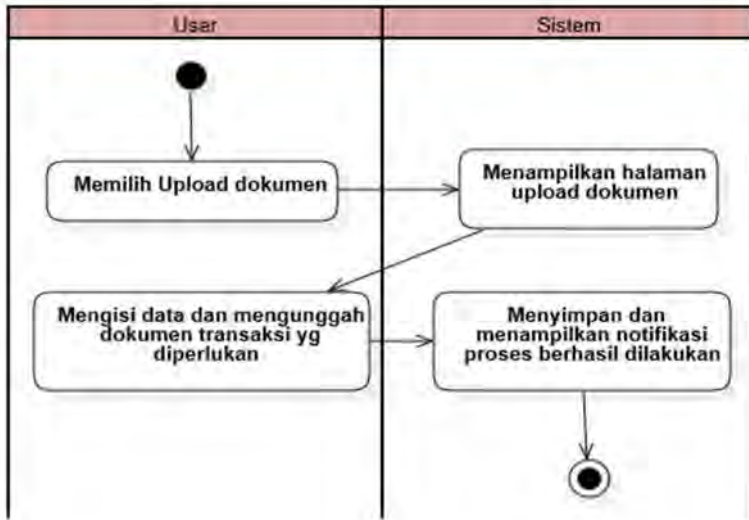
penggunaan tersebut dapat dilihat pada Tabel 3.12 dan diagram aktivitas terdapat pada Gambar 3. 12.

**Tabel 3. 12 Kasus Penggunaan Upload Berkas Dokumen Transaksi**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Upload Berkas Dokumen Transaksi
<b>Nomor</b>	UC-009
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana aktor melakukan upload berkas pembuktian pembayaran transaksi ketika transaksi yang dilakukan telah disetujui
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Customer, Distributor
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Aktor melakukan upload berkas bukti pembayaran.
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih Upload dokumen</li> <li>2. Sistem menampilkan halaman Upload dokumen</li> <li>3. Aktor mengisi data dan mengunggah dokumen transaksi yang diperlukan</li> <li>4. Sistem menyimpan dan menampilkan notifikasi bahwa proses berhasil dilakukan.</li> </ol>

#### **3.1.4.14 Kasus Penggunaan Menerima Proses Permintaan Order**

Pada kasus penggunaan ini, aktor menerima permintaan order yang masuk. Aktor akan melakukan persetujuan atau menolak permohonan order yang diterima. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 13 dan diagram aktivitas terdapat Gambar 3. 13.



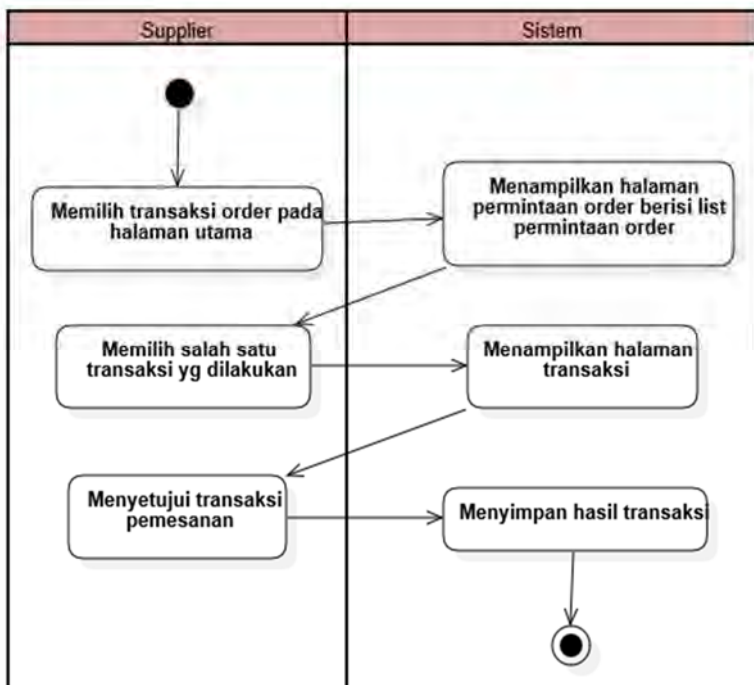
Gambar 3. 12 Diagram Aktivitas Upload Berkas Dokumen Transaksi

Tabel 3. 13 Kasus Penggunaan Menerima Proses Permintaan Order

Komponen	Deskripsi
<b>Nama</b>	Menerima proses permintaan order
<b>Nomor</b>	UC-010
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana Supplier menerima permintaan order yang datang dari Customer maupun Distributor.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Supplier
<b>Kondisi Awal</b>	Transaksi belum disetujui atau ditolak oleh supplier
<b>Kondisi Akhir</b>	Transaksi telah disetujui atau ditolak oleh supplier
<b>Alur Normal</b>	1. Supplier memilih Transaksi Order pada halaman utama.



	<ol style="list-style-type: none"> <li>2. Sistem menampilkan halaman transaksi order berisi list permintaan order dari Customer maupun Distributor.</li> <li>3. Supplier memilih salah satu transaksi yang akan dilakukan.</li> <li>4. Sistem menampilkan halaman transaksi.</li> <li>5. Supplier menyetujui transaksi pemesanan.</li> <li>6. Sistem menyimpan hasil transaksi.</li> </ol>
--	--



Gambar 3. 13 Diagram Aktivitas Menerima Proses Permintaan Order

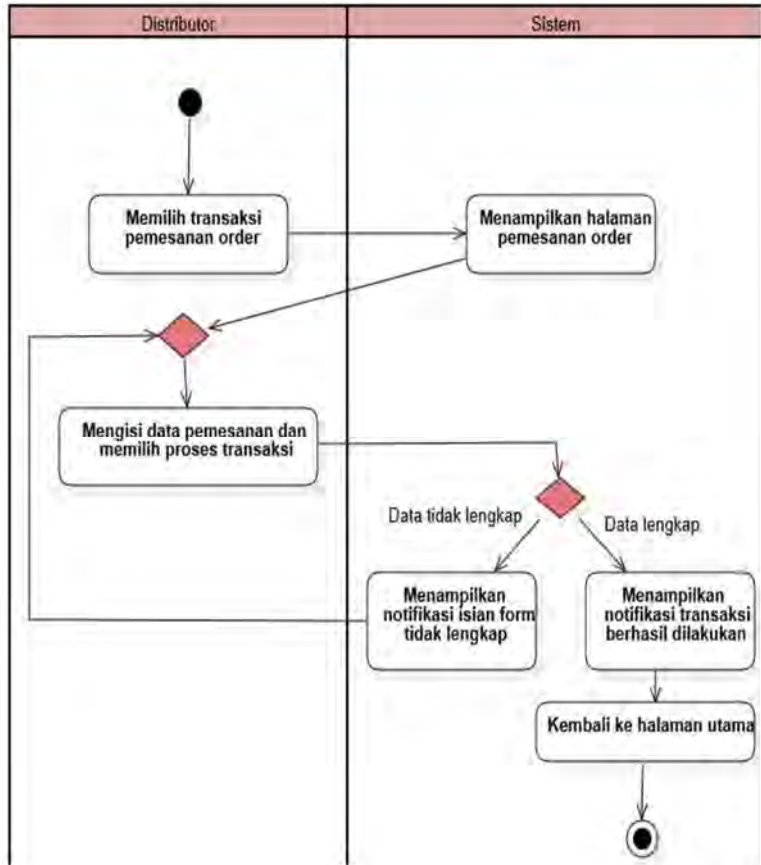
### 3.1.4.15 Kasus Penggunaan Melakukan Permintaan Order

Pada kasus penggunaan ini, aktor telah bisa melakukan transaksi yaitu pemesanan produk. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 14 dan diagram aktivitas pada Gambar 3. 14.

**Tabel 3. 14 Kasus Penggunaan Melakukan Permintaan Order**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Melakukan Pemesanan Produk
<b>Nomor</b>	UC-011
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana Distributor melakukan transaksi pemesanan product ke Supplier.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Distributor
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Aktor melakukan transaksi pemesanan produk
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Distributor memilih transaksi pemesanan order.</li> <li>2. Sistem menampilkan halaman pemesanan product.</li> <li>3. Distributor mengisi data pemesanan dan memilih poses transaksi. [Alternative A3]</li> <li>4. Sistem menampilkan notifikasi bahwa transaksi berhasil dilakukan dan kembali ke halaman transaksi.</li> </ol>
<b>Alur Alternative</b>	<p>A.3. Isian form tidak lengkap</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan notifikasi bahwa isian form tidak lengkap.</li> <li>2. Distributor kembali mengisi data pemesanan dan memilih proses transaksi.</li> </ol>

3. Sistem menampilkan notifikasi bahwa transaksi berhasil dilakukan dan kembali ke halaman utama.



Gambar 3. 14 Diagram Aktivitas Melakukan Permintaan Order

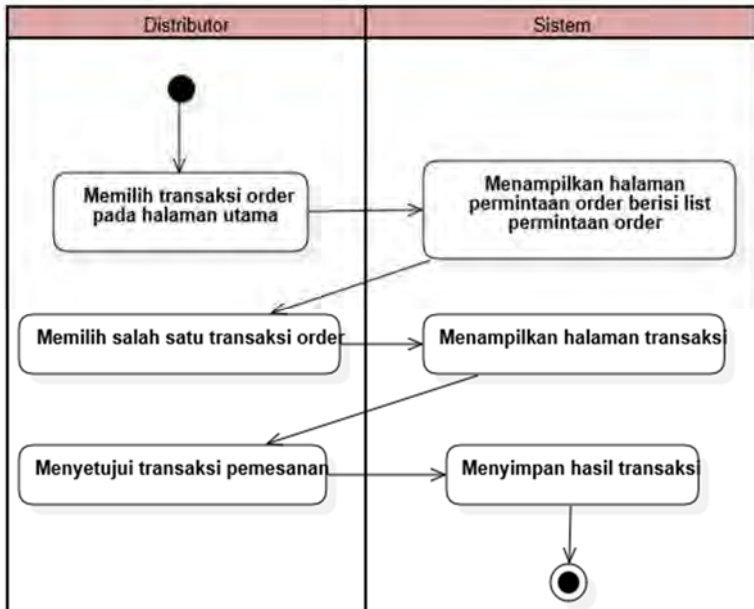
### 3.1.4.16 Kasus Penggunaan Menerima Pemesanan Produk

Pada kasus penggunaan ini, aktor menerima permintaan order yang masuk. Aktor akan melakukan persetujuan atau

menolak permohonan order yang diterima. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 15 dan diagram aktivitas terdapat pada Gambar 3. 15.

**Tabel 3. 15 Kasus Penggunaan Menerima Pemesanan Produk**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Menerima proses permintaan order
<b>Nomor</b>	UC-012
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana Distributor menerima permintaan order yang datang dari Customer.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Distributor
<b>Kondisi Awal</b>	Transaksi belum disetujui atau ditolak oleh distributor
<b>Kondisi Akhir</b>	Transaksi telah disetujui atau ditolak oleh distributor
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Distributor memilih transaksi order pada halaman utama.</li> <li>2. Sistem menampilkan halaman transaksi order berisi list permintaan order dari Customer.</li> <li>3. Distributor memilih salah satu transaksi yang akan dilakukan.</li> <li>4. Sistem menampilkan halaman transaksi.</li> <li>5. Distributor menyetujui transaksi pemesanan.</li> <li>6. Sistem menyimpan hasil transaksi.</li> </ol>



Gambar 3. 15 Diagram Aktivitas Menerima Pemesanan Produk

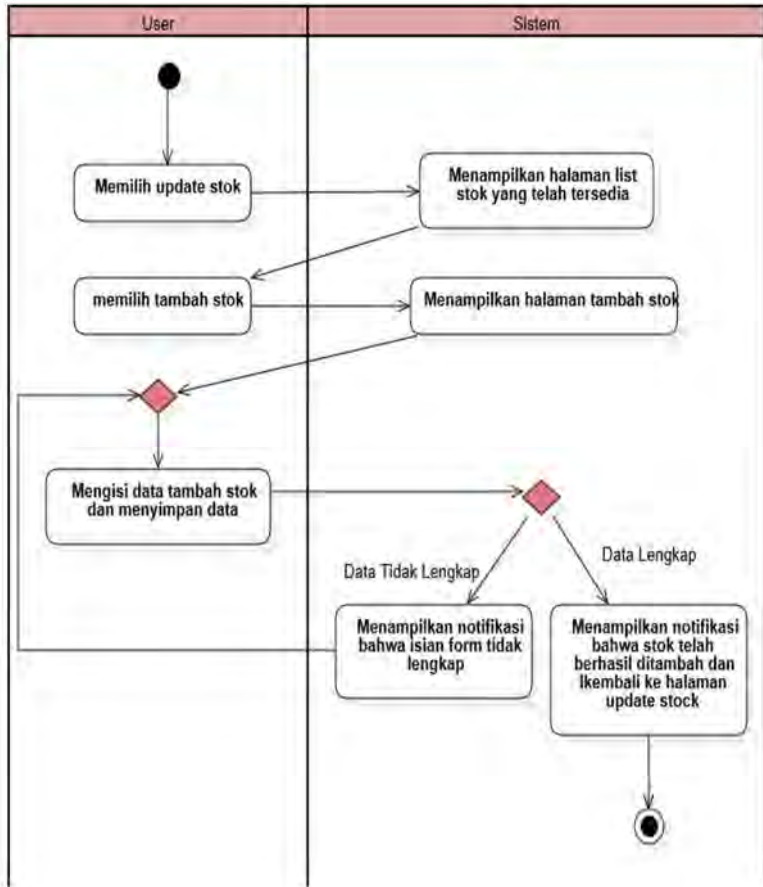
### 3.1.4.17 Kasus Penggunaan Menambah Jumlah Stok Tersedia

Pada kasus penggunaan ini, aktor menambah jumlah stok yang tersedia pada hari data ditambah. Sehingga nantinya customer bisa melihat jumlah stok barang yang tersedia dan memesan dengan jumlah barang yang tersedia. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 16 dan diagram aktivitas terdapat pada Gambar 3. 16.

Tabel 3. 16 Kasus Penggunaan Menambah Jumlah Stok Tersedia

Komponen	Deskripsi
Nama	Menambah jumlah stok tersedia
Nomor	UC-013

<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana aktor menambah jumlah stok yang tersedia pada hari tersebut.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Supplier, Distributor
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Aktor menambah data stok yang tersedia
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih update stok.</li> <li>2. Sistem menampilkan halaman list stok yang telah tersedia.</li> <li>3. Aktor memilih tambah stok.</li> <li>4. Sistem menampilkan halaman tambah stok.</li> <li>5. Aktor mengisi data tambah stok dan menyimpan data. [Alternative A5]</li> <li>6. Sistem menampilkan notifikasi bahwa stok telah berhasil ditambah dan kembali ke halaman update stok.</li> </ol>
<b>Alur Alternative</b>	<p>A.5. Isian form tidak lengkap</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan notifikasi bahwa isian halaman tidak lengkap.</li> <li>2. Aktor kembali mengisi data tambah stok dan menyimpan data.</li> <li>3. Sistem menampilkan notifikasi bahwa transaksi berhasil dilakukan dan kembali ke halaman update stok.</li> </ol>



**Gambar 3. 16 Diagram Aktivitas Menambah Jumlah Stock Tersedia**

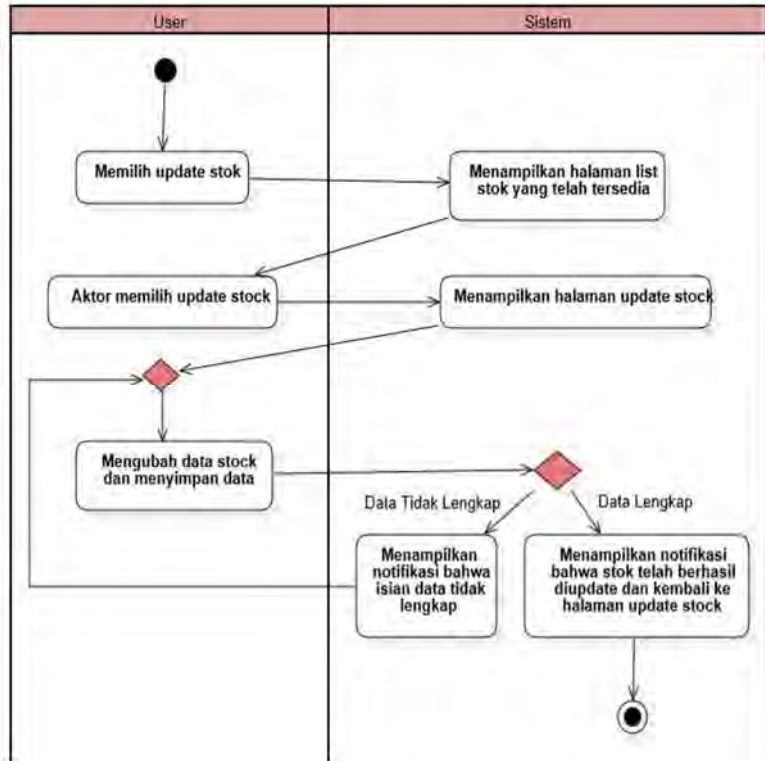
### 3.1.4.18 Kasus Penggunaan Mengubah Jumlah Stock Tersedia

Pada kasus penggunaan ini, aktor mengupdate jumlah stock yang tersedia berdasarkan hari data di update. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 17 dan diagram aktivitas terdapat pada Gambar 3. 17.

**Tabel 3. 17 Kasus Penggunaan Mengubah Jumlah Stock Tersedia**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Mengubah jumlah stok tersedia
<b>Nomor</b>	UC-014
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana aktor mengupdate jumlah stok yang tersedia pada hari tersebut.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Supplier, Distributor
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Aktor update data stok yang tersedia
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih update stok.</li> <li>2. Sistem menampilkan halaman list stok yang telah tersedia.</li> <li>7. Aktor memilih update stok.</li> <li>8. Sistem menampilkan halaman update stok.</li> <li>9. Aktor mengubah data stok dan menyimpan data. [Alternative A5]</li> <li>10. Sistem menampilkan notifikasi bahwa stok telah berhasil diupdate dan kembali ke halaman update stock.</li> </ol>
<b>Alur Alternative</b>	<p>A.5. Isian form tidak lengkap</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan notifikasi bahwa isian form tidak lengkap.</li> <li>2. Aktor kembali mengisi data update stok dan menyimpan data.</li> <li>3. Sistem menampilkan notifikasi bahwa stok berhasil diupdate dan kembali ke halaman update stok.</li> </ol>





Gambar 3. 17 Diagram Aktivitas Megubah Jumlah Stock Tersedia

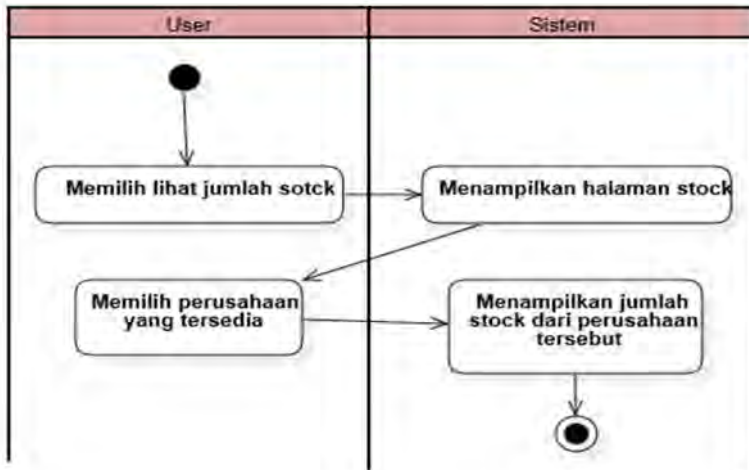
### 3.1.4.19 Kasus Penggunaan Melihat Jumlah Stock Tersedia

Pada kasus penggunaan ini, aktor dapat melihat jumlah stock yang tersedia pada supplier atau distributor terdaftar. Rincian kasus penggunaan tersebut dapat dilihat pada Tabel 3. 18 dan diagram aktivitas terdapat pada Gambar 3. 18.

Tabel 3. 18 Kasus Penggunaan Melihat Jumlah Stock Tersedia

Komponen	Deskripsi
Nama	Melihat jumlah stok tersedia
Nomor	UC-015

<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana aktor melihat jumlah stock barang yang tersedia berdasarkan perusahaan yang dipilih
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Distributor, Customer
<b>Kondisi Awal</b>	Tampilan list jumlah stock belum ada
<b>Kondisi Akhir</b>	List jumlah stock tersedia
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih lihat jumlah stock.</li> <li>2. Sistem menampilkan halaman stock.</li> <li>3. Aktor memilih perusahaan yang tersedia.</li> <li>4. Sistem menampilkan jumlah stock dari perusahaan tersebut.</li> </ol>



Gambar 3. 18 Diagram Aktivitas Melihat Jumlah Stock Tersedia

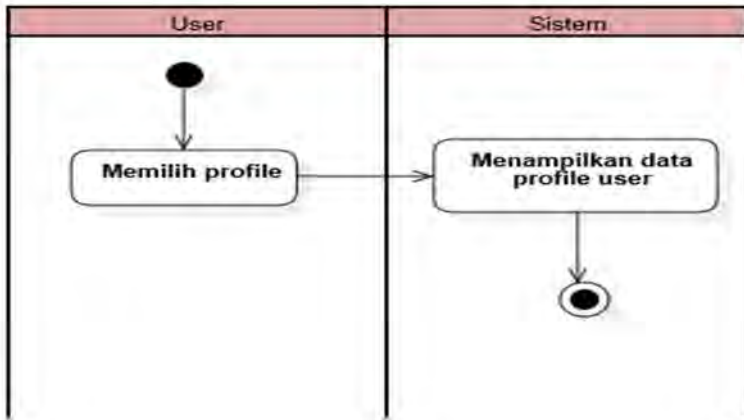
### 3.1.4.20 Kasus Penggunaan Melihat Profile User

Pada kasus penggunaan ini, aktor dapat melihat data profile sendiri. Rincian kasus penggunaan tersebut dapat dilihat

pada Tabel 3. 19 dan diagram aktivitas terdapat pada Gambar 3. 19.

**Tabel 3. 19 Kasus Penggunaan Melihat Profile User**

Komponen	Deskripsi
<b>Nama</b>	Melihat Profile User
<b>Nomor</b>	UC-016
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana User melihat data profile mereka sendiri
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Supplier, Distributor, Customer
<b>Kondisi Awal</b>	-
<b>Kondisi Akhir</b>	Tampilan data profile tersedia
<b>Alur Normal</b>	1. User memilih profile. 2. Sistem menampilkan Data-data profile.



**Gambar 3. 19 Diagram Aktivitas Melihat Profile User**

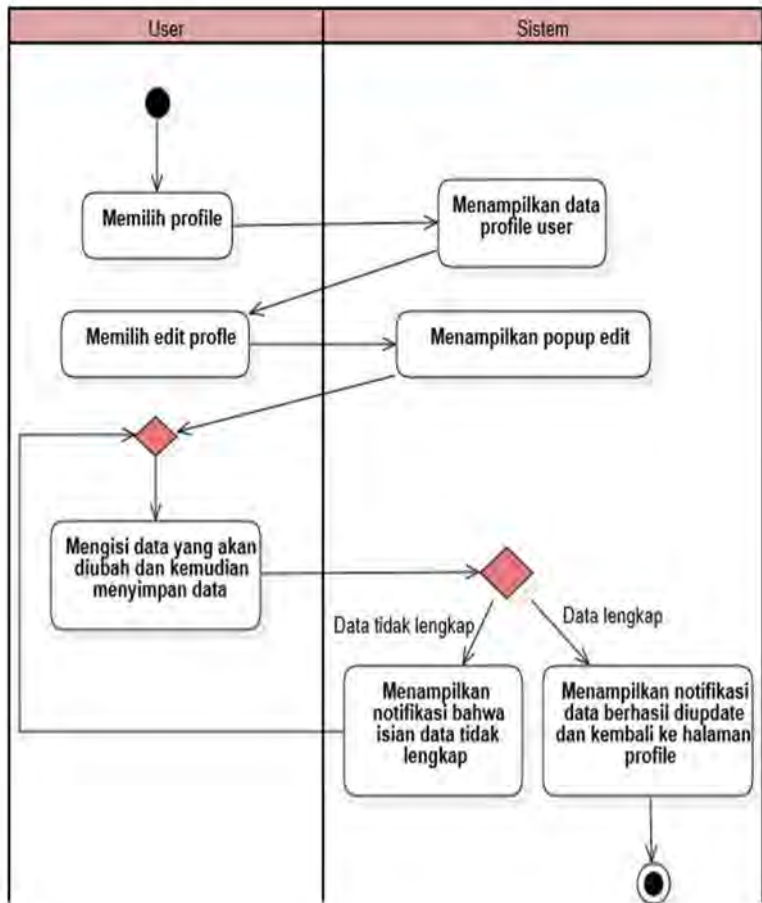
### 3.1.4.21 Kasus Penggunaan Mengubah Data User

Pada kasus penggunaan ini, aktor dapat mengubah data profile sendiri. Rincian kasus penggunaan tersebut dapat dilihat

pada Tabel 3. 20 dan diagram aktivitas terdapat pada Gambar 3. 20.

**Tabel 3. 20 Kasus Penggunaan Mengubah Data User**

<b>Komponen</b>	<b>Deskripsi</b>
<b>Nama</b>	Mengubah Data User
<b>Nomor</b>	UC-017
<b>Deskripsi</b>	Use-case ini mendeskripsikan bagaimana aktor mengubah data profile masing-masing.
<b>Tipe</b>	Fungsional
<b>Aktor</b>	Supplier, Distributor, Customer
<b>Kondisi Awal</b>	Data profile masih yang lama.
<b>Kondisi Akhir</b>	Data profile baru telah tersimpan
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih profile.</li> <li>2. Sistem menampilkan halaman profile user.</li> <li>3. Aktor memilih edit profile.</li> <li>4. Sistem menampilkan popup edit.</li> <li>5. Aktor mengisi data yang akan diubah dan menyimpan data. [Alternative A5]</li> <li>6. Sistem menampilkan notifikasi data berhasil diupdate dan kembali ke halaman profile.</li> </ol>
<b>Alur Alternative</b>	<p>A.5. Isian data tidak lengkap</p> <ol style="list-style-type: none"> <li>1. Sistem menampilkan notifikasi bahwa isian data tidak lengkap.</li> <li>2. Aktor kembali mengisi data update profile dan menyimpan data.</li> <li>3. Sistem menampilkan notifikasi bahwa data berhasil diupdate dan kembali ke halaman profile.</li> </ol>



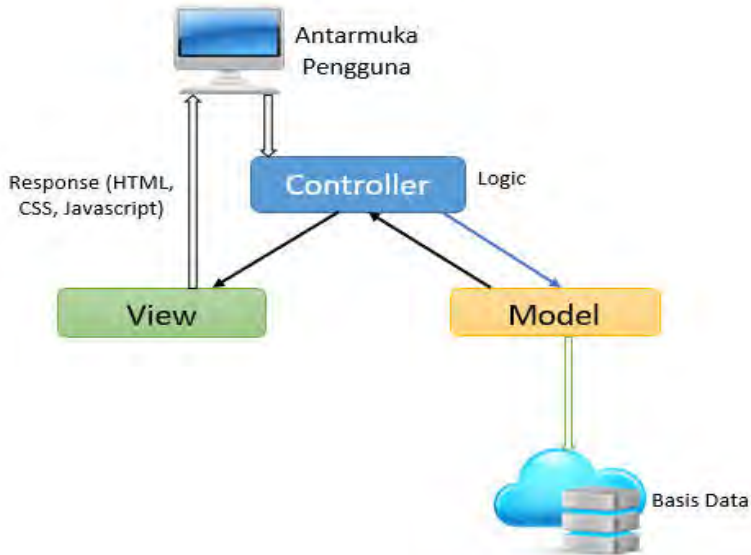
Gambar 3. 20 Diagram Aktivitas Mengubah Data User

### 3.2 Perancangan

Pada subbab perancangan akan dijelaskan mengenai arsitektur sistem yang digunakan, perancangan basis data dan perancangan antarmuka pengguna.

### 3.2.1 Perancangan Arsitektur Sistem

Arsitektur sistem yang digunakan pada tugas akhir ini menggunakan arsitektur sistem *framework* CodeIgniter berbasis MVC pada setiap modul. Ilustrasi arsitektur CodeIgniter terdapat pada Gambar 3.21.



**Gambar 3.21 Ilustrasi Arsitektur Model Sistem Informasi Distribusi Daging Sapi**

Antarmuka pengguna merupakan lapisan yang berhubungan langsung dengan pengguna. *Controller* adalah penghubung antara antarmuka dengan model yang berisi query ke database. Pada *controller* berisi *logic-logic* agar sistem dapat berjalan sebagaimana mestinya. *Controller* akan meminta data-data yang diperlukan ke Model. Model berisi query-query untuk mengambil data dari database yang kemudian akan dikembalikan ke *controller*. Data yang telah didapatkan dari model selanjutnya akan ditampilkan pada view melalui *Controller*. View merupakan tampilan yang akan dilihat oleh pengguna.

### 3.2.2 Perancangan Diagram Kelas

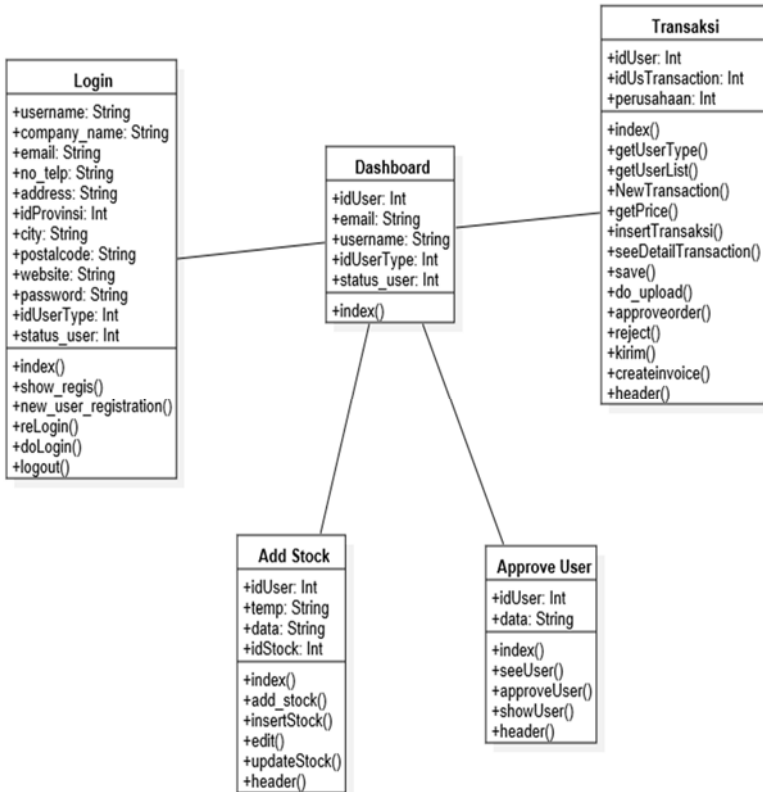
Pada Gambar 3. 22 Perancangan Diagram Kelas menunjukkan diagram kelas sistem informasi distribusi daging sapi yang telah digabung menjadi 5 fungsi utama. Dalam pembuatan sistem informasi ini terdapat total 16 kelas. Kelas Login merupakan *controller* dari aktivitas login pengguna. Pada kelas login ini menyimpan data-data pengguna yang melakukan registrasi, pengguna melakukan login, pemberian *session user*, dan sampai *user* logout dari sistem. Kelas Dashboard berfungsi untuk membedakan tampilan antarmuka tiap pengguna berdasarkan tipe *user*. Kelas Transaksi merupakan kelas yang menangani transaksi, yaitu menangani transaksi baru, melakukan persetujuan atau menolak transaksi, konfirmasi pengiriman barang serta mencetak invoice ketika transaksi telah selesai dilakukan. Kelas *add stock* merupakan kelas yang menangani penambahan stock barang serta update stock barang terbaru. Kelas *add stock* juga berfungsi untuk melihat data barang yang tersedia. Kelas *approve user* berfungsi untuk melakukan persetujuan pengguna yang telah melakukan registrasi. Perancangan diagram kelas dapat dilihat pada Gambar 3. 22.

### 3.2.3 Perancangan Basis Data

Dalam pembuatan sistem informasi manajemen rantai pasok distribusi daging sapi ini menggunakan basis data MySQL. Penggunaan basis data MySQL dipilih karena penggunaannya yang mudah dan *opensource*. Perancangan basis data terdapat pada Gambar 3. 23 serta berikut ini adalah penjelasan secara detail tabel-tabel yang ada.

#### 1. Tabel User

Tabel User adalah tabel yang digunakan untuk menyimpan data-data user yang telah melakukan registrasi. Pada tabel user terdapat foreign key ke tabel *usertype* dan tabel *province*.



**Gambar 3. 22 Perancangan Diagram Kelas**

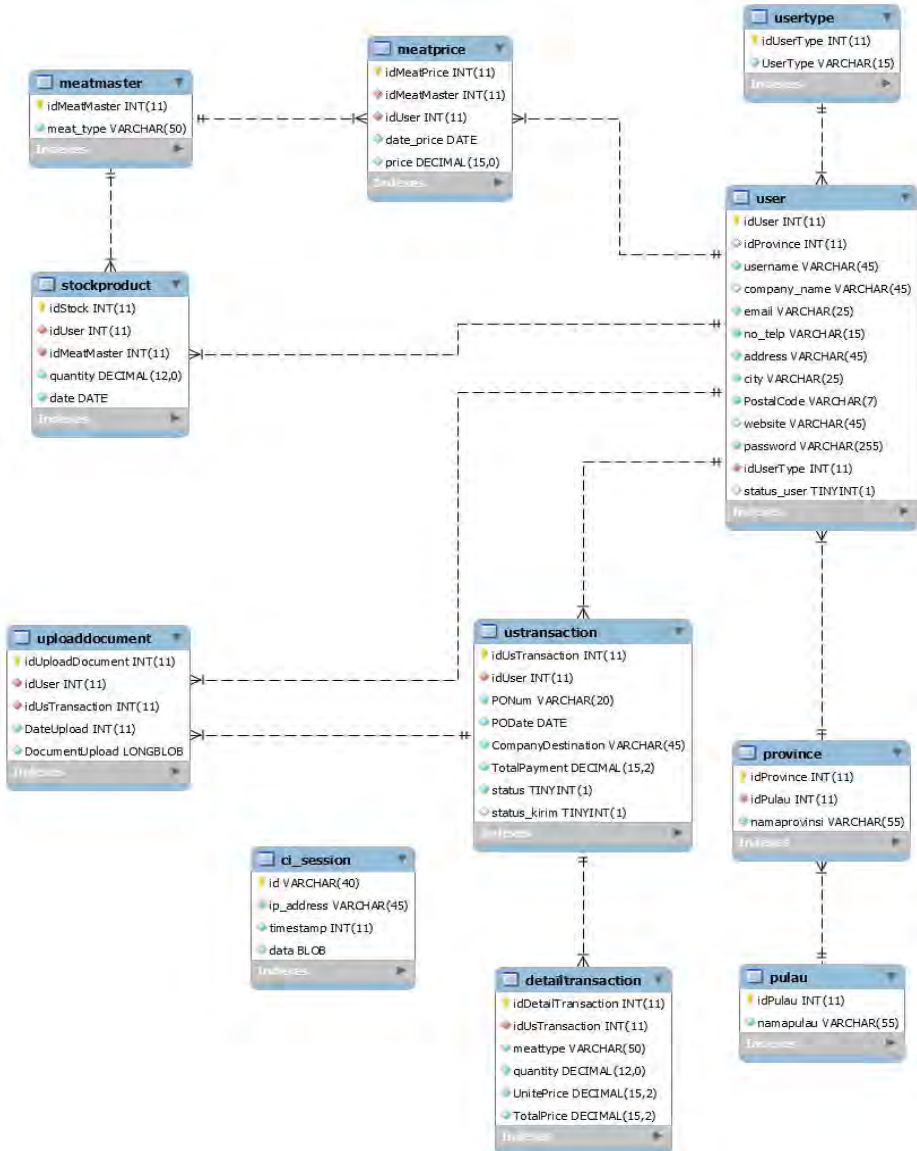
## 2. Tabel Usertype

Tabel usertype adalah tabel yang berisi tipe dari user dan menyimpan informasi hak akses untuk tiap user.

## 3. Tabel Province

Tabel province menyimpan data semua provinsi yang ada di Indonesia. Tabel ini berguna untuk pendataan transaksi jika di filter berdasarkan provinsi.





Gambar 3. 23 Rancangan Basis Data SQL

#### 4. Tabel Pulau

Tabel pulau menyimpan data-data pulau yang ada di Indonesia. Tabel pulau berhubungan dengan tabel province dimana setiap pulau berhubungan dengan provinsi yang berkaitan.

#### 5. Tabel UsTransaction

Tabel UsTransaction merupakan tabel transaksi yang dilakukan oleh user. Semua data transaksi yang sudah dilakukan akan masuk ke tabel UsTransaction. Setiap transaksi bisa memiliki banyak item transaksi sehingga tabel UsTransaction berhubungan dengan tabel DetailTransaction.

#### 6. Tabel DetailTransaction

Tabel DetailTransaction menyimpan data-data jumlah barang yang dibeli tiap 1 kali transaksi. Setiap user melakukan transaksi, tabel detailtransaction akan menyimpan data jumlah item yang dibeli oleh user.

#### 7. Tabel UploadDocument

Tabel uploaddocument berguna untuk menyimpan berkas dokumen bukti pembayaran yang telah dilakukan user.

#### 8. Tabel MeatMaster

Tabel MeatMaster menyimpan data-data master mengenai jenis daging sapi.

#### 9. Tabel MeatPrice

Tabel MeatPrice menyimpan semua data-data yang berkaitan dengan harga dari setiap daging sapi. Semua data yang dimasukkan dan juga yang telah diupdate akan masuk ke tabel MeatPrice.

#### 10. Tabel StockProduct

Tabel stockproduct berisi data mengenai jumlah stok tiap item yang tersedia.

#### 11. Tabel ci\_session

Tabel ci\_session berguna untuk menampung session dari tiap pengguna yang login.

#### 12. Perancangan Antarmuka Pengguna

Perancangan antarmuka membahas rancangan antarmuka pengguna pada sistem informasi manajemen rantai pasok distribusi

daging sapi. Perancangan antarmuka pengguna merupakan hal yang penting dalam melakukan perancangan aplikasi. Antarmuka pengguna yang berhubungan langsung dengan aktor harus memiliki kemudahan-kemudahan dan tampilan yang rapi dan menarik bagi penggunanya. Terdapat beberapa rancangan tampilan antarmuka yang akan dijelaskan sebagai berikut.

### 3.2.3.1 Rancangan Antarmuka Halaman Registrasi User

Halaman ini digunakan untuk user melakukan registrasi jika mereka belum menjadi member dari sistem ini. Pada halaman ini terdapat form untuk registrasi. User dapat mengisikan data-data perusahaan mereka pada halaman ini. Halaman antarmuka dapat dilihat pada Gambar 3. 24.

The image shows a web form for user registration. The form is titled "Form Registrasi" and "Create Account". It contains several input fields and a dropdown menu. The fields are labeled as follows:

- Nama
- Perusahaan
- Email (with a placeholder example: email (email@domain.com))
- Telephone
- Alamat Perusahaan
- Provinsi (with a dropdown menu labeled "Pilih Provinsi")
- Kota
- Kode Pos
- Website URL (with a placeholder example: www.website.com)
- Password
- Konfirmasi Password
- Hak Akses (with a dropdown menu labeled "Supplier")

At the bottom of the form, there is a "Submit" button and a link that says "Saya sudah memiliki akun". Below the form, the text "Distribusi Sapi 2016" is visible.

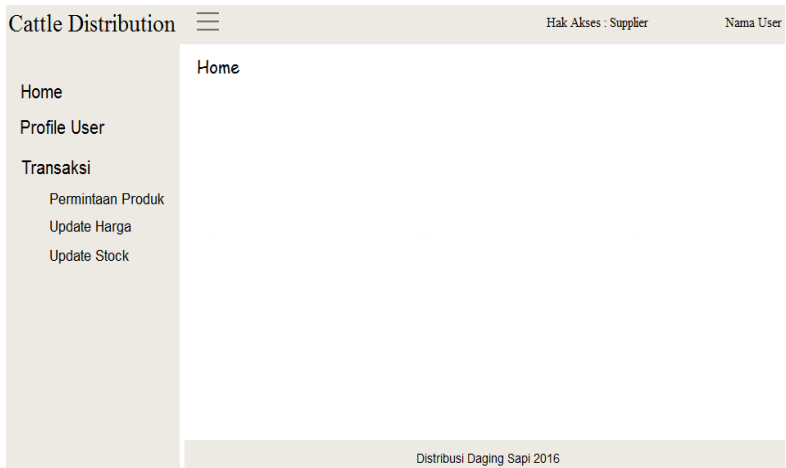
Gambar 3. 24 Rancangan Antarmuka Registrasi User

### 3.2.3.2 Rancangan Antarmuka Halaman Utama

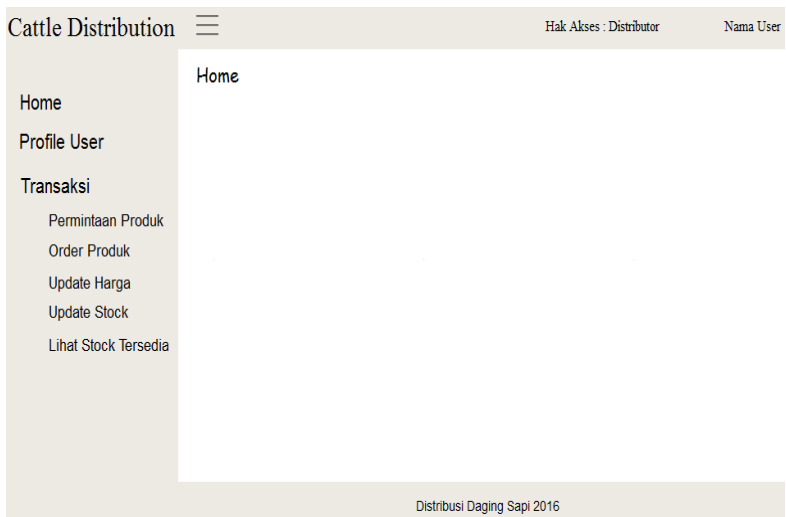
Halaman ini merupakan tampilan utama ketika user telah melakukan login. Pada halaman ini telah terdapat berbagai menu pilihan berdasarkan tipe user yang login. Tampilan rancangan antarmuka halaman admin dapat dilihat pada Gambar 3. 25. Pada tampilan antarmuka halaman utama user ada perbedaan ketika akun user telah di setujui oleh admin dan ketika akun user belum disetujui oleh admin. Tampilan antarmuka user yang telah disetujui admin, untuk halaman supplier dapat dilihat pada Gambar 3. 26, halaman distributor dapat dilihat pada Gambar 3. 27 serta halaman customer dapat dilihat pada Gambar 3. 28. Sedangkan tampilan utama user yang belum disetujui oleh admin dapat dilihat pada Gambar 3. 29.



**Gambar 3. 25 Rancangan Antarmuka Halaman Utama Admin**



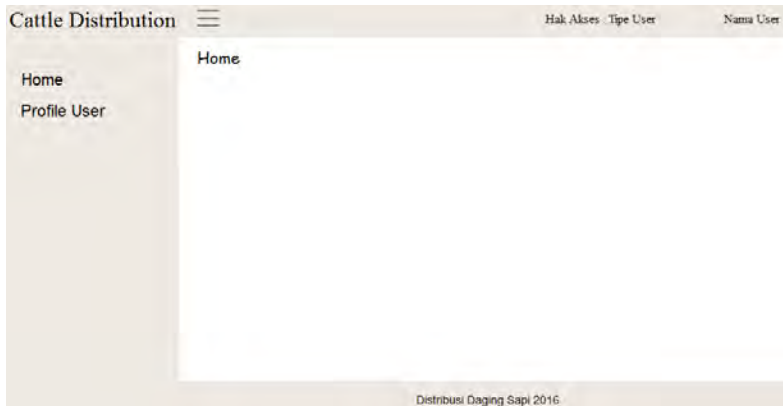
**Gambar 3. 26 Rancangan Antarmuka Halaman Utama Supplier**



**Gambar 3. 27 Rancangan Antarmuka Halaman Utama Distributor**



**Gambar 3. 28 Rancangan Antarmuka Halaman Customer**



**Gambar 3. 29 Rancangan Antarmuka Halaman User**

### 3.2.3.3 Rancangan Antarmuka Halaman Approval User

Halaman ini digunakan untuk kasus penggunaan approval user. Pada halaman ini terdapat list user yang sudah melakukan registrasi dan admin bisa menyetujui atau menolak

akun yang sudah ter-registrasi tersebut. Selain itu pada halaman ini admin juga bisa melihat detail profil dari user yang akan di approve akunya. Halaman antarmuka approve user dapat dilihat pada Gambar 3. 30 dan halaman antarmuka detail user dapat dilihat pada Gambar 3. 31.

Cari:

Nama	Perusahaan	Email	Kota	Website	Status User	Tipe User	Aksi
Superindo	Superindo	superindo@gmail	Surabaya	-	User Baru	Customer	<input type="button" value="Lihat Detail"/> <input type="button" value="Setuju"/> <input type="button" value="Tolak"/>

**Gambar 3. 30 Rancangan Antarmuka Halaman Approval User**

## Approval User

Home / Approval User / Lihat Detail User

Data User	<input type="button" value="Setuju"/> <input type="button" value="Tolak"/> <input type="button" value="Kembali"/>
<p>Nama</p> <p>Perusahaan :</p> <p>Email :</p> <p>Telephone :</p> <p>Alamat Perusahaan :</p> <p>Kota :</p> <p>Kode Pos :</p> <p>Website :</p> <p>Status User :</p>	

**Gambar 3. 31 Rancangan Antarmuka Halaman Lihat Detail User**

### 3.2.3.4 Rancangan Antarmuka Halaman Mengelola Data Admin

Halaman ini digunakan untuk mengelola data-data admin, baik menambah, mengubah, melihat, maupun menghapus data admin yang ada. Tampilan antarmuka halaman mengelola data admin dapat dilihat pada Gambar 3. 32 dan halaman antarmuka menambah admin dapat dilihat pada Gambar 3. 33.

### 3.2.3.5 Rancangan Antarmuka Halaman Melakukan Pemesanan Produk

Halaman ini digunakan user untuk melakukan transaksi. Pada halaman ini user dapat melakukan transaksi pemesanan produk yang diinginkan. Halaman dapat dilihat pada Gambar 3. 34, Gambar 3. 35 dan Gambar 3. 36.

The screenshot shows a web interface for managing admin data. At the top left is a blue button labeled 'Tambah Admin'. To its right is a search bar with the placeholder text 'Cari: Cari'. Below these elements is a table with the following structure:

Nama	Email	Nomor Telephone	Aksi
Admin	Admin@gmail.com	103272	<input type="button" value="Lihat Detail"/> <input type="button" value="Edit"/> <input type="button" value="Hapus"/>

**Gambar 3. 32 Rancangan Antarmuka Mengelola Data Admin**



### Data Admin Baru

Nama :

Email :

Telephone :

Alamat :

Provinsi :  ▼

Kota :

Kode Pos :

Password :

Konfirmasi Password :

Hak Akses :  ▼




**Gambar 3. 33 Rancangan Antarmuka Menambah Data Admin**

Transaksi | Status Pengiriman

ID Order	Nomor Order	Tanggal Order	Perusahaan Tujuan	Total Pembayaran	Slip Transfer	Status Order	Aksi
1	KE001	2016-05-08	PT.X	Rp 280.000	-	Belum Disetujui	<input type="button" value="Lihat Detail"/>

**Gambar 3. 34 Rancangan Antarmuka Melakukan Pemesanan Produk**



Order Baru	
Pilih Jenis Perusahaan	Supplier
Pilih Perusahaan	
Nomor Order	
Tanggal Order	
<input type="button" value="Cancel"/> <input type="button" value="Next"/>	

Gambar 3. 35 Rancangan Antarmuka Transaksi Baru

### 3.2.3.6 Rancangan Antarmuka Halaman Menerima Pemesanan Produk

Halaman ini digunakan user untuk menerima data-data pemesanan barang yang masuk. Pada halaman ini user bisa melakukan persetujuan atau menolak data pemesanan yang masuk serta dapat melihat detail transaksi. Pada halaman ini user juga bisa mengubah status pengiriman menjadi dikirim serta mencetak invoice. Halaman antarmuka dapat dilihat pada Gambar 3. 37, Gambar 3. 38 serta Gambar 3. 39.

Transaksi

---

Perusahaan Tujuan  Tanggal Order

Nomor Order

Detail Order Transaksi

[Tambah Baris Input](#)

Jenis Daging	Jumlah Pemesanan	Harga Barang	Total Harga
<input type="text" value="Pilih Daging"/> ▾	<input type="text"/>	<input type="text"/>	<input type="text"/>

[Submit](#) [Cancel](#)

**Gambar 3. 36 Rancangan Antarmuka Detail Transaksi**

Permintaan Transaksi [Konfirmasi Pengiriman](#) [Cetak Invoice](#)

ID Transaksi	Perusahaan	Tanggal Order	Total Pembayaran	Slip Pembayaran	Aksi
1	XY	2016-05-08	Rp 280.000	-	<a href="#">Lihat Detail Order</a>

**Gambar 3. 37 Rancangan Antarmuka Halaman Terima Order**

Permintaan Order Kembali

---

Perusahaan Tujuan  Tanggal Order

Nomor Order  Total Pembayaran

Jenis Daging	Jumlah Pemesanan (Kg)	Harga Barang (Rp)	Total Harga	Status Order
Daging Punuk / Blade	3	93.000	280.000	Belum Disetujui

Setuju

**Gambar 3. 38 Rancangan Antarmuka Halaman Detail Order**

Invoice Pengiriman Produk

---

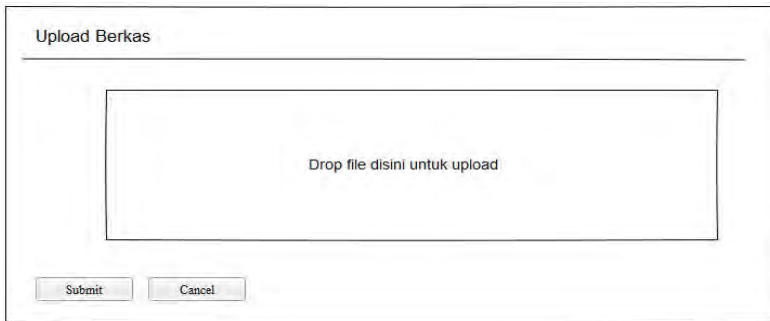
Order	Kepada	Dikirim Ke
Tanggal Kirim	Nama	Nama Perusahaan
Nomor Order	Nama Perusahaan	
Tanggal Order	Alamat	
	Email	

Jenis Daging	Jumlah Pemesanan (Kg)	Harga Barang (Rp)	Total Harga (Rp)
Daging Punuk / Blade	3	93.000	280.000
Total Pembayaran			Rp 280.000

**Gambar 3. 39 Rancangan Halaman Invoice**

### 3.2.3.7 Rancangan Antarmuka Halaman Upload Berkas Dokumen Transaksi

Halaman ini digunakan untuk user mengupload dokumen bukti pembayaran ketika transaksi telah disetujui oleh perusahaan tujuan. Halaman antarmuka dapat dilihat pada Gambar 3. 40.



**Gambar 3. 40 Rancangan Antarmuka Halaman Upload Berkas**

### 3.2.3.8 Rancangan Antarmuka Halaman Menambah Jumlah Stock Tersedia

Halaman ini digunakan untuk menambah jumlah stok barang yang tersedia. Pada halaman ini user bisa melakukan insert jumlah barang yang tersedia pada perusahaan mereka. Halaman antarmuka dapat dilihat pada Gambar 3. 41 dan Gambar 3. 42.

Tambah Stock

Cari:

Tipe Daging	Stock (Kg)	Tanggal Stock	Aksi
Daging Punuk / Blade	30	2016-05-17	<input type="button" value="Update Stock"/>

**Gambar 3. 41 Rancangan Antarmuka Halaman Mengelola Jumlah Stock**

Tambah Stock Baru

Tanggal Sekarang

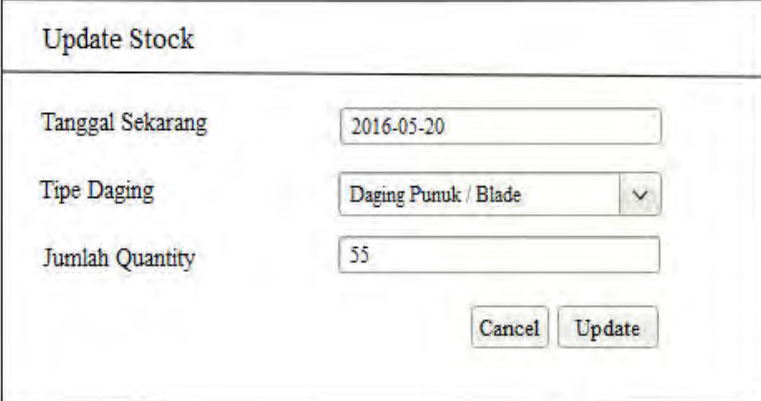
Tipe Daging

Jumlah Quantity

**Gambar 3. 42 Rancangan Antarmuka Halaman Tamabah Stock Barang Tersedia**

### 3.2.3.9 Rancangan Antarmuka Halaman Update Jumlah Stock Tersedia

Halaman ini digunakan user untuk meng-update jumlah stok barang yang tersedia pada hari itu. Halaman antarmuka dapat dilihat pada Gambar 3. 43.



Update Stock	
Tanggal Sekarang	<input type="text" value="2016-05-20"/>
Tipe Daging	<input style="border: none; background-color: #f0f0f0;" type="text" value="Daging Punuk / Blade"/> ▾
Jumlah Quantity	<input type="text" value="55"/>
<input type="button" value="Cancel"/> <input type="button" value="Update"/>	

**Gambar 3. 43 Rancangan Antarmuka Halaman Update Jumlah Stock Tersedia**

### **3.2.3.10 Rancangan Antarmuka Halaman Melihat Jumlah Stock Tersedia**

Halaman ini digunakan untuk melihat jumlah stock barang dari perusahaan-perusahaan yang dipilih. Halaman antarmuka dapat dilihat pada Gambar 3. 44.

### **3.2.3.11 Rancangan Antarmuka Halaman Mengelola Data User**

Halaman ini digunakan untuk melihat profile user. Pada halaman ini user juga dapat mengubah data- data yang diinginkan. Halaman antarmuka dapat dilihat pada Gambar 3. 45.

### **3.2.3.12 Rancangan Antarmuka History Transaksi**

Halaman ini digunakan untuk melihat history transaksi yang telah dilakukan oleh distributor maupun customer. Pada halaman ini admin bisa melihat transaksi yang telah dilakukan oleh user. Halaman antarmuka dapat dilihat pada Gambar 3. 46.

**Stock Tersedia**

---

Pilih Perusahaan Terkait :  ▼

---

**Laporan Jumlah Stock**

Tipe Daging	Stock (Kg)	Tanggal Stock
Daging Punuk / Blade	550	2016-05-13

**Gambar 3. 44 Rancangan Antarmuka Halaman Melihat Jumlah Stock Tersedia**

**Profile**

---

Nama : X

Perusahaan : PT X

Email : x@gmail.com

Telephone : 123

Alamat : Surabaya

Provinsi : Jawa Timur

Kota : Surabaya

Kode Pos : 123

Website : -

Password : xxx

Status User : Akun sudah disetujui

**Gambar 3. 45 Rancangan Antarmuka Halaman Mengelola Data User**



Lihat History Transaksi

---

Pilih Perusahaan Terkait  ▼

---

Tanggal Order	Perusahaan	Jenis Produk	Jumlah Pemesanan (Kg)
2016-05-13	Superindo	Daging Punuk	200

**Gambar 3. 46 Rancangan Antarmuka Halaman History Transaksi**

*[Halaman ini sengaja dikosongkan]*

## **BAB IV IMPLEMENTASI**

Pada bab ini akan dibahas mengenai implementasi sistem sesuai dengan analisis dan perancangan proses bisnis secara umum pada sistem informasi manajemen rantai pasok distribusi daging sapi yang telah dijabarkan pada bab sebelumnya.

Implementasi yang dijelaskan meliputi lingkungan pembangunan perangkat lunak, kode sumber utama dan implementasi antarmuka perangkat lunak. Bahasa pemrograman yang digunakan adalah PHP dengan arsitektur sistem MVC dengan *framework* CodeIgniter.

### **4.1 Lingkungan Implementasi**

Lingkungan implementasi yang akan digunakan untuk melakukan implementasi pada tugas akhir ini dilakukan pada lingkungan dengan kaskas sebagai berikut.

1. Prosesor Intel® Core™ i5-32 10M 2.5 GHz
2. Sistem operasi Windows 10 Home Professional 64 bit.
3. Sublime Text editor sebagai kaskas bantu pembuatan program.
4. MySQL Workbench 6.3 CE untuk merancang basis data.
5. StarUML Beta digunakan untuk membuat diagram kasus penggunaan dan diagram aktivitas serta Pencil sebagai tools dalam perancangan antarmuka sistem.
6. XAMPP sebagai web server.

### **4.2 Implementasi *Model-View-Controller***

Sistem yang dibuat memiliki lapisan-lapisan yang direpresentasikan dalam kelas, yaitu *view* sebagai lapisan antarmuka pengguna, *controller* sebagai tempat untuk menerima *request* yang dikirim oleh aplikasi *client* atau browser, *model* sebagai tempat untuk melakukan pengelolaan terhadap basis data.

#### 4.2.1 Implementasi View

*View* pada sistem ini bertugas untuk menampilkan data dari basis data yang diakses melalui *Model* dan fungsi-fungsi di *Controller*. *View* yang diimplementasikan pada sistem ini dapat dilihat pada Tabel 4. 1. Serta terdapat view untuk menampilkan grafik dari basis data. Kode sumber tercantum pada Kode Sumber 4. 1.

**Tabel 4. 1 Implementasi View**

No	View		Deskripsi View
1	Login	Login_view.php	Menampilkan halaman login
		Registrasi.php	Menampilkan halaman registrasi user
2	Supplier	Home_supplier.php	Menampilkan halaman utama tipe user supplier
		S_addPrice.php	Menampilkan halaman untuk menambah harga barang
		S_addStock.php	Menampilkan halaman untuk menambah stock barang
		S_approveOrder.php	Menampilkan halaman untuk menyetujui order dari customer atau distributor
		S_historytransaksi.php	Menampilkan halaman untuk melihat transaksi yang sudah selesai dilakukan

No	View		Deskripsi View
		S_konfirmasiPengiriman.php	Menampilkan halaman untuk mengkonfirmasi pengiriman transaksi
		S_lihatOrder.php	Menampilkan halaman untuk melihat order transaksi
		S_updatePrice.php	Menampilkan halaman untuk update harga barang
		S_updateStock.php	Menampilkan halaman untuk update stock
		Show_supplier.php	Menampilkan halaman untuk melihat profile user
3	Distributor	Home_Distributor.php	Menampilkan halaman utama distributor ketika sudah melakukan login
		Show_Distributor.php	Menampilkan halaman untuk melihat profile distributor
		D_addPrice.php	Menampilkan halaman untuk menambah harga barang
		D_addStock.php	Menampilkan halaman untuk menambah stock barang
		D_approveOrder.php	Menampilkan halaman untuk

No	View	Deskripsi View
		menyetujui order transaksi
	D_konfirmasiPengiriman.php	Menampilkan halaman untuk konfirmasi pengiriman dari transaksi yang telah disetujui
	D_lihatDetailTransaksi.php	Menampilkan halaman untuk melihat detail transaksi
	D_lihatOrder.php	Menampilkan halaman untuk melihat order transaksi yang telah dilakukan
	D_NewTransaksi.php	Menampilkan halaman untuk melihat permintaan transaksi dari customer yang baru
	D_updatePrice.php	Menampilkan halaman untuk update harga barang
	D_updateStock.php	Menampilkan halaman untuk update stock barang
	D_uplodaBerkas.php	Menampilkan halaman untuk upload berkas bukti pembayaran transaksi yang telah dilakukan
	Transaksi_Distributor.php	Menampilkan halaman order barang

No	View		Deskripsi View
4	Customer	Home_Customer.php	Menampilkan halaman utama customer ketika sudah melakukan login
		Show_Customer.php	Menampilkan halaman untuk melihat profile user
		Transaksi_customer.php	Menampilkan halaman untuk melakukan transaksi order baru
		C_newTransaksi.php	Menampilkan halaman untuk mengisi detail transaksi
		C_lihatDetailTransaksi.php	Menampilkan halaman untuk melihat detail transaksi yang telah dilakukan
		C_lihatStock.php	Menampilkan halaman untuk melihat stock barang tersedia
		C_uploadBerkas.php	Menampilkan halaman untuk upload berkas bukti pembayaran.
5	Admin	Home_admin.php	Menampilkan halaman utama ketika user telah melakukan login
		Edit_admin.php	Menampilkan halaman untuk edit profile user

No	View	Deskripsi View
	Lihat_admin.php	Menampilkan halaman untuk melihat list admin yang tersedia
	Lihat_user.php	Menampilkan halaman list user yang telah melakukan registrasi
	Show_user_approve.php	Menampilkan halaman list user yang telah di approve

```

1 <script type="text/javascript">
2   $(function () {
3     $('#container').highcharts({
4       title: {
5         text: 'Data Jumlah Produksi Daging Sapi',
6         x: -20 //center
7       },
8       subtitle: {
9         text: 'Produksi Daging Sapi',
10        x: -20
11      },
12      xAxis: {
13        categories: [
14          <?php foreach ($jumlahproduksi[1] as $key =>
15            $value): ?>
16            <?php echo $value->tahun ?>,
17          <?php endforeach ?>
18        ]
19      },
20      yAxis: {
21        title: {
22          text: 'Jumlah Produksi (Ton)'
23        },
24        plotLines: [{
25          value: 0,
26          width: 1,
27          color: '#808080'
28        }]
29      },
30      tooltip: {

```



```

31         valueSuffix: ' Ton'
32     },
33     legend: {
34         layout: 'vertical',
35         align: 'right',
36         verticalAlign: 'middle',
37         borderWidth: 0
38     },
39     series: [
40         <?php foreach ($jumlahproduksi as $key =>
41 $value): ?>
42             {
43                 name: '<?php echo $value[0]-
44 >namapulau?>',
45                 data: [
46                     <?php foreach ($value as $val): ?>
47                     <?php echo $val-
48 >jumlah_produksi ?>,
49                     <?php endforeach ?>
50                 ]
51             },
52         <?php endforeach ?>
53     ]
54     });
55 });
56 </script>

```

**Kode Sumber 4. 1 Kode Sumber View Grafik**

## 4.2.2 Implementasi Kelas *Model*

Model pada sistem ini berguna untuk mengambil, mengubah dan memasukkan data ke dalam basis data. Pada subbab ini akan dijelaskan fungsi-fungsi umum yang terdapat pada kelas *Model*.

### 4.2.2.1 Fungsi *isDuplicate*

Fungsi ini digunakan untuk mengecek email user yang melakukan registrasi apakah ada yang sama atau tidak. Fungsi ini memiliki 1 parameter yaitu menggunakan email. Kode sumber tercantum pada Kode Sumber 4. 2.

### 4.2.2.2 Fungsi Login User

Fungsi ini digunakan untuk mengambil data dari database dan meng-cek apakah username dan password pengguna sudah cocok. Kode sumber tercantum pada Kode Sumber 4. 3.

```

1 public function isDuplicate($email)
2 {
3     $result = $this->db->get_where('user',
4         array('email' => $email)
5     );
6     if ($result->num_rows() > 0)
7         return FALSE;
8     else
9         return TRUE;
10 }

```

**Kode Sumber 4. 2 Fungsi *isDuplicate***

```

1 public function login_user($email, $password)
2 {
3     $this->db->select('*');
4     $this->db->from('user');
5     $this->db->where('email', $email);
6     $this->db->where('password', $password);
7
8     $query = $this->db->get();
9     return $query;
10 }

```

**Kode Sumber 4. 3 Fungsi *Login User***

#### 4.2.2.3 Fungsi *Get List Admin*

Fungsi ini digunakan untuk mengambil list data admin yang ada pada database. Query ini digunakan untuk menampilkan data admin yang tersedia. Kode sumber tercantum pada Kode Sumber 4. 4.

```

1 public function getAdmin()
2 {
3     $this->db->select("*");
4     $this->db->from("usertype");
5     $this->db->where("idUserType = 1");
6
7     $query = $this->db->get();
8     if($query->num_rows() > 0){

```

```

9         foreach ($query->result() as $row) {
10             $data[] = $row;
11         }
12         return $data;
13     }
14     else
15         return FALSE;
16 }

```

**Kode Sumber 4. 4 Fungsi *Get List Admin***

#### 4.2.2.4 Fungsi *Get Transaksi List*

Fungsi ini digunakan untuk mengambil list transaksi yang tersedia di database. Kode sumber tercantum pada Kode Sumber 4. 5.

```

1 public function getTransaksiList($idUser)
2 {
3     $this->db->select("idUsTransaction, idUser, PONum,
4     PODate, CompanyDestination, TotalPayment, status,
5     status_kirim, tanggal_kirim, documentUpload");
6     $this->db->from("ustransaction");
7     $this->db->where("idUser = '$idUser'");
8
9     $query = $this->db->get();
10    if($query->num_rows() > 0){
11        foreach ($query->result() as $row) {
12            $data[] = $row;
13        }
14        return $data;
15    }
16    else
17        return FALSE;
18 }

```

**Kode Sumber 4. 5 Fungsi *Get List Transaksi***

#### 4.2.2.5 Fungsi *Update Pembayaran*

Fungsi ini digunakan untuk update total pembayaran dari keseluruhan transaksi yang telah dilakukan. Pada fungsi ini terdapat 2 parameter yaitu idtransaksi dan total yang digunakan untuk update total pembayaran. Kode sumber tercantum pada Kode Sumber 4. 6.

```

1 public function updatePembayaran($idTransaksi, $total)
2 {
3     $this->db->where('idUsTransaction', $idTransaksi);
4     return $this->db->update('ustransaction',
5         array('TotalPayment' => $total));
6 }

```

**Kode Sumber 4. 6 Update Pembayaran**

#### 4.2.2.6 Fungsi *Insert to Transaction*

Fungsi ini digunakan untuk memasukkan data transaksi ke tabel transaksi dan tabel detail transaksi. Pada fungsi ini memasukkan data transaksi ke dalam database. Kode sumber tercantum pada Kode Sumber 4. 7.

```

1 public function insertToTransaction($data)
2 {
3     $this->db->insert('ustransaction', $data);
4     $last_id = $this->db->insert_id();
5     return $last_id;
6 }
7 public function insertdetailtransaksi($data)
8 {
9     return $this->db->insert('detailtransaction',
10    $data);
11 }

```

**Kode Sumber 4. 7 Fungsi *Insert to Transaction***

#### 4.2.2.7 Fungsi *Transaksi*

Fungsi ini digunakan untuk melakukan persetujuan transaksi serta update tanggal pengiriman transaksi. Pada fungsi ini berisi query untuk melakukan persetujuan transaksi dengan mengubah status transaksi. Kode sumber tercantum pada Kode Sumber 4. 8.

```

1 public function approve($idUsTransaction)
2 {
3     $this->db->where('idUsTransaction',
4     $idUsTransaction);
5     $this->db->update('ustransaction', array('status'
6     => 1));
7 }

```

```

8 public function reject($idUserTransaction)
9 {
10     $this->db->where('idUserTransaction',
11 $idUserTransaction);
12     $this->db->update('ustransaction', array('status'
13 => 2));
14 }
15 public function kirim($idUserTransaction, $tanggal,
16 $replace)
17 {
18     $this->db->where('idUserTransaction',
19 $idUserTransaction);
20     $this->db->update('ustransaction', array(
21         'status_kirim' => 1,
22         'tanggal_kirim' => $tanggal,
23         'PONum' => $replace)
24     );
25 }

```

**Kode Sumber 4. 8 Fungsi Transaksi**

#### 4.2.2.8 Fungsi *Add Stock*

Fungsi ini digunakan untuk mengelola data stock barang yang tersedia. Pada fungsi ini bisa mengambil data stock, memasukkan data stock serta update data ke dalam database. Kode Sumber tercantum pada Kode Sumber 4. 9.

```

1 public function getstock($idUser)
2 {
3     $this->db->select("sp.idStock, sp.quantity,
4 sp.date, mm.meat_type, sp.idUser");
5     $this->db->from("stockproduct as sp, meatmaster as
6 mm");
7     $this->db->where("sp.idUser = '$idUser' and
8 sp.idMeatMaster = mm.idMeatMaster");
9
10    $query = $this->db->get();
11    if($query->num_rows() > 0){
12        foreach ($query->result() as $row) {
13            $data[] = $row;
14        }
15        return $data;
16    }
17    else
18        return FALSE;

```

```

19 }
20 public function insertStock($data)
21 {
22     return $this->db->insert('stockproduct', $data);
23 }
24 public function updateStock($data, $idStock)
25 {
26     $this->db->where('idStock', $idStock);
27     $this->db->update('stockproduct', $data);
28 }

```

**Kode Sumber 4.9 Fungsi *Add Stock***

#### 4.2.2.9 Fungsi *Manage User*

Fungsi ini digunakan untuk mengelola data user. Pada fungsi ini dapat mengambil data user, serta query untuk melakukan persetujuan user. Kode sumber tercantum pada Kode Sumber 4.10.

```

1 public function getDataUser($idUser)
2 {
3     $this->db->select("user.idUser, user.username,
4 user.company_name, user.email, user.no_telp, user.address,
5 province.namaprovinsi, user.city, user.PostalCode,
6 user.website, user.password,
7 usertype.UserType, user.status_user");
8     $this->db->from("user, UserType, province");
9     $this->db->where("idUser = '$idUser' and
10 user.idUserType = usertype.idUserType and user.idProvince
11 = province.idProvince");
12
13     $query = $this->db->get();
14
15     if($query->num_rows() > 0){
16         foreach ($query->result() as $row) {
17             $data[] = $row;
18         }
19         return $data;
20     }
21     else
22         return false;
23 }
24 public function approve($idUser)
25 {
26     $this->db->where('idUser', $idUser);

```

```

27     $this->db->update('user', array('status_user' =>
28     1));
29     }
30     public function reject($idUser)
31     {
32         $this->db->where('idUser', $idUser);
33         $this->db->update('user', array('status_user' =>
34         2));
35     }
36     public function updateUser($data, $email)
37     {
38         $this->db->where('email', $email);
39         $this->db->update('user', $data);
40     }

```

**Kode Sumber 4. 10 Fungsi *Manage User***

### 4.2.3 Implementasi Kelas *Controller*

Kelas *controller* pada sistem ini berfungsi untuk menghubungkan *View* dan *Model*. *Controller* berisikan fungsi-fungsi yang digunakan untuk pemrosesan data dan menjalankan fitur-fitur sistem serta untuk menangkap *request* data dari *client* lalu memberikan *response* berupa antarmuka. Pada subbab ini, akan dijelaskan fungsi-fungsi umum yang ada pada kelas *controller*.

#### 4.2.3.1 Fungsi Registrasi

Fungsi ini bertugas untuk memanggil halaman registrasi. Fungsi ini berisi *logic* untuk melakukan registrasi serta memasukkan data pengguna kedalam database. Kode sumber tercantum pada Kode Sumber 4. 11.

```

1     public function show_regis(){
2         $data['groups'] = $this->login_database-
3     >getUserType();
4         $data['province'] = $this->login_database-
5     >getProvince();
6         $this->load->view('User/register_view', $data);
7     }
8     public function new_user_registration()
9     {
10        $username = $this->input->post('tbName');

```

```

11     $company = $this->input->post('tbCompany');
12     $email = $this->input->post('tbEmail');
13     $telp = $this->input->post('tbPhone');
14     $address = $this->input->post('tbStreet');
15     $provinsi = $this->input->post('ddlProvince');
16     $city = $this->input->post('tbCity');
17     $postalcode = $this->input->post('tbPostalCode');
18     $website = $this->input->post('tbWeb');
19     $pass = $this->input->post('password');
20     $usertype = $this->input->post('ddlAkses');
21
22     $data = array(
23         'username' => $username,
24         'company_name' => $company,
25         'email' => $email,
26         'no_telp' => $telp,
27         'address' => $address,
28         'idProvince' => $provinsi,
29         'city' => $city,
30         'PostalCode' => $postalcode,
31         'website' => $website,
32         'password' => $pass,
33         'idUserType' => $usertype,
34         'status_user' => '0'
35     );
36
37     if ($this->login_database->isDuplicate($email)) {
38         $this->login_database-
39 >registration_model($data);
40         echo '<script language="javascript">';
41         echo 'alert("Registrasi berhasil dilakukan");
42 window.location.href = "" . site_url('index.php/login') .
43 ""';
44         echo '</script>';
45     }
46     else{
47         echo '<script language="javascript">';
48         echo 'alert("Email sudah digunakan");
49 window.history.back();';
50         echo '</script>';
51     }
52 }

```

**Kode Sumber 4. 11 Fungsi Registrasi**



### 4.2.3.2 Fungsi *getPrice*

Fungsi ini digunakan untuk mengambil harga yang telah dimasukkan perusahaan untuk ditampilkan pada halaman transaksi sehingga bisa ditampilkan secara otomatis ketika user melakukan transaksi. Kode sumber tercantum pada Kode Sumber 4. 12.

```

1 public function getPrice()
2 {
3     $idMeatType = $this->input->post('id', TRUE);
4     $idPerusahaan = $this->input->post('tbCompany',
5 TRUE);
6
7     //run the query for the price we specified earlier
8     $data['priceDrop'] = $this->order_request-
9 >getList($idMeatType, $idPerusahaan);
10    $output = null;
11    foreach ($data['priceDrop'] as $row)
12    {
13        //here we build a dropdown item line for each
14 query result
15        $output = $row->price;
16    }
17    echo $output;
18 }

```

**Kode Sumber 4. 12 Fungsi *getPrice***

### 4.2.3.3 Class Login

Class ini digunakan untuk mengelola data login pengguna. Class ini digunakan untuk masuk ke halaman utama sistem. Pada class ini terdapat fungsi-fungsi yang digunakan untuk login pengguna, pembuatan session serta logout dari sistem. Kode sumber tercantum pada Kode Sumber 4. 13.

### 4.2.3.4 Fungsi *insertToTransaction*

Fungsi ini digunakan untuk memasukkan data transaksi ke tabel transaksi. Kode sumber tercantum pada Kode Sumber 4. 14.

```

1 public function reLogin()
2 {
3     if($this->session->userdata('isLogin') == TRUE){
4         redirect('index.php/dashboard');
5     }
6     $data['error'] = TRUE;
7     $this->load->view('User/login_view', $data);
8 }
9 public function doLogin()
10 {
11     if($this->session->userdata('isLogin') == TRUE){
12         redirect('index.php/dashboard');
13     }
14
15     $email = $this->input->post('tbEmail');
16     $password = $this->input->post('tbPassword');
17
18     $query = $this->login_database->login_user($email,
19 $password);
20     if ($query->num_rows() > 0){
21         foreach ($query->result() as $row) {
22             $data = array(
23                 'idUser' => $row->idUser,
24                 'email' => $row->email,
25                 'username' => $row->username,
26                 'idUserType' => $row->idUserType,
27                 'status_user' => $row->status_user,
28                 'isLogin' => TRUE
29             );
30             $this->session->set_userdata($data);
31             redirect('index.php/dashboard');
32         }
33     }
34     else{
35         echo '<script language="javascript">';
36         echo 'window.location.href = "reLogin";';
37         echo '</script>';
38     }
39 }
40 public function logout()
41 {
42     $this->session->unset_userdata('isLogin');
43     session_destroy();
44     redirect('index.php/login', 'refresh');
45 }

```

**Kode Sumber 4. 13 Class Login**

```

1 public function NewTransaction()
2 {
3     $this->header();
4     $this->load->model('manage_supplier');
5     $data = array(
6         'idUser' => $this->input->post('tbUser'),
7         'PONum' => $this->input-
8 >post('tbPONumber'),
9         'PODate' => $this->input-
10 >post('tbPODate'),
11         'CompanyDestination' => $this->input-
12 >post('ddlUserlist'),
13         'status' => 0,
14         'status_kirim' => 0
15     );
16     $idtransaksi = $this->order_request-
17 >insertToTransaction($data);
18
19     $data['meattype'] = $this->manage_supplier-
20 >getMeatType();
21     $data['idtransaksi'] = $idtransaksi;
22     $temp = $this->manage_supplier-
23 >getPerusahaan($this->input->post('ddlUserlist'));
24
25     $data['company_id'] = $temp[0]->idUser;
26
27     $this->load->view('Customer/C_NewTransaction',
28 $data);
29     $this->load->view('Footer/footer');
30     $this->load-
31 >view('Footer/js/js_manajemen_datatable');
32     $this->load->view('Footer/js/js_dropdown');
33 }

```

**Kode Sumber 4. 14 Fungsi *insertToTransaction***

#### 4.2.3.5 Fungsi *do\_Upload*

Fungsi ini digunakan untuk melakukan upload berkas bukti pembayaran untuk transaksi yang telah dilakukan. Kode sumber tercantum pada Kode Sumber 4. 15.

```

1 public function save($idUsTransaction)
2 {
3     $idUser = $this->session->userdata('idUser');
4     $url = $this->do_upload();
5 }

```

```

6      $this->order_request->saveFile($idUserTransaction,
7      $url);
8
9      $temp = "../getUserType/" . $idUser;
10     echo '<script language="javascript">';
11     echo 'alert("Berkas Berhasil di upload");';
12     echo 'window.location.href = "' . $temp . "'";';
13     echo '</script>';
14 }
15 private function do_upload()
16 {
17     $type = explode('.', $_FILES["userfile"]["name"]);
18     $type = $type[count($type) - 1];
19     $url = "./upload/.uniqid(rand())." . $type;
20
21     if(in_array($type, array( "jpg", "jpeg", "png",
22 "gif", "pdf")))
23
24         if(is_uploaded_file($_FILES["userfile"]["tmp_name"]))
25             if
26 (move_uploaded_file($_FILES["userfile"]["tmp_name"], $url))
27                 return $url;
28     return "";
29 }

```

**Kode Sumber 4. 15 Kode Sumber *do\_Upload***

#### 4.2.3.6 Class Add Stock

Class ini berfungsi untuk melakukan penambahan stock barang serta update stock barang. Kode sumber tercantum pada Kode Sumber 4. 16.

```

1 public function S_insertStock()
2 {
3     $idUser = $this->session->userdata('idUser');
4     $data = array(
5         'date' => $this->input->post('tbDate'),
6         'idMeatMaster' => $this->input-
7     >post('ddlMeat'),
8         'idUser' => $this->input->post('tbUser'),
9         'quantity' => $this->input-
10    >post('tbQuantity')
11    );
12
13    $this->addStock->insertStock($data);
14    $temp = "Sadd_stock/" . $idUser;

```

```

15         echo '<script language="javascript">';
16         echo 'alert("Stock berhasil ditambah"); location.href
17         = "" . $temp . ""';
18         echo '</script>';
19     }
20     public function S_updateStock($idStock)
21     {
22         $idUser = $this->session->userdata('idUser');
23         $data = array(
24             'date' => $this->input->post('tbDate'),
25             'idMeatMaster' => $this->input-
26             >post('ddlMeat'),
27             'idUser' => $this->input->post('tbUser'),
28             'quantity' => $this->input-
29             >post('tbQuantity')
30         );
31         $this->addStock->updateStock($data, $idStock);
32
33         $temp = "../Sadd_Stock/" . $idUser;
34         echo '<script language="javascript">';
35         echo 'alert("Stock berhasil diupdate");
36         location.href = "" . $temp . ""';
37         echo '</script>';
38     }

```

**Kode Sumber 4. 16 Class Add Stock**

### 4.3 Implementasi Antarmuka Pengguna

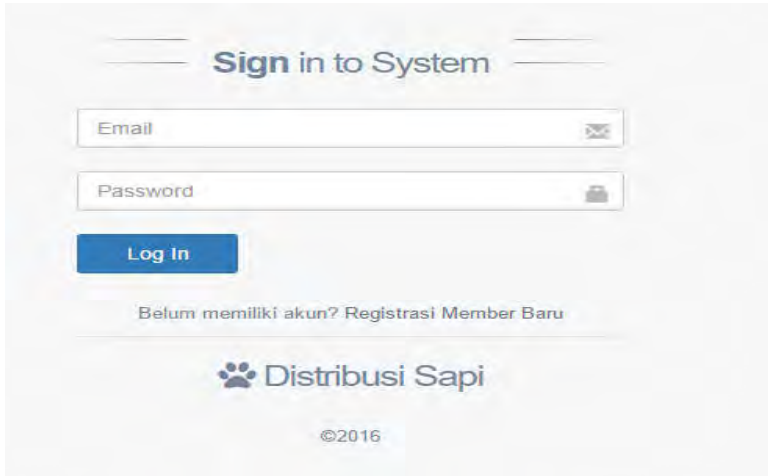
Pada subbab ini akan menjelaskan dan menampilkan tampilan halaman antarmuka sesuai dengan rancangan antarmuka yang terdapat pada bab 3.

#### 4.3.1 Halaman Login

Halaman ini merupakan halaman antarmuka pertama kali ketika pengguna membuka sistem distribusi daging sapi. Pada halaman ini terdapat form untuk login. Pengguna perlu memasukkan email dan password yang telah terdaftar sebelumnya. Pada halaman ini juga terdapat link ke menu registrasi jika pengguna belum terdaftar sebagai member. Implementasi antarmuka halaman login dapat dilihat pada Gambar 4. 1.

### 4.3.2 Halaman Registrasi User

Halaman ini merupakan halaman antarmuka yang digunakan user untuk melakukan registrasi member baru. Implementasi halaman login dapat dilihat pada Gambar 4. 2.



**Gambar 4. 1 Implementasi Antarmuka Halaman Login**

### 4.3.3 Halaman Approve User

Halaman ini merupakan halaman antarmuka yang digunakan untuk melakukan *approval user* yang telah melakukan registrasi. Halaman antarmuka *approve user* ditunjukkan pada Gambar 4. 3. Lihat detail user ditunjukkan pada Gambar 4. 4. Serta untuk melihat detail user yang telah disetujui atau ditolak ditunjukkan pada Gambar 4. 5.

### 4.3.4 Halaman Mengelola Data Admin

Halaman ini merupakan halaman antarmuka mengelola data admin. Pada halaman ini dapat melihat data admin, mengubah serta menambah admin baru. Halaman antarmuka mengelola data

admin dapat dilihat pada Gambar 4. 6. Halaman antarmuka menambah data admin dapat dilihat pada Gambar 4. 7. Melihat data admin ditunjukkan pada Gambar 4. 8. Serta mengubah data admin dapat dilihat pada Gambar 4. 9.

The image shows a registration form titled "Form Registrasi" and "CreateAccount". The form includes the following fields:

- Nama \* (Name) with a person icon
- Perusahaan \* (Company) with a building icon
- Email \* (Email (email@domain.com)) with an envelope icon
- Telephone \* (Telephone) with a telephone icon
- Alamat Perusahaan \* (Company Address) with a building icon
- Provinsi \* (Province) with a dropdown menu showing "Pilih Provinsi"
- Kota \* (City) with a building icon
- Kode Pos \* (Post Code) with a building icon
- Website URL (Website URL) with a globe icon and the value "www.website.com"
- Password \* (Password) with a lock icon
- Konfirmasi Password \* (Confirm Password) with a lock icon
- Hak Akses \* (Access Rights) with a dropdown menu showing "Supplier"

At the bottom of the form, there is a blue "Submit" button and a link "Saya sudah memiliki akun" (I already have an account).

**Gambar 4. 2 Implementasi Antarmuka Halaman Registrasi**

List User

Show 10 entries Cari:

Nama	Perusahaan	Email	Kota	Website	Status User	Tipe User	Aksi
PT. Superindo Utama	PT. Superindo Utama	superindo@gmail.com	Surabaya	-	User baru	Customer	<a href="#">Lihat Detail</a> <a href="#">Setuju</a> <a href="#">Tolak</a>

Showing 1 to 1 of 1 entries Previous Next

**Gambar 4. 3 Implementasi Antarmuka Halaman *Approve User***

Approval User

Home | Approval User | Lihat Detail User

User Record 
[Setuju](#)
[Tolak](#)
[Kembali](#)

Nama : PT. Superindo Utama

Perusahaan : PT. Superindo Utama

Email : superindo@gmail.com

Telephone : 085731046030

Alamat Perusahaan : Jl. Arief Rahman Hakim no 139

Kota : Surabaya

Kode Pos : 60111

Website : -

Status User : User belum di konfirmasi

**Gambar 4. 4 Implementasi Antarmuka Halaman Lihat Detail User**



List User

Supplier Distributor Customer

Nama	Perusahaan	Email	Status User	Aksi
cs	cs	cs@gmail.com	Disetujui	<a href="#">Lihat Detail</a>
PT. Superindo Utama	PT. Superindo Utama	superindo@gmail.com	Disetujui	<a href="#">Lihat Detail</a>
PT Superindo Utama	PT Superindo Utama	qsthiiaulia1996@gmail.com	Disetujui	<a href="#">Lihat Detail</a>

**Gambar 4. 5 Implementasi Antarmuka Halaman List User**

List Admin

Tampilkan Admin

Show 10 entries

Nama	Email	Nomor Telephone	Aksi
Admin	admin1@gmail.com	783451	<a href="#">Lihat Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Admin	admin@gmail.com	081320349854	<a href="#">Lihat Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

**Gambar 4. 6 Implementasi Antarmuka Halaman Mengelola Data Admin**

### 4.3.5 Halaman Transaksi

Halaman ini merupakan halaman antarmuka transaksi. Pada halaman ini terdapat halaman tambah transaksi baru serta halaman input detail transaksi. Halaman antarmuka tambah transaksi baru dapat dilihat pada Gambar 4. 10 serta halaman antarmuka input detail transaksi dapat dilihat pada Gambar 4. 11.

Data Admin Baru

Nama \*

Email \*

Telephone \*

Alamat Admin \*

Provinsi \*

Kota \*

Kode Pos \*

Password \*

Konfirm Password \*

Hak Akses \*

**Gambar 4. 7 Implementasi Antarmuka Halaman Menambah Data Admin**

Management Admin

# Home / Management Admin / Lihat Detail Admin

Data Administrator

Nama : Admin

Email : admin1@gmail.com

Telephone : 783451

Alamat Admin : Surabaya

Provinsi : Jawa Timur

Kota : Surabaya

Kode Pos : 60111

Password : aaaa

**Gambar 4. 8 Implementasi Antarmuka Halaman Melihat Data Admin**

Management Admin

Home Management Admin Edit Admin Record

Update Record

Nama \* Admin

Email \* admin1@gmail.com

Telephone \* 783451

Alamat Admin \* Surabaya

Provinsi \* Jawa Timur

Kota \* Surabaya

Kode Pos \* 60111

Password \* ....

Konfirm Password \* ....

Hak Akses \* Admin

Update Cancel

**Gambar 4. 9 Implementasi Antarmuka Halaman Mengubah Data Admin**

Order Baru

Pilih Jenis Perusahaan \* Supplier

Pilih Perusahaan \* Its

Nomor Order \* ST150620161

Tanggal Order \* 06/18/2016

Cancel Next

**Gambar 4. 10 Implementasi Antarmuka Halaman Transaksi Baru**

Transaksi

---

Perusahaan Tujuan  Tanggal Order

Nomor Order

Detail Order Transaksi

Jenis Daging	Jumlah Pemesanan (Kg)	Harga Barang	Total Harga
Daging Puncuk / Blade *	<input type="text" value="100"/>	<input type="text" value="120000"/>	<input type="text" value="12000000"/>
Daging Paha Depan / Chuck *	<input type="text" value="100"/>	<input type="text" value="130000"/>	<input type="text" value="13000000"/>

**Gambar 4. 11 Implementasi Antarmuka Halaman Detail Transaksi**

### 4.3.6 Halaman Terima Transaksi

Halaman ini merupakan halaman antarmuka untuk menampilkan transaksi yang ditujukan oleh pelanggan. Pada halaman ini terdapat menu untuk melakukan persetujuan transaksi, melakukan konfirmasi pengiriman serta melakukan cetak invoice. Halaman antarmuka persetujuan transaksi dapat dilihat pada Gambar 4. 12. Halaman antarmuka konfirmasi pengiriman dapat dilihat pada Gambar 4. 13. Serta untuk melakukan cetak invoice dapat dilihat pada Gambar 4. 14.

Permintaan Order Kembali

---

Perusahaan Tujuan  Tanggal Order

Nomor Order  Total Pembayaran

Jenis Daging	Jumlah Pemesanan (Kg)	Harga Barang (Rp)	Total Harga (Rp)	Status Order
Daging Puncuk / Blade	100	120000.00	12000000.00	Belum Ditetapkan
Daging Paha Depan / Chuck	100	130000.00	13000000.00	Belum Ditetapkan

Setuju  Tolak

**Gambar 4. 12 Implementasi Antarmuka Halaman Persetujuan Transaksi**

Permintaan Order Kembali

---

Perusahaan Tujuan  Tanggal Order

Nomor Order  Total Pembayaran

Status Order

Jenis Daging	Jumlah Pemesanan (Kg)	Harga Barang (Rp)	Total Harga (Rp)
Daging Puncuk / Blade	100	120000.00	12000000.00
Daging Paha Depan / Chuck	100	130000.00	13000000.00

Kirim

**Gambar 4. 13 Implementasi Antarmuka Halaman Konfirmasi Pengiriman**

Invoice Pengiriman Produk			
<b>Order</b>		<b>Kepada</b>	<b>Dikirim Ke</b>
Tanggal Kirim	2016-06-20	PT. Superindo Utama	PT. Superindo Utama
Nomor Order	DO150620161	PT. Superindo Utama	
Tanggal Order	2016-06-18	Surabaya superindo@gmail.com	
Jenis Daging	Jumlah Pemesanan (Kg)	Harga Barang	Total Harga
Daging Punuk / Blade	120	Rp 120000.00	Rp 14400000.00
Daging Paha Depan / Chuck	100	Rp 130000.00	Rp 13000000.00
<b>Total Pembayaran</b>			<b>Rp 27400000.00</b>

**Gambar 4. 14 Implementasi Halaman Cetak Invoice**

### 4.3.7 Halaman Upload Berkas Bukti Pembayaran

Halaman ini merupakan halaman antarmuka untuk melakukan upload berkas bukti pembayaran untuk transaksi yang telah dilakukan. Halaman antarmuka upload berkas dapat dilihat pada Gambar 4. 15.

Upload Berkas Pembayaran

Home / Transaksi Produk / Upload Berkas

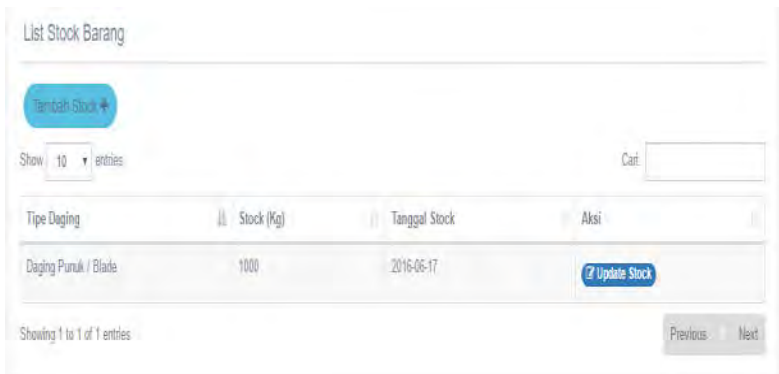
Upload Berkas

No file chosen

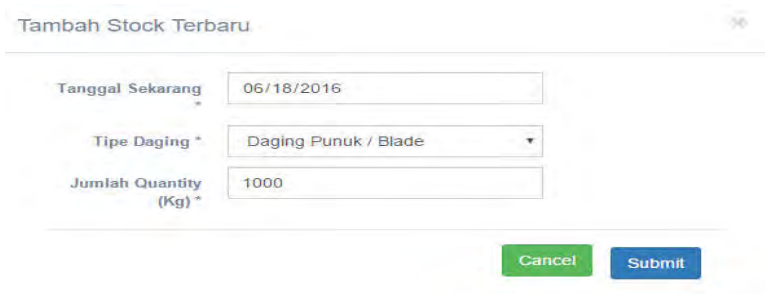
**Gambar 4. 15 Implementasi Antarmuka Halaman Upload Berkas**

### 4.3.8 Halaman Mengelola Stock Barang

Halaman ini merupakan halaman antarmuka untuk mengelola data stock. Pada halaman ini terdapat menu untuk menambah stock barang, update stock serta melihat stock tersedia. Untuk halaman antarmuka mengelola data stock barang dapat dilihat pada Gambar 4. 16. Halaman antarmuka menambah stock barang dapat dilihat pada Gambar 4. 17. Halaman antarmuka update stock dapat dilihat pada Gambar 4. 18. Serta halaman antarmuka melihat stock tersedia dapat dilihat pada Gambar 4. 19.



**Gambar 4. 16 Implementasi Antarmuka Halaman Mengelola Data Stock**



**Gambar 4. 17 Implementasi Antarmuka Halaman Menambah Data Stock**

Update Data

Tanggal Sekarang \* 06/17/2016

Tipe Daging \* Daging Punuk / Blade

Quantity (Kg) \* 1000

Update Cancel

**Gambar 4. 18 Implementasi Antarmuka Halaman Update Stock**

Pilih Perusahaan Terkait \*

Show 10 entries

Tipe Daging	Stock (Kg)	Tanggal Stock
Daging Punuk / Blade	1000	2016-06-17

**Gambar 4. 19 Implementasi Antarmuka Halaman Melihat Stock Tersedia**

### 4.3.9 Halaman History Transaksi

Halaman ini merupakan halaman antarmuka untuk melihat transaksi yang telah dilakukan pengguna. Halaman history transaksi dapat dilihat pada Gambar 4. 20.

### 4.3.10 Halaman Grafik Produksi Daging Sapi

Halaman ini merupakan halaman antarmuka untuk melihat grafik produksi daging sapi di Indonesia berdasarkan pembagian pulau. Halaman grafik produksi dapat dilihat pada Gambar 4. 21.

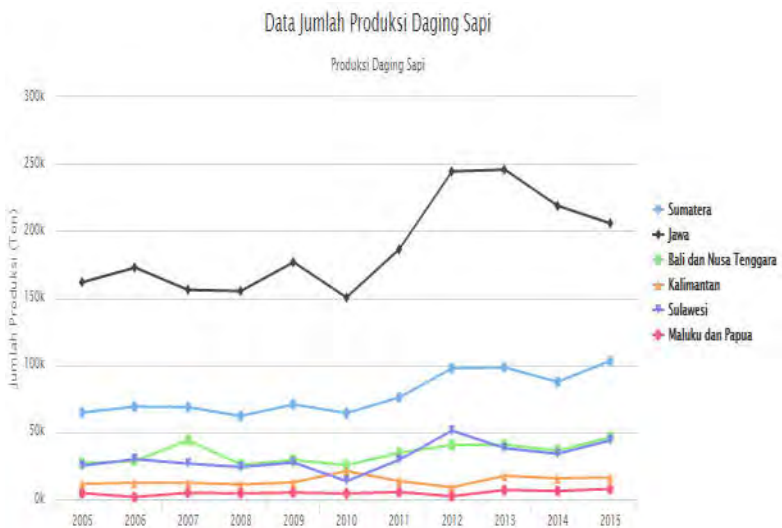


Pilih Perusahaan Terkait\*

Show  entries

Tanggal Order	Perusahaan	Jenis Barang	Jumlah Pemesanan
2016-06-09	its	Daging Puncuk / Blade	100
2016-06-09	its	Daging Paha Depan / Chuck	100

**Gambar 4. 20** Implementasi Antarmuka Halaman History Transaksi



**Gambar 4. 21** Implementasi Antarmuka Halaman Grafik Produksi Daging Sapi

*[Halaman ini sengaja dikosongkan]*

## **BAB V**

### **PENGUJIAN DAN EVALUASI**

Pada bab ini akan dijelaskan pengujian dan evaluasi yang telah dilakukan pada aplikasi yang telah dikerjakan serta analisa dari uji coba yang telah dilakukan. Pembahasan pengujian meliputi lingkungan uji coba, skenario uji coba yang memperhatikan terpenuhinya kebutuhan fungsionalitas dari sistem serta evaluasi pengujian.

#### **5.1 Lingkungan Pengujian**

Lingkungan pengujian meliputi perangkat keras maupun perangkat lunak tempat pengujian sistem dilakukan. Pengujian sistem informasi ini dilakukan pada komputer dengan spesifikasi pada Tabel 5. 1.

**Tabel 5. 1 Spesifikasi Sistem Pengujian**

<b>Spesifikasi</b>	<b>Deskripsi</b>
CPU	Intel® Core™ i5-32 10M 2.5 GHz
RAM	4.00 GB
Sistem Operasi	Windows 10 Home Professional 64 bit

#### **5.2 Skenario Pengujian**

Pada subbab ini dijelaskan mengenai skenario pengujian yang digunakan. Seperti yang sudah dijelaskan pada pengantar di bab 5, bahwa pengujian merupakan pengujian fungsionalitas yang menggunakan metode *blackbox*. Metode *blackbox* merupakan metode dimana pengujian ditekankan pada pola *input* dan *output* yang sesuai dengan skenario. Pengujian dilakukan dengan mengacu kasus penggunaan yang dijelaskan pada bab 3, analisis dan perancangan sistem.

Pengujian juga dilakukan kepada responden yang mewakili pengguna dari tiap aktor di sistem.

### 5.2.1 Kasus Pengujian Melakukan Registrasi User

Pada kasus pengujian ini pengguna akan melakukan registrasi sebelum masuk kedalam sistem. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir terdapat pada Tabel 5. 2. Tampilan pengisian data terdapat pada Gambar 5. 1 dan Gambar 5. 2.

**Tabel 5. 2 Kasus Uji Melakukan Registrasi User**

<b>ID</b>	<b>UJ-001</b>
<b>Kasus Penggunaan</b>	Melakukan Registrasi User.
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji apakah sistem dapat melakukan registrasi <i>user</i> .
<b>Skenario</b>	Pengguna melakukan registrasi.
<b>Kondisi Awal</b>	Akun <i>user</i> belum ada.
<b>Langkah Pengujian</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu registrasi.</li> <li>2. Aktor mengisi data-data perusahaan yang diperlukan.</li> <li>3. Aktor memilih tombol submit.</li> </ol>
<b>Hasil yang Diharapkan</b>	Pengguna telah memiliki akun untuk masuk kedalam sistem.
<b>Hasil yang Didapat</b>	Data pengguna telah tersimpan di basis data.
<b>Kondisi Akhir</b>	Pengguna telah ter-registrasi.
<b>Kesimpulan</b>	Berhasil

**CreateAccount**

Nama *	PT Superindo Utama
Perusahaan *	PT Superindo Utama
Email *	qisthiaulia1990@gmail.com
Telephone *	085731846030
Alamat Perusahaan *	Jl. Arief Rahman Hakim no 139
Provinsi *	Jawa Timur
Kota *	Surabaya
Kode Pos *	60111
Website URL	www.website.com
Password *	****
Konfirmasi Password *	****
Hak Akses *	Customer

[Saya sudah memiliki akun](#)

**Gambar 5. 1 Tampilan Pengujian Pengisian Data Registrasi User**



**Gambar 5. 2 Tampilan Hasil Pengujian Registrasi User**

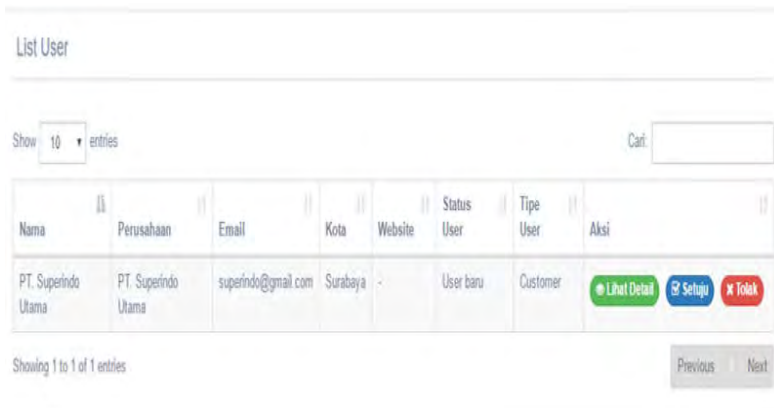
### 5.2.2 Kasus Pengujian Approval Data User

Pada kasus pengujian ini akan menjelaskan kasus uji untuk melakukan approval pengguna. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir terdapat pada Tabel 5. 3. Tampilan pengujian data terdapat pada Gambar 5. 3 dan Gambar 5. 4.

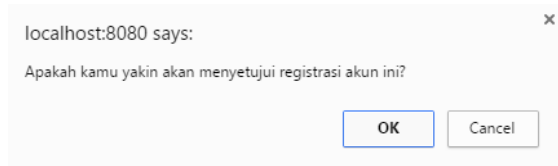
**Tabel 5. 3 Kasus Uji Approval User**

<b>ID</b>	<b>UJ-002</b>
-----------	---------------

<b>Kasus Penggunaan</b>	Approval User
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji apakah administrator dapat melakukan persetujuan terhadap member baru yang telah terdaftar
<b>Skenario</b>	Admin melihat detail data pengguna dan melakukan persetujuan.
<b>Kondisi Awal</b>	Data user belum disetujui oleh admin.
<b>Langkah Pengujian</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu approval user.</li> <li>2. Aktor memilih tombol lihat detail.</li> <li>3. Aktor memilih tombol setuju atau tolak.</li> </ol>
<b>Hasil yang Diharapkan</b>	Akun pengguna yang telah ter-registrasi disetujui atau ditolak.
<b>Hasil yang Didapat</b>	Data pengguna berhasil disetujui.
<b>Kondisi Akhir</b>	Pengguna dapat melakukan transaksi.
<b>Kesimpulan</b>	Berhasil



**Gambar 5. 3 Tampilan Persetujuan Pengguna**



**Gambar 5. 4 Tampilan Persetujuan Pengguna**

### 5.2.3 Kasus Pengujian Melihat History Proses Transaksi

Pada kasus pengujian ini akan menjelaskan kasus uji untuk melihat history proses transaksi yang telah dilakukan oleh pengguna sistem. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir yang terdapat pada Tabel 5. 4. Tampilan pengujian data terdapat pada Gambar 5. 5.

**Tabel 5. 4 Kasus Uji Melihat History Proses Transaksi**

<b>ID</b>	<b>UJ-003</b>
<b>Kasus Penggunaan</b>	Melihat History Proses Transaksi
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji apakah administrator dapat melihat history transaksi distribusi daging sapi berdasarkan perusahaan yang dipilih.
<b>Skenario</b>	Admin memilih jenis perusahaan yang akan dilihat history transaksi yang telah dilakukan.
<b>Kondisi Awal</b>	History transaksi belum tersedia.
<b>Langkah Pengujian</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu history transaksi.</li> <li>2. Aktor memilih perusahaan yang akan dilihat transaksinya.</li> </ol>
<b>Hasil yang Diharapkan</b>	Terdapat list data transaksi yang telah dilakukan pengguna.
<b>Hasil yang Didapat</b>	Data transaksi yang telah dilakukan pengguna dapat ditampilkan.
<b>Kondisi Akhir</b>	History transaksi tersedia.
<b>Kesimpulan</b>	Berhasil

Pilih Perusahaan Terkait\*

Show 10 entries

Tanggal Order	Perusahaan	Jenis Barang	Jumlah Pemesanan
2016-05-09	its	Daging Punuk / Blade	100
2016-05-09	its	Daging Paha Depan / Chuck	100

**Gambar 5. 5 Tampilan Pengujian History Transaksi**

### 5.2.4 Kasus Pengujian Mengelola Data Admin

Pada kasus pengujian ini akan dijelaskan kasus uji untuk penambahan admin baru. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir yang terdapat pada Tabel 5. 5.

**Tabel 5. 5 Kasus Uji Penambahan Admin Baru**

<b>ID</b>	<b>UJ-004</b>
<b>Kasus Penggunaan</b>	Penambahan Data Admin Baru
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji <i>input</i> data pada penambahan admin baru
<b>Skenario</b>	Admin menambahkan data anggota admin baru
<b>Kondisi Awal</b>	Data admin baru belum tersedia.
<b>Langkah Pengujian</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu add admin.</li> <li>2. Aktor memilih tombol tambah admin dan mengisi data admin.</li> <li>3. Aktor memilih tombol simpan.</li> </ol>
<b>Hasil yang Diharapkan</b>	Terdapat data admin yang baru.
<b>Hasil yang Didapat</b>	Program merespon data dan menambahkan data admin yang baru.



<b>Kondisi Akhir</b>	Data Admin baru tersedia.
<b>Kesimpulan</b>	Berhasil

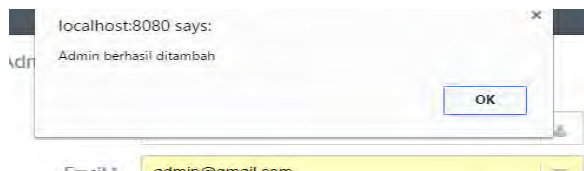
Bentuk pengisian dilakukan dengan *input* pada sistem berupa teks. Hasil dari skenario adalah data admin yang baru tersimpan dalam basis data dan ditampilkan dalam bentuk *datatable*. Tampilan pengisian data dan kondisi akhir dari skenario terdapat pada Gambar 5. 6, Gambar 5. 7 dan Gambar 5. 8.

The image shows a web form titled "Data Admin Baru" with the following fields and values:

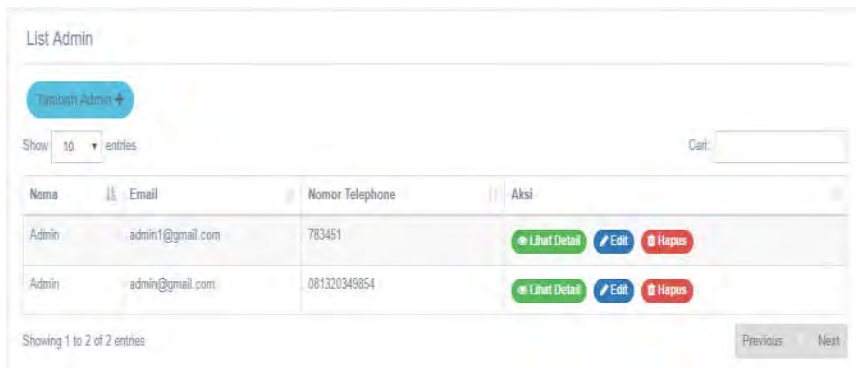
- Nama \*: Admin
- Email \*: admin@gmail.com
- Telephone \*: 081320349854
- Alamat Admin \*: Jalan Semolowaru Tengah II
- Provinsi \*: Jawa Timur
- Kota \*: Surabaya
- Kode Pos \*: 60111
- Password \*: \*\*\*\*
- Konfirm Password \*: \*\*\*\*
- Hak Akses \*: Admin

At the bottom right of the form are two buttons: "Cancel" (green) and "Submit" (blue).

**Gambar 5. 6 Pengisian Data Uji Penambahan Admin**



**Gambar 5. 7 Tampilan Notifikasi Keberhasilan Penambahan Data**



**Gambar 5. 8 Tampilan Hasil Pengujian Penambahan Admin**

### 5.2.5 Kasus Pengujian Melakukan Pemesanan Produk

Pada kasus pengujian ini akan dijelaskan kasus uji untuk melakukan transaksi pemesanan produk. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir yang terdapat pada Tabel 5. 6.

**Tabel 5. 6 Kasus Uji Melakukan Pemesanan Produk**

<b>ID</b>	<b>UJ-005</b>
<b>Kasus Penggunaan</b>	Melakukan Pemesanan Produk
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Pengujian untuk melakukan transaksi apakah pengguna dapat meminta pemesanan produk dengan tepat.
<b>Skenario</b>	Pengguna melakukan pemesanan barang.
<b>Kondisi Awal</b>	Pengguna berada pada halaman order produk
<b>Langkah Pengujian</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu order produk.</li> <li>2. Aktor memilih tombol order baru dan mengisi data transaksi.</li> <li>3. Aktor memilih tombol <i>next</i>.</li> </ol>

	<p>4. Aktor mengisi data-data transaksi berupa jumlah kuantiti barang yang diinginkan.</p> <p>5. Aktor memilih tombol submit.</p>
<b>Hasil yang Diharapkan</b>	Terdapat data pemesanan baru.
<b>Hasil yang Didapat</b>	Data pemesanan baru telah tersimpan di basisdata.
<b>Kondisi Akhir</b>	Data Transaksi pemesanan tersedia.
<b>Kesimpulan</b>	Berhasil

Bentuk pengisian dilakukan dengan *input* pada sistem berupa teks. Hasil dari skenario adalah pemesanan produk tersimpan dalam basis data dan ditampilkan dalam bentuk *datatable*. Tampilan pemesanan produk dan kondisi akhir dari skenario terdapat pada Gambar 5. 9, Gambar 5. 10, Gambar 5. 11 dan Gambar 5. 12.

**Gambar 5. 9 Pengisian Data Uji Pemesanan Produk**

Transaksi

---

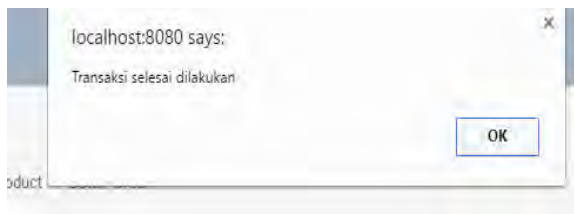
Perusahaan Tujuan:  Tanggal Order:

Nomor Order:

Detail Order Transaksi

Jenis Daging	Jumlah Pesanan (Kg)	Harga Barang	Total Harga
Daging Puncuk / Blade	<input type="text" value="100"/>	<input type="text" value="120000"/>	<input type="text" value="12000000"/>
Daging Paha Depan / Chuck	<input type="text" value="100"/>	<input type="text" value="130000"/>	<input type="text" value="13000000"/>

**Gambar 5. 10 Pengujian Data Uji Pemesanan Produk**



**Gambar 5. 11 Tampilan Notifikasi Keberhasilan Transaksi**

The screenshot shows a web interface for a transaction. At the top, there is a header 'Transaksi' and a navigation bar with 'Transaksi' and 'Status Pengiriman'. Below this is a blue button labeled 'Daftar Menu'. The main content is a table with the following data:

Nomor Order	Tanggal Order	Perusahaan Tujuan	Total Pembayaran	Slip Transfer	Status Order	Aksi
ST150628162	2016-06-18	its	Rp 15400000.00	Belum bisa upload bukti pembayaran	Belum Dikerjakan	<a href="#">Lihat Detail</a>

**Gambar 5. 12 Tampilan Hasil Pengujian Pemesanan Produk**

### 5.2.6 Kasus Pengujian Menerima Pemesanan Produk

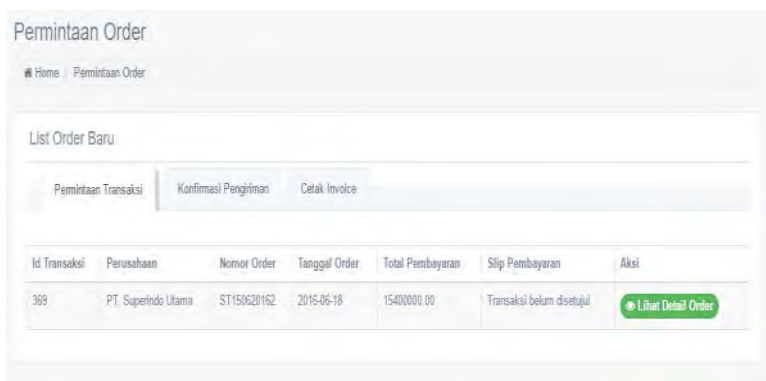
Pada kasus pengujian ini akan dijelaskan kasus uji untuk menerima pemesanan produk yang telah dilakukan oleh pihak *customer*. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir yang terdapat pada Tabel 5. 7.

**Tabel 5. 7 Kasus Uji Menerima Pemesanan Produk**

<b>ID</b>	<b>UJ-006</b>
<b>Kasus Penggunaan</b>	Menerima Pemesanan Produk
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji apakah sistem dapat menampilkan pemesanan produk yang telah dilakukan pengguna serta melakukan persetujuan transaksi.
<b>Skenario</b>	Pengguna menerima pemesanan barang serta melakukan persetujuan transaksi.
<b>Kondisi Awal</b>	Pengguna berada pada halaman daftar permintaan order
<b>Langkah Pengujian</b>	1. Aktor memilih menu daftar permintaan order.

	<ol style="list-style-type: none"> <li>2. Aktor memilih tombol lihat detail transaksi.</li> <li>3. Aktor menyetujui atau menolak transaksi.</li> </ol>
<b>Hasil yang Diharapkan</b>	Data transaksi ditampilkan dan transaksi dapat disetujui atau ditolak.
<b>Hasil yang Didapat</b>	Data transaksi tersedia.
<b>Kondisi Akhir</b>	Transaksi telah disetujui.
<b>Kesimpulan</b>	Berhasil

Hasil dari skenario adalah tampilan penerimaan pemesanan produk dari *customer* yang telah melakukan transaksi permintaan produk. Tampilan pemesanan produk dan kondisi akhir dari skenario terdapat pada Gambar 5. 13, Gambar 5. 14 dan Gambar 5. 15.



**Gambar 5. 13 Tampilan List Permintaan Produk**

Permintaan Order Kembali

---

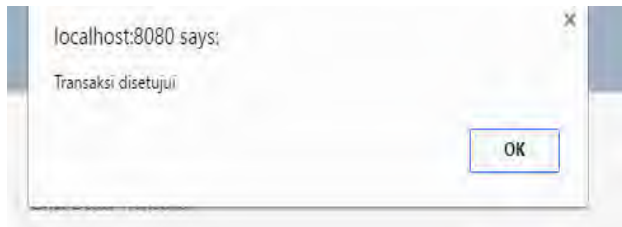
Perusahaan Tujuan: 
 Tanggal Order:

Nomor Order: 
 Total Pembayaran:

Jenis Daging	Jumlah Pemesanan (Kg)	Harga Barang (Rp)	Total Harga (Rp)	Status Order
Daging Puncuk / Blade	100	120000.00	12000000.00	Belum Disetujui
Daging Paha Depan / Chuck	100	130000.00	13000000.00	Belum Disetujui

Setuju
  Tolak

**Gambar 5. 14 Tampilan Halaman Persetujuan Transaksi**



**Gambar 5. 15 Tampilan Notifikasi Keberhasilan Persetujuan Transaksi**

### 5.2.7 Kasus Pengujian Upload Berkas Bukti Pembayaran

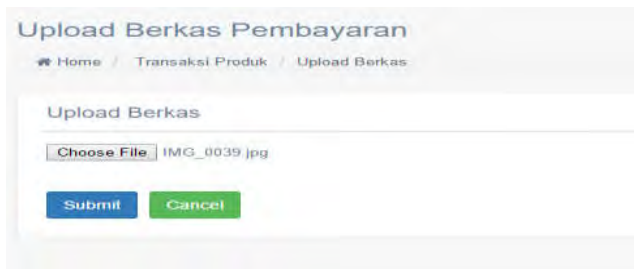
Pada kasus pengujian ini akan dijelaskan kasus uji untuk melakukan *upload* berkas pembayaran yang telah dilakukan serta menampilkan pada halaman pengguna. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir yang terdapat pada Tabel 5. 8.

**Tabel 5. 8 Kasus Uji Melakukan Upload Berkas Bukti Pembayaran**

<b>ID</b>	<b>UJ-007</b>
-----------	---------------

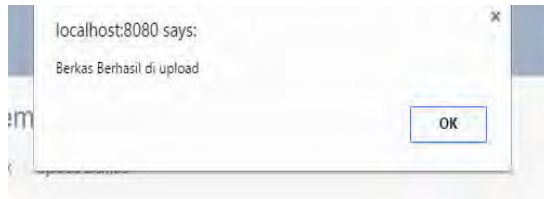
<b>Kasus Penggunaan</b>	Upload Berkas Bukti Pembayaran
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji apakah sistem dapat menyimpan hasil upload berkas dalam bentuk image atau file.
<b>Skenario</b>	Pengguna melakukan upload berkas bukti telah melakukan pembayaran.
<b>Kondisi Awal</b>	Bukti pembayaran belum tersimpan di basis data.
<b>Langkah Pengujian</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih button upload bukti pembayaran.</li> <li>2. Aktor memilih file berkas yang ingin diupload.</li> <li>3. Aktor memilih tombol submit.</li> </ol>
<b>Hasil yang Diharapkan</b>	Data berkas pembayaran tersimpan dibasis data serta dapat ditampilkan di tampilan antarmuka.
<b>Hasil yang Didapat</b>	Berkas berhasil dimasukkan ke basis data.
<b>Kondisi Akhir</b>	Hasil upload berkas tersedia pada menu status transaksi.
<b>Kesimpulan</b>	Berhasil

Hasil dari skenario adalah tampilan bukti pembayaran pada halaman pengguna. Tampilan upload bukti pembayaran dan kondisi akhir dari skenario terdapat pada Gambar 5. 16, Gambar 5. 17, Gambar 5. 18.



**Gambar 5. 16 Pengujian Data Upload Berkas Pembayaran**





**Gambar 5. 17 Tampilan Notifikasi Keberhasilan Upload Berkas**



**Gambar 5. 18 Tampilan Hasil Upload Bukti Pembayaran**

## 5.2.8 Kasus Pengujian Menambah Jumlah Stock Barang

Pada kasus pengujian ini akan dijelaskan kasus uji untuk melakukan penambahan stock barang yang tersedia serta melakukan pembaharuan stock. Detail pengujian berupa skenario, kondisi awal dan kondisi akhir yang terdapat pada Tabel 5. 9 dan Tabel 5. 10.

**Tabel 5. 9 Kasus Uji Menambah Stock barang**

<b>ID</b>	<b>UJ-008</b>
<b>Kasus Penggunaan</b>	Menambah Jumlah Stock Tersedia
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji apakah sistem dapat menambah stock barang yang tersedia.
<b>Skenario</b>	Pengguna mengisi data stock barang yang tersedia.
<b>Kondisi Awal</b>	Stock barang belum tersedia.
<b>Langkah Pengujian</b>	1. Aktor memilih menu update stock. 2. Aktor memilih button tambah stock.

	<ol style="list-style-type: none"> <li>3. Aktor mengisi jumlah stock yang tersedia.</li> <li>4. Aktor memilih tombol submit.</li> </ol>
<b>Hasil yang Diharapkan</b>	Data stock barang tersimpan dibasis data.
<b>Hasil yang Didapat</b>	Data berhasil dimasukkan ke basis data.
<b>Kondisi Akhir</b>	Stock barang telah tersedia.
<b>Kesimpulan</b>	Berhasil

Hasil dari skenario adalah tampilan jumlah ketersediaan stock barang pada halaman pengguna. Tampilan penambahan stock dan kondisi akhir dari skenario terdapat pada Gambar 5. 19, Gambar 5. 20 dan Gambar 5. 21.

**Gambar 5. 19 Pengujian Data Menambah Jumlah Stock Barang**

**Gambar 5. 20 Tampilan Notifikasi Keberhasilan Proses**

List Stock Barang

Tambah Stock +

Show 10 entities

Tipe Daging	Stock (Kg)	Tanggal Stock	Aksi
Daging Paha Depan / Chuck	1000	2016-06-15	Update Stock
Daging Puncuk / Blade	1000	2016-06-09	Update Stock
Has Dalam / Tenderloin	500	2016-06-18	Update Stock
Lamuir / Cube Roll	1500	2016-06-15	Update Stock

Showing 1 to 4 of 4 entries

Previous Next

**Gambar 5. 21 Tampilan Hasil Uji Penambahan Stock Barang**

**Tabel 5. 10 Kasus Uji Memperbaharui Stock Barang**

<b>ID</b>	<b>UJ-009</b>
<b>Kasus Penggunaan</b>	Memperbaharui Jumlah Stock Tersedia
<b>Sub Kasus</b>	-
<b>Tujuan Pengujian</b>	Menguji apakah sistem dapat memperbaharui stock barang yang tersedia.
<b>Skenario</b>	Pengguna mengubah data stock barang yang tersedia.
<b>Kondisi Awal</b>	Stock barang belum di <i>update</i> .
<b>Langkah Pengujian</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu update stock.</li> <li>2. Aktor memilih tombol update stock.</li> <li>3. Aktor mengisi jumlah stock yang tersedia.</li> <li>4. Aktor memilih tombol submit.</li> </ol>
<b>Hasil yang Diharapkan</b>	Data stock barang tersimpan dibasis data.
<b>Hasil yang Didapat</b>	Data berhasil dimasukkan ke basis data.
<b>Kondisi Akhir</b>	Stock barang telah di <i>update</i> .

<b>Kesimpulan</b>	Berhasil
-------------------	----------

Hasil dari skenario adalah data jumlah stock barang telah terupdate dengan data terbaru. Tampilan pembaharuan stock dan kondisi akhir dari skenario terdapat pada Gambar 5. 22 dan Gambar 5. 23.

Update Data

---

Tanggal Sekarang \*

Tipe Daging \*

Quantity (Kg) \*

---

**Gambar 5. 22 Pengujian Data Pembaharuan Stock**



**Gambar 5. 23 Tampilan Notifikasi Keberhasilan Pembaharuan Stock**

### 5.3 Evaluasi Pengujian

Pada bab ini dijelaskan mengenai hasil pengujian fungsionalitas. Evaluasi yang diberikan adalah evaluasi pengujian fungsionalitas dan *feedback* yang didapatkan dari kuesioner.

Pada Tabel 5. 11 merupakan evaluasi fungsionalitas yang dilakukan oleh Mbak Qiqi dari pihak PT. Superindo Utama selaku mewakili *customer* yang melakukan *entry* data utama sistem distribusi daging sapi.

**Tabel 5. 11 Evaluasi Pengujian oleh Mbak Qiqi Sebagai Customer**

Aktor : Mbak Qiqi Peran : Customer		
No.	Kode Kasus Pengujian	Terpenuhi
1	UJ-001	√
2	UJ-005	√
3	UJ-007	√
4	UJ-008	√

Untuk peran admin, supplier serta distributor diperankan oleh Bu Erma yang memahami berjalannya sistem distribusi daging sapi. Pada Tabel 5. 12 tersedia evaluasi fungsionalitas yang mewakili administrator. Pada Tabel 5. 13 tersedia evaluasi fungsionalitas yang mewakili supplier, serta pada Tabel 5. 14 tersedia evaluasi fungsionalitas yang mewakili distributor.

**Tabel 5. 12 Evaluasi Pengujian oleh Bu Erma Sebagai Administrator**

Aktor : Bu Erma Peran : Administrator		
No.	Kode Kasus Pengujian	Terpenuhi
1	UJ-002	√
2	UJ-003	√

**Tabel 5. 13 Evaluasi Pengujian oleh Bu Erma Sebagai Supplier**

Aktor : Bu Erma Peran : Supplier		
No.	Kode Kasus Pengujian	Terpenuhi
1	UJ-001	√
2	UJ-006	√
3	UJ-008	√

**Tabel 5. 14 Evaluasi Pengujian oleh Bu Erma Sebagai Distributor**

Aktor : Bu Erma Peran : Distributor		
--	--	--

No.	Kode Kasus Pengujian	Terpenuhi
1	UJ-001	√
2	UJ-005	√
3	UJ-006	√
4	UJ-007	√
5	UJ-008	√

Berdasarkan hasil pengujian diatas, seluruh skenario sudah dapat terpenuhi. Evaluasi pengujian fungsionalitas yang dilaksanakan di sisi klien memiliki rincian sebagai berikut.

1. Seluruh kasus uji pada aktor sistem informasi distribusi daging sapi terpenuhi.
2. Seluruh kasus uji pada aktor *customer* telah terpenuhi.
3. Seluruh kasus uji pada aktor administrator telah terpenuhi.
4. Seluruh kasus uji pada aktor supplier telah terpenuhi.
5. Seluruh kasus uji pada aktor distributor telah terpenuhi.

## LAMPIRAN

**Kuisisioner Tugas Akhir**  
Rancang Bangun Aplikasi Sistem Informasi Manajemen Rantai Pasok  
Distribusi Daging Sapi Nasional

Nama : *Quji*  
Peran : *Grup Individu*

No	Kebutuhan Fungsional	Terpenuhi
1.	Sistem dapat melakukan proses pemesanan produk	✓
2.	Sistem dapat menampilkan balasan persetujuan transaksi	✓
3.	Sistem dapat menampilkan jumlah ketersediaan stok barang	✓
4.	Sistem dapat melakukan persetujuan data user	✓
5.	User dapat melakukan registrasi ke sistem	✓

Gambar A. 1 Evaluasi Pengujian Fungsionalitas Sebagai Customer

**Kuisisioner Tugas Akhir**  
Rancang Bangun Aplikasi Sistem Informasi Manajemen Rantai Pasok  
Distribusi Daging Sapi Nasional

Nama : *Erma Suryani*  
Peran : *Distributor*

No	Kebutuhan Fungsional	Terpenuhi
1.	Sistem dapat menampilkan list permintaan order	✓
2.	Dapat melakukan persetujuan transaksi	✓
3.	Dapat melakukan transaksi	✓
4.	Dapat melakukan upload dokumen	✓
5.	Dapat melihat stock barang yang tersedia	✓
6.	Dapat melakukan penambahan stock barang	✓
7.	Dapat melakukan update stock	✓

Gambar A. 2 Evaluasi Pengujian Fungsionalitas Sebagai Distributor

**Kuisisioner Tugas Akhir**

Rancang Bangun Aplikasi Sistem Informasi Manajemen Rantai Pasok  
Distribusi Daging Sapi Nasional

Nama : *Erma Suryani*  
Peran : *Supplier*

No	Kebutuhan Fungsional	Terpenuhi
1.	Sistem dapat menampilkan list permintaan order	✓
2.	Dapat melakukan persetujuan transaksi	✓
3.	Dapat melakukan penambahan stock barang	✓
4.	Dapat melakukan update stock	✓

**Gambar A. 3 Evaluasi Pengujian Fungsionalitas Sebagai Supplier**

**Kuisisioner Tugas Akhir**

Rancang Bangun Aplikasi Sistem Informasi Manajemen Rantai Pasok  
Distribusi Daging Sapi Nasional

Nama : *Erma Suryani*  
Peran : *Admin Web*

No	Kebutuhan Fungsional	Terpenuhi
1.	Sistem dapat menampilkan list user yang telah terregistrasi	✓
2.	Dapat melakukan persetujuan user	✓

**Gambar A. 4 Evaluasi Pengujian Fungsionalitas Sebagai Administrator**

**Lembar Feedback**

Nama : *Erma Suryani*  
Peran : *Supplier*

Pertanyaan

1. Desain antar muka aplikasi ini mudah untuk dimengerti dan membantu anda untuk menggunakan aplikasi ini.  
a. Sangat setuju  b. Setuju  c. Kurang Setuju  d. Tidak Setuju

2. Aplikasi memiliki respon yang cepat dalam hal pengambilan data dan pengiriman data.  
a. Sangat setuju  b. Setuju  c. Kurang Setuju  d. Tidak Setuju

**Gambar A. 5 Lembar *Feedback* Dari Sisi Administrator**



**Lembar Feedback**

Nama: Erma Suryani  
Peran: SupPLIER

**Pertanyaan**

1. Desain antar muka aplikasi ini mudah untuk dimengerti dan membantu anda untuk menggunakan aplikasi ini.  
b. Sangat setuju  b. Setuju c. Kurang Setuju d. Tidak Setuju
2. Aplikasi memiliki respon yang cepat dalam hal pengambilan data dan pengiriman data.  
b. Sangat setuju  b. Setuju c. Kurang Setuju d. Tidak Setuju

Gambar A. 6 Lembar *Feedback* Dari Sisi Supplier

**Lembar Feedback**

Nama: Erma Suryani  
Peran: Distributor

**Pertanyaan**

1. Desain antar muka aplikasi ini mudah untuk dimengerti dan membantu anda untuk menggunakan aplikasi ini.  
c. Sangat setuju  b. Setuju c. Kurang Setuju d. Tidak Setuju
2. Aplikasi memiliki respon yang cepat dalam hal pengambilan data dan pengiriman data.  
c. Sangat setuju  b. Setuju c. Kurang Setuju d. Tidak Setuju

Gambar A. 7 Lembar *Feedback* Dari Sisi Distributor

**Lembar Feedback**

Nama: Septi / qiqi  
Peran: Superindo

Pertanyaan

1. Aplikasi transaksi distribusi daging sapi dapat menyelesaikan permasalahan dengan cepat dibandingkan sistem sebelumnya.  
a. Sangat setuju  b. Setuju c. Kurang Setuju d. Tidak Setuju
2. Desain antar muka aplikasi ini mudah untuk dimengerti dan membantu anda untuk menggunakan aplikasi ini.  
a. Sangat setuju  b. Setuju c. Kurang Setuju d. Tidak Setuju
3. Aplikasi memiliki respon yang cepat dalam hal pengambilan data dan pengiriman data.  
a. Sangat setuju  b. Setuju c. Kurang Setuju d. Tidak Setuju

**Gambar A. 8 Lembar *Feedback* Dari Sisi Customer Yaitu Pihak Superindo**

## **BAB VI**

### **KESIMPULAN DAN SARAN**

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga terdapat saran yang ditujukan untuk pengembangan penelitian lebih lanjut.

#### **6.1 Kesimpulan**

Dari hasil uji coba yang telah dilakukan terhadap pembuatan sistem informasi akademik pada modul kurikulum, dapat diambil kesimpulan sebagai berikut:

1. Model bisnis dari sistem distribusi daging sapi merupakan sebuah sistem berbasis Manajemen Rantai Pasok yang telah disesuaikan dengan kebutuhan pengguna.
2. Pembagian aktor pada aplikasi dapat membantu aksesibilitas informasi ketersediaan stock komoditas bagi para *stakeholder* terlibat.
3. Aplikasi sistem informasi distribusi daging sapi sudah melalui analisis dan perancangan sistem dengan melihat kebutuhan fungsional dari proses bisnis distribusi daging sapi.
4. Aplikasi sistem informasi dapat melakukan transaksi berdasarkan jumlah stock yang tersedia. Pengguna dapat melakukan pemesanan produk berdasarkan jumlah stock yang tersedia serta pengguna dapat melihat status transaksi yang telah dilakukan.
5. Aplikasi sudah memenuhi semua kebutuhan fungsional dengan melakukan pengujian fungsionalitas terhadap perwakilan dari tiap pengguna sistem.

## 6.2 Saran

Saran yang diberikan untuk pengembangan aplikasi ini adalah:

1. Perlu dikembangkan sistem untuk melakukan pembayaran online sehingga transaksi dapat berjalan lebih baik.
2. Adanya fitur notifikasi untuk tiap status. Baik status pemesanan, status konfirmasi pengiriman maupun status persetujuan pengguna sehingga dapat memudahkan pengguna dalam menggunakan sistem ini.
3. Perlu dirancang sistem *Supply Chain Management* yang dapat mengintegrasikan distribusi daging sapi secara nasional.

## DAFTAR PUSTAKA

- [1] "Kementerian Pertanian," 2015. [Online]. Available: <http://www.pertanian.go.id/>. [Accessed 6 12 2015].
- [2] C. Dr. Zaroni, "Implementasi Pusat Distribusi Regional," 2015. [Online]. Available: <http://supplychainindonesia.com/new/implementasi-pusat-distribusi-regional/>. [Accessed 15 05 2016].
- [3] Setijadi, "Introduction to SCM," 2014. [Online]. Available: [http://supplychainindonesia.com/new/wp-content/files/1\\_Introduction\\_to\\_SCM\\_Man.\\_Rantai\\_Pasok.pdf](http://supplychainindonesia.com/new/wp-content/files/1_Introduction_to_SCM_Man._Rantai_Pasok.pdf). [Accessed 8 12 2015].
- [4] S. Erma, R. A. Hendrawan, M. Isnaini and L. P. Dewi, "DYNAMIC SIMULATION MODEL OF BEEF SUPPLY CHAIN TO FULFILL NATIONAL DEMAND," *Jurnal Teknologi UTM*, p. 373–381, 2016.
- [5] D. S. Wiyono, "Analisis dan Desain Aplikasi Web dan Mobile Supply Chain Management untuk Mendukung Distribusi Komoditas Padi Paska-Panen," vol. 8, pp. 1-10, 2009.
- [6] K. Pertanian, Rencana Strategis Kementerian Pertanian 2015-2019, Jakarta: Kementerian Pertanian, 2015.
- [7] H. Soeprapto and Z. Abidin, "Cara Tepat Penggemukan Sapi Potong," Jakarta, PT Agromedia Pustaka, 2006.
- [8] H. Min and G. Zhou, "Supply Chain Modeling: past, present and future," *Journal of Computers and Industrial Engineering*, 2002.
- [9] D. F. Ross, Introduction to e-Supply Chain Management, USA: St.Lucis Press, 2003.
- [10] "CodeIgniter," 2016. [Online]. Available: <https://en.wikipedia.org/wiki/CodeIgniter>. [Accessed 7 04 2016].

- [11] "PHP," 2016. [Online]. Available: <https://en.wikipedia.org/wiki/PHP>. [Accessed 7 4 2016].
- [12] "MySQL," 2016. [Online]. Available: <https://en.wikipedia.org/wiki/MySQL>. [Accessed 7 04 2016].
- [13] "Web Server," 2016. [Online]. Available: [https://id.wikipedia.org/wiki/Server\\_web](https://id.wikipedia.org/wiki/Server_web). [Accessed 7 04 2016].

## BIODATA PENULIS



Lidra Trifidya, lahir di Sungai Pakning, tanggal 29 Oktober 1994 (Riau). Penulis menempuh pendidikan mulai dari SD Negeri 1 Sungai Pakning (2000-2006), SMP Negeri 1 Bukit Batu (2006-2009), SMA Negeri 1 Bukit Batu (2009-2012) dan S1 Teknik Informatika Fakultas Teknologi Informasi Institut Teknologi Sepuluh Nopember, Surabaya (2012-2016).

Selama masa kuliah, penulis aktif dalam organisasi Himpunan Mahasiswa Teknik Computer (HMTTC) dan UKM PLH SIKLUS ITS. Diantaranya adalah menjadi staf Departemen Hubungan Luar 2013-2014 serta menjadi bendahara umum II PLH SIKLUS ITS. Penulis juga aktif dalam kepanitiaan schematics serta beberapa kepanitiaan yang diadakan oleh PLH SIKLUS ITS.

Selama kuliah di Teknik Informatika ITS, penulis mengambil bidang minat Rekayasa Perangkat Lunak (RPL) dengan ketertarikan penulis dibidang teknologi informasi. Penulis dapat dihubungi melalui alamat email [trifidya29@gmail.com](mailto:trifidya29@gmail.com).