



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KI141502

RANCANG BANGUN SISTEM *ENTERPRISE RESOURCE PLANNING* PADA MODUL *SALES AND DISTRIBUTION* YANG BERORIENTASI *MULTI-TENANCY* DENGAN BASIS DATA TERDISTRIBUSI

VICKY CHRYSIAN SUGIARTO
NRP 5112 100 006

Dosen Pembimbing I
Dwi Sunaryono, S.Kom, M.Kom.

Dosen Pembimbing II
Prof. Drs. Ec. Ir. Rivanarto Sarno, M.Sc., Ph.D.

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

[Halaman ini sengaja dikosongkan]



TUGAS AKHIR - KI141502

RANCANG BANGUN SISTEM *ENTERPRISE RESOURCE PLANNING* PADA MODUL *SALES AND DISTRIBUTION* YANG BERORIENTASI *MULTI-TENANCY* DENGAN BASIS DATA TERDISTRIBUSI

VICKY CHRYSTIAN SUGIARTO
NRP 5112 100 006

Dosen Pembimbing I
Dwi Sunaryono, S.Kom, M.Kom.

Dosen Pembimbing II
Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc., Ph.D.

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

[Halaman ini sengaja dikosongkan]



FINAL PROJECT - KI141502

**DESIGN AND IMPLEMENTATION OF
ENTERPRISE RESOURCE PLANNING IN SALES
AND DISTRIBUTION WITH MULTI-TENANCY
ORIENTED USING DISTRIBUTED DATABASE**

VICKY CHRYSIAN SUGIARTO
NRP 5112 100 006

Supervisor I
Dwi Sunaryono, S.Kom, M.Kom.

Supervisor II
Prof. Drs. Ec. Ir. Rivanarto Sarno, M.Sc., Ph.D.

DEPARTMENT OF INFORMATICS
Faculty of Information Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2016

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

RANCANG BANGUN SISTEM *ENTERPRISE RESOURCE PLANNING* PADA MODUL *SALES AND DISTRIBUTION* YANG BERORIENTASI *MULTI-TENANCY* DENGAN BASIS DATA TERDISTRIBUSI

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Rumpun Mata Kuliah Manajemen Informasi
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh

VICKY CHRYSIAN SUGIARTO

NRP. 5112 100 006

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Dwi Sunaryono, S.Kom., M.Kom.
NIP: 19720528 199702 1 001

Prof. Drs. Ec. Ir. Riyanarto Sarno, M.Sc.,
Ph.D.

NIP: 19590803 198601 1 001



(pembimbing 1)

(pembimbing 2)

**SURABAYA
JUNI, 2016**

[Halaman ini sengaja dikosongkan]

RANCANG BANGUN SISTEM *ENTERPRISE RESOURCE PLANNING* PADA MODUL SALES AND DISTRIBUTION YANG BERORIENTASI MULTI-TENANCY DENGAN BASIS DATA TERDISTRIBUSI

Nama : Vicky Chrystian Sugiarto
NRP : 5112100006
Jurusan : Teknik Informatika – FTIf ITS
Dosen Pembimbing I : Dwi Sunaryono, S.Kom., M.Kom.
Dosen Pembimbing II : Prof. Drs. Ec. Ir. Rivanarto Sarno, M.Sc., Ph.D.

ABSTRAK

Enterprise Resource Planning adalah sebuah sistem yang digunakan untuk mengelola sebuah perusahaan besar yang dirancang untuk mengkoordinasikan semua sumber daya, informasi dan aktifitas yang diperlukan untuk proses bisnis lengkap. Salah satu modul yang terpenting dan yang telah dikerjakan dalam tugas akhir ini, yaitu *Sales and Distribution*. *Sales and distribution* adalah modul yang menangani penjualan dan pengiriman barang ke pelanggan untuk mencapai tujuan bisnisnya. Penggunaan modul *sales and distribution* ditujukan agar mempermudah proses penjualan kepada pelanggan sesuai dengan minat pelanggan terhadap barang dan jasa, mempermudah pengecekan penjualan barang dan pengiriman barang, serta mempermudah pendataan pelanggan yang melakukan pembelian barang atau jasa yang disediakan oleh organisasi dan menentukan pelayanan yang tepat untuk pelanggan-pelanggan tertentu.

Dalam tugas akhir ini, pada modul *sales and distribution* diusulkan metode untuk melakukan prakiraan data penjualan untuk perusahaan tiap bulannya dengan menggunakan 2 (dua) metode, yaitu *Metode Holt-Winters additive* dan *Metode Holt-Winters multiplicative* yang menggunakan *exponential smoothing*.

Hasil eksperimen menunjukkan bahwa sistem ERP yang dirancang menggunakan basis data terdistribusi dapat tetap berjalan dengan baik ketika salah satu server lain mati. Pada sales and distribution, dengan adanya prakiraan data penjualan, dapat meminimalisir jumlah permintaan barang atau jasa yang diminta oleh pelanggan tanpa harus melakukan produksi secara mendadak. Serta monitoring terhadap transaksi yang dilakukan oleh pelanggan, sehingga mudah untuk mengetahui bagaimana pelayanan yang harus diberikan kepada pelanggan-pelanggan yang ada.

Kata kunci: Distributed Database, Enterprise Resource Planning (ERP), Sales and Distribution, Multitenancy, Peramalan.

DESIGN AND IMPLEMENTATION OF ENTERPRISE RESOURCE PLANNING IN SALES AND DISTRIBUTION WITH MULTI-TENANCY ORIENTED USING DISTRIBUTED DATABASE

Student Name : Vicky Chrystian Sugiarto
NRP : 5112100006
Major : Teknik Informatika – FTIf ITS
Supervisor I : Dwi Sunaryono, S.Kom., M.Kom.
Supervisor II : Prof. Drs. Ec. Ir. Rivanarto Sarno, M.Sc., Ph.D.

ABSTRACT

Enterprise Resource Planning is a system used to manage a large enterprise that is designed to coordinate all the resources, information, and activities needed to complete business processes. One of the most important modules, and that has been done in this final project, namely Sales and Distribution. Sales and distribution is a module that handles the sale and delivery of goods to customers to achieve their business objectives. The use of sales and distribution module is intended to simplify the process of selling to customers in accordance with the interests of customers for goods and services, makes it easy to check the sale of goods and delivery of goods, and facilitate the collection of customers who make a purchase of goods or services provided by the organization and determines the appropriate services to customers.

In this final project, sales and distribution module is proposed a method for forecasting the sales data for the company each month by using 2 (two) methods, Holt-Winters additive method and Holt-Winters multiplicative method which uses exponential smoothing.

The experimental results showed that ERP systems are designed using a distributed database can still run well when one of the other servers fail. In sales and distribution, with forecast sales data, can minimize the amount of demand for goods or

services requested by customers without having to make a sudden production. As well as monitoring of transactions made by customers, so it's easy to see how the service should be provided to existing customers.

Keywords: Distributed Database, Enterprise Resource Planning (ERP), Sales and Distribution, Multitenancy, Forecast.

KATA PENGANTAR

Puji syukur kepada Tuhan Yesus Kristus, karena hanya atas berkat, kasih dan penyertaan-Nya penulis dapat menyelesaikan tugas akhir yang berjudul “Rancang Bangun Sistem *Enterprise Resource Planning* pada Modul *Sales and Distribution* yang Berorientasi *Multi-Tenancy* dengan Basis Data Terdistribusi” ini dapat selesai sesuai dengan waktu yang telah ditentukan.

Pengerjaan tugas akhir ini menjadi sebuah sarana untuk penulis memperdalam ilmu yang telah didapatkan selama menempuh pendidikan di kampus perjuangan Institut Teknologi Sepuluh Nopember Surabaya, khususnya dalam disiplin ilmu Teknik Informatika. terselesaikannya buku tugas akhir ini tidak terlepas dari bantuan dan dukungan semua pihak. Pada kesempatan kali ini penulis ingin mengucapkan terima kasih kepada:

1. Bapak Riyanarto Sarno dan Bapak Dwi Sunaryono selaku dosen pembimbing yang telah bersedia meluangkan waktu untuk memberikan petunjuk selama proses pengerjaan tugas akhir ini.
2. Papa, mama, kakak, adik dan keluarga yang selalu memberikan dukungan penuh untuk menyelesaikan tugas akhir ini.
3. Richard, Andrean, Prasetyo, William, Ardi, Anton, Ryan, dan Felix yang selalu memberikan bantuan dan dukungan selama menempuh perkuliahan.
4. Tim ERP 2016 Al, Akbar, Arya, Fina, Melinda, Yusi, Nicko, dan Hari.
5. Bapak dan Ibu dosen Jurusan Teknik Informatika ITS yang telah banyak memberikan ilmu dan bimbingan yang tak ternilai harganya bagi penulis.
6. Teman-teman Angkatan 2012 yang selalu mendukung, menyemangati dan membantu selama proses pengerjaan tugas akhir.
7. Serta semua pihak yang turut membantu penulis dalam menyelesaikan tugas akhir ini.

Pada akhirnya, penulis menyadari ketidaksempurnaan buku tugas akhir ini sehingga saran dan kritik yang bersifat membangun sangat penulis harapkan untuk menjadi pelajaran pada kesempatan yang akan datang. Penulis juga hendak meminta maaf apabila terdapat kesalahan yang penulis lakukan secara sengaja maupun tidak sengaja sepanjang pengerjaan tugas akhir ini. Akhir kata, penulis mengucapkan terima kasih kepada semua pihak atas dukungannya, dan semoga tugas akhir yang telah terselesaikan ini dapat bermanfaat bagi masyarakat, bangsa, dan negara.

Surabaya, Juni 2016

Vicky Chrystian Sugiarto

DAFTAR ISI

LEMBAR PENGESAHAN.....	v
ABSTRAK	vii
<i>ABSTRACT</i>	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxiii
DAFTAR KODE SUMBER	xxv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Permasalahan	3
1.3 Batasan Permasalahan	4
1.4 Tujuan	4
1.5 Manfaat	5
1.6 Metodologi	5
1.7 Sistematika Penulisan	6
BAB II DASAR TEORI.....	9
2.1 Penelitian Terkait.....	9
2.2 <i>Enterprise Resource Planning (ERP)</i>	10
2.3 <i>Sales and Distribution</i>	11
2.3.1 <i>Inquiry</i>	13
2.3.2 <i>Quotation</i>	13
2.3.3 <i>Sales Order</i>	13
2.3.4 <i>Return Order</i>	14
2.3.5 <i>Shipping and Delivery</i>	14
2.3.6 <i>Forecasting</i>	14
2.4 Basis Data Terdistribusi	15
2.4.1 Replikasi	15

2.4.2	Fragmentasi.....	15
2.5	<i>Multitenancy</i>	15
2.5.1	<i>Separated Database, Separated Schema</i>	16
2.5.2	<i>Shared Database, Separated Schema</i>	16
2.5.3	<i>Shared Database, Shared Schema</i>	17
2.6	<i>Role Based Access Control (RBAC)</i>	18
2.7	<i>Metode Holt-Winters</i>	18
2.7.1	<i>Metode Holt-Winters Additive</i>	19
2.7.2	<i>Metode Holt-Winters Multiplicative</i>	20
2.8	<i>Hypertext Preprocessor (PHP)</i>	21
2.9	<i>Kerangka Kerja Yii 2.0</i>	22
2.10	<i>MySQL Cluster</i>	22
 BAB III ANALISIS DAN PERANCANGAN SISTEM.....		23
3.1	<i>Analisis</i>	23
3.1.1	<i>Analisis Proses Bisnis</i>	23
3.1.2	<i>Analisis Data</i>	31
3.1.3	<i>Analisis Peramalan Penjualan</i>	31
3.2	<i>Deskripsi Umum Sistem</i>	33
3.3	<i>Identifikasi Pengguna</i>	34
3.4	<i>Spesifikasi Kebutuhan Perangkat Lunak</i>	34
3.5	<i>Kasus Penggunaan</i>	36
3.5.1	<i>Kasus Penggunaan Mengelola Rute Pengiriman</i>	38
3.5.2	<i>Kasus Penggunaan Mengelola Narahubung</i>	41
3.5.3	<i>Kasus Penggunaan Mengelola Pelanggan</i>	43
3.5.4	<i>Kasus Penggunaan Mengelola Inquiry</i>	45
3.5.5	<i>Kasus Penggunaan Mengelola Quotation</i>	47
3.5.6	<i>Kasus Penggunaan Mengelola Sales Order</i>	50
3.5.7	<i>Kasus Penggunaan Mengelola Return Order</i>	53
3.5.8	<i>Kasus Penggunaan Mengelola Pengiriman</i>	56
3.5.9	<i>Kasus Penggunaan Mengelola Peramalan penjualan</i>	59

3.5.10 Kasus Penggunaan Melihat Rute Pengiriman.....	61
3.5.11 Kasus Penggunaan Melihat Narahubung	62
3.5.12 Kasus Penggunaan Melihat Pelanggan.....	63
3.5.13 Kasus Penggunaan Melihat <i>Inquiry</i>	64
3.5.14 Kasus Penggunaan Melihat <i>Quotation</i>	65
3.5.15 Kasus Penggunaan Melihat <i>Sales order</i>	66
3.5.16 Kasus Penggunaan Melihat <i>Return order</i>	67
3.5.17 Kasus Penggunaan Melihat Pengiriman	68
3.5.18 Kasus Penggunaan Melihat Peramalan penjualan	69
3.5.19 Kasus Penggunaan Melihat Laporan Penjualan	70
3.5.20 Kasus Penggunaan Melihat Laporan Pelanggan.....	71
3.6 Perancangan	72
3.6.1 Perancangan <i>Multitenancy</i>	72
3.6.2 Perancangan Basis Data	73
3.6.3 Perancangan <i>Role Based Access Control (RBAC)</i>	78
3.6.4 Perancangan Antarmuka Sistem	79
BAB IV IMPLEMENTASI SISTEM.....	83
4.1 Lingkungan Pengembangan Sistem	83
4.2 Implementasi Basis Data Terdistribusi	83
4.2.1 Instalasi Data <i>SQL Node</i> pada <i>Node 1</i> dan <i>Node 2</i>	83
4.2.2 Pemasangan <i>Node</i> Manajemen Pada <i>Node 3</i>	85
4.2.3 Konfigurasi Manajemen <i>Node</i>	86
4.2.4 Konfigurasi Data dan <i>SQL Node</i>	86
4.2.5 Memulai <i>MySQL Cluster</i>	87
4.3 Implementasi <i>Role Base Access Control (RBAC)</i>	88
4.3.1 Membuat Tabel Pengguna	88
4.3.2 Membuat Tabel Autentifikasi RBAC dan Tabel Pengguna.....	88
4.3.3 Membuat Modul <i>Admin</i> dan Konfigurasi Autentifikasi	89

4.3.4	Membuat Model Tabel Autentifikasi, <i>Controller</i> dan <i>View</i> Pengguna	89
4.3.5	Menambahkan Kode pada <i>Usercontroller</i>	90
4.4	Implementasi <i>Multitenancy</i>	91
4.4.1	Membuat Halaman <i>Tenant</i>	92
4.4.2	Menambahkan Basis Data untuk <i>Tenant</i> Baru.....	92
4.4.3	<i>Login Tenant</i>	93
4.5	Implementasi Program pada Modul <i>Sales and Distribution</i>	93
4.5.1	Halaman Utama Modul	94
4.5.2	Melihat Daftar Data Submodul	94
4.5.3	Melihat Detail Data Submodul.....	94
4.5.4	Menambah dan Menyunting Data Submodul.....	94
4.5.5	Menghapus Data Submodul.....	95
4.5.6	Menambah <i>Forecast</i>	95
4.6	Implementasi Antarmuka Pengguna <i>Sales and Distribution</i>	95
4.6.1	Antarmuka Halaman Utama Modul <i>Sales and</i> <i>Distribution</i>	95
4.6.2	Antarmuka Melihat Daftar <i>Shipping</i>	96
4.6.3	Antarmuka Melihat Daftar <i>Contact Person</i>	97
4.6.4	Antarmuka Melihat Daftar <i>Customer</i>	97
4.6.5	Antarmuka Melihat Daftar <i>Inquiry</i>	97
4.6.6	Antarmuka Melihat Daftar <i>Quotation</i>	98
4.6.7	Antarmuka Melihat Daftar <i>Sales Order</i>	98
4.6.8	Antarmuka Melihat Daftar <i>Return Order</i>	99
4.6.9	Antarmuka Melihat Daftar <i>Shipping and Delivery</i>	99
4.6.10	Antarmuka Melihat Daftar <i>Forecasting</i>	100
BAB V PENGUJIAN DAN EVALUASI		101
5.1	Lingkungan Pengujian	101
5.2	Skenario Pengujian	101
5.2.1	Pengujian Perencanaan Bisnis.....	102

5.2.2	Pengujian Fungsionalitas Sistem	106
5.2.3	Pengujian Fungsional <i>Role Based Access Control</i> (RBAC)	128
5.2.4	Pengujian Fungsional <i>Multitenancy</i>	132
5.2.5	Pengujian Fungsionalitas Basis Data Terdistribusi	133
5.2.6	Pengujian Peramalan Penjualan	138
5.3	Evaluasi Pengujian	139
5.3.1	Evaluasi Pengujian Fungsionalitas Sistem	140
BAB VI KESIMPULAN DAN SARAN		143
6.1	Kesimpulan	143
6.2	Saran	144
DAFTAR PUSTAKA		145
A.	LAMPIRAN A – PROSES BISNIS	147
A.1	Proses Bisnis Odoo	147
A.2	Proses Bisnis InoERP	148
A.3	Proses Bisnis Adempiere	149
A.4	Proses Bisnis <i>Make-To-Order</i> ERP 2016	150
A.5	Proses Bisnis <i>Make-To-Stock</i> ERP 2016	151
B.	LAMPIRAN B – DATA PERAMALAN PENJUALAN .	153
B.1	Data Masukkan Peramalan Penjualan	153
C.	LAMPIRAN C – KODE SUMBER	155
C.1	Kode Sumber Halaman Utama Modul	155
C.2	Kode Sumber Melihat Daftar Data Submodul	156
C.3	Kode Sumber Melihat Detail Data Submodul	156
C.4	Kode Sumber Menambah dan Menyunting Data Submodul	157
C.5	Kode Sumber Menghapus Data Submodul	158
C.6	Kode Sumber Menambah <i>Forecast</i>	159
D.	LAMPIRAN D – HASIL UJI PERAMALAN	163
D.1	Hasil Uji Peramalan Metode <i>Holt-Winters Additive</i> ..	163

D.2 Hasil Uji Peramalan Metode <i>Holt-Winters</i>	
<i>Multiplicative</i>	164
BIODATA PENULIS.....	165

DAFTAR GAMBAR

Gambar 2.1. Diagram Integrasi ERP Secara Umum	12
Gambar 2.2. Arsitektur Data <i>Separated Database</i>	16
Gambar 2.3. Arsitektur Data <i>Shared Database, Separated Schema</i>	17
Gambar 2.4. Arsitektur Data <i>Shared Database, Shared Schema</i>	17
Gambar 3.1. Proses Bisnis ERP <i>Level 0</i>	25
Gambar 3.2. Proses Bisnis <i>Level 1 Make-To-Order</i>	25
Gambar 3.3. Proses Bisnis <i>Level 1 Make-To-Stock</i>	25
Gambar 3.4. Proses Bisnis <i>Level 2 Make-To-Order</i>	26
Gambar 3.5. Proses Bisnis <i>Level 2 Make-To-Order</i>	27
Gambar 3.6. Proses Bisnis <i>Level 3 Make-To-Order</i>	29
Gambar 3.7. Proses Bisnis <i>Level 3 Make-To-Stock</i>	30
Gambar 3.8. Diagram Alur Proses Peramalan Penjualan	32
Gambar 3.9. Rancang Basis Data Terdistribusi.....	33
Gambar 3.10. Diagram Kasus Penggunaan	37
Gambar 3.11. Diagram Aktifitas Mengelola Rute Pengiriman ...	40
Gambar 3.12. Diagram Aktifitas Mengelola Narahubung	42
Gambar 3.13. Diagram Aktifitas Mengelola Pelanggan	44
Gambar 3.14. Diagram Aktifitas Mengelola <i>Inquiry</i>	46
Gambar 3.15. Diagram Aktifitas Mengelola <i>Quotation</i>	49
Gambar 3.16. Diagram Aktifitas Mengelola <i>Sales Order</i>	52
Gambar 3.17. Diagram Aktifitas Mengelola <i>Return Order</i>	55
Gambar 3.18. Diagram Aktifitas Mengelola Pengiriman.....	58
Gambar 3.19. Diagram Aktifitas Mengelola Peramalan Penjualan	60
Gambar 3.20. Diagram Aktifitas Melihat Rute Pengiriman	61
Gambar 3.21. Diagram Aktifitas Melihat Narahubung	62
Gambar 3.22. Diagram Aktifitas Melihat Pelanggan	63
Gambar 3.23. Diagram Aktifitas Melihat <i>Inquiry</i>	64
Gambar 3.24. Diagram Aktifitas Melihat <i>Quotation</i>	65
Gambar 3.25. Diagram Aktifitas Melihat <i>Sales Order</i>	66
Gambar 3.26. Diagram Aktifitas Melihat <i>Return Order</i>	67
Gambar 3.27. Diagram Aktifitas Melihat Pengiriman	68

Gambar 3.28. Diagram Aktifitas Melihat Peramalan Penjualan	69
Gambar 3.29. Diagram Aktifitas Melihat Laporan Penjualan	70
Gambar 3.30. Diagram Aktifitas Melihat Laporan Pelanggan	71
Gambar 3.31. Perancangan <i>Multitenancy</i>	72
Gambar 3.32. <i>Physical Data Model</i> Modul <i>Sales and Distribution</i>	75
Gambar 3.33. Rancang Antarmuka <i>Login</i>	78
Gambar 3.34. Rancang Antarmuka Menambahkan Pengguna Baru	79
Gambar 3.35. Rancang Antarmuka Halaman Utama Modul	80
Gambar 3.36. Rancang Antarmuka Daftar Data Submodul	80
Gambar 3.37. Rancang Antarmuka Detail Data Submodul	81
Gambar 3.38. Rancang Antarmuka Menambah dan Menyunting Data Submodul	82
Gambar 4.1. Antarmuka Halaman Utama Modul <i>Sales and Distribution</i>	96
Gambar 4.2. Antarmuka Melihat Daftar <i>Shipping</i>	96
Gambar 4.3. Antarmuka Melihat Daftar <i>Contact Person</i>	97
Gambar 4.4. Antarmuka Melihat Daftar <i>Customer</i>	97
Gambar 4.5. Antarmuka Melihat Daftar <i>Inquiry</i>	98
Gambar 4.6. Antarmuka Melihat Daftar <i>Quotation</i>	98
Gambar 4.7. Antarmuka Melihat Daftar <i>Sales Order</i>	99
Gambar 4.8. Antarmuka Melihat Daftar <i>Return Order</i>	99
Gambar 4.9. Antarmuka Melihat Daftar <i>Shipping and Delivery</i>	99
Gambar 4.10. Antarmuka Melihat Daftar <i>Forecasting</i>	100
Gambar 5.1. Proses <i>Login User Admin</i>	130
Gambar 5.2. Tampilan Awal Setelah <i>Login</i> Berhasil Dilakukan	130
Gambar 5.3. <i>Admin</i> Menambahkan <i>User</i> Baru	131
Gambar 5.4. Proses <i>User Admin</i> Menyunting Data <i>User</i>	131
Gambar 5.5. Proses <i>User Admin</i> Menghapus Data <i>User</i>	132
Gambar 5.6. Pengujian <i>Multitenancy</i> Berada di Halaman <i>Login</i>	133
Gambar 5.7. Hasil Pengujian <i>Multitenancy</i>	133
Gambar 5.8. Pengujian Fitur Replikasi pada Sistem	134

Gambar 5.9. Pengujian Replikasi pada Database Server1	135
Gambar 5.10. Pengujian Fitur Replikasi pada Database Server2	135
Gambar 5.11. Pengujian Fitur <i>High-Availability</i>	136
Gambar 5.12. Pengecekan <i>Availability</i> pada <i>Node Server</i>	136
Gambar 5.13. Pengecekan Basis Data pada Node1	137
Gambar 5.14. Pengecekan <i>Availability</i> pada <i>Node Server</i>	137
Gambar 5.15. Mematikan Basis Data <i>Server</i> pada Node2	138
Gambar 5.16. Hasil Uji Coba Menggunakan Metode <i>Holt-Winters Additive</i>	139
Gambar 5.17. Hasil Uji Coba Menggunakan Metode <i>Holt-Winters Multiplicative</i>	139
Gambar A.1. Proses Bisnis Odoo	147
Gambar A.2. Proses Bisnis InoERP	148
Gambar A.3. Proses Bisnis Adempiere	149
Gambar A.4. Proses Bisnis <i>Make-To-Order</i> ERP 2016	150
Gambar A.5. Proses Bisnis <i>Make-To-Stock</i> ERP 2016	151

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 2.1. Perbedaan ERP 2013 dan ERP 2016 pada Modul <i>Sales and Distribution</i>	9
Tabel 3.1. Perbandingan Proses Bisnis ERP 2016, Odoo, InoERP, dan Adempiere pada modul <i>Sales and Distribution</i>	23
Tabel 3.2. Deskripsi Proses Bisnis <i>Level 1</i>	26
Tabel 3.3. Deskripsi Proses Bisnis <i>Level 2</i>	27
Tabel 3.4. Daftar Kebutuhan Fungsional Sistem.....	34
Tabel 3.5. Keterangan Kode Kasus Penggunaan.....	37
Tabel 3.6. Kasus Penggunaan Mengelola Rute Pengiriman.....	38
Tabel 3.7. Kasus Penggunaan Mengelola Narahubung.....	41
Tabel 3.8. Kasus Penggunaan Mengelola Pelanggan	43
Tabel 3.9. Kasus Penggunaan Mengelola <i>Inquiry</i>	45
Tabel 3.10. Kasus Penggunaan Mengelola <i>Quotation</i>	47
Tabel 3.11. Kasus Penggunaan Mengelola <i>Sales Order</i>	50
Tabel 3.12. Kasus Penggunaan Mengelola <i>Return Order</i>	53
Tabel 3.13. Kasus Penggunaan Mengelola Pengiriman	56
Tabel 3.14. Kasus Penggunaan Mengelola Peramalan Penjualan	59
Tabel 3.15. Kasus Penggunaan Melihat Rute Pengiriman	61
Tabel 3.16. Kasus Penggunaan Melihat Narahubung	62
Tabel 3.17. Kasus Penggunaan Melihat Pelanggan.....	63
Tabel 3.18. Kasus Penggunaan Melihat <i>Inquiry</i>	64
Tabel 3.19. Kasus Penggunaan Melihat <i>Quotation</i>	65
Tabel 3.20. Kasus Penggunaan Melihat <i>Sales Order</i>	66
Tabel 3.21. Kasus Penggunaan Melihat <i>Return Order</i>	67
Tabel 3.22. Kasus Penggunaan Melihat Pengiriman.....	68
Tabel 3.23. Kasus Penggunaan Melihat Peramalan Penjualan....	69
Tabel 3.24. Kasus Penggunaan Melihat Laporan Penjualan	70
Tabel 3.25. Kasus Penggunaan Melihat Laporan Pelanggan	71
Tabel 3.26. Keterangan PDM.....	73
Tabel 5.1. Daftar Barang Jadi.....	102
Tabel 5.2. Daftar Aset	102
Tabel 5.3. Daftar Bahan Baku	103

Tabel 5.4. Daftar Material	104
Tabel 5.5. Perencanaan Keuangan Perusahaan.....	105
Tabel 5.6. Pengujian Mengelola Rute Pengiriman	106
Tabel 5.7. Pengujian Mengelola Narahubung	108
Tabel 5.8. Pengujian Mengelola Pelanggan	109
Tabel 5.9. Pengujian Mengelola <i>Inquiry</i>	111
Tabel 5.10. Pengujian Mengelola <i>Quotation</i>	112
Tabel 5.11. Pengujian Mengelola <i>Sales Order</i>	114
Tabel 5.12. Pengujian Mengelola <i>Return Order</i>	115
Tabel 5.13. Pengujian Mengelola Pengiriman.....	117
Tabel 5.14. Pengujian Mengelola Peramalan penjualan.....	118
Tabel 5.15. Pengujian Melihat Rute Pengiriman.....	120
Tabel 5.16. Pengujian Melihat Narahubung.....	121
Tabel 5.17. Pengujian Melihat Pelanggan	121
Tabel 5.18. Pengujian Melihat <i>Inquiry</i>	122
Tabel 5.19. Pengujian Melihat <i>Quotation</i>	123
Tabel 5.20. Pengujian Melihat <i>Sales Order</i>	123
Tabel 5.21. Pengujian Melihat <i>Return Order</i>	124
Tabel 5.22. Pengujian Melihat Pengiriman	125
Tabel 5.23. Pengujian Melihat Peramalan penjualan	126
Tabel 5.24. Pengujian Melihat Laporan Penjualan.....	126
Tabel 5.25. Pengujian Mengelola Laporan Pelanggan	127
Tabel 5.26. Pengujian Fitur <i>Role Based Access Control</i> (RBAC)	128
Tabel 5.27. Pengujian Fitur <i>Multitenancy</i>	132
Tabel 5.28. Evaluasi Pengujian Fungsionalitas Sistem	140
Tabel B.1. Data Latih	153
Tabel B.2. Data Uji.....	154
Tabel D.1. Hasil Uji Metode <i>Holt-Winter Additive</i>	163
Tabel D.2. Hasil Uji Metode Holt-Winters Multiplicative.....	164

DAFTAR KODE SUMBER

Kode Sumber 4.1. Membuat Grup <i>Mysql</i> Pengguna Baru dan Menambah <i>User</i> Baru	84
Kode Sumber 4.2. Mengubah Lokasi Direktori, Mengubah Arsip, dan Menciptakan <i>Symlink</i>	84
Kode Sumber 4.3. Mengubah Lokasi Direktori	84
Kode Sumber 4.4. Mengatur Perizinan <i>Server Mysql</i>	84
Kode Sumber 4.5. Menyalin <i>Script Startup Mysql</i>	85
Kode Sumber 4.6. Mengubah Lokasi Direktori	85
Kode Sumber 4.7. Mengubah Lokasi Direktori	85
Kode Sumber 4.8. Membuat Direktori	86
Kode Sumber 4.9. Mengatur <i>File</i> “config.ini”	86
Kode Sumber 4.10. Data dan <i>SQL Node</i>	87
Kode Sumber 4.11. Memulai Proses Manajemen <i>Node</i>	87
Kode Sumber 4.12. Memulai Proses <i>Ndbd</i> dan <i>Mysql Cluster</i> ...	87
Kode Sumber 4.13. Mengaktifkan <i>MySQL</i>	87
Kode Sumber 4.14. <i>Generate</i> Tabel Autentifikasi	88
Kode Sumber 4.15. Konfigurasi Autentifikasi.....	89
Kode Sumber 4.16. Menambahkan Data Ke Tabel <i>Auth_Item</i> ...	90
Kode Sumber 4.17. Menambahkan Data ke Tabel <i>Auth_Item_Child</i>	91
Kode Sumber 4.18. Menambahkan Data ke Tabel <i>Auth_Assignment</i>	91
Kode Sumber 4.19. Pembuatan Halaman Depan <i>Tenant</i>	92
Kode Sumber 4.20. Penambahan Basis Data untuk <i>Tenant</i> Baru	92
Kode Sumber 4.21. <i>Login Tenant</i>	93
Kode Sumber C.1. Halaman Utama Modul.....	155
Kode Sumber C.2. Melihat Daftar Data Submodul.....	156
Kode Sumber C.3. Melihat Detail Data Submodul	157
Kode Sumber C.4. Menambah dan Menyunting Data Submodul	158
Kode Sumber C.5. Menghapus Data Submodul.....	158
Kode Sumber C.6. Menambah Forecast.....	161

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Pada bab ini akan dibahas mengenai garis besar tugas akhir yang meliputi latar belakang, rumusan dan batasan permasalahan, tujuan dan manfaat, metodologi pengerjaan tugas akhir, dan sistematika penulisan tugas akhir.

1.1 Latar Belakang

Kondisi dunia bisnis saat ini telah berkembang menjadi semakin kompleks, semakin kompetitif, bergerak dengan cepat serta semakin sulit untuk diprediksi [1]. Agar dapat bersaing dan sukses dalam persaingan bisnis, perusahaan perlu memadukan bisnis dan sumber daya teknologi informasi yang dimiliki agar dapat mengakomodasi adanya perubahan untuk kemudian dilakukan adaptasi terhadap perubahan tersebut secara cepat dan tepat. Dengan adanya permasalahan tersebut, disimpulkan bahwa kebutuhan akan aplikasi *Enterprise Resource Planning* (ERP) dalam perusahaan merupakan kebutuhan yang perlu agar perusahaan dapat memadukan bisnis dan sumber daya. Pada ERP terdapat banyak sekali modul-modul yang dibuat dan diintegrasikan, antara lain adalah *Finance/Accounting, Sales and Distribution, Inventory, Production and Planning, Human Resource Management*, dan modul-modul lainnya. Pada tugas akhir ini, akan diimplementasikan modul aplikasi *Sales and Distribution*.

Penjualan *menunjukkan* adanya pertukaran dari suatu barang atau jasa yang dimiliki oleh sebuah organisasi atau perusahaan tertentu untuk ditukarkan dengan uang atau dengan barang lain yang memiliki nilai sejenis. Dalam penjualan, sebuah organisasi atau perusahaan dapat mencapai tujuan bisnisnya. Dengan demikian, kemampuan sebuah organisasi atau perusahaan untuk mengelola penjualan merupakan kegiatan yang sangat penting [1]. Selain harus memiliki fungsi yang kuat dalam penjualan, sebuah organisasi atau perusahaan juga harus memiliki

saluran pendistribusian yang efisien untuk membuat produk yang selalu tersedia ke pelanggan. Untuk dapat mencapai fungsi penjualan yang efisien, diperlukan adanya implementasi sistem informasi yang canggih [1].

Sales and distribution adalah fungsi dalam sebuah organisasi atau perusahaan untuk mengatur masalah pelayanan pelanggan, penjualan barang, dan pendistribusian barang ke tangan pelanggan agar tujuan bisnis perusahaan dapat tercapai. *Sales and Distribution* juga membantu perusahaan untuk menyelesaikan masalah yang terjadi di dalam lingkup penjualan dan pelanggan [2]. Apabila dalam sebuah perusahaan tidak memiliki manajemen penjualan dan pendistribusian, maka perusahaan tersebut akan sulit sekali berkembang dan tidak dapat menyelesaikan masalah yang terjadi dalam hal penjualan dan pendistribusian barang ke tangan pelanggan.

Masalah yang dihadapi oleh *Sales and Distribution* adalah bagaimana cara melayani pelanggan dan meyakinkan pelanggan untuk melakukan transaksi dengan perusahaan [2]. *Sales and Distribution* mengatasi masalah dengan cara memberikan submodul-submodul, seperti: *inquiry*, *quotation*, dan *customer*. Tujuan dari submodul-submodul ini adalah untuk mencapai tujuan bisnis dari perusahaan tersebut. *inquiry* adalah submodul yang digunakan untuk mencatat daftar permintaan barang yang ditanyakan oleh pelanggan ke perusahaan, pencatatan daftar permintaan barang digunakan untuk menghitung kemungkinan pelanggan untuk melakukan transaksi dengan perusahaan. *Quotation* adalah submodul yang digunakan untuk memberikan harga barang yang dipesan oleh pelanggan, harga yang diberikan termasuk harga diskon dan pajak. *Quotation* berasal dari *inquiry* atau dapat dibuat secara langsung tanpa *inquiry*. *Customer* adalah submodul yang digunakan untuk membedakan masing-masing pelanggan yang melakukan transaksi dengan perusahaan, dalam melakukan transaksi terdapat pelanggan yang kemungkinan hanya melakukan pembelian sekali saja, pelanggan yang sering melakukan transaksi, pelanggan yang masuk ke dalam daftar

hitam, dan jenis-jenis pelanggan lainnya. Dengan melakukan penggolongan terhadap pelanggan, diharapkan perusahaan dapat melakukan pelayanan yang efisien, sehingga pelanggan tetap melakukan transaksi dengan perusahaan.

Pada pengembangan sebelumnya, telah dibuat sebuah ERP, yaitu ERP 2013. ERP 2013, menggunakan *Service Oriented Architecture* (SOA) dengan sistem *Model-View-Controller* (MVC) dan *workflow* untuk .NET. Pada aplikasi tersebut telah terdapat modul *Sales and Distribution*, akan tetapi tidak terdapat kemampuan untuk melakukan perencanaan untuk melakukan penjualan kedepan. Sehingga pada ERP 2016 ditambahkan fitur untuk melakukan perencanaan, yaitu peramalan penjualan. Dengan adanya peramalan penjualan, jumlah permintaan barang yang diminta oleh pelanggan dapat ditangani tanpa terjadinya kekurangan barang. Selain itu, pada ERP 2016 diimplemetasikan dengan berorientasi *multitenancy* dengan Basis Data Terdistribusi.

1.2 Rumusan Permasalahan

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana membuat proses bisnis dari modul *Sales and Distribution* yang terdiri dari submodul-submodul *shipping, customer, inquiry, quotation, sales order, return order, shipping and delivery*, dan *forecasting*?
2. Bagaimana mengaplikasikan peramalan penjualan di dalam aplikasi ERP pada modul *Sales and Distribution*?
3. Bagaimana mengaplikasikan *multitenancy* pada basis data terdistribusi terkait modul *Sales and Distribution* pada ERP?
4. Bagaimana menangani kegagalan sistem pada sebuah *database server* sehingga kinerja *tenant* lainnya tidak terganggu?

1.3 Batasan Permasalahan

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Bahasa pemrograman yang digunakan adalah PHP dengan kerangka kerja Yii 2.0 dan sistem basis data yang digunakan adalah *MySQL Cluster*.
2. Aplikasi yang dikembangkan merupakan aplikasi berbasis *website*.
3. Sistem ERP yang dibangun berorientasikan *multitenancy* dengan basis data terdistribusi.
4. Hasil dari tugas akhir ini adalah sebuah aplikasi ERP modul *Sales and Distribution* yang terdiri dari submodul-submodul: *Shipping, Customer, Inquiry, Quotation, Sales order, Return order, Shipping and delivery*, dan *Forecasting*.
5. Metode yang diimplementasikan untuk peramalan penjualan adalah metode *Holt-Winter Additive* dan *Holt-Winters Multiplicative*.
6. Pada proses *shipping and delivery*, proses pengiriman dilakukan sekali.

1.4 Tujuan

Tujuan dari tugas akhir ini adalah sebagai berikut:

1. Membuat proses bisnis dari modul *Sales and Distribution* pada sistem ERP yang terdiri dari submodul-submodul *shipping, customer, inquiry, quotation, sales order, return order, shipping and delivery*, dan *forecasting*.
2. Mengimplementasikan peramalan penjualan melalui submodul *forecasting* menggunakan metode *Holt-Winters Additive* dan *Holt-Winters Multiplicative*.
3. Mengimplementasikan *multitenancy* pada modul sales and distribution.
4. Mengimplementasikan Basis Data Terdistribusi dengan melakukan replikasi dan fragmentasi menggunakan *MySQL Cluster*.

1.5 Manfaat

Manfaat dari hasil tugas akhir ini adalah memberikan kontribusi untuk membangun aplikasi ERP yang berguna untuk kalangan UKM (Usaha Kecil Menengah) di Indonesia demi meningkatkan kemampuan bisnis. Dengan adanya aplikasi ini diharapkan aplikasi ERP lebih berkembang untuk penggunaan skala UKM.

1.6 Metodologi

Langkah-langkah yang ditempuh dalam pengerjaan tugas akhir ini yaitu:

1. Studi literatur

Pada tahap ini akan dilakukan studi literatur yang berfokus pada pembuatan aplikasi PHP dengan kerangka kerja Yii 2.0 sesuai dengan proses bisnis model dari ERP, implementasi *multitenancy* dengan basis data terdistribusi serta proses bisnis yang akan digunakan seperti *Sales and Distribution*. Selain itu, juga dibantu beberapa literatur lain yang dapat *menunjang* proses penyelesaian tugas akhir ini.

2. Analisis dan desain perangkat lunak

Pada tahap analisis dan desain dilakukan perancangan sistem berdasarkan studi literatur dan pembelajaran konsep teknologi dari perangkat lunak yang digunakan. Tahap ini mendefinisikan alur dari implementasi, perancangan database, perancangan *prototype* sistem. Desain perangkat lunak akan menggunakan sistem rancang MVC (*Model-View-Controller*).

3. Implementasi perangkat lunak

Pada tahap ini implementasi dilakukan dengan cara menerapkan algoritma-algoritma pada sistem dibangun menggunakan bahasa pemrograman PHP dengan kerangka kerja Yii 2.0. Pengolahan database menggunakan database *MySQL Cluster*.

4. Pengujian dan evaluasi

Pada tahap ini dilakukan pengujian terhadap elemen perangkat lunak dengan menggunakan data uji yang telah dipersiapkan. Pengujian dan evaluasi perangkat lunak dilakukan untuk mengevaluasi jalannya perangkat lunak, mengevaluasi fitur utama, mengevaluasi fitur-fitur tambahan, mencari kesalahan yang timbul pada saat perangkat lunak aktif, dan mengadakan perbaikan jika ada kekurangan.

5. Penyusunan buku tugas akhir

Pada tahap ini dilakukan pendokumentasian dan pelaporan dari seluruh konsep, dasar teori, implementasi, proses yang telah dilakukan, dan hasil-hasil yang telah didapatkan selama pengerjaan tugas akhir.

1.7 Sistematika Penulisan

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. 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 ini berisi latar belakang masalah, tujuan dan manfaat pembuatan tugas akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

Bab II Dasar Teori

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan yang menjadi dasar dari pembuatan tugas akhir ini.

Bab III Analisis dan Perancangan Sistem

Bab ini membahas mengenai perancangan perangkat lunak. Perancangan perangkat lunak meliputi perancangan alur, proses dan perancangan antarmuka pada perangkat lunak.

Bab IV Implementasi Sistem

Bab ini berisi implementasi dari perancangan perangkat lunak perangkat lunak dan implementasi fitur-fitur penunjang perangkat lunak.

Bab V Pengujian dan Evaluasi

Bab ini membahas pengujian dengan metode pengujian subjektif untuk mengetahui penilaian aspek kegunaan (*usability*) dari perangkat lunak dan pengujian fungsionalitas yang dibuat dengan memperhatikan keluaran yang dihasilkan serta evaluasi terhadap fitur-fitur perangkat lunak.

Bab VI Kesimpulan dan Saran

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan. Bab ini membahas saran-saran untuk pengembangan sistem lebih lanjut.

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan tugas akhir.

Lampiran

Merupakan bab tambahan yang berisi dokumen tambahan yang penting pada perangkat lunak.

[Halaman ini sengaja dikosongkan]

BAB II

DASAR TEORI

Pada bab ini akan dibahas mengenai teori-teori yang menjadi dasar dari pembuatan tugas akhir. Teori-teori tersebut meliputi penelitian terkait, *Enterprise Resource Planning*, *Sales and Distribution*, Basis Data Terdistribusi, *Multitenancy*, *Role Based Access Control*, Metode *Holt-Winters*, Kerangka kerja Yii 2.0, dan *MySQL Cluster*.

2.1 Penelitian Terkait

Tugas akhir ini merupakan pengembangan riset berkelanjutan dari riset atau penelitian tentang ERP. Pada tugas akhir sebelumnya telah dikembangkan sebuah ERP bernama ERP 2013 yang menangani penjualan dan pelanggan dengan modul *Customer Relationship Management*. Pada pengembangan ERP kali ini, akan dikembangkan sebuah ERP bernama ERP 2016 yang menangani penjualan dan pelanggan dengan modul *Sales and Distribution*. Perbedaan antara ERP 2013 dan ERP 2016 dapat dilihat pada Tabel 2.1.

Tabel 2.1. Perbedaan ERP 2013 dan ERP 2016 pada Modul *Sales and Distribution*

No	ERP 2013	ERP 2016	Keterangan
1	<i>Sales Order</i>	<i>Sales Order</i>	Memiliki fungsi yang sama untuk menangani masalah penjualan
2	<i>Customer</i>	<i>Customer</i>	Memiliki fungsi yang sama untuk menangani pelanggan
3	<i>Return</i>	<i>Return Order</i>	Memiliki fungsi yang sama untuk menangani pengembalian barang
4	<i>Distribution</i>	<i>Shipping and Delivery</i>	Memiliki fungsi yang sama untuk menangani pengiriman barang, akan tetapi pada ERP 2016 pengiriman barang

No	ERP 2013	ERP 2016	Keterangan
			menjelaskan tentang kendaraan dan rute pengiriman barang
5	-	<i>Inquiry</i> dan <i>Quotation</i>	Memiliki fungsi untuk menangani aktifitas awal penjualan
6	-	<i>Forecasting</i>	Memiliki fungsi untuk melakukan peramalan penjualan
7	<i>Invoice</i>	-	Memiliki fungsi untuk memberikan jumlah tagihan penjualan ke pelanggan. Pada ERP 2016, <i>invoice</i> berada pada modul <i>Account Receivable (AR Transaction)</i>

Pada modul *Sales and Distribution* ERP 2013 terdapat submodul-submodul seperti *sales order* untuk menangani penjualan, *return* untuk menangani pengembalian barang, *customer* untuk menangani pelanggan, *distribution* untuk menangani pengiriman, dan *invoice* untuk memberikan tagihan penjualan ke pelanggan [3]. ERP 2016 memiliki modul yang juga dapat menangani penjualan dan pelanggan, yaitu modul *Sales and Distribution*. Modul *Sales and Distribution* pada ERP 2016 memiliki submodul-submodul *inquiry* dan *quotation* untuk menangani aktifitas awal penjualan, *sales order* untuk menangani penjualan, *return order* untuk menangani pengembalian barang, *customer* untuk menangani pelanggan, *shipping and delivery* untuk menangani pengiriman barang, dan *forecasting* untuk peramalan penjualan.

2.2 *Enterprise Resource Planning (ERP)*

ERP atau Perencanaan Sumber Daya Perusahaan merupakan suatu sistem terintegrasi yang terdiri dari berbagai macam modul proses bisnis perusahaan pada umumnya. Sistem ERP disebut sebagai suatu sistem yang saling terkait dikarenakan penggunaan

perangkat lunak sebagai sarana pengintegrasian antar bagian yang diinginkan untuk diintegrasikan. Sistem ERP ini seringkali digunakan perusahaan untuk mengelola data serta meneliti data-data krusial perusahaan yang tersebar di berbagai area bisnisnya, semisal data-data dari bagian keuangan, pemasaran, produksi, sumber daya manusia, dan sebagainya [4].

Keuntungan penggunaan ERP diantaranya adalah Integrasi data keuangan, standarisasi proses operasi, standarisasi data dan informasi, penurunan inventaris dan tenaga kerja, peningkatan pelayanan dan kontrol keuangan dan penurunan waktu yang dibutuhkan untuk mendapatkan informasi. Ada pun departemen-departemen atau bagian-bagian yang pada umumnya terintegrasikan meliputi:

- a. Pengelolaan keuangan (*Financial Management*) dalam aplikasi ERP 2016 ini diwakili oleh modul *Account Payable*, *Account Receivable*, *Fixed Asset*, *Cost Management*, dan *General Ledger*.
- b. Pengelolaan rantai pasokan (*Supply Chain Management*) diwakili oleh modul *Purchasing*.
- c. Perencanaan produksi dan manufaktur (*Manufacturing Resource Planning*) diwakili oleh modul *Production dan Inventory*.
- d. Pengelolaan Sumber Daya Manusia (*Human Resource Management*) diwakili oleh modul *Human Resource*.
- e. Pengelolaan relasi dengan pelanggan (*Customer Relationship Management*) diwakili oleh modul *Sales and Distribution*.

Departemen-departemen tersebut diintegrasikan melalui suatu sistem ERP terpusat, seperti yang ditunjukkan dalam Gambar 2.1.

2.3 Sales and Distribution

Proses bisnis adalah kumpulan dari tugas-tugas atau aktivitas-aktivitas yang saling membentuk menjadi hasil yang diinginkan. Setiap proses dipicu oleh sebuah kejadian, seperti terjadinya pembelian oleh pelanggan membuat penambahan

barang di dalam gudang. Sistem *enterprise* merupakan sistem untuk membantu menyelesaikan proses-proses yang ada di dalam perusahaan [2] [4] [5].



Gambar 2.1. Diagram Integrasi ERP Secara Umum

Menurut Simha R. Magal dan Jeffrey Word (2011), proses bisnis penjualan merupakan proses perusahaan menjual barang kepada pelanggan. Proses ini merupakan awal dalam proses *Make-to-Order*, dimana perusahaan membuat barang dengan adanya pembelian barang oleh pelanggan. Proses bisnis ini dimulai dengan perusahaan melakukan *presales activity*, kemudian membuat dokumen *sales order* untuk mengelola siapa yang membeli barang, barang yang dibeli, jumlah barang, harga barang, dan total harga. Kemudian proses dilanjutkan dengan mempersiapkan pengiriman barang. Barang akan dikirim dan muncul dokumen *shipment*. Setelah barang diterima oleh *customer*, maka perusahaan mengirimkan tagihan atau *invoice* kepada pelanggan.

Proses bisnis penjualan atau *Sales and Distribution* terdiri dari *presales activity (inquiry dan quotation)*, *sales order*, *return order*, *shipping and delivery*, dan *forecasting* yang akan dijelaskan sebagai berikut.

2.3.1 Inquiry

Inquiry adalah sebuah dokumen yang mencatat permintaan pelanggan ke perusahaan mengenai informasi penjualan, seperti: harga barang, ketersediaan barang, dan tanggal pengiriman [5]. *Inquiry* bersifat tidak mengikat, sehingga pelanggan dapat melakukan pembatalan transaksi sewaktu-waktu. Tujuan *Inquiry* adalah untuk *menunjukkan* minat dari pelanggan yang akan melakukan transaksi dengan perusahaan.

2.3.2 Quotation

Quotation adalah sebuah perjanjian yang ditawarkan oleh perusahaan ke pelanggan. Perusahaan menyetujui jumlah biaya untuk jumlah barang tertentu yang akan dibebankan ke pelanggan dalam batas waktu yang terbatas dan apabila *quotation* melewati batas waktu yang telah ditentukan, maka *quotation* tidak berlaku lagi [5]. *Quotation* bersifat mengikat, sehingga pelanggan tidak dapat melakukan pembatalan transaksi, pembatalan transaksi hanya dapat dilakukan apabila sesuai dengan kondisi yang telah disepakati.

2.3.3 Sales Order

Sales order adalah dokumen konfirmasi yang dikirim ke pelanggan sebelum memberikan barang atau penyelesaian jasa. *Sales order* dapat dibuat sekali sebagai kutipan yang diterima oleh calon pelanggan dari *purchase order* yang dikirim oleh pelanggan untuk diproses lebih lanjut. Setelah menerima *sales order*, pelanggan dapat meminta tanggal pengiriman barang atau jasa. Setelah pesanan dikirimkan ke pelanggan, tagihan dibuat berdasarkan pesanan penjualan untuk tujuan penagihan. Secara umum, *sales order* berisi nomor, tanggal, barang yang dijual,

termasuk jumlah dan harga berdasarkan *purchase order*, alamat penagihan, alamat pengiriman, dan Syarat & Ketentuan [6]. *Sales order* memiliki fungsi utama sebagai pencatatan pesanan dari pelanggan mengenai informasi barang yang di pesan, kuantitas barang beserta spesifikasinya termasuk harga jual.

2.3.4 Return Order

Return order adalah dokumen informasi yang berisi informasi pengembalian barang dari perusahaan ke pelanggan sesuai dengan barang yang dikembalikan oleh pelanggan. *Return order* berisi tentang jumlah barang yang dikembalikan dan alasan pengembalian barang [6].

2.3.5 Shipping and Delivery

Shipping and delivery adalah sebuah proses yang berhubungan dengan pengiriman barang atau penyampaian jasa ke pelanggan. Tujuan *shipping and delivery* adalah untuk mencatat setiap pengiriman barang yang dilakukan, sehingga pengiriman barang dapat dilakukan dengan cepat dan efisien [5].

2.3.6 Forecasting

Forecasting berfungsi untuk memperkirakan permintaan barang yang akan dijual kepada pelanggan dalam jangka waktu tertentu, peramalan penjualan juga dapat digunakan sebagai perencanaan bisnis di sebuah perusahaan dalam mengambil keputusan kedepan [7]. Dengan adanya peramalan penjualan, diharapkan jumlah dari barang yang diminta oleh pelanggan akan meningkat, sehingga dengan meningkatnya permintaan, akan mempengaruhi jumlah barang yang akan diproduksi. Peramalan penjualan juga dapat digunakan untuk menghindari kelebihan atau kekurangan pasokan barang di gudang, dan juga bertindak sebagai perencanaan ke depan untuk pembelian bahan baku, penganggaran, dan jumlah karyawan yang dibutuhkan dalam produksi [7].

2.4 Basis Data Terdistribusi

Basis Data Terdistribusi adalah suatu basis data yang memiliki kontrol terpusat pada *server* manajemen dengan distribusi penyimpanan data yang tersebar di beberapa *server* data [8]. Basis Data Terdistribusi diterapkan dalam rangka untuk menjaga prinsip *high availability* pada aplikasi ERP. Basis Data Terdistribusi memungkinkan aplikasi tetap berjalan jika terdapat sebuah *server* basis data mengalami *system failure*. Terdapat 2 mekanisme utama dalam Basis Data Terdistribusi, yaitu replikasi dan fragmentasi.

2.4.1 Replikasi

Replikasi adalah mekanisme penyalinan seluruh atau sebagian tabel basis data ke beberapa *server data* [8]. Setiap transaksi (penambahan, penghapusan, dan perubahan data) akan dieksekusi pada semua *server* data yang menyusun sistem Basis Data Terdistribusi. Hal ini menyebabkan mekanisme replikasi membebani performa sistem, namun memiliki tingkat kompleksitas yang paling sederhana.

2.4.2 Fragmentasi

Fragmentasi adalah mekanisme penyalinan sebagian data atau struktur dari setiap tabel basis data [8]. Transaksi basis data yang terjadi harus diolah terlebih dahulu oleh *server* manajemen untuk menentukan letak *server* data dari tabel yang akan terpengaruh. Hal ini menyebabkan mekanisme fragmentasi memiliki tingkat kompleksitas yang tinggi, namun memiliki performa yang lebih baik dibandingkan dengan mekanisme replikasi.

2.5 Multitenancy

Sebuah arsitektur data pada sistem perangkat lunak dimana *tenant* memungkinkan mengakses data yang dimiliki sendiri dalam basis data yang sama. Arsitektur data ini didesain untuk *multitenancy* dimana setiap *tenant* mendapat *sharing* data,

konfigurasi, manajemen pengguna, fungsionalitas setiap *tenant*, dan properti *non-fungsional* [9]. Arsitektur data *multitenancy* digunakan karena kuat dan cukup aman untuk mengatasi masalah kurangnya kepercayaan *tenant* untuk menyerahkan kontrol data bisnis ke pihak ketiga, dan juga efektif dan efisien dalam pengelolaan dan pemeliharaan. Arsitektur data *multitenancy* terbagi menjadi tiga, yaitu *separated database*, *separated schema*, *shared database*, *separated schema*, dan *shared database*, *shared schema*.

2.5.1 *Separated Database, Separated Schema*

Separated database adalah arsitektur data dimana data setiap *tenant* disimpan pada basis data yang terpisah dengan *tenant* lain. Keuntungan arsitektur ini adalah mudah untuk mengatur kembali model data aplikasi yang digunakan. Tetapi memerlukan biaya yang cukup tinggi untuk menjaga peralatan *server* dan juga *back up* data dari setiap *tenant* [9]. Model arsitektur *separated database* ditunjukkan pada Gambar 2.2.

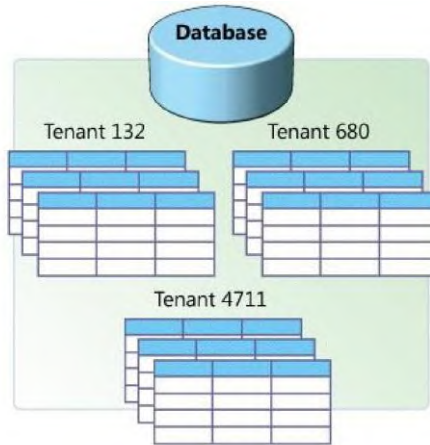


Gambar 2.2. Arsitektur Data *Separated Database*

2.5.2 *Shared Database, Separated Schema*

Shared database, *separated schema* adalah arsitektur data dimana data setiap *tenant* disimpan pada satu basis data tetapi pada *schema* yang terpisah dengan *tenant* lain. Keuntungan dari arsitektur ini adalah mudah untuk digunakan karena tabel yang dibentuk pada awalnya merupakan tabel standar, dan selanjutnya dapat diubah sesuai keinginan *tenant*. Akan tetapi apabila terjadi

kegagalan maka perlu dilakukan perbaikan untuk semua tabel yang ada dalam basis data [9]. Model arsitektur *shared database, separated schema* ditunjukkan pada Gambar 2.3.



Gambar 2.3. Arsitektur Data *Shared Database, Separated Schema*

2.5.3 *Shared Database, Shared Schema*

Shared database, shared schema adalah arsitektur data dimana data setiap *tenant* disimpan pada satu basis data dan satu *schema*. Keuntungan dari arsitektur ini adalah tidak memerlukan biaya yang tinggi, akan tetapi apabila terjadi kegagalan maka perlu dilakukan perbaikan untuk semua tabel yang ada dalam basis data [9]. Model arsitektur *shared database, shared schema* ditunjukkan pada Gambar 2.4.

TenantID	CustName	Address
4	TenantID	ProductID ProductName
1	4	TenantID Shipment Date
6	1	4711 324965 2006-02-21
4	6	132 115468 2006-04-08
4	680	654109 2006-03-27
	4711	324956 2006-02-23

Gambar 2.4. Arsitektur Data *Shared Database, Shared Schema*

2.6 *Role Based Access Control (RBAC)*

Role Based Access Control (RBAC) adalah sistem yang diterapkan pada aplikasi yang berhubungan dengan pengontrolan akses sumber daya. RBAC memberikan hak akses untuk peran (*roles*) [10]. Perancang kebijakan atau administrator sangat berperan dalam memberikan hak kepada para pengguna, sehingga subjek akan mendapatkan akses melalui *role* yang telah diberikan oleh administrator. Fitur RBAC dapat mengorganisir subjek dan *role* secara alamiah sesuai dengan struktur yang diterapkan pada perusahaan tersebut. Hubungan antara hak akses dengan para pelaku yang diterapkan pada perusahaan tersebut, pertama RBAC memberikan tugas keamanan pada kontrol akses sebagai prioritas tertinggi untuk mengontrol akses ke sumber daya. Hal tersebut mengakibatkan RBAC akan menerapkan keamanan yang sangat ketat dalam melakukan kontrol akses ke sumber daya. Kedua RBAC dalam menerapkan hak akses kepada pengguna membutuhkan waktu yang singkat, dengan cara menghubungkan subjek dengan *role*, sehingga memerlukan penunggasan hak akses untuk *role* pada setiap subjek. Struktur RBAC meliputi *roles*, *permission*, *user*, dan *session*.

2.7 *Metode Holt-Winters*

Holt (1957) dan Winters (1960) adalah pengembangan lebih lanjut dari metode sebelumnya, yaitu metode *Holt* [7]. Data *time series* menunjukkan gejala musiman. Musiman mengacu pada kecenderungan data *time series* menunjukkan gejala berulang pada setiap periode waktu tertentu atau pada setiap periode t [7]. Metode *Holt-Winters* digunakan untuk memodelkan data dengan pola musiman, baik mengandung *trend* maupun tidak. Titik berat metode ini adalah pada nilai ramalan (α), kemiringan *slope* (β), maupun efek musiman (γ). Pada metode *Holt-Winters*, tersusun atas 3 (tiga) persamaan pemulusan, yaitu: pemulusan (L_t), *trend* (B_t), dan musiman (S_t). Terdapat 2 (dua) jenis metode *Holt-Winters*, yaitu *Holt-Winters Additive* dan *Holt-Winters Multiplicative*.

2.7.1 Metode *Holt-Winters Additive*

Metode ini dapat diterapkan ketika data *time-series* memiliki komponen musiman (*seasonal*). Metode ini mengasumsikan bahwa data *time-series* terdiri dari siklus *trend* dan musiman yang linier, yang tersusun dari 3 (tiga) rangkaian proses statistik yang berkorelasi (pemulusan, *trend*, dan musiman) dan memproyeksikan *trend* serta komponen musiman ke depan [11]. Persamaan metode *Holt-Winters Additive* ditunjukkan pada Persamaan 2.1, Persamaan 2.2, Persamaan 2.3, dan Persamaan 2.4.

$$F_{t+h|t} = L_t + hB_t + S_{t-m} \quad (2.1)$$

Pada Persamaan 2.1, ditunjukkan persamaan peramalan, dimana $F_{t+h|t}$ adalah hasil peramalan, L_t adalah nilai tingkat pemulusan pada periode t , h adalah periode, B_t adalah nilai *trend* pada periode t , dan S_{t-m} adalah nilai musiman pada periode sebelumnya yang sama.

$$L_t = \alpha(Y_t - S_{t-m}) + (1 - \alpha)(L_{t-1} + B_{t-1}) \quad (2.2)$$

Pada Persamaan 2.2, ditunjukkan persamaan tingkat pemulusan (L_t), dimana α adalah konstanta pemulusan yang berada di antara nilai 0 dan 1, Y_t adalah data aktual pada periode t , S_{t-m} adalah nilai musiman pada periode sebelumnya yang sama, L_{t-1} adalah nilai tingkat pemulusan pada periode $t-1$, dan B_{t-1} adalah nilai *trend* pada periode $t-1$.

$$B_t = \beta(L_t - L_{t-1}) + (1 - \beta)B_{t-1} \quad (2.3)$$

Pada Persamaan 2.3, ditunjukkan persamaan *trend* (B_t), dimana β adalah konstanta pemulusan yang berada di antara nilai 0 dan 1, L_t adalah nilai tingkat pemulusan pada periode t , L_{t-1} adalah

nilai tingkat pemulusan pada periode $t-1$, dan B_{t-1} adalah nilai *trend* pada periode $t-1$.

$$S_t = \gamma(Y_t - L_{t-1} - B_{t-1}) + (1 - \gamma)S_{t-m} \quad (2.4)$$

Pada Persamaan 2.4, ditunjukkan persamaan musiman (S_t), dimana γ adalah konstanta pemulusan yang berada di antara nilai 0 dan 1, Y_t adalah data aktual pada periode t , S_{t-m} adalah nilai musiman pada periode sebelumnya yang sama, L_{t-1} adalah nilai tingkat pemulusan pada periode $t-1$, dan B_{t-1} adalah nilai *trend* pada periode $t-1$.

2.7.2 Metode *Holt-Winters Multiplicative*

Metode ini dapat diterapkan pada data *time-series* musiman sama halnya dengan metode *additive*, tetapi pada model ini diasumsikan bahwa komponen-komponen data *time-series* (pemulusan data, *trend*, dan musiman), dikalikan satu sama lain sehingga menghasilkan data *time-series* yang lebih aktif [11]. Persamaan metode *Holt-Winters Multiplicative* ditunjukkan pada Persamaan 2.5, Persamaan 2.6, Persamaan 2.7, dan Persamaan 2.8.

$$F_{t+h|t} = (L_t + hB_t)S_{t-m} \quad (2.5)$$

Pada Persamaan 2.5, ditunjukkan persamaan peramalan, dimana $F_{t+h|t}$ adalah hasil peramalan, L_t adalah nilai tingkat pemulusan pada periode t , h adalah periode, B_t adalah nilai *trend* pada periode t , dan S_{t-m} adalah nilai musiman pada periode sebelumnya yang sama.

$$L_t = \alpha \left(\frac{Y_t}{S_{t-m}} \right) + (1 - \alpha)(L_{t-1} + B_{t-1}) \quad (2.6)$$

Pada Persamaan 2.6, ditunjukkan persamaan tingkat pemulusan (L_t), dimana α adalah konstanta pemulusan yang berada di antara nilai 0 dan 1, Y_t adalah data aktual pada periode t , S_{t-m} adalah nilai musiman pada periode sebelumnya yang sama, L_{t-1} adalah nilai tingkat pemulusan pada periode $t-1$, dan B_{t-1} adalah nilai *trend* pada periode $t-1$.

$$B_t = \beta(L_t - L_{t-1}) + (1 - \beta)B_{t-1} \quad (2.7)$$

Pada Persamaan 2.7, ditunjukkan persamaan *trend* (B_t), dimana β adalah konstanta pemulusan yang berada di antara nilai 0 dan 1, L_t adalah nilai tingkat pemulusan pada periode t , L_{t-1} adalah nilai tingkat pemulusan pada periode $t-1$, dan B_{t-1} adalah nilai *trend* pada periode $t-1$.

$$S_t = \gamma \left(\frac{Y_t}{L_{t-1} - B_{t-1}} \right) + (1 - \gamma)S_{t-m} \quad (2.8)$$

Pada Persamaan 2.8, ditunjukkan persamaan musiman (S_t), dimana γ adalah konstanta pemulusan yang berada di antara nilai 0 dan 1, Y_t adalah data aktual pada periode t , S_{t-m} adalah nilai musiman pada periode sebelumnya yang sama, L_{t-1} adalah nilai tingkat pemulusan pada periode $t-1$, dan B_{t-1} adalah nilai *trend* pada periode $t-1$.

2.8 *Hypertext Preprocessor (PHP)*

PHP adalah singkatan dari *Hypertext Preprocessor* yakni salah satu bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs *web* dan bisa digunakan bersamaan dengan HTML. PHP diciptakan oleh Rasmus Lerdorf pertama kali tahun 1994¹. Pada awalnya PHP

¹ <http://php.net/manual/en/history.php.php>

adalah singkatan dari *Personal Home Page Tools*. Selanjutnya diganti FI (*Forms Interpreter*). Sejak versi 3.0, nama bahasa ini diubah menjadi PHP (*Hypertext Preprocessor*) dengan singkatan PHP. PHP versi terbaru adalah versi ke-7.

2.9 Kerangka Kerja Yii 2.0

Yii 2.0 adalah sebuah kerangka kerja untuk bahasa pemrograman PHP dimana kerangka kerja ini telah didukung oleh aspek *modularity* dan *eloquent database*². Arsitektur perangkat lunak ini menggunakan *Model-View-Controller*, namun dengan aspek *modularity*, maka dapat dibuat modul dimana setiap modul memiliki MVC tersendiri. Sedangkan *eloquent database* adalah basis data berupa objek di dalam sebuah bahasa pemrograman PHP.

2.10 MySQL Cluster

MySQL Cluster merupakan sebuah tipe basis data yang dapat beroperasi dalam ukuran data yang relatif besar. *MySQL Cluster* adalah sebuah teknologi baru untuk memungkinkan *clustering* di dalam *memory* basis data dalam sebuah sistem *share-nothing*. Arsitektur *share-nothing* memungkinkan sistem dapat bekerja dengan perangkat keras yang sangat murah, dan tidak membutuhkan perangkat keras dan lunak dengan spesifikasi khusus. Arsitektur tersebut juga handal karena masing-masing komponen mempunyai *memory* dan *disk* tersendiri. *MySQL Cluster* menggabungkan *MySQL Server* biasa dengan sebuah mesin penyimpanan *in-memory ter-cluster* yang dinamakan NDB. NDB berarti bagian dari suatu rangkaian yang dikhususkan sebagai mesin penyimpanan, sedangkan *MySQL Cluster* diartikan sebagai kombinasi atau gabungan dari *MySQL* dan mesin penyimpanan yang baru tersebut³.

² <http://www.yiiframework.com/doc-2.0/guide-intro-yii.html>

³ <https://www.mysql.com/products/cluster/>

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini akan dibahas tahap analisis dan perancangan sistem yang akan dibangun. Analisis membahas semua persiapan yang akan menjadi pokok pikiran pembuatan aplikasi. Perancangan sistem membahas hal-hal yang berkaitan dengan dasar pembuatan aplikasi.

3.1 Analisis

Tahap analisis dibagi menjadi beberapa bagian antara lain: analisis proses bisnis, analisis data, analisis peramalan penjualan, deskripsi umum sistem, identifikasi pengguna, spesifikasi kebutuhan perangkat lunak, dan kasus penggunaan.

3.1.1 Analisis Proses Bisnis

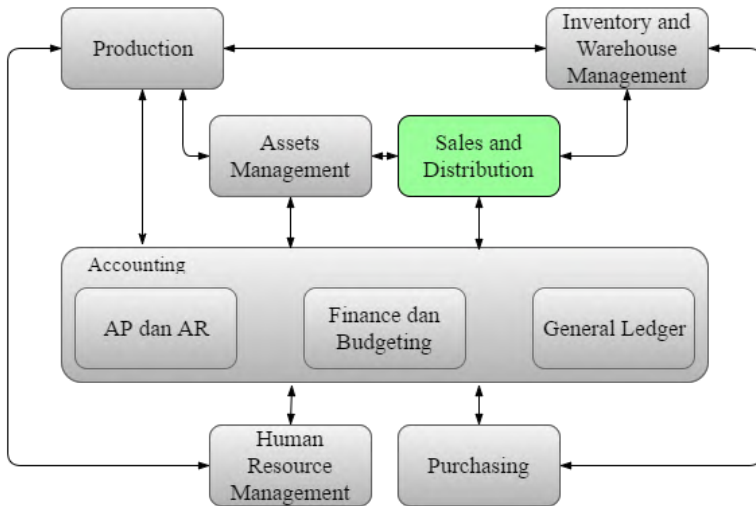
Setiap ERP memiliki proses bisnis yang berbeda-beda. Pada tugas akhir ini, aplikasi ERP yang dikembangkan memiliki proses bisnis tersendiri dan dibandingkan dengan aplikasi ERP lainnya yang cukup banyak digunakan oleh perusahaan-perusahaan, seperti Odoo, Adempiere, dan InoERP. Pada subbab ini akan dijelaskan tentang analisa proses bisnis yang telah ada. Proses bisnis yang dimiliki oleh Odoo, Adempiere, dan InoERP ditunjukkan pada Lampiran A.1, Lampiran A.2, dan Lampiran A.3. Perbandingan masing-masing ERP di tunjukkan pada Tabel 3.1.

Tabel 3.1. Perbandingan Proses Bisnis ERP 2016, Odoo, InoERP, dan Adempiere pada modul *Sales and Distribution*

No	Nama Modul	Odoo	InoERP	Adempiere
1	<i>Sales and Distribution</i>	Kontra: Pada proses <i>creating delivery</i> , apabila barang yang dipesan	Kontra: Pada proses <i>creating delivery</i> , apabila barang yang dipesan	Kontra: Pada proses <i>creating delivery</i> , apabila barang yang dipesan

No	Nama Modul	Odoo	InoERP	Adempiere
		tidak tersedia di gudang, <i>delivery</i> tetap dapat dilakukan	tidak tersedia di gudang, <i>delivery</i> tetap dapat dilakukan	tidak tersedia di gudang, <i>delivery</i> tetap dapat dilakukan
		Tidak memiliki proses <i>presales activity</i>		
		Pro: Memiliki halaman <i>front end</i> untuk pemasaran secara langsung	Pro: Dapat menghitung probabilitas pembelian barang dari pelanggan	Pro: -

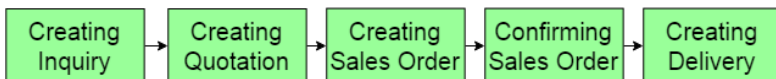
Pada Tabel 3.1 menunjukkan kelebihan dan kekurangan dari masing-masing ERP pada modul *Sales and Distribution*. Berdasarkan kekurangan yang dimiliki oleh setiap ERP maka perlu dikembangkannya sebuah aplikasi ERP yang sesuai dengan kebutuhan pada perencanaan bisnis. Terdapat dua jenis proses bisnis yang digunakan sesuai dengan subbab 2.3, yaitu *Make-To-Order* (MTO) dan *Make-To-Stock* (MTS). MTO adalah sebuah proses bisnis dimana produksi didasarkan pada permintaan yang dilakukan oleh *sales order*. MTS adalah sebuah proses bisnis dimana produksi didasarkan oleh peramalan penjualan. Proses bisnis yang dirancang sesuai dengan kebutuhan *business plan* dimana proses bisnis ini telah mengakomodasi kekurangan Odoo, Adempiere, dan InoERP ditunjukkan pada Gambar 3.1, Gambar 3.2, Gambar 3.3, Gambar 3.4, Gambar 3.5, Gambar 3.6, dan Gambar 3.7.



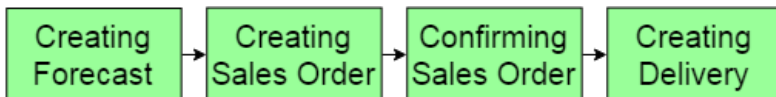
Gambar 3.1. Proses Bisnis ERP Level 0

Pada Gambar 3.1, modul yang berlatar belakang hijau merupakan modul yang dikerjakan dalam tugas akhir ini, yaitu modul *Sales and Distribution*.

Proses bisnis ERP level 0 dapat dipecah kembali menjadi proses-proses bisnis yang lebih rinci berdasarkan subbab 2.3. Proses bisnis secara keseluruhan ditunjukkan pada Lampiran A.4 dan Lampiran A.5. Hasil pemecahan proses bisnis ditunjukkan pada Gambar 3.2 dan Gambar 3.3.



Gambar 3.2. Proses Bisnis Level 1 Make-To-Order



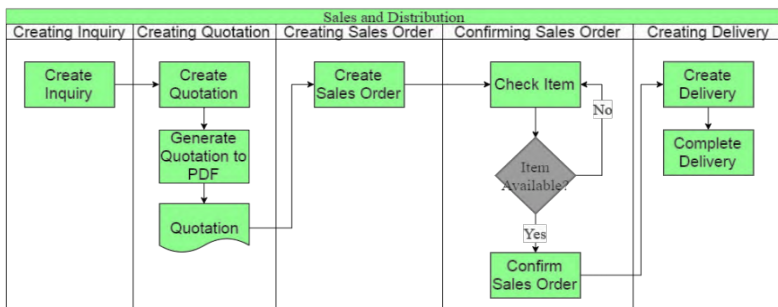
Gambar 3.3. Proses Bisnis Level 1 Make-To-Stock

Pada Gambar 3.2 dan Gambar 3.3 ditunjukkan proses bisnis *level 1* MTO dan proses bisnis *level 1* MTS. Penjelasan proses bisnis *level 1* terdapat pada Tabel 3.2.

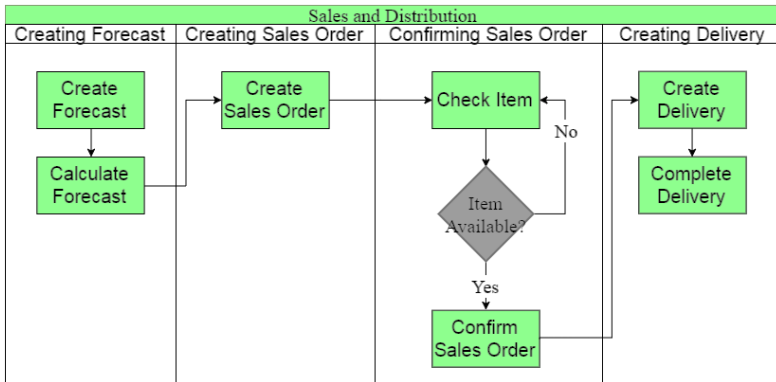
Tabel 3.2. Deskripsi Proses Bisnis Level 1

No	Proses Bisnis	Keterangan
1	<i>Creating Inquiry</i>	Merupakan proses bisnis untuk membuat <i>inquiry</i> .
2	<i>Creating Quotation</i>	Merupakan proses bisnis untuk membuat <i>quotation</i> .
3	<i>Creating Sales Order</i>	Merupakan proses bisnis untuk membuat <i>sales order</i> .
4	<i>Confirming Sales Order</i>	Merupakan proses bisnis untuk memastikan jumlah barang yang dipesan telah tersedia sebelum dilakukan.
5	<i>Creating Delivery</i>	Merupakan proses bisnis untuk membuat pengiriman barang ke pelanggan.
6	<i>Creating Forecast</i>	Merupakan proses bisnis untuk membuat peramalan penjualan.

Dari masing-masing proses bisnis *level 1* dapat dipecah kembali menjadi proses bisnis yang lebih rinci lagi. Proses bisnis ditunjukkan pada Gambar 3.4 dan Gambar 3.5.



Gambar 3.4. Proses Bisnis Level 2 Make-To-Order



Gambar 3.5. Proses Bisnis Level 2 Make-To-Order

Pada Gambar 3.4 ditunjukkan proses bisnis *level 2* MTO dan Gambar 3.5 ditunjukkan proses bisnis *level 2* MTS. Penjelasan proses bisnis *level 2* terdapat pada Tabel 3.3.

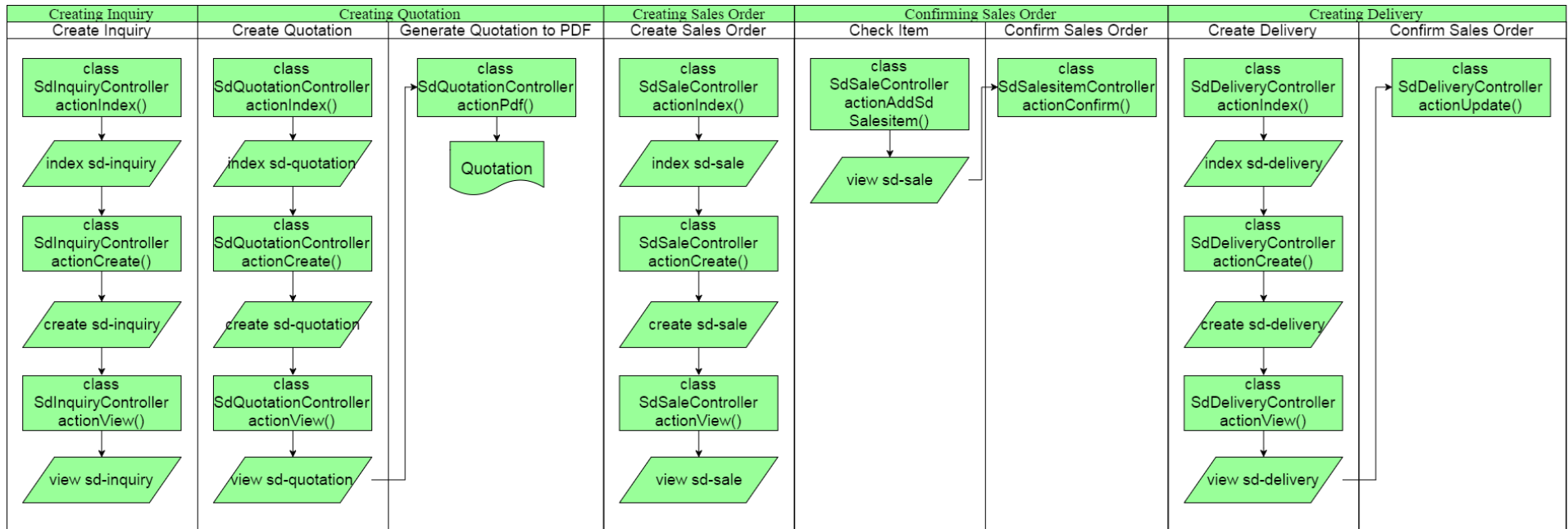
Tabel 3.3. Deskripsi Proses Bisnis Level 2

No	Proses Bisnis	Keterangan
1	<i>Create Inquiry</i>	Merupakan proses bisnis untuk membuat <i>inquiry</i> .
2	<i>Create Quotation</i>	Merupakan proses bisnis untuk membuat <i>quotation</i> .
3	<i>Generate Quotation to PDF</i>	Merupakan proses bisnis untuk membuat dokumen <i>quotation</i> .
4	<i>Quotation</i>	Dokumen <i>quotation</i> .
5	<i>Create Sales Order</i>	Merupakan proses bisnis untuk membuat <i>sales order</i> .

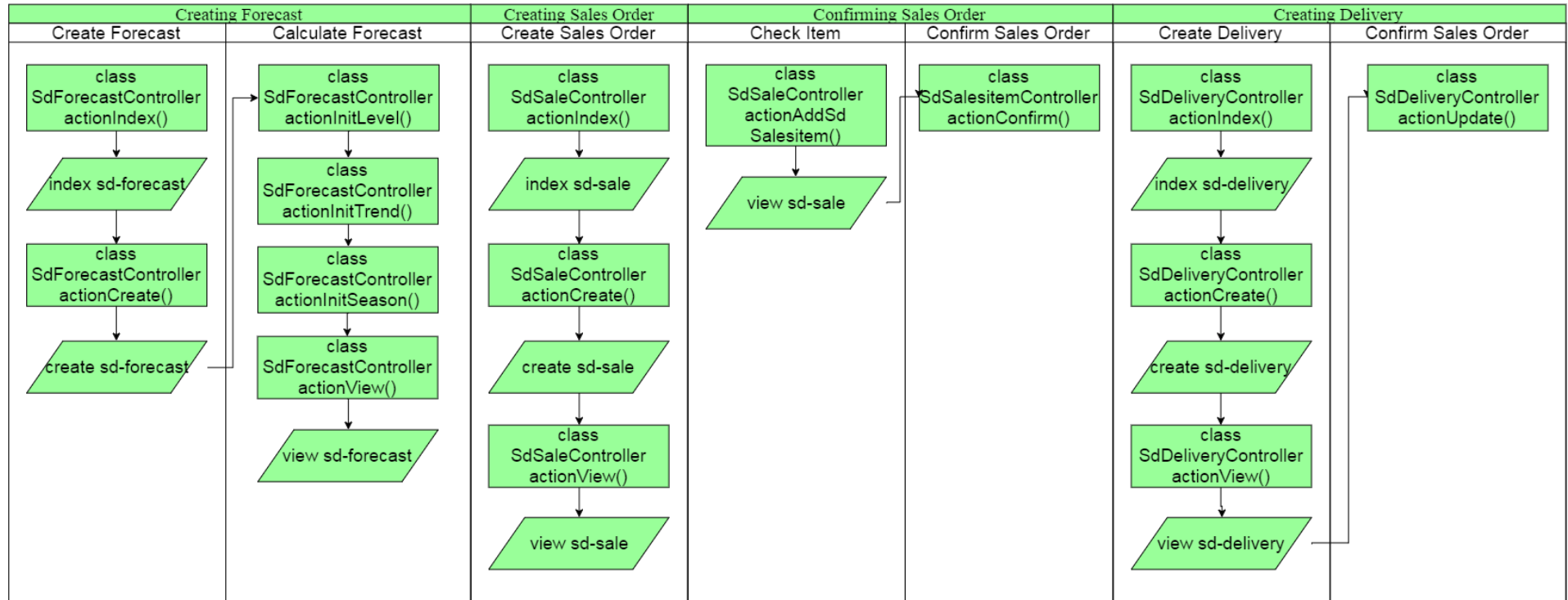
No	Proses Bisnis	Keterangan
6	<i>Check Item</i>	Merupakan proses bisnis untuk pengecekan ketersediaan barang.
7	<i>Confirm Sales Order</i>	Merupakan proses bisnis untuk memastikan jumlah barang yang dipesan telah tersedia sebelum dilakukan.
8	<i>Create Delivery</i>	Merupakan proses bisnis untuk membuat pengiriman barang ke pelanggan.
9	<i>Complete Delivery</i>	Merupakan proses bisnis untuk memastikan bahwa barang yang dikirim telah sampai ke pelanggan.
10	<i>Create Forecast</i>	Merupakan proses bisnis untuk membuat peramalan penjualan.
11	<i>Calculate Forecast</i>	Merupakan proses bisnis untuk melakukan perhitungan permalan penjualan dari data penjualan aktual.

Dari masing-masing proses bisnis *level 2* dapat dipecah kembali menjadi proses bisnis yang lebih rinci lagi. Proses bisnis ditunjukkan pada Gambar 3.6 dan Gambar 3.7.

Gambar 3.6 merupakan proses bisnis *level 3* MTO dan Gambar 3.7 merupakan proses bisnis *level 3* MTS. Pada proses bisnis *level 3* menjelaskan detail dari kelas, fungsi, dan halaman yang ditampilkan untuk melakukan proses bisnis yang terdapat pada *level 2*.



Gambar 3.6. Proses Bisnis Level 3 Make-To-Order



Gambar 3.7. Proses Bisnis Level 3 Make-To-Stock

3.1.2 Analisis Data

Konsep autentikasi dan otorisasi data yang digunakan adalah *multitenancy separated database, separated schema* yang dijelaskan pada subbab 2.5.1. Pada konsep ini, data milik setiap *tenant* akan disimpan di basis data yang terpisah. Sehingga tidak ada 2 (dua) atau lebih *tenant* menggunakan tabel dan basis data yang sama. Penambahan *tenant* baru akan menyebabkan penambahan sebuah basis data baru. Penggunaan konsep *separated database, separated schema* dimaksudkan agar data milik setiap *tenant* dapat terjamin kerahasiaannya. Keuntungan lainnya adalah kerumitan proses *maintenance* basis data dapat dikurangi.

Konsep basis data terdistribusi yang diterapkan adalah replikasi yang dijelaskan pada subbab 2.4.1. Replikasi memungkinkan penyalinan setiap tabel basis data ke *node-node* penyusun basis data terdistribusi. Kegagalan sebuah *node* tidak akan menyebabkan basis data berhenti bekerja. Sebaliknya, hal tersebut akan memicu mekanisme sinkronisasi jika *node* yang mati kembali hidup.

3.1.3 Analisis Peramalan Penjualan

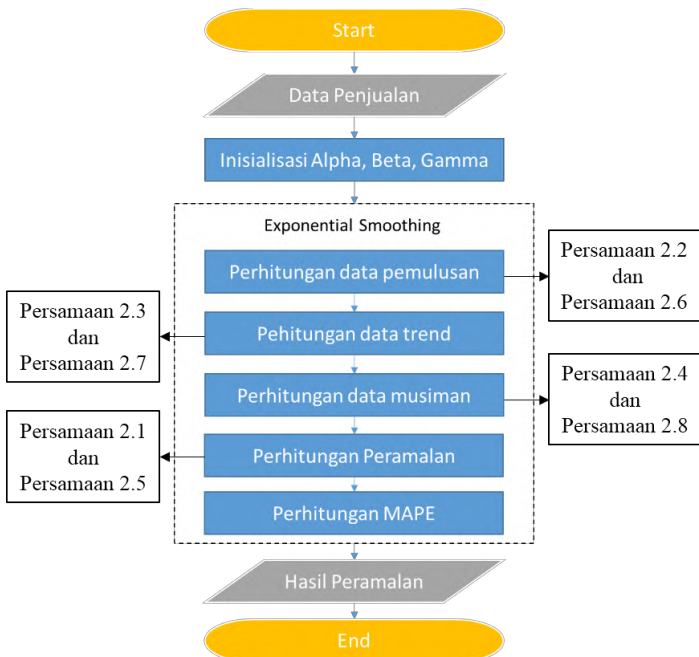
Pada bagian ini akan dijelaskan mengenai perancangan peramalan penjualan yang terdapat pada sistem ERP 2016 menggunakan metode *Holt-Winters Additive* dan *Holt-Winters Multiplicative*.

3.1.3.1 Data Masukkan dan Data Keluaran

Data masukkan pada peramalan penjualan merupakan data penjualan bulanan selama 5 tahun yang berupa 60 data yang bersifat *time-series*. Terdiri dari 48 data latihan dan 12 data uji. Data masukkan ditunjukkan pada Lampiran B.1. Data keluaran pada peramalan penjualan berupa hasil peramalan penjualan selama beberapa bulan sejumlah periode.

3.1.3.2 Proses Peramalan Penjualan

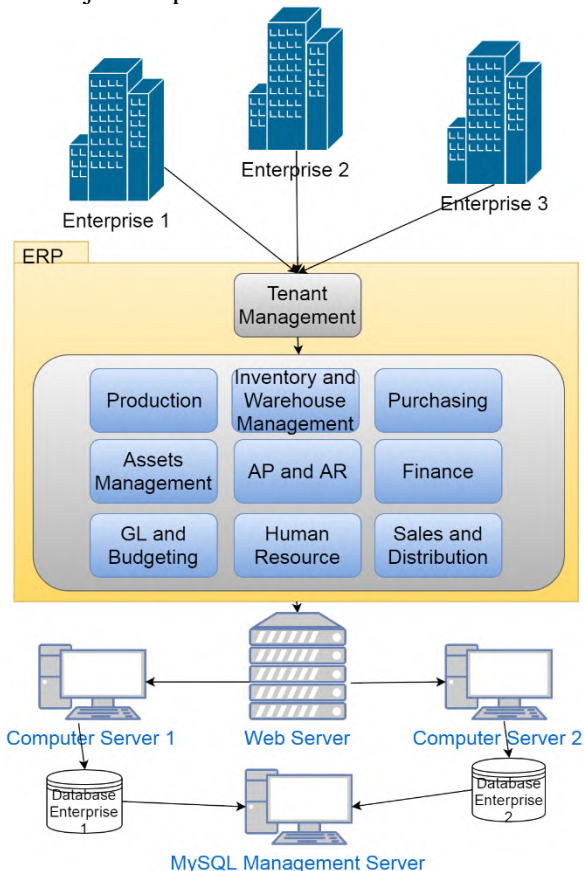
Proses peramalan penjualan terdiri dari beberapa tahap, yaitu inialisasi nilai alpha, beta, dan gamma untuk menentukan nilai awal dan inialisasi pada peramalan dengan menggunakan metode *Holt-Winters*, perhitungan pemulusan eksponensial untuk menghitung pemulusan eksponensial dengan menghitung dari data pemulusan (S_t) yang dijelaskan pada Persamaan 2.2 dan Persamaan 2.6, perhitungan data *trend* (B_t) yang dijelaskan pada Persamaan 2.3 dan Persamaan 2.7, dan perhitungan data musiman (L_t) yang dijelaskan pada Persamaan 2.4 dan Persamaan 2.8, yang kemudian dilakukan tahapan untuk menghitung akurasi peramalan dengan menggunakan MAPE (*Mean Absolute Percentage Error*), pelatihan *smoothing* untuk menemukan nilai yang alpha, beta, dan gamma yang nilai keluaran yang diharapkan. Proses-proses tersebut ditunjukkan pada Gambar 3.8.



Gambar 3.8. Diagram Alur Proses Peramalan Penjualan

3.2 Deskripsi Umum Sistem

Pada tugas akhir ini dibangun aplikasi *Sales and Distribution* yang terintegrasi ke dalam sebuah sistem *Enterprise Resource Planning* (ERP). Tujuan dari aplikasi *Sales and Distribution* adalah untuk mengelola dan melakukan monitoring terhadap penjualan dan pelanggan. Sistem yang dibangun berorientasi *multitenancy* dengan Basis Data Terdistribusi. Desain basis data terdistribusi dan *multitenancy* yang digunakan dalam sistem ini ditunjukkan pada Gambar 3.9.



Gambar 3.9. Rancang Basis Data Terdistribusi

Pada Gambar 3.9, dijelaskan bahwa perusahaan 1, perusahaan 2, dan perusahaan 3 dapat menggunakan ERP secara bersamaan. Setiap perusahaan mempunyai 2 basis data yang telah direplikasi yaitu *Computer Server 1* dengan *IP Enterprise 1* dan *Computer Server 2* dengan *IP Enterprise 2*. Komputer 3 berfungsi sebagai *server cluster* untuk melakukan replikasi, sehingga apabila basis data 1 atau *Computer Server 1* dalam keadaan mati, sistem ERP masih tetap berjalan.

3.3 Identifikasi Pengguna

Pada sistem yang akan dibangun, terdapat 2 (dua) aktor yang akan menjadi pengguna sistem ERP, yaitu staf *Sales and Distribution* dan manajer *Sales and Distribution*. Staf *Sales and Distribution* memiliki hak akses dalam pengelolaan *Sales and Distribution*, yang terdiri atas membuat (*create*), menyunting (*update*), melihat (*view*) dan menghapus (*delete*). Sedangkan manajer mempunyai hak akses melihat (*view*) data yang telah dikelola oleh staf.

3.4 Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan dalam sistem ini mencakup kebutuhan fungsional. Kebutuhan fungsional berisikan proses-proses yang dibutuhkan dalam sistem dan harus dijalankan. Kebutuhan fungsional sistem dideskripsikan dalam Tabel 3.4.

Tabel 3.4. Daftar Kebutuhan Fungsional Sistem

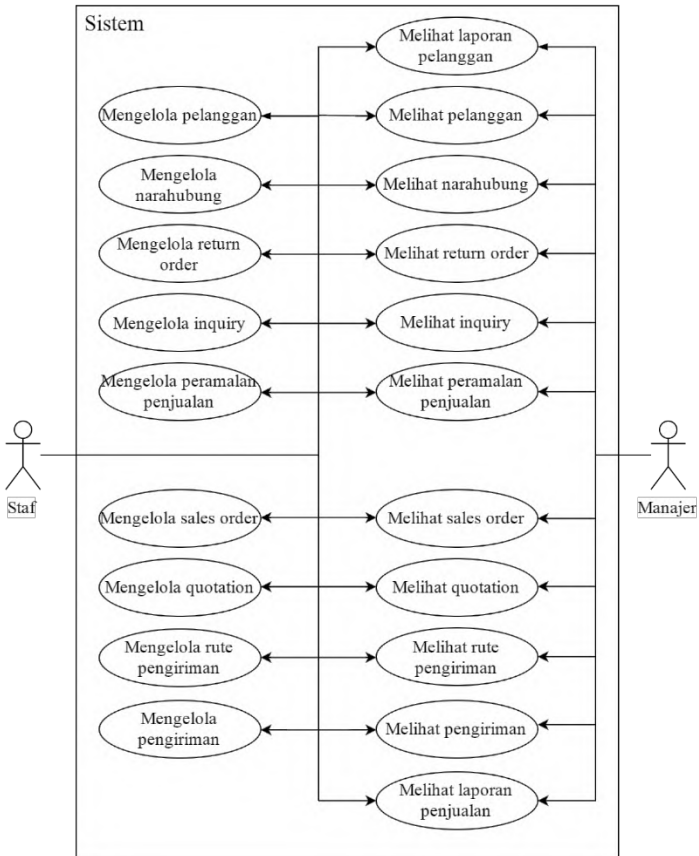
Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
F-001	Mengelola rute pengiriman	Pengguna dapat mengelola rute pengiriman yang meliputi proses: <i>create, update, delete</i>
F-002	Mengelola narahubung	Pengguna dapat mengelola narahubung yang meliputi proses: <i>create, update, delete</i>

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
F-003	Mengelola pelanggan	Pengguna dapat mengelola pelanggan yang meliputi proses: <i>create, update, delete</i>
F-004	Mengelola <i>inquiry</i>	Pengguna dapat mengelola <i>inquiry</i> yang meliputi proses: <i>create, update, delete</i>
F-005	Mengelola <i>quotation</i>	Pengguna dapat mengelola <i>quotation</i> yang meliputi proses: <i>create, update, delete</i>
F-006	Mengelola <i>sales order</i>	Pengguna dapat mengelola <i>sales order</i> yang meliputi proses: <i>create, update, delete</i>
F-007	Mengelola <i>return order</i>	Pengguna dapat mengelola <i>return order</i> yang meliputi proses: <i>create, update, delete</i>
F-008	Mengelola pengiriman	Pengguna dapat mengelola pengiriman yang meliputi proses: <i>create, update, delete</i>
F-009	Mengelola peramalan penjualan	Pengguna dapat mengelola peramalan penjualan yang meliputi proses: <i>create, update, delete</i>
F-010	Melihat rute pengiriman	Pengguna dapat melihat data rute pengiriman
F-011	Melihat narahubung	Pengguna dapat melihat data narahubung
F-012	Melihat pelanggan	Pengguna dapat melihat data pelanggan

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
F-013	Melihat <i>inquiry</i>	Pengguna dapat melihat data <i>inquiry</i>
F-014	Melihat <i>quotation</i>	Pengguna dapat melihat data <i>quotation</i>
F-015	Melihat <i>sales order</i>	Pengguna dapat melihat data <i>sales order</i>
F-016	Melihat <i>return order</i>	Pengguna dapat melihat data <i>return order</i>
F-017	Melihat pengiriman	Pengguna dapat melihat data pengiriman
F-018	Melihat peramalan penjualan	Pengguna dapat melihat hasil peramalan penjualan untuk beberap bulan kedepan
F-019	Melihat laporan penjualan	Pengguna dapat melihat laporan penjualan
F-020	Melihat laporan pelanggan	Pengguna dapat melihat laporan pelanggan

3.5 Kasus Penggunaan

Kasus penggunaan yang dibutuhkan pada sistem sesuai dengan analisa yang telah dilakukan. Diagram kasus penggunaan dapat dilihat pada Gambar 3.10 dan kode kasus penggunaan terdapat pada Tabel 3.5.



Gambar 3.10. Diagram Kasus Penggunaan

Tabel 3.5. Keterangan Kode Kasus Penggunaan

No.	Kode	Nama kasus pengguna
1	UC-001	Mengelola rute pengiriman
2	UC-002	Mengelola narahubung
3	UC-003	Mengelola pelanggan
4	UC-004	Mengelola <i>inquiry</i>

No.	Kode	Nama kasus pengguna
5	UC-005	Mengelola <i>quotation</i>
6	UC-006	Mengelola <i>sales order</i>
7	UC-007	Mengelola <i>return order</i>
8	UC-008	Mengelola pengiriman
9	UC-009	Mengelola peramalan penjualan
10	UC-010	Melihat rute pengiriman
11	UC-011	Melihat narahubung
12	UC-012	Melihat pelanggan
13	UC-013	Melihat <i>inquiry</i>
14	UC-014	Melihat <i>quotation</i>
15	UC-015	Melihat <i>sales order</i>
16	UC-016	Melihat <i>return order</i>
17	UC-017	Melihat pengiriman
18	UC-018	Melihat peramalan penjualan
19	UC-019	Melihat laporan penjualan
20	UC-020	Melihat laporan pelanggan

3.5.1 Kasus Penggunaan Mengelola Rute Pengiriman

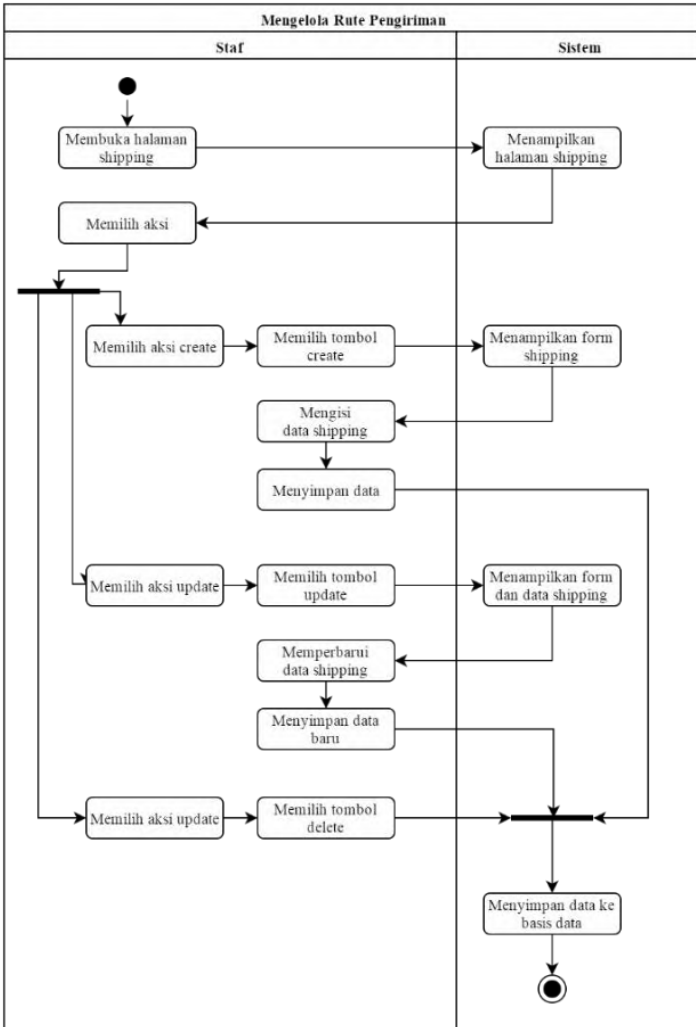
Pada kasus penggunaan ini, sistem menampilkan halaman *shipping*. Pengguna dapat melakukan penambahan rute pengiriman dengan mengisi *form* yang tersedia. Kasus penggunaan mengelola rute pengiriman ditunjukkan pada Tabel 3.6 dan Gambar 3.11.

Tabel 3.6. Kasus Penggunaan Mengelola Rute Pengiriman

Nama	Mengelola rute pengiriman
Kode	UC-001

Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data rute pengiriman
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar rute pengiriman 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah rute pengiriman baru A2. Pengguna memilih menyunting rute pengiriman tertentu A3. Pengguna memilih menghapus rute pengiriman tertentu 3. Kasus penggunaan berakhir
Alur Alternatif	<ol style="list-style-type: none"> A1. Pengguna memilih menambah rute pengiriman baru <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form shipping</i> 2. Pengguna memasukkan data rute pengiriman baru 3. Sistem menyimpan data rute pengiriman 4. Berlanjut ke alur normal langkah 3 A2. Pengguna memilih menyunting rute pengiriman tertentu <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form shipping</i> 2. Pengguna menyunting data rute pengiriman 3. Sistem memperbaharui data rute pengiriman 4. Berlanjut ke alur normal langkah 3 A3. Pengguna memilih menghapus rute pengiriman tertentu <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data rute pengiriman 2. Pengguna menyetujui penghapusan data

	3. Sistem menghapus data rute pengiriman 4. Berlanjut ke alur normal langkah 3
Kondisi Akhir	Sistem menampilkan daftar rute pengiriman



Gambar 3.11. Diagram Aktifitas Mengelola Rute Pengiriman

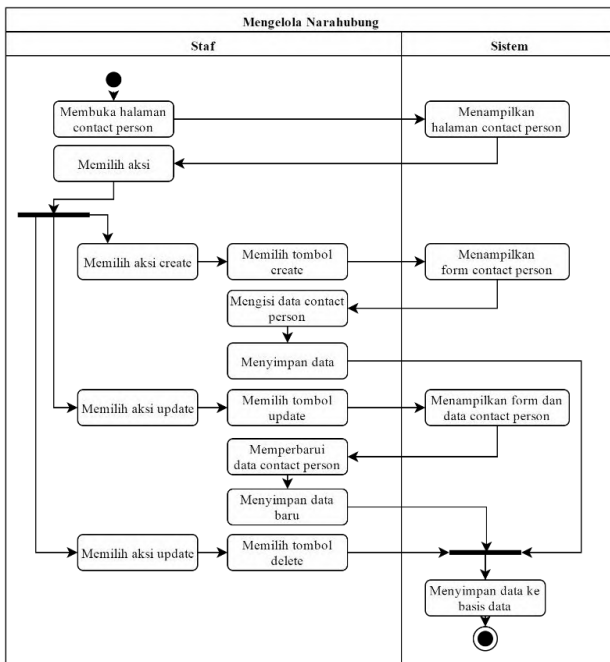
3.5.2 Kasus Penggunaan Mengelola Narahubung

Pada kasus penggunaan ini, sistem menampilkan halaman *contact person*. Pengguna dapat melakukan penambahan narahubung dengan mengisi *form* yang tersedia. Kasus penggunaan mengelola narahubung ditunjukkan pada Tabel 3.7 dan Gambar 3.12.

Tabel 3.7. Kasus Penggunaan Mengelola Narahubung

Nama	Mengelola narahubung
Kode	UC-002
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data narahubung
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu contact person</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar narahubung 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah narahubung baru A2. Pengguna memilih menyunting narahubung tertentu A3. Pengguna memilih menghapus narahubung tertentu 3. Kasus penggunaan berakhir
Alur Alternatif	<ol style="list-style-type: none"> A1. Pengguna memilih menambah narahubung baru <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form contact person</i> 2. Pengguna memasukkan data narahubung baru 3. Sistem menyimpan data narahubung 4. Berlanjut ke alur normal langkah 3 A2. Pengguna memilih menyunting narahubung tertentu

	<ol style="list-style-type: none"> 1. Sistem menampilkan <i>form contact person</i> 2. Pengguna menyunting data narahubung 3. Sistem memperbaharui data narahubung 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus narahubung tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data narahubung 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data narahubung 4. Berlanjut ke alur normal langkah 3
Kondisi Akhir	Sistem menampilkan daftar narahubung



Gambar 3.12. Diagram Aktifitas Mengelola Narahubung

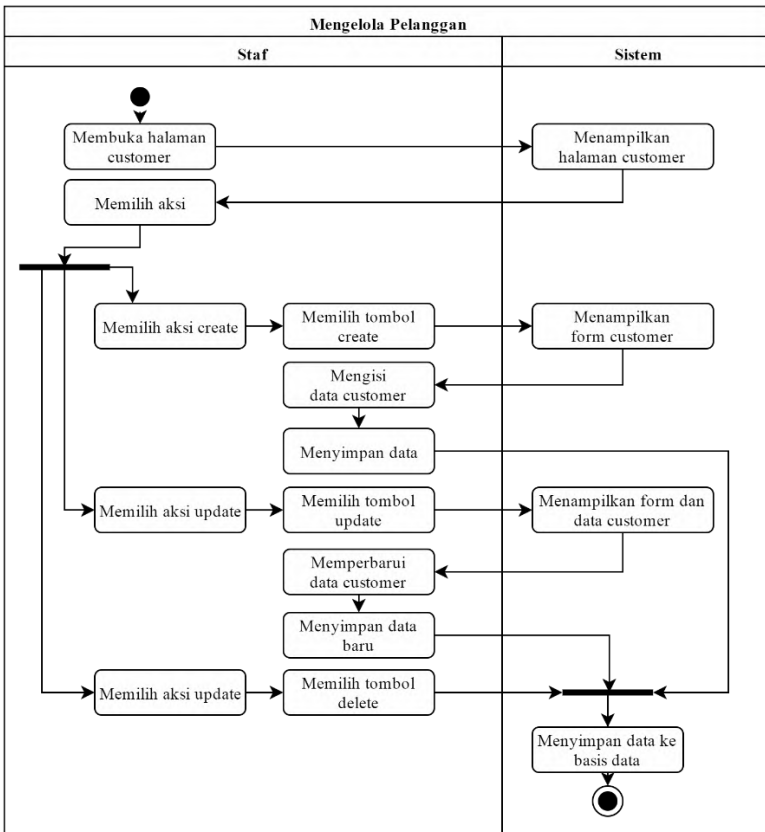
3.5.3 Kasus Penggunaan Mengelola Pelanggan

Pada kasus penggunaan ini, sistem menampilkan halaman pelanggan. Pengguna dapat melakukan penambahan pelanggan dengan mengisi *form* yang tersedia. Kasus penggunaan mengelola pelanggan ditunjukkan pada Tabel 3.8 dan Gambar 3.13.

Tabel 3.8. Kasus Penggunaan Mengelola Pelanggan

Nama	Mengelola pelanggan
Kode	UC-003
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data pelanggan
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu customer</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar pelanggan 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah pelanggan baru A2. Pengguna memilih menyunting pelanggan tertentu A3. Pengguna memilih menghapus pelanggan tertentu 3. Kasus penggunaan berakhir
Alur Alternatif	<ol style="list-style-type: none"> A1. Pengguna memilih menambah pelanggan baru <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form customer</i> 2. Pengguna memasukkan data pelanggan baru 3. Sistem menyimpan data pelanggan 4. Berlanjut ke alur normal langkah 3 A2. Pengguna memilih menyunting pelanggan tertentu <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form customer</i> 2. Pengguna menyunting data pelanggan 3. Sistem memperbaharui data pelanggan

	<ol style="list-style-type: none"> 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus pelanggan tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data pelanggan 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data pelanggan 4. Berlanjut ke alur normal langkah 3
Kondisi Akhir	Sistem menampilkan daftar pelanggan



Gambar 3.13. Diagram Aktifitas Mengelola Pelanggan

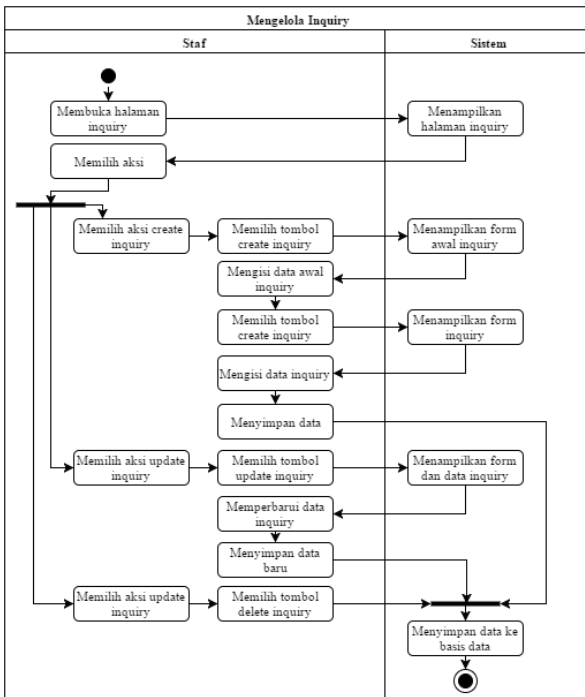
3.5.4 Kasus Penggunaan Mengelola *Inquiry*

Pada kasus penggunaan ini, sistem menampilkan halaman untuk melakukan pembuatan *inquiry*. Pengguna dapat melakukan pembuatan *inquiry* dengan mengisi *form* yang tersedia. Kasus penggunaan mengelola *inquiry* ditunjukkan pada Tabel 3.9 dan Gambar 3.14.

Tabel 3.9. Kasus Penggunaan Mengelola *Inquiry*

Nama	Mengelola <i>inquiry</i>
Kode	UC-004
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data <i>inquiry</i>
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu inquiry</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>inquiry</i> 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah <i>inquiry</i> baru A2. Pengguna memilih menyunting <i>inquiry</i> tertentu A3. Pengguna memilih menghapus <i>inquiry</i> tertentu 3. Kasus penggunaan berakhir
Alur Alternatif	<ol style="list-style-type: none"> A1. Pengguna memilih menambah <i>inquiry</i> baru <ol style="list-style-type: none"> 1. Sistem menampilkan kotak dialog <i>inquiry</i> 2. Pengguna memasukkan jenis <i>inquiry</i> dan memilih tombol <i>create</i> 3. Sistem menampilkan <i>form inquiry</i> 4. Pengguna memasukkan data <i>inquiry</i> baru 5. Sistem menyimpan data <i>inquiry</i> 6. Berlanjut ke alur normal langkah 3

	<p>A2. Pengguna memilih menyunting <i>inquiry</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form inquiry</i> 2. Pengguna menyunting data <i>inquiry</i> 3. Sistem memperbaharui data <i>inquiry</i> 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus <i>inquiry</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data <i>inquiry</i> 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data <i>inquiry</i> 4. Berlanjut ke alur normal langkah 3
<p>Kondisi Akhir</p>	<p>Sistem menampilkan daftar <i>inquiry</i></p>



Gambar 3.14. Diagram Aktifitas Mengelola *Inquiry*

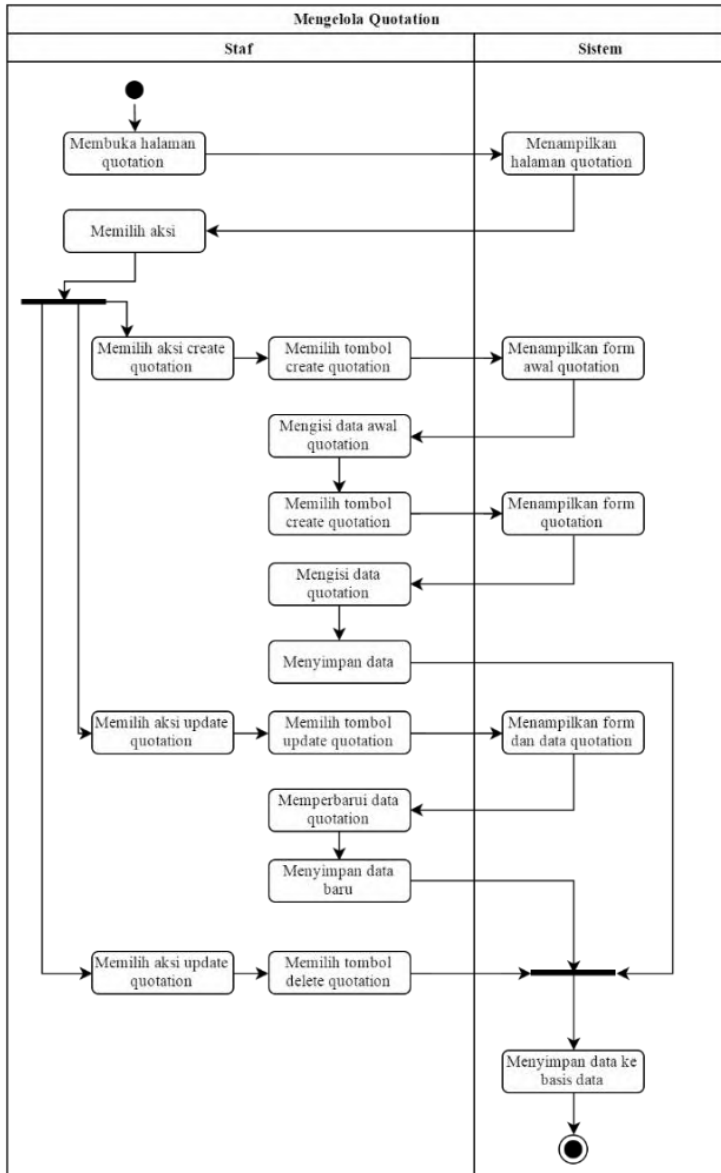
3.5.5 Kasus Penggunaan Mengelola *Quotation*

Pada kasus penggunaan ini, sistem menampilkan halaman untuk melakukan pembuatan *quotation*. Pengguna dapat melakukan pembuatan *quotation* dengan mengisi *form* yang tersedia. Pengguna dapat menambahkan informasi barang yang akan dibeli oleh pelanggan, pajak yang akan dibebankan pada pelanggan, transportasi yang digunakan untuk mengirimkan barang ke pelanggan, dan informasi potongan harga maupun tambahan harga untuk barang yang akan dipesan. Kasus penggunaan mengelola *quotation* ditunjukkan pada Tabel 3.10 dan Gambar 3.15.

Tabel 3.10. Kasus Penggunaan Mengelola *Quotation*

Nama	Mengelola <i>quotation</i>
Kode	UC-005
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data <i>quotation</i>
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu quotation</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>quotation</i> 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah <i>quotation</i> baru A2. Pengguna memilih menyunting <i>quotation</i> tertentu A3. Pengguna memilih menghapus <i>quotation</i> tertentu 3. Kasus penggunaan berakhir

Alur Alternatif	<p>A1. Pengguna memilih menambah <i>quotation</i> baru</p> <ol style="list-style-type: none"> 1. Sistem menampilkan kotak dialog <i>quotation</i> 2. Pengguna memasukkan jenis <i>quotation</i> dan memilih tombol <i>create</i> 3. Sistem menampilkan <i>form quotation</i> 4. Pengguna memasukkan data <i>quotation</i> baru 5. Sistem menyimpan data <i>quotation</i> 6. Berlanjut ke alur normal langkah 3 <p>A2. Pengguna memilih menyunting <i>quotation</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form quotation</i> 2. Pengguna menyunting data <i>quotation</i> 3. Sistem memperbaharui data <i>quotation</i> 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus <i>quotation</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data <i>quotation</i> 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data <i>quotation</i> 4. Berlanjut ke alur normal langkah 3
Kondisi Akhir	Sistem menampilkan daftar <i>quotation</i>



Gambar 3.15. Diagram Aktifitas Mengelola *Quotation*

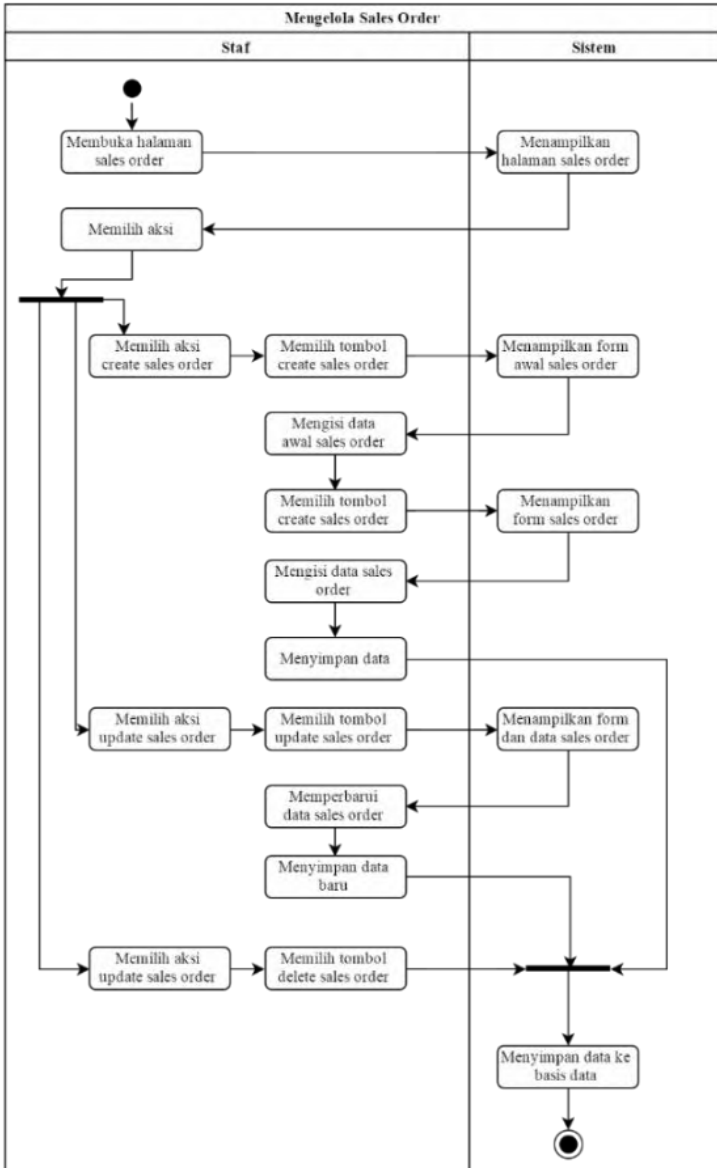
3.5.6 Kasus Penggunaan Mengelola *Sales Order*

Pada kasus penggunaan ini, sistem menampilkan halaman untuk melakukan pembuatan *sales order*. Pengguna dapat melakukan pembuatan *sales order* dengan mengisi *form* yang tersedia. Pengguna dapat menambahkan informasi barang yang akan dibeli oleh pelanggan, pajak yang akan dibebankan pada pelanggan, transportasi yang digunakan untuk mengirimkan barang ke pelanggan, dan informasi potongan harga maupun tambahan harga untuk barang yang akan dipesan. Kasus penggunaan mengelola *sales order* ditunjukkan pada Tabel 3.11 dan Gambar 3.16.

Tabel 3.11. Kasus Penggunaan Mengelola *Sales Order*

Nama	Mengelola <i>sales order</i>
Kode	UC-006
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data <i>sales order</i>
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu sales order</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>sales order</i> 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah <i>sales order</i> baru A2. Pengguna memilih menyunting <i>sales order</i> tertentu A3. Pengguna memilih menghapus <i>sales order</i> tertentu 3. Kasus penggunaan berakhir

Alur Alternatif	<p>A1. Pengguna memilih menambah <i>sales order</i> baru</p> <ol style="list-style-type: none"> 1. Sistem menampilkan kotak dialog <i>sales order</i> 2. Pengguna memasukkan jenis <i>sales order</i> dan memilih tombol <i>create</i> 3. Sistem menampilkan <i>form sales order</i> 4. Pengguna memasukkan data <i>sales order</i> baru 5. Sistem menyimpan data <i>sales order</i> 6. Berlanjut ke alur normal langkah 3 <p>A2. Pengguna memilih menyunting <i>sales order</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form sales order</i> 2. Pengguna menyunting data <i>sales order</i> 3. Sistem memperbaharui data <i>sales order</i> 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus <i>sales order</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data <i>sales order</i> 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data <i>sales order</i> 4. Berlanjut ke alur normal langkah 3
Kondisi Akhir	Sistem menampilkan daftar <i>sales order</i>



Gambar 3.16. Diagram Aktifitas Mengelola Sales Order

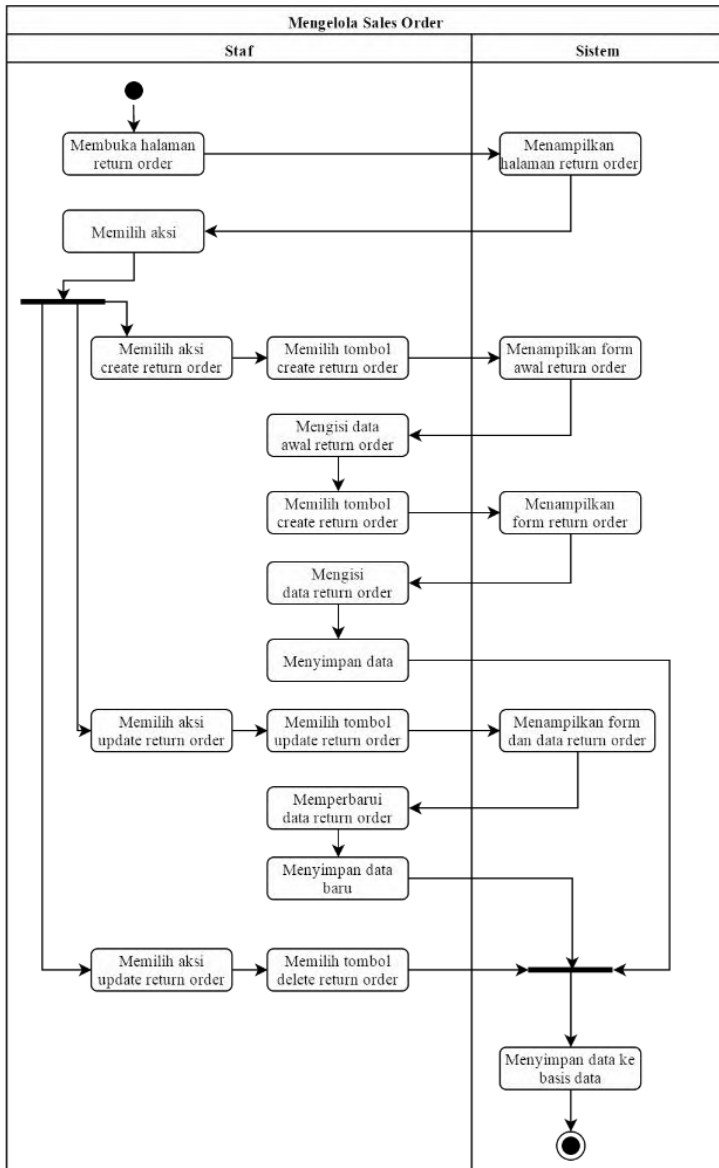
3.5.7 Kasus Penggunaan Mengelola *Return Order*

Pada kasus penggunaan ini, sistem menampilkan halaman untuk melakukan pembuatan *return order*. Pengguna dapat melakukan pembuatan *return order* dengan mengisi *form* yang tersedia. Pengguna dapat menambahkan informasi barang yang akan dibeli oleh pelanggan, pajak yang akan dibebankan pada pelanggan, transportasi yang digunakan untuk mengirimkan barang ke pelanggan, dan informasi potongan harga maupun pemberian biaya pengiriman secara gratis kepada pelanggan untuk pengembalian barang yang dikembalikan. Kasus penggunaan mengelola *return order* ditunjukkan pada Tabel 3.12 dan Gambar 3.17.

Tabel 3.12. Kasus Penggunaan Mengelola *Return Order*

Nama	Mengelola <i>return order</i>
Kode	UC-007
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data <i>return order</i>
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu return order</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>return order</i> 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah <i>return order</i> baru A2. Pengguna memilih menyunting <i>return order</i> tertentu A3. Pengguna memilih menghapus <i>return order</i> tertentu

	3. Kasus penggunaan berakhir
Alur Alternatif	<p>A1. Pengguna memilih menambah <i>return order</i> baru</p> <ol style="list-style-type: none"> 1. Sistem menampilkan kotak dialog <i>return order</i> 2. Pengguna memasukkan jenis <i>return order</i> dan memilih tombol <i>create</i> 3. Sistem menampilkan <i>form return order</i> 4. Pengguna memasukkan data <i>return order</i> baru 5. Sistem menyimpan data <i>return order</i> 6. Berlanjut ke alur normal langkah 3 <p>A2. Pengguna memilih menyunting <i>return order</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form return order</i> 2. Pengguna menyunting data <i>return order</i> 3. Sistem memperbaharui data <i>return order</i> 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus <i>return order</i> tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data <i>return order</i> 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data <i>return order</i> 4. Berlanjut ke alur normal langkah 3
Kondisi Akhir	Sistem menampilkan daftar <i>return order</i>



Gambar 3.17. Diagram Aktifitas Mengelola Return Order

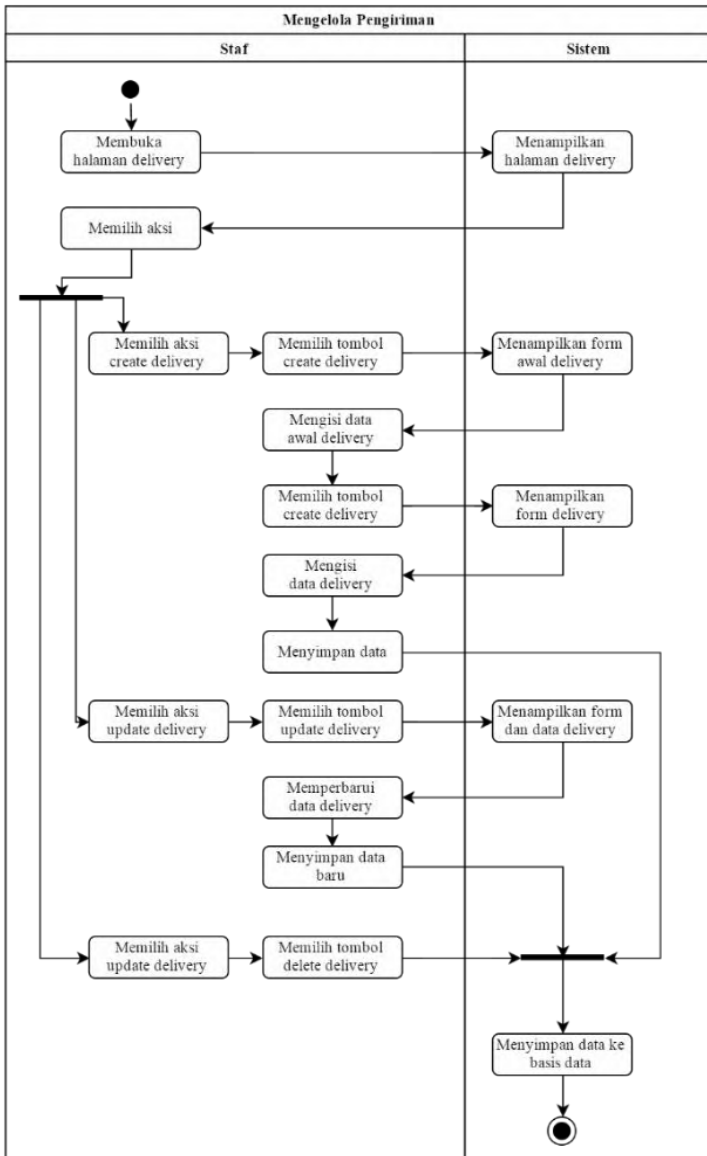
3.5.8 Kasus Penggunaan Mengelola Pengiriman

Pada kasus penggunaan ini, sistem menampilkan halaman untuk melakukan proses pengiriman barang. Pengguna dapat melakukan proses pengiriman barang dengan mengisi *form* yang tersedia berdasarkan *sales order* yang telah dibuat sebelumnya. Pengguna dapat menambahkan transportasi yang digunakan untuk mengirimkan barang ke pelanggan, dan mengubah lokasi pengiriman barang ke pelanggan. Kasus penggunaan mengelola pengiriman ditunjukkan pada Tabel 3.13 dan Gambar 3.18.

Tabel 3.13. Kasus Penggunaan Mengelola Pengiriman

Nama	Mengelola pengiriman
Kode	UC-008
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data pengiriman
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu shipping and delivery</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar pengiriman 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah pengiriman baru A2. Pengguna memilih menyunting pengiriman tertentu A3. Pengguna memilih menghapus pengiriman tertentu 3. Kasus penggunaan berakhir

Alur Alternatif	<p>A1. Pengguna memilih menambah pengiriman baru</p> <ol style="list-style-type: none"> 1. Sistem menampilkan kotak dialog <i>shipping and delivery</i> 2. Pengguna memasukkan kode <i>sales order</i> yang akan dikirim dan memilih tombol <i>create</i> 3. Sistem menampilkan <i>form shipping and delivery</i> 4. Pengguna memasukkan data pengiriman baru 5. Sistem menyimpan data pengiriman 6. Berlanjut ke alur normal langkah 3 <p>A2. Pengguna memilih menyunting pengiriman tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form shipping and delivery</i> 2. Pengguna menyunting data pengiriman 3. Sistem memperbaharui data pengiriman 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus pengiriman tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data pengiriman 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data pengiriman 4. Berlanjut ke alur normal langkah 3
Kondisi Akhir	Sistem menampilkan daftar pengiriman



Gambar 3.18. Diagram Aktifitas Mengelola Pengiriman

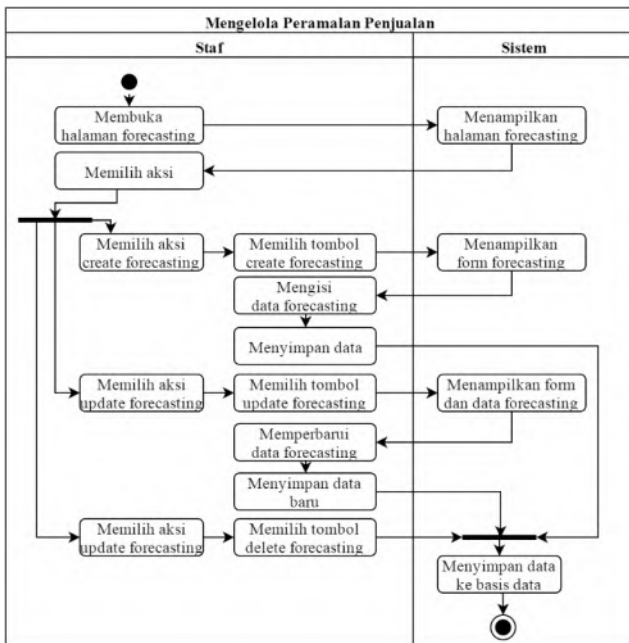
3.5.9 Kasus Penggunaan Mengelola Peramalan penjualan

Pada kasus penggunaan ini, sistem menampilkan halaman peramalan. Pengguna dapat melakukan penambahan peramalan dengan mengisi *form* yang tersedia. Kasus penggunaan mengelola peramalan ditunjukkan pada Tabel 3.14 dan Gambar 3.19.

Tabel 3.14. Kasus Penggunaan Mengelola Peramalan Penjualan

Nama	Mengelola peramalan penjualan
Kode	UC-009
Deskripsi	Kasus penggunaan ini digunakan untuk menambah, menyunting, dan menghapus data peramalan penjualan
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu forecasting</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar peramalan penjualan 2. Pengguna memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Pengguna memilih menambah peramalan penjualan baru A2. Pengguna memilih menyunting peramalan penjualan tertentu A3. Pengguna memilih menghapus peramalan penjualan tertentu 3. Kasus penggunaan berakhir
Alur Alternatif	<ol style="list-style-type: none"> A1. Pengguna memilih menambah peramalan penjualan baru <ol style="list-style-type: none"> 1. Sistem menampilkan <i>form forecasting</i> 2. Pengguna memasukkan data peramalan penjualan baru 3. Sistem menyimpan data peramalan penjualan 4. Berlanjut ke alur normal langkah 3 A2. Pengguna memilih menyunting peramalan penjualan tertentu

	<ol style="list-style-type: none"> 1. Sistem menampilkan <i>form forecasting</i> 2. Pengguna menyunting data peramalan penjualan 3. Sistem memperbaharui data peramalan penjualan 4. Berlanjut ke alur normal langkah 3 <p>A3. Pengguna memilih menghapus peramalan penjualan tertentu</p> <ol style="list-style-type: none"> 1. Sistem menampilkan konfirmasi penghapusan data peramalan penjualan 2. Pengguna menyetujui penghapusan data 3. Sistem menghapus data peramalan penjualan 4. Berlanjut ke alur normal langkah 3
<p>Kondisi Akhir</p>	<p>Sistem menampilkan daftar peramalan penjualan</p>



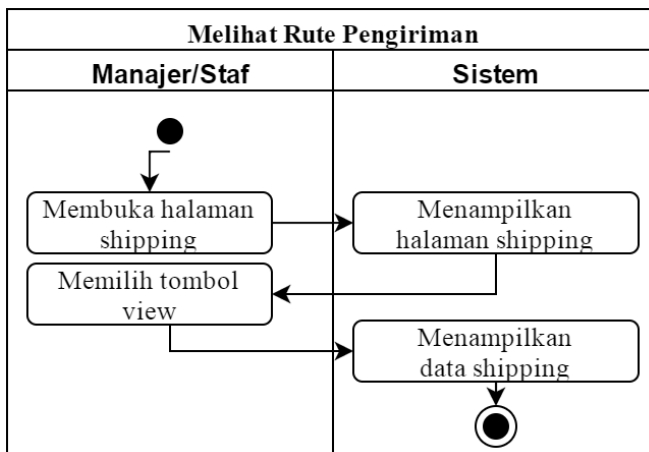
Gambar 3.19. Diagram Aktifitas Mengelola Peramalan Penjualan

3.5.10 Kasus Penggunaan Melihat Rute Pengiriman

Pada kasus penggunaan ini, sistem menampilkan halaman *shipping*. Pengguna dapat melihat detail dari rute pengiriman. Kasus penggunaan melihat rute pengiriman ditunjukkan pada Tabel 3.15 dan Gambar 3.20.

Tabel 3.15. Kasus Penggunaan Melihat Rute Pengiriman

Nama	Melihat rute pengiriman
Kode	UC-010
Deskripsi	Kasus penggunaan ini digunakan untuk melihat rute pengiriman tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu shipping</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar rute pengiriman 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari rute pengiriman yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail rute pengiriman ditampilkan



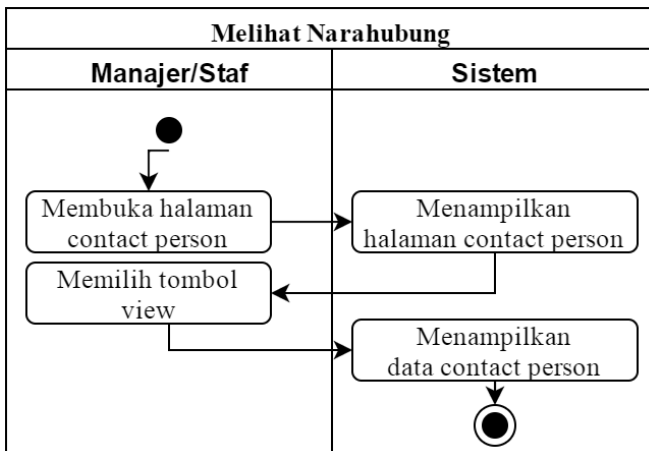
Gambar 3.20. Diagram Aktifitas Melihat Rute Pengiriman

3.5.11 Kasus Penggunaan Melihat Narahubung

Pada kasus penggunaan ini, sistem menampilkan halaman *contact person*. Pengguna dapat melihat detail dari narahubung. Kasus penggunaan melihat narahubung ditunjukkan pada Tabel 3.16 dan Gambar 3.21.

Tabel 3.16. Kasus Penggunaan Melihat Narahubung

Nama	Melihat narahubung
Kode	UC-011
Deskripsi	Kasus penggunaan ini digunakan untuk melihat narahubung tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu contact person</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar narahubung 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari narahubung yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail narahubung ditampilkan



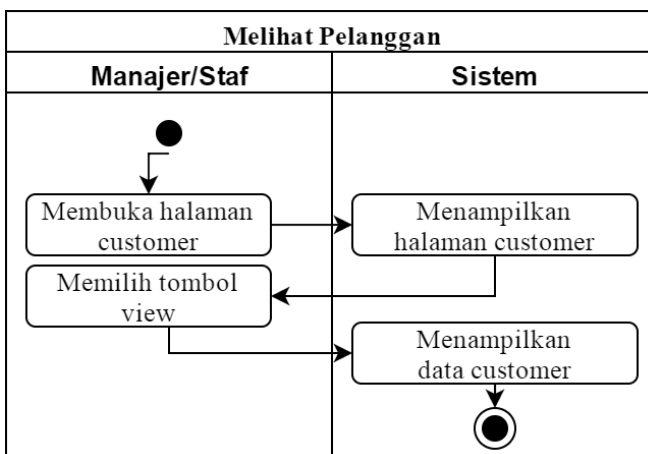
Gambar 3.21. Diagram Aktifitas Melihat Narahubung

3.5.12 Kasus Penggunaan Melihat Pelanggan

Pada kasus penggunaan ini, sistem menampilkan halaman pelanggan. Pengguna dapat melihat detail dari pelanggan. Kasus penggunaan melihat pelanggan ditunjukkan pada Tabel 3.17 dan Gambar 3.22.

Tabel 3.17. Kasus Penggunaan Melihat Pelanggan

Nama	Melihat pelanggan
Kode	UC-012
Deskripsi	Kasus penggunaan ini digunakan untuk melihat pelanggan tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu customer</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar pelanggan 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari pelanggan yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail pelanggan ditampilkan



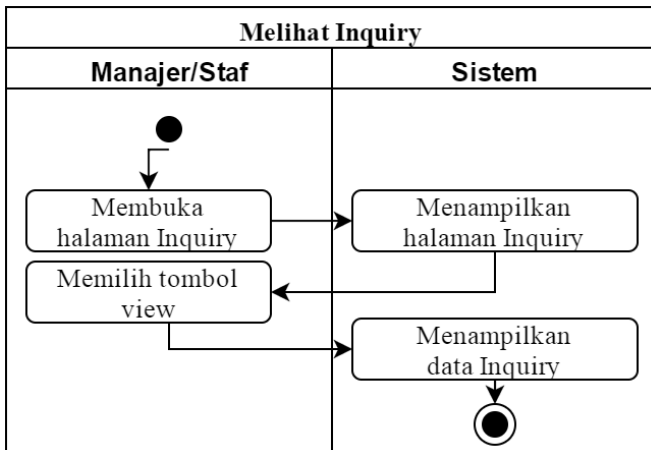
Gambar 3.22. Diagram Aktifitas Melihat Pelanggan

3.5.13 Kasus Penggunaan Melihat *Inquiry*

Pada kasus penggunaan ini, sistem menampilkan halaman *inquiry*. Pengguna dapat melihat detail dari *inquiry*. Kasus penggunaan melihat *inquiry* ditunjukkan pada Tabel 3.18 dan Gambar 3.23.

Tabel 3.18. Kasus Penggunaan Melihat *Inquiry*

Nama	Melihat <i>inquiry</i>
Kode	UC-013
Deskripsi	Kasus penggunaan ini digunakan untuk melihat <i>inquiry</i> tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu inquiry</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>inquiry</i> 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari <i>inquiry</i> yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail <i>inquiry</i> ditampilkan



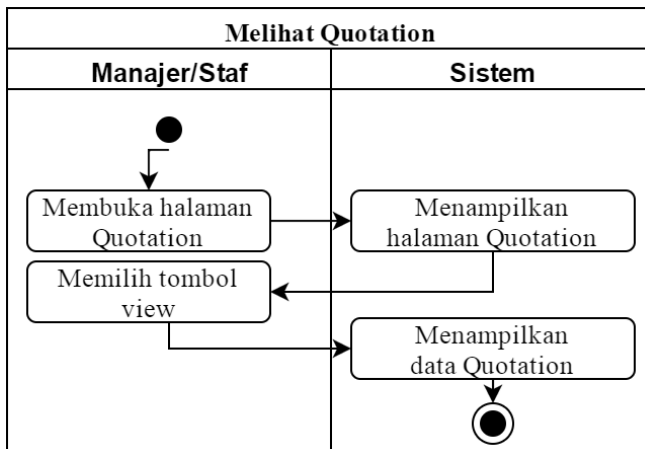
Gambar 3.23. Diagram Aktifitas Melihat *Inquiry*

3.5.14 Kasus Penggunaan Melihat *Quotation*

Pada kasus penggunaan ini, sistem menampilkan halaman *quotation*. Pengguna dapat melihat detail dari *quotation*. Kasus penggunaan melihat *quotation* ditunjukkan pada Tabel 3.19 dan Gambar 3.24.

Tabel 3.19. Kasus Penggunaan Melihat *Quotation*

Nama	Melihat <i>quotation</i>
Kode	UC-014
Deskripsi	Kasus penggunaan ini digunakan untuk melihat <i>quotation</i> tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu quotation</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>quotation</i> 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari <i>quotation</i> yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail <i>quotation</i> ditampilkan



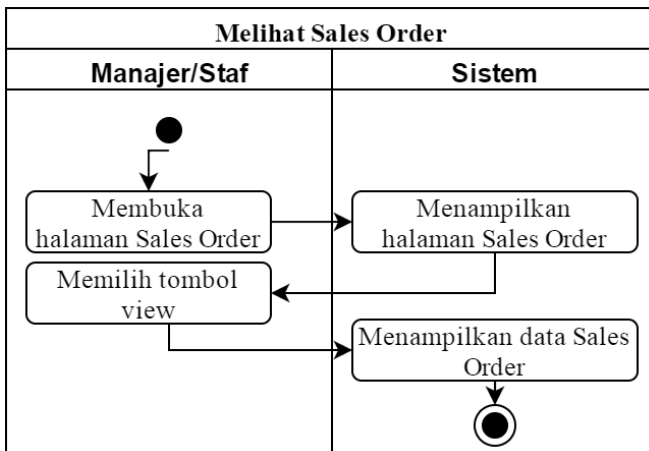
Gambar 3.24. Diagram Aktifitas Melihat *Quotation*

3.5.15 Kasus Penggunaan Melihat *Sales order*

Pada kasus penggunaan ini, sistem menampilkan halaman *sales order*. Pengguna dapat melihat detail dari *sales order*. Kasus penggunaan melihat *sales order* ditunjukkan pada Tabel 3.20 dan Gambar 3.25.

Tabel 3.20. Kasus Penggunaan Melihat *Sales Order*

Nama	Melihat <i>sales order</i>
Kode	UC-015
Deskripsi	Kasus penggunaan ini digunakan untuk melihat <i>sales order</i> tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu sales order</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>sales order</i> 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari <i>sales order</i> yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail <i>sales order</i> ditampilkan



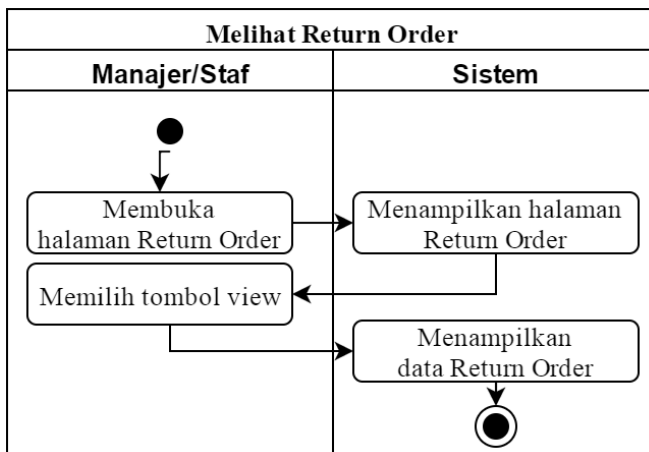
Gambar 3.25. Diagram Aktifitas Melihat *Sales Order*

3.5.16 Kasus Penggunaan Melihat *Return order*

Pada kasus penggunaan ini, sistem menampilkan halaman *return order*. Pengguna dapat melihat detail dari *return order*. Kasus penggunaan melihat *return order* ditunjukkan pada Tabel 3.21 dan Gambar 3.26.

Tabel 3.21. Kasus Penggunaan Melihat *Return Order*

Nama	Melihat <i>return order</i>
Kode	UC-016
Deskripsi	Kasus penggunaan ini digunakan untuk melihat <i>return order</i> tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu return order</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar <i>return order</i> 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari <i>return order</i> yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail <i>return order</i> ditampilkan



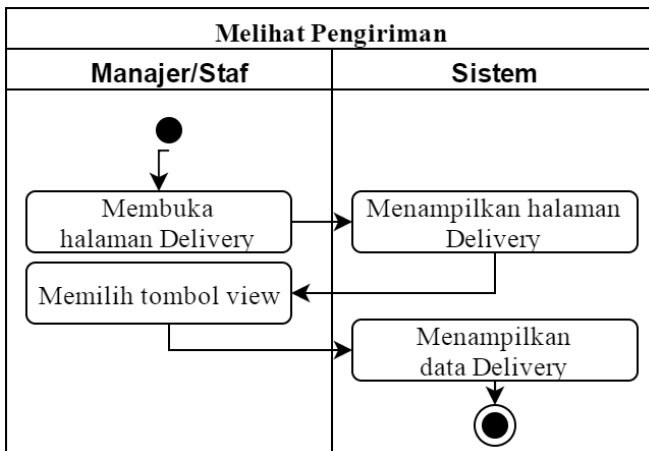
Gambar 3.26. Diagram Aktifitas Melihat *Return Order*

3.5.17 Kasus Penggunaan Melihat Pengiriman

Pada kasus penggunaan ini, sistem menampilkan halaman *shipping and delivery*. Pengguna dapat melihat detail dari pengiriman. Kasus penggunaan melihat pengiriman ditunjukkan pada Tabel 3.22 dan Gambar 3.27.

Tabel 3.22. Kasus Penggunaan Melihat Pengiriman

Nama	Melihat pengiriman
Kode	UC-017
Deskripsi	Kasus penggunaan ini digunakan untuk melihat pengiriman tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu shipping and delivery</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar pengiriman 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari pengiriman yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail pengiriman ditampilkan



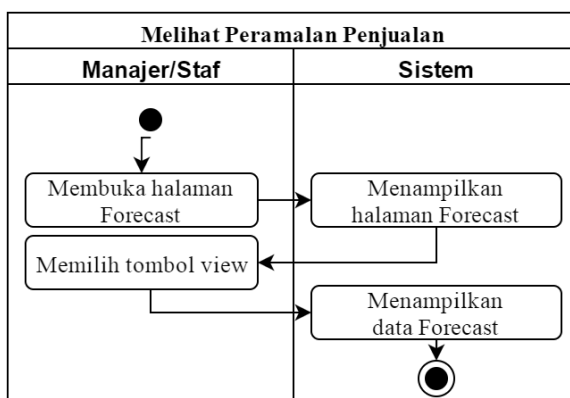
Gambar 3.27. Diagram Aktifitas Melihat Pengiriman

3.5.18 Kasus Penggunaan Melihat Peramalan penjualan

Pada kasus penggunaan ini, sistem menampilkan halaman *forecasting*. Pengguna dapat melihat detail dari peramalan penjualan. Kasus penggunaan melihat peramalan penjualan ditunjukkan pada Tabel 3.23 dan Gambar 3.28.

Tabel 3.23. Kasus Penggunaan Melihat Peramalan Penjualan

Nama	Melihat peramalan penjualan
Kode	UC-018
Deskripsi	Kasus penggunaan ini digunakan untuk melihat peramalan penjualan tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu forecasting</i>
Alur Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan daftar peramalan penjualan 2. Pengguna memilih tombol <i>view</i> 3. Sistem menampilkan halaman dan data dari peramalan penjualan yang dipilih
Alur Alternatif	-
Kondisi Akhir	Halaman detail peramalan penjualan ditampilkan



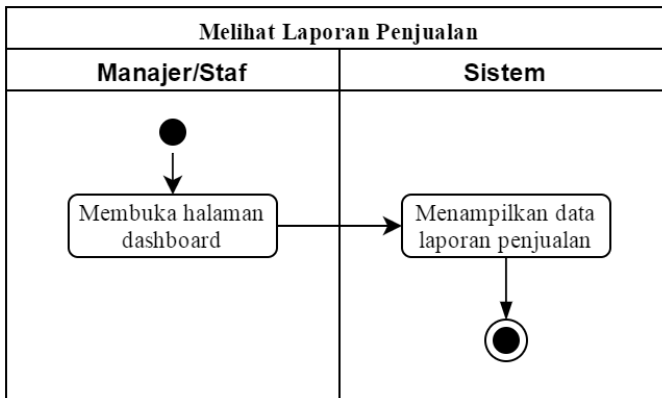
Gambar 3.28. Diagram Aktifitas Melihat Peramalan Penjualan

3.5.19 Kasus Penggunaan Melihat Laporan Penjualan

Pada kasus penggunaan ini, sistem menampilkan halaman utama modul. Pengguna dapat melihat laporan penjualan. Kasus penggunaan melihat laporan penjualan ditunjukkan pada Tabel 3.24 dan Gambar 3.29.

Tabel 3.24. Kasus Penggunaan Melihat Laporan Penjualan

Nama	Melihat laporan penjualan
Kode	UC-019
Deskripsi	Kasus penggunaan ini digunakan untuk melihat laporan penjualan tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Sales and Distribution</i>
Alur Normal	1. Sistem menampilkan data laporan penjualan
Alur Alternatif	-
Kondisi Akhir	Halaman detail laporan penjualan ditampilkan



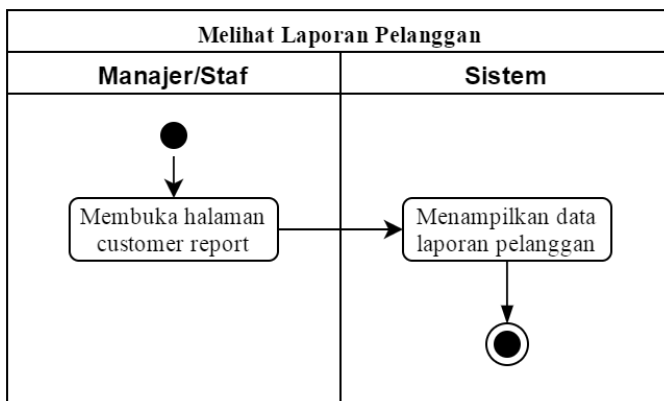
Gambar 3.29. Diagram Aktifitas Melihat Laporan Penjualan

3.5.20 Kasus Penggunaan Melihat Laporan Pelanggan

Pada kasus penggunaan ini, sistem menampilkan halaman *customer report*. Pengguna dapat melihat laporan pelanggan. Kasus penggunaan melihat laporan pelanggan ditunjukkan pada Tabel 3.25 dan Gambar 3.30.

Tabel 3.25. Kasus Penggunaan Melihat Laporan Pelanggan

Nama	Melihat laporan pelanggan
Kode	UC-020
Deskripsi	Kasus penggunaan ini digunakan untuk melihat laporan pelanggan tertentu
Tipe	Fungsional
Aktor	Staf <i>Sales and Distribution</i> dan manager <i>Sales and Distribution</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu customer report</i>
Alur Normal	1. Sistem menampilkan data laporan pelanggan
Alur Alternatif	-
Kondisi Akhir	Halaman detail laporan pelanggan ditampilkan



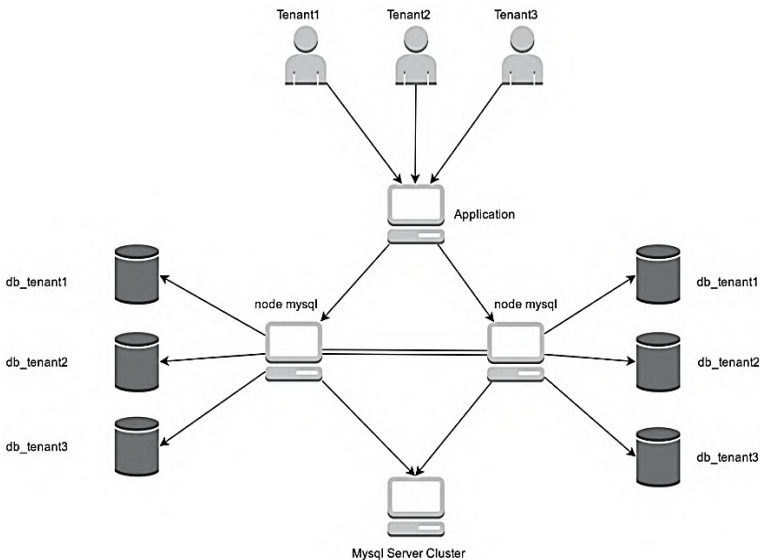
Gambar 3.30. Diagram Aktifitas Melihat Laporan Pelanggan

3.6 Perancangan

Tahap perancangan perangkat lunak dibagi menjadi beberapa bagian yaitu perancangan *multitenancy*, perancangan basis data, perancangan RBAC, dan perancangan antarmuka sistem.

3.6.1 Perancangan *Multitenancy*

Pada bagian ini akan dijelaskan mengenai perancangan *multitenancy* yang terdapat pada sistem ERP 2016. Secara rinci mengenai *multitenancy* dijabarkan pada Gambar 3.31.



Gambar 3.31. Perancangan *Multitenancy*

Gambar 3.31 menunjukkan alur *multitenancy* yang terjadi pada sistem ERP. Pembagian *node MySQL* yang telah dijelaskan pada subbab 3.2.1, yaitu dengan cara melakukan *clustering* pada *computer server* menjadi 2 basis data yang berbeda. Basis data terdistribusi merupakan penunjang dari *multitenancy*, yang

berfungsi untuk mengatur *session* setiap perusahaan yang akan mengakses sistem ERP.

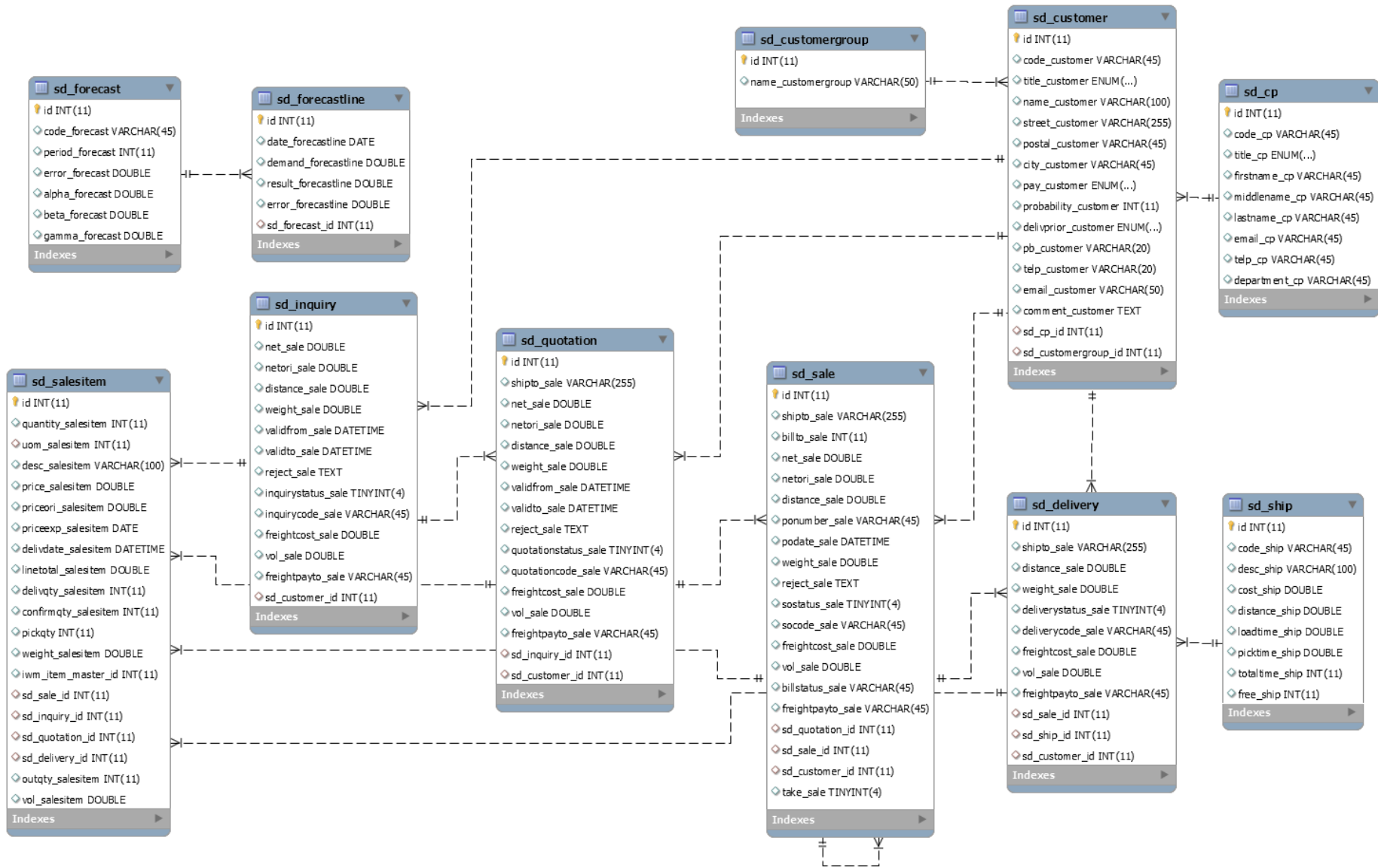
3.6.2 Perancangan Basis Data

Pada Subbab ini akan dijelaskan bagaimana rancangan basis data yang digunakan pada sistem. Basis data pada sistem ini menggunakan sistem manajemen basis data *MySQL Cluster*. *Physical Data Model* (PDM) dari basis data sistem ini ditunjukkan pada Gambar 3.32. Perancangan PDM pada Gambar 3.32 dijelaskan pada tabel 3.26.

Tabel 3.26. Keterangan PDM

No	Tabel	Atribut	Keterangan
1	sd_custom ergroup	id : <i>integer</i>	Tabel untuk menyimpan data grup pelanggan
		name_customergroup : <i>varchar</i>	
2	sd_cp	id : <i>integer</i>	Tabel untuk menyimpan data narahubung
		code_cp : <i>varchar</i>	
		title_cp : <i>enum</i>	
		firstname_cp : <i>varchar</i>	
		middlename_cp : <i>varchar</i>	
		lastname_cp : <i>varchar</i>	
		email_cp : <i>varchar</i>	
		telp_cp : <i>varchar</i>	
3	sd_custom er	id : <i>integer</i>	Tabel untuk menyimpan data pelanggan
		code_customer : <i>varchar</i>	
		title_customer : <i>enum</i>	
		name_customer : <i>varchar</i>	
		street_customer : <i>varchar</i>	
		postal_customer : <i>varchar</i>	
		city_customer : <i>varchar</i>	
		pay_customer : <i>enum</i>	
		probability_customer : <i>int</i>	
		delivprior_customer : <i>enum</i>	
	pb_customer : <i>varchar</i>		

No	Tabel	Atribut	Keterangan	
		telp_customer : <i>varchar</i>		
		email_customer : <i>varchar</i>		
		comment_customer : <i>text</i>		
4	sd_inquiry, sd_quotation, sd_sale, sd_delivery	id : <i>integer</i>	Tabel untuk menyimpan data <i>inquiry, quotation, sales order, dan delivery</i>	
		net_sale : <i>double</i>		
		netori_sale : <i>double</i>		
		distance_sale : <i>double</i>		
		weight_sale : <i>double</i>		
		validfrom_sale : <i>datetime</i>		
		validto_sale : <i>datetime</i>		
		reject_sale : <i>text</i>		
		inquirystatus_sale : <i>integer</i>		
		inquirycode_sale : <i>varchar</i>		
		freightcost_sale : <i>double</i>		
		vol_sale : <i>double</i>		
		freightpayto_sale : <i>varchar</i>		
		shipto_sale : <i>varchar</i>		
		quotationstatus_sale : <i>integer</i>		
		quotationcode_sale : <i>varchar</i>		
		billto_sale : <i>integer</i>		
		ponumber_sale : <i>varchar</i>		
		podate_sale : <i>datetime</i>		
		sostatus_sale : <i>integer</i>		
		socode_sale : <i>varchar</i>		
		billstatus_sale : <i>varchar</i>		
		take_sale : <i>integer</i>		
		deliverystatus_sale : <i>integer</i>		
		deliverycode_sale : <i>varchar</i>		



Gambar 3.32. Physical Data Model Modul Sales and Distribution

[Halaman ini sengaja dikosongkan]

No	Tabel	Atribut	Keterangan
5	sd_salesitem	id : <i>integer</i>	Tabel untuk menyimpan data barang yang dipesan oleh pelanggan
		quantity_salesitem : <i>integer</i>	
		uom_salesitem : <i>integer</i>	
		desc_salesitem : <i>varchar</i>	
		price_salesitem : <i>double</i>	
		priceori_salesitem : <i>double</i>	
		priceexp_salesitem : <i>date</i>	
		delivdate_salesitem : <i>datetime</i>	
		linetotal_salesitem : <i>double</i>	
		delivqty_salesitem : <i>integer</i>	
		confirmqty_salesitem : <i>integer</i>	
		pickqty : <i>integer</i>	
		weight_salesitem : <i>double</i>	
outqty_salesitem : <i>integer</i>			
6	sd_ship	id : <i>integer</i>	Tabel untuk menyimpan data pengiriman
		code_ship : <i>varchar</i>	
		desc_ship : <i>varchar</i>	
		cost_ship : <i>double</i>	
		distance_ship : <i>double</i>	
		loadtime_ship : <i>double</i>	
		picktime_ship : <i>double</i>	
		totaltime_ship : <i>integer</i>	
		free_ship : <i>integer</i>	
7	sd_forecast	id : <i>integer</i>	Tabel untuk menyimpan data <i>header</i> peramalan penjualan
		code_forecast : <i>varchar</i>	
		period_forecast : <i>integer</i>	
		error_forecast : <i>double</i>	
		alpha_forecast : <i>double</i>	
		beta_forecast : <i>double</i>	
gamma_forecast : <i>double</i>			

No	Tabel	Atribut	Keterangan
8	sd_forecast line	id : <i>integer</i>	Tabel untuk menyimpan data <i>line</i> peramalan penjualan
		date_forecastline : <i>date</i>	
		demand_forecastline : <i>double</i>	
		result_forecastline : <i>double</i>	
		error_forecastline : <i>double</i>	

3.6.3 Perancangan *Role Based Access Control* (RBAC)

Pada bagian ini akan dijelaskan mengenai perancangan RBAC yang terdapat pada sistem ERP 2016. Perincian mengenai RBAC dijabarkan sebagai berikut.

3.6.3.1 Perancangan Antarmuka *Login*

Rancangan antarmuka *login* dibagi menjadi 3 (tiga) bagian, di bagian paling atas terdapat *header* yang berisi logo aplikasi dan tombol *login*, di bagian atas terdapat navigasi (*breadcrumb*) untuk mengetahui pengguna sedang berada pada halaman tertentu, dan di bagian tengah terdapat *form login* yang memiliki *username*, *password*, dan pemilihan basis data yang digunakan sebagai *input*. Perancangan antarmuka *login* ditunjukkan pada gambar 3.33.

The image shows a login form with the following elements:

- Title: Login
- Username: [Text input field]
- Password: [Text input field]
- Select DB: [Dropdown menu]
- Login: [Blue button]

Gambar 3.33. Rancang Antarmuka *Login*

3.6.3.2 Perancangan Antarmuka Menambahkan Pengguna Baru

Rancangan antarmuka menambahkan pengguna baru ditampilkan dalam sebuah kotak dialog yang berisi *username*, *email*, dan *password* yang digunakan sebagai *input* data untuk menambahkan pengguna baru. Perancangan antarmuka menambahkan pengguna baru ditunjukkan pada Gambar 3.34.



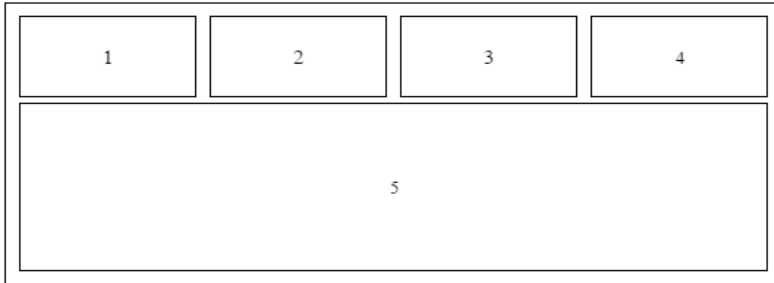
Gambar 3.34. Rancang Antarmuka Menambahkan Pengguna Baru

3.6.4 Perancangan Antarmuka Sistem

Pada bagian ini, akan dijelaskan mengenai perancangan antarmuka yang terdapat pada modul *Sales and Distribution* di sistem ERP 2016. Perincian mengenai perancangan antarmuka *Sales and Distribution* dijabarkan sebagai berikut.

3.6.4.1 Perancangan Antarmuka Halaman Utama Modul

Antarmuka halaman utama submodul dirancang dengan tujuan untuk mempermudah melihat laporan dan informasi lainnya mengenai *Sales and Distribution*. Rancang antarmuka halaman utama modul ditunjukkan pada Gambar 3.35.



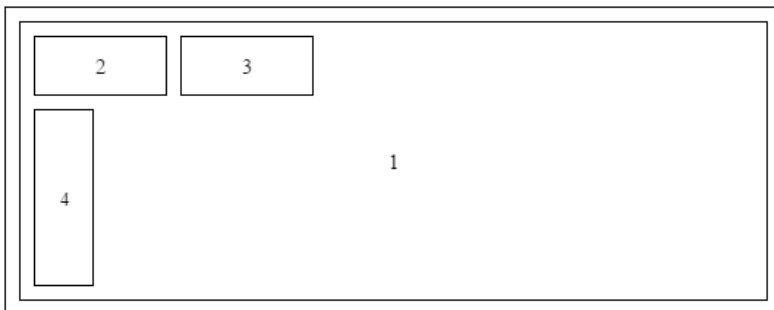
Gambar 3.35. Rancang Antarmuka Halaman Utama Modul

Komponen-komponen yang terdapat pada Gambar 3.35 adalah sebagai berikut:

1. Jumlah pelanggan yang terdaftar di ERP
2. Jumlah *sales order* yang dibuat
3. Jumlah *return order* yang dibuat
4. Total pendapatan penjualan bulanan

3.6.4.2 Perancangan Antarmuka Daftar Data Submodul

Antarmuka daftar data submodul dirancang dengan tujuan untuk mempermudah melihat daftar data yang terdapat pada masing-masing submodul di dalam modul *Sales and Distribution*. Perancangan antarmuka daftar data submodul ditunjukkan pada Gambar 3.36.



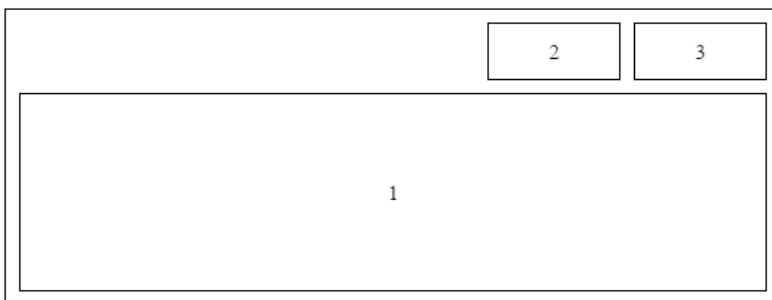
Gambar 3.36. Rancang Antarmuka Daftar Data Submodul

Komponen-komponen yang terdapat pada Gambar 3.36 adalah sebagai berikut:

1. Daftar data submodul
2. Tombol *Create*
3. Tombol *Search*
4. Tombol *View, Update, dan Delete*.

3.6.4.3 Perancangan Antarmuka Detail Data Submodul

Antarmuka detail data submodul dirancang dengan tujuan untuk mempermudah melihat detail informasi yang terdapat pada masing-masing submodul di dalam modul *Sales and Distribution*. Rancang antarmuka detail data submodul ditunjukkan pada Gambar 3.37.



Gambar 3.37. Rancang Antarmuka Detail Data Submodul

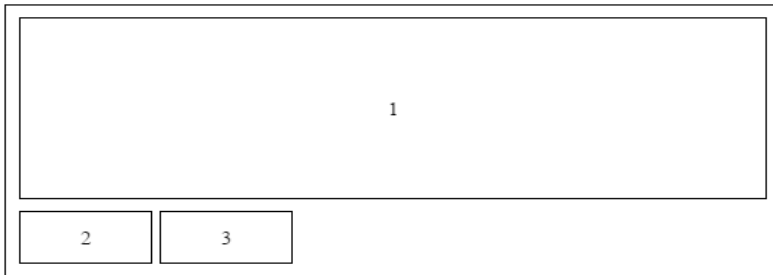
Komponen-komponen yang terdapat pada Gambar 3.37 adalah sebagai berikut:

1. Detail data submodul
2. Tombol *Update*
3. Tombol *Delete*

3.6.4.4 Perancangan Antarmuka Menambah dan Menyunting Data Submodul

Antarmuka menambah dan menyunting data submodul dirancang dengan tujuan untuk menambah dan menyunting data

submodul. Rancang antarmuka menambah dan menyunting data submodul ditunjukkan pada Gambar 3.38.



The image shows a wireframe of a user interface for adding and editing sub-module data. It consists of a large rectangular area labeled '1' at the top, which is the data form. Below this form are two smaller rectangular buttons labeled '2' and '3', representing the 'Create' and 'Cancel' actions respectively.

Gambar 3.38. Rancang Antarmuka Menambah dan Menyunting Data Submodul

Komponen-komponen yang terdapat pada Gambar 3.38 adalah sebagai berikut:

1. Form data submodul
2. Tombol *Create*
3. Tombol *Cancel*.

BAB IV

IMPLEMENTASI SISTEM

Pada bab ini akan dibahas implementasi dari perancangan sistem ERP 2016. Di dalamnya mencakup penjelasan lingkungan pengembangan sistem serta proses implementasi Basis Data Terdistribusi, RBAC, *multitenancy*, dan antarmuka pengguna.

4.1 Lingkungan Pengembangan Sistem

Lingkungan pengembangan sistem yang digunakan untuk mengembangkan tugas akhir ini dilakukan pada lingkungan dan kaskas sebagai berikut:

1. Basis data yang digunakan pada *server* adalah *MySQL Cluster*.
2. 2 PC untuk *server database* menggunakan Sistem Operasi Ubuntu 12.04.
3. PC untuk *server* menggunakan Intel® Core™ i3-2120 @3.30GHz , RAM 4GB dengan Sistem Operasi Windows 8.1 Enterprise x64.
4. Mozilla Firefox 46.0.1 dan Chrome 49.0 sebagai antarmuka untuk pengujian aplikasi klien.

4.2 Implementasi Basis Data Terdistribusi

Pada bagian ini akan dijelaskan mengenai implementasi basis data terdistribusi yang terdapat pada sistem ERP 2016.

4.2.1 Instalasi Data SQL Node pada Node 1 dan Node 2

Pada implementasi instalasi *SQL node* pada *node1* dan *node2* untuk implementasi Basis Data Terdistribusi, dilakukan langkah-langkah sebagai berikut.

1. Membuat grup di dalam *MySQL* sebagai pengguna baru, kemudian menambah *user* di dalam *MySQL*. Kode implementasi membuat grup di dalam *MySQL* dan menambah *user* baru yang dimaksud ditunjukkan oleh Kode Sumber 4.1.

```

1 | shell> groupadd mysql
2 | shell> useradd-g mysql mysql

```

Kode Sumber 4.1. Membuat Grup *MySQL* Pengguna Baru dan Menambah *User* Baru

2. Mengubah lokasi ke dalam direktori yang berisi *file* yang telah diunduh, kemudian mengubah arsip dan menciptakan *symlink* ke dalam direktori *mysql* yang bernama “mysql”. File yang sebenarnya dan nama direktori bervariasi sesuai dengan jumlah *cluster* versi *MySQL*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.2.

```

1 | shell> cd/usr/local
2 | shell> /usr/local$ tar xzvf mysql-cluster-gpl-
   | 7.1.34-linux-x86_64-glibc23
3 | shell> ln-s/usr/local/mysql-cluster-gpl-7.1.34
   | linux-x86-glibc23/usr/local/mysql
4 | shell> export PATH = $PATH:/usr/local/mysql/bin
5 | shell> echo "export PATH=\$PATH:/usr/local/
   | mysql/ bin">> /etc/bash.bashrc

```

Kode Sumber 4.2. Mengubah Lokasi Direktori, Mengubah Arsip, dan Menciptakan *Symlink*

3. Mengubah lokasi ke dalam direktori *MySQL* dan menjalankan *script* untuk menciptakan *database system*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.3.

```

1 | shell> cd mysql
2 | shell> ./scripts/mysql install db-user = mysql

```

Kode Sumber 4.3. Mengubah Lokasi Direktori

4. Mengatur izin yang diperlukan oleh *server MySQL*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.4.

```

1 | shell> chown-R root.
2 | shell> chown-R mysql data
3 | shell> chgrp-R mysql.

```

Kode Sumber 4.4. Mengatur Perizinan *Server MySQL*

5. Menyalin *script startup MySQL* ke direktori yang sesuai, mengubah menjadi *executable*, dan memulai ketika sistem beroperasi. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.5.

1	shell> cp support-files/mysql.server/etc/ init.d/mysql
2	shell> chmod + x/etc/init.d/mysql
3	shell> update-rc.d mysql defaults

Kode Sumber 4.5. Menyalin Script Startup Mysql

4.2.2 Pemasangan Node Manajemen Pada Node 3

Pemasangan *node* manajemen memerlukan manajemen *server MySQL Cluster* (*ndb_mgmd*). Untuk memasang *ndb_mgmd* dan *ndb_mgm* pada *host Cluster*, sistem sebagai *root* melakukan langkah-langkah sebagai berikut.

1. Mengubah lokasi ke dalam direktori */var/tmp* direktori, dan mengekstrak *ndb_mgm* dan *ndb_mgmd* dari arsip ke direktori yang sesuai seperti */usr/local/bin*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.6.

1	shell> cd/usr/local
2	shell> tar-zxvf mysql-cluster-gpl-7.1.34-linux-x86-glibc23.tar.gz
3	shell> cd/usr/local/mysql-cluster-gpl-7.1.34-linux-x86-glibc235
4	shell> cp bin / ndb mgm */usr/local/bin

Kode Sumber 4.6. Mengubah Lokasi Direktori

2. Mengubah lokasi ke dalam direktori tempat *file* disalin, kemudian dieksekusi. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.7.

1	shell> cd/usr/local/bin
2	shell> chmod +x ndb mgm*

Kode Sumber 4.7. Mengubah Lokasi Direktori

4.2.3 Konfigurasi Manajemen *Node*

Konfigurasi pada manajemen *node* dilakukan dengan langkah-langkah sebagai berikut.

1. Membuat direktori tempat *file* konfigurasi ditemukan kemudian membuat *file* itu sendiri. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.8.

1	shell> mkdir/var/lib/mysql-cluster
2	shell> cd/var/lib/mysql-cluster
3	shell> vi config.ini

Kode Sumber 4.8. Membuat Direktori

2. Mengatur *file* “config.ini”. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.9.

1	[ndbd default]
2	NoOfReplicas=2
3	DataMemory=10G
4	IndexMemory=2G
5	[tcp default]
6	[ndb_mgmd]
7	hostname=10.151.64.182
8	datadir=/var/lib/mysql-cluster
9	[ndbd]
10	hostname=10.151.64.182
11	datadir=/usr/local/mysql/data
12	[ndbd]
13	hostname=10.151.64.203
14	datadir=/usr/local/mysql/data
15	[mysqld]
16	MaxNoOfAttributes=10000
17	hostname=10.151.64.182
18	[mysqld]
19	MaxNoOfAttributes=10000
20	hostname=10.151.64.203

Kode Sumber 4.9. Mengatur *File* “config.ini”

4.2.4 Konfigurasi Data dan *SQL Node*

Konfigurasi data dan *SQL Node* dilakukan dengan cara menyunting *file* my.cnf pada direktori /etc/. Untuk setiap data *node* dan *SQL node* yang diatur pada my.cnf. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.10.

1	[client]
2	port=3306
3	socket=/tmp/mysql.sock
4	[mysqld]
5	port=3306
6	socket=/tmp/mysql.sock
7	ndbcluster
8	ndb-connectstring=10.151.64.181
9	[mysql_cluster]
10	ndb-connectstring=10.151.64.181

Kode Sumber 4.10. Data dan SQL Node

4.2.5 Memulai MySQL Cluster

Setiap proses *node cluster* harus dimulai secara terpisah. Manajemen *node* harus dimulai terlebih dahulu, kemudian *node* data. Pada setiap *node SQL* dilakukan langkah sebagai berikut.

1. Pada *node 3 (host manajemen)*, untuk memulai proses manajemen *node* dari *shell* sistem dilakukan perintah berikut. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.11.

1	shell> ndb_mgmd -f /var/lib/mysql-cluster/ config.ini-configdir=/var/lib/mysql-cluster/
---	--

Kode Sumber 4.11. Memulai Proses Manajemen Node

2. Jalankan perintah untuk memulai *ndbd* dan proses *mysql server* pada masing-masing *Data/host SQL*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.12.

1	shell> /usr/local/mysql/bin/ndbd
---	----------------------------------

Kode Sumber 4.12. Memulai Proses Ndbd dan MySQL Cluster

3. Mengaktifkan *MySQL* pada data *node*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.13.

1	shell> / etc / init.d / mysql start
---	-------------------------------------

Kode Sumber 4.13. Mengaktifkan MySQL

4.3 Implementasi *Role Base Access Control* (RBAC)

Pada bagian ini akan dijelaskan mengenai implementasi *RBAC* yang terdapat pada sistem ERP 2016. Secara rinci mengenai implementasi *RBAC* dijabarkan sebagai berikut:

4.3.1 Membuat Tabel Pengguna

Pembuatan tabel pengguna dilakukan dengan menjalankan perintah “*yii migrate*” pada folder aplikasi. Apabila proses migrasi berhasil dilakukan, maka akan muncul keterangan “*migrate successfully*” dan akan muncul tabel *user* di dalam basis data.

4.3.2 Membuat Tabel Autentifikasi RBAC dan Tabel Pengguna

Pada tahap ini dibutuhkan 4 tabel autentifikasi yang terdiri dari:

1. Tabel *Item*, tabel yang menyimpan daftar otorisasi, standar nama tabel adalah *auth_item*.
2. Tabel *Child*, tabel yang menyimpan hirarki daftar otoristas, standar nama tabel adalah *auth_item_child*.
3. Tabel *Assignment*, tabel yang menyimpan penetapan user untuk daftar otorisasi, standar nama tabel adalah *auth_assignment*.
4. Tabel *Rule*, tabel yang menyimpan aturan-aturan autentifikasi, standar nama tabel adalah *auth_rule*.

Cara untuk membuat tabel-tabel tersebut yaitu dengan menjalankan perintah yang ditunjukkan pada Kode Sumber 4.14.

1	<code>Yii migrate --migrationPath=@yii/rbac/migrations</code>
---	---

Kode Sumber 4.14. Generate Tabel Autentifikasi

Apabila proses migrasi berhasil yang maka akan terbuat 4 tabel pada basis data, yaitu tabel *auth_assignment*, tabel *auth_item*, tabel *auth_item_child* dan tabel *auth_rule*.

4.3.3 Membuat Modul *Admin* dan Konfigurasi Autentifikasi

Proses ini bertujuan untuk meletakkan konfigurasi pengguna dan masing-masing model dari 4 tabel autentifikasi. Kemudian dilakukan konfigurasi pada *folder config file/web.php*. Terdapat 3 konfigurasi, yaitu: menambahkan *admin* ke daftar modul, dengan tujuan menambah akses ke modul *admin*, menambahkan *authManager* pada *component* sebagai autentifikasi di Yii 2.0 dan mengatur *role* awal sebagai *guest*, dan menambahkan *Session Time Out* untuk mengatur durasi *time out session* selama 5 menit atau 300 detik. Masing-masing konfigurasi tersebut ditunjukkan pada Kode Sumber 4.15.

```

1  $config = [
2      'modules' => [
3          'admin' => [
4              'class' => 'app\modules\admin\AdminModule',
5          ],
6      ],
7      'components' => [
8          'authManager' => [
9              'class' => 'yii\rbac\DbManager',
10             'defaultRoles' => ['guest'],
11         ],
12         'session' => [
13             'timeout' => 300,
14         ],
15     ],
16 ];

```

Kode Sumber 4.15. Konfigurasi Autentifikasi

4.3.4 Membuat Model Tabel Autentifikasi, *Controller* dan *View* Pengguna

Pada tahap ini dilakukan *generate* kelas model dari masing-masing tabel autentifikasi pada modul *admin* yang diperlukan pada tahap sebelumnya. Kemudian ditambahkan *generate* kelas *controller* dan *view* pada tabel pengguna. Proses *generate* ini menggunakan *yii generator* yang telah disediakan oleh kerangka kerja Yii 2.0.

4.3.5 Menambahkan Kode pada *Usercontroller*

Pada modul *admin*, file *usercontroller.php* ditambahkan kode fungsi untuk semua tabel autentifikasi. Masing-masing fungsi ditunjukkan pada Kode Sumber 4.16, Kode Sumber 4.17, dan Kode Sumber 4.18.

```

1 public function actionAuthItem()
2 {
3     $auth = Yii::$app->authManager;
4
5     //menambahkan akses sebagai admin ke tabel
6     auth_item
7     $admin = $auth->createPermission('admin');
8     $admin->description = 'Allow user to access
9     all page';
10    $auth->add($admin);
11
12    //menambahkan akses sebagai sales and
13    distribution manajer ke tabel auth_item
14    $sd_manager = $auth->createPermission('sd-
15    manager');
16    $sd_manager->description = 'Allow user as
17    Sales and Distribution Manager';
18    $auth->add($sd_manager);
19
20    //menambahkan akses sebagai sales and
21    distribution staff ke tabel auth_item
22    $sd_staff = $auth->createPermission('sd-
23    staff');
24    $sd_staff->description = 'Allow user as Sales
25    and Distribution Staff';
26    $auth->add($sd_staff);
27 };
```

Kode Sumber 4.16. Menambahkan Data Ke Tabel Auth_Item

Kode Sumber 4.16 menunjukkan fungsi untuk menambahkan data ke tabel *auth_item* yang telah dibuat sebelumnya pada subbab 4.3.2. Setiap kode yang terdapat di dalam fungsi mendaftarkan setiap modul yang ada pada sistem ERP dan juga hak akses sebagai staf atau manajer.

```

1 public function actionItemChild()
2 {
3     $auth = Yii::$app->authManager;
```

```

4
5 //admin dapat mengakses semua daftar izin akses
6 $sd_manager = $auth->createPermission('sd-
manager');
7 $sd_staff = $auth->createPermission('sd-
staff');
8
9 $admin = $auth->createRole('admin');
10 $auth->add($admin);
11 $auth->addChild($admin, $sd_manager);
12 $auth->addChild($admin, $sd_staff);
13 }

```

Kode Sumber 4.17. Menambahkan Data ke Tabel Auth_Item_Child

Kode Sumber 4.17 *menunjukkan* fungsi untuk menambahkan data ke tabel `auth_item_child` yang telah dibuat sebelumnya pada subbab 4.3.2. Setiap kode yang terdapat di dalam fungsi bertujuan untuk mengatur *user admin* untuk dapat mengakses setiap modul yang telah terdaftar di dalam tabel `auth_item`.

```

1 public function actionAuthAssignment()
2 {
3     $auth = Yii::$app->authManager;
4     $admin = $auth->createRole('admin');
5     $auth->assign($admin, 1);
6 }

```

Kode Sumber 4.18. Menambahkan Data ke Tabel Auth_Assignment

Kode Sumber 4.18 *menunjukkan* fungsi untuk menambahkan data ke tabel `auth_assignment` yang telah dibuat sebelumnya pada subbab 4.3.2. Setiap kode yang terdapat di dalam fungsi bertujuan untuk menetapkan *user admin* yang memiliki id dengan nilai 1 (satu) pada tabel *user* dapat mengakses semua modul yang terdaftar di dalam tabel `auth_item`.

4.4 Implementasi *Multitenancy*

Pada bagian ini akan dijelaskan mengenai implementasi *multitenancy* yang terdapat pada sistem ERP 2016. Secara rinci mengenai implementasi *multitenancy* dijabarkan sebagai berikut.

4.4.1 Membuat Halaman *Tenant*

Pada implementasi pembuatan halaman *tenant*, dilakukan pembuatan halaman *tenant* secara sederhana, kemudian ditambahkan pembuatan basis data untuk setiap *tenant*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.19.

```

1 // mendapatkan nilai yang dimasukkan dari view
2 $host = "10.151.64.182";
3 $tenant = Yii::$app->request->post('tenant');
4 $database = Yii::$app->request->post('database');
5
6 // membuka koneksi pada untuk memasukkan data tenant
7 $koneksi =
8 mysqli_connect($host,"root","","multitenant");
9 if (!$koneksi->connect_error) {
10 die("Connection failed: ".$koneksi->connect_error);
11 }
12
13 // query untuk memasukkan data tenant
14 $insert = "insert into tenant
15 (tenant,database_name,created_at)
16 values('".$tenant."','".$database."',NOW())";
17 if($tenant != "" && $database != ""){
18 $koneksi->query($insert);
19 }
20
21 // memutuskan koneksi ke server
22 $koneksi->close();

```

Kode Sumber 4.19. Pembuatan Halaman Depan *Tenant*

4.4.2 Menambahkan Basis Data untuk *Tenant* Baru

Pada tahap ini dilakukan penambahan basis data untuk *tenant* baru dengan cara melakukan konfigurasi pada *server node MySQL*. Kode implementasi yang dimaksud ditunjukkan oleh Kode Sumber 4.20.

```

1 shell> mysql -u root -p;
2 shell> create database <nama database>;
3 shell> use <nama database>;
4 shell> source /tmp/mysql-dump/final-db.sql
5 shell> mysql -u root -p;
6 shell> create database <nama database>;

```

Kode Sumber 4.20. Penambahan Basis Data untuk *Tenant* Baru

4.4.3 Login Tenant

Setelah proses pembuatan basis data dan replikasi pada tahap 4.4.2. selesai, maka *tenant* melakukan *login* dengan memilih salah satu basis data, nama basis data yang dipilih tersebut disimpan dalam *session* dan akan digunakan untuk koneksi yang akan dibuat. Kode implementasi yang dimaksud ditunjukkan oleh kode sumber 4.21.

```

1  $host = '10.151.64.182'; // node mysql
2  $port = 3306;
3  $waitTimeoutInSeconds = 1;
4  $src=@fsockopen($host,$port,$errCode,$errStr,$waitTimeout
5  InSeconds);
6  if(is_resource($src)){
7      $_SESSION['dbserver_ip'] = "10.151.64.182";
8
9      // session multitenancy berdasarkan nama basis
10     data
11     $dbname =
12     isset($_SESSION['database_name'])?$_SESSION
13     ['database_name'] : 'test';
14 } else {
15     $_SESSION['dbserver_ip'] = "10.151.64.203";
16 }
17 $connection = [
18     'class' => 'yii\db\Connection',
19     'dbname'=> ".$dbname.",
20     'username' => 'root',
21     'password' => '',
22     'charset' => 'utf8'
23 ];
24 return $connection;

```

Kode Sumber 4.21. Login Tenant

4.5 Implementasi Program pada Modul *Sales and Distribution*

Pada bagian akan dijelaskan mengenai implementasi yang terdapat pada modul *Sales and Distribution* yang terbagi menjadi tampilan halaman utama atau *dashboard* dan beberapa submodul, antara lain: *shipping*, *customer*, *inquiry*, *quotation*, *sales order*, *return order*, *shipping and delivery*, dan *forecasting*. Secara rinci

mengenai implementasi lapisan antarmuka modul *Sales and Distribution* dijabarkan sebagai berikut:

4.5.1 Halaman Utama Modul

Halaman utama modul *Sales and Distribution* menampilkan ringkasan informasi mengenai penjualan dan pelanggan yang terdapat pada modul *Sales and Distribution*. Pengaturan halaman utama *Sales and Distribution* diatur pada fungsi `actionIndex()` yang ditunjukkan pada Lampiran C.1.

4.5.2 Melihat Daftar Data Submodul

Pada implementasi melihat daftar data submodul, sistem menampilkan seluruh daftar data *shipping, contact person, customer, inquiry, quotation, sales order, return order, shipping and delivery, dan forecasting* yang disimpan dalam basis data. Pengaturan melihat daftar data submodul diatur pada fungsi `actionIndex()` yang ditunjukkan pada Lampiran C.2.

4.5.3 Melihat Detail Data Submodul

Pada implementasi melihat detail data submodul, sistem menampilkan detail data *shipping, contact person, customer, inquiry, quotation, sales order, return order, shipping and delivery, dan forecasting* yang disimpan dalam basis data. Pengaturan melihat detail data submodul diatur pada kelas fungsi `actionView()` yang ditunjukkan pada Lampiran C.3.

4.5.4 Menambah dan Menyunting Data Submodul

Pada implementasi menambah dan menyunting data submodul, sistem menambahkan *shipping, contact person, customer, inquiry, quotation, sales order, return order, shipping and delivery, dan forecasting* baru ke basis data dan menyunting *shipping, contact person, customer, inquiry, quotation, sales order, return order, shipping and delivery, dan forecasting* yang telah ada di dalam basis data. Pengaturan menambahkan menambah dan

menyunting diatur pada kelas fungsi `actionCreate()` dan `actionUpdate()` yang ditunjukkan pada Lampiran C.4.

4.5.5 Menghapus Data Submodul

Pada implementasi menghapus data submodul, sistem menghapus data *shipping*, *contact person*, *customer*, *inquiry*, *quotation*, *sales order*, *return order*, *shipping and delivery*, dan *forecasting* yang telah ada di dalam basis data. Pengaturan menghapus data diatur pada masing-masing fungsi `actionDelete()` yang ditunjukkan pada Lampiran C.5.

4.5.6 Menambah Forecast

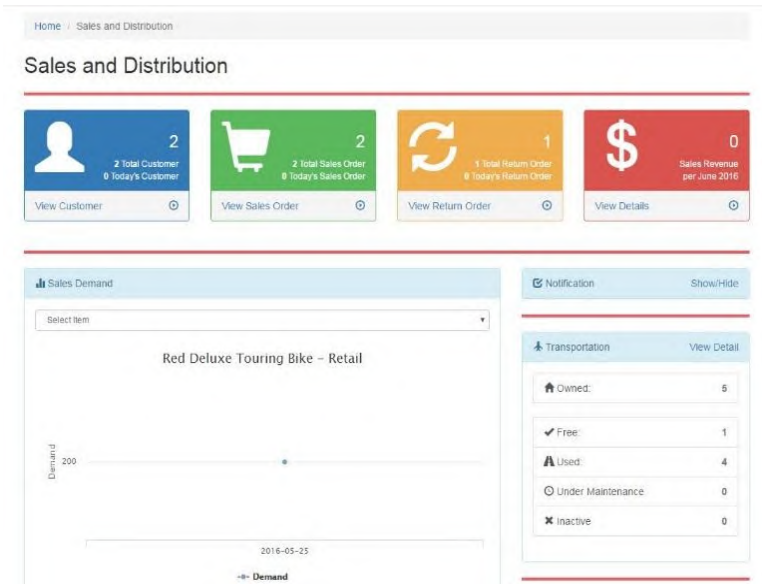
Pada implementasi menambah *forecast*, sistem menambahkan peramalan penjualan baru ke basis data. Pengaturan menambahkan dan perhitungan *forecast* diatur pada kelas fungsi `actionCreate()` yang ditunjukkan pada Lampiran C.6.

4.6 Implementasi Antarmuka Pengguna *Sales and Distribution*

Pada subbab ini akan dibahas mengenai implementasi antarmuka yang terdapat pada modul *Sales and Distribution*. Implementasi antarmuka ditunjukkan sebagai berikut.

4.6.1 Antarmuka Halaman Utama Modul *Sales and Distribution*

Pada antarmuka ini pengguna dapat melihat hasil penjualan untuk produk tertentu, jumlah pelanggan, jumlah *sales order*, jumlah *return order*, status transportasi yang digunakan untuk mengirim barang, notifikasi status pemesanan barang, total pembelian yang dilakukan oleh pelanggan, dan pendapatan penjualan. Tampilan antarmuka ini dapat dilihat pada Gambar 4.1.



Gambar 4.1. Antarmuka Halaman Utama Modul *Sales and Distribution*

4.6.2 Antarmuka Melihat Daftar *Shipping*

Pada antarmuka ini pengguna dapat melihat daftar rute pengiriman yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus rute pengiriman tertentu serta membuat rute pengiriman baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.2.

Shipping


#	Shipping Code	Shipping Description	Shipping Cost	Load Time	Pick Time	Total Shipping Time	Shipping Distance	Shipping Free
1	SHIP00	Standard shipping using small truck	IDR 100/kg	5 Hours	5 Hours	20 Hours	100.00 Km	1,000.00 Kg

Gambar 4.2. Antarmuka Melihat Daftar *Shipping*

4.6.3 Antarmuka Melihat Daftar *Contact Person*

Pada antarmuka ini pengguna dapat melihat daftar narahubung yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus narahubung tertentu serta membuat narahubung baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.3.

Contact Person



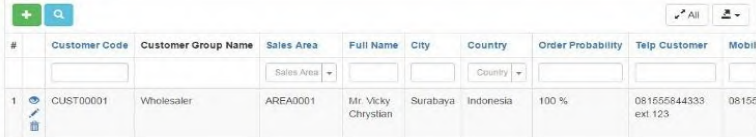
#	Contact Person Code	Person Id	Full Name	Email	Phone	Mobile	Department
1	CP00001	E005	Mr. RENANDA AGUSTIAN TORO	renan@yahoo.com	ext.		Production

Gambar 4.3. Antarmuka Melihat Daftar *Contact Person*

4.6.4 Antarmuka Melihat Daftar *Customer*

Pada antarmuka ini pengguna dapat melihat daftar pelanggan yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus pelanggan tertentu serta membuat pelanggan baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.4.

Customer



#	Customer Code	Customer Group Name	Sales Area	Full Name	City	Country	Order Probability	Telp Customer	Mobil
1	CUST00001	Wholesaler	AREA0001	Mr. Vicky Chrystian	Surabaya	Indonesia	100 %	081555844333 ext. 123	08155

Gambar 4.4. Antarmuka Melihat Daftar *Customer*

4.6.5 Antarmuka Melihat Daftar *Inquiry*

Pada antarmuka ini pengguna dapat melihat daftar *inquiry* yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus

inquiry tertentu serta membuat *inquiry* baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.5.

Inquiry

#	Inquiry Code	Inquiry Status	Sales Type	Sales Area	Sold To	Total Price	Shipping Cost	Shipping Pay To
1	IN00003	Expanded	MTO	AREA0001	CUST00001	IDR 424.108.031	IDR 940.900	Free

Gambar 4.5. Antarmuka Melihat Daftar *Inquiry*

4.6.6 Antarmuka Melihat Daftar *Quotation*

Pada antarmuka ini pengguna dapat melihat daftar *quotation* yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus *quotation* tertentu serta membuat *quotation* baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.6.

Quotation

#	Quotation Code	Quotation Status	Reference	Sales Type	Sales Area	Sold To	Payment Method
1	Q000001	Expanded	IN00003	MTO	AREA0001	CUST00001	Cash

Gambar 4.6. Antarmuka Melihat Daftar *Quotation*

4.6.7 Antarmuka Melihat Daftar *Sales Order*

Pada antarmuka ini pengguna dapat melihat daftar *sales order* yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus *sales order* tertentu serta membuat *sales order* baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.7.

Sales Order

#	Sales Order Code	Sales Order Status	Payment Status	Reference	Sales Type	Sales Area	Sold To
1	SO00002	incomplete	incomplete	No Reference	MTO	AREA0001	CUST00002

Gambar 4.7. Antarmuka Melihat Daftar *Sales Order*

4.6.8 Antarmuka Melihat Daftar *Return Order*

Pada antarmuka ini pengguna dapat melihat daftar *return order* yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus *return order* tertentu serta membuat *return order* baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.8.

Return Order

#	Sales Order Code	Return Status	From SO	Sales Type	Sales Area	Sold To	Ship To
1	RE00003	incomplete	No Reference	RE	AREA0001	(not set)	(not set)

Gambar 4.8. Antarmuka Melihat Daftar *Return Order*

4.6.9 Antarmuka Melihat Daftar *Shipping and Delivery*

Pada antarmuka ini pengguna dapat melihat daftar pengiriman yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus pengiriman tertentu serta membuat pengiriman baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.9.

Delivery

#	Delivery Code	Delivery Status	Sales Order	Sold To	Ship To	Bill To	Shipping Cost	Distance
1	DV00001	Shipping Pending	SO00001	CUST00001	Jalan Raya	1	IDR 0	100.00 Km

Gambar 4.9. Antarmuka Melihat Daftar *Shipping and Delivery*

4.6.10 Antarmuka Melihat Daftar *Forecasting*

Pada antarmuka ini pengguna dapat melihat daftar peramalan penjualan yang terdapat pada sistem. Pengguna juga dapat memilih melakukan kegiatan menyunting, melihat detail, dan menghapus peramalan penjualan tertentu serta membuat peramalan penjualan baru. Tampilan antarmuka ini dapat dilihat pada Gambar 4.10.

Forecast



#	Forecast Code	Item Name	Forecast Period	
	<input type="text"/>	<input type="text" value="Item Name"/>	<input type="text"/>	
1	RED-DEL-	Red Deluxe Touring Bike	30	 

Gambar 4.10. Antarmuka Melihat Daftar *Forecasting*

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas hasil dan pembahasan pada aplikasi yang dikembangkan. Pada bab ini akan dijelaskan tentang data yang digunakan, hasil yang didapatkan dari penggunaan perangkat lunak dan uji coba yang dilakukan pada perangkat lunak yang telah dikerjakan untuk menguji apakah fungsionalitas aplikasi telah diimplementasikan dengan benar dan berjalan sebagaimana mestinya.

5.1 Lingkungan Pengujian

Lingkungan uji coba menjelaskan lingkungan yang digunakan untuk menguji implementasi pembuatan sistem pada tugas akhir ini. Lingkungan uji coba meliputi perangkat keras dan perangkat lunak yang dijelaskan sebagai berikut:

1. Basis data yang digunakan pada *server* adalah *MySQL Cluster*.
2. 2 PC untuk *server* basis data menggunakan Sistem Operasi Ubuntu 12.04.
3. PC untuk *server* menggunakan Intel® Core™ i3-2120 @3.30GHz , RAM 4GB dengan Sistem Operasi Windows 8.1 Enterprise x64.
4. Mozilla Firefox 46.0.1 dan Chrome 49.0 sebagai antarmuka untuk pengujian aplikasi klien.

5.2 Skenario Pengujian

Pada bagian ini akan dijelaskan tentang skenario pengujian yang dilakukan. Pengujian dilakukan dalam 3 (tiga) tahap, yaitu pengujian kebutuhan fungsionalitas, pengujian kegunaan sistem, dan pengujian perbandingan *metrics*. Pengujian kebutuhan fungsionalitas menggunakan metode kotak hitam (*black box*). Metode ini menekankan pada hasil keluaran sistem.

5.2.1 Pengujian Perencanaan Bisnis

Pengujian perencanaan bisnis dilakukan dengan memasukkan data-data perencanaan bisnis yang telah dibuat ke dalam aplikasi. Perencanaan bisnis ditunjukkan pada Tabel 5.1, Table 5.2, Tabel 5.3, Tabel 5.4, dan tabel 5.5.

5.2.1.1 *Daftar Barang Jadi*

Perencanaan bisnis berupa daftar barang jadi yang dihasilkan dan dijual oleh perusahaan ditunjukkan pada Tabel 5.1.

Tabel 5.1. Daftar Barang Jadi

No	Nama Barang
1	<i>Red Deluxe Touring Bike</i>
2	<i>Red Profesional Touring Bike</i>

Pada Tabel 5.1, ditunjukkan bahwa perusahaan ini memproduksi dua jenis sepeda. Dua jenis sepeda tersebut memiliki bahan dasar yang berbeda, *Deluxe Touring Bike* berbahan dasar aluminium, sedangkan *Profesional Touring Bike* berbahan dasar karbon.

5.2.1.2 *Daftar Aset*

Perencanaan bisnis berupa daftar aset yang dimiliki oleh perusahaan ditunjukkan pada Tabel 5.1.

Tabel 5.2. Daftar Aset

No.	Nama	Kuantitas
1	Tanah	180 x 150 m ²
2	Kantor	50 x 50 m ²
3	Parkir	60 x 50 m ²
4	Kantin	40 x 60 m ²
5	Pabrik	108 x 80 m ²
6	Pengolahan Limbah	20 x 40 m ²
7	<i>Raw Materials Inventory</i>	40 x 60 m ²
8	<i>Semi-Finished Good Inventory</i>	40 x 60 m ²
9	<i>Finished Good Inventory</i>	40 x 60 m ²

10	<i>Welding Machine</i>	2 baris
11	<i>Molding Machine</i>	2 baris
12	<i>Laser Cutting Machine</i>	2 baris
13	<i>Spray Painting Machine</i>	2 baris
14	<i>Testing Machine</i>	2 baris
15	Truk	10 buah
16	<i>Forklift</i>	6 buah
17	<i>Heavy Forklift</i>	6 buah

Pada Tabel 5.2, perusahaan memiliki 17 aset pada perencanaannya dengan kuantitas dari masing-masing aset yang ada.

5.2.1.3 *Daftar Bahan Baku*

Perencanaan bisnis berupa daftar bahan baku yang dimiliki oleh perusahaan ditunjukkan pada Tabel 5.3.

Tabel 5.3. Daftar Bahan Baku

No.	Nama bahan baku	Kuantitas	Satuan unit
1	<i>Tire</i>	1.2	<i>Pcs</i>
2	<i>Seat Kit</i>	0.5	<i>Pcs</i>
3	<i>Chain</i>	1.5	<i>Pcs</i>
4	<i>Gear</i>	1.8	<i>Pcs</i>
5	<i>Brake</i>	0.6	<i>Pcs</i>
6	<i>Handle Bar</i>	1.2	<i>Pcs</i>
7	<i>Pedal</i>	0.4	<i>Pcs</i>
8	<i>Aluminium</i>	2	m
9	<i>Carbon Fiber</i>	2	m
10	<i>Red Paint 20KG</i>	20	<i>Big Drum</i>
11	<i>Velg</i>	2.3	<i>Pcs</i>
12	<i>Tube</i>	0.4	<i>Pcs</i>
13	<i>Hex Nut 5mm</i>	0.04	<i>Pcs</i>
14	<i>Lock Washer 5mm</i>	0.06	<i>Pcs</i>
15	<i>Socket Head Bolt 5mm</i>	0.03	<i>Pcs</i>

Perencanaan yang diadakan oleh perusahaan ini akan menggunakan 15 macam bahan baku seperti yang dijabarkan pada

Tabel 5.3. 15 bahan baku ini ditentukan dari daftar material yang akan diterangkan di Tabel 5.4.

5.2.1.4 *Daftar Material*

Perencanaan bisnis berupa daftar material yang dibutuhkan oleh perusahaan untuk membuat 1 (satu) barang jadi ditunjukkan pada Tabel 5.4.

Tabel 5.4. Daftar Material

No.	Bahan yang dibutuhkan	Kuantitas	Satuan unit	Bahan yang dihasilkan	
1	<i>Aluminium</i>	5	m	Kerangka untuk <i>Red Deluxe Touring Bike</i>	
2	<i>Paint drum with Red Color 20KG</i>	0.05	<i>Big drum</i>		
3	<i>Carbon Fiber</i>	5	m	Kerangka untuk <i>Red Profesional Touring Bike</i>	
4	<i>Paint drum with Red Color 20KG</i>	0.05	<i>Big drum</i>		
5	<i>Hex Nut 5 mm</i>	2	<i>Pcs</i>	Roda sepeda	
6	<i>Lock Washer 5 mm</i>	2	<i>Pcs</i>		
7	<i>Socket Head Bolt 5mm</i>	1	<i>Pcs</i>		
8	<i>Tire's Bike</i>	1	<i>Pcs</i>		
9	<i>Wheel's Tube</i>	1	<i>Pcs</i>		
10	<i>Wheel's Velg</i>	1	<i>Pcs</i>		
11	<i>Frame for Deluxe Touring Bike with Red Color</i>	1	<i>Pcs</i>		<i>Red Deluxe Touring Bike</i>
12	<i>Wheel's Bike</i>	2	<i>Pcs</i>		
13	<i>Chain's Bike</i>	1	<i>Pcs</i>		
14	<i>Handle Bar's Bike</i>	1	<i>Pcs</i>		
15	<i>Seat Kit's Bike</i>	1	<i>Pcs</i>		
16	<i>Pedal's Bike</i>	2	<i>Pcs</i>		
17	<i>Gear's Bike</i>	1	<i>Pcs</i>		
18	<i>Front and Rear Brake's Bike</i>	1	<i>Pcs</i>		

19	<i>Frame for Deluxe Touring Bike with Red Color</i>	1	<i>Pcs</i>	<i>Red Professional Touring Bike</i>
20	<i>Wheel's Bike</i>	2	<i>Pcs</i>	
21	<i>Chain's Bike</i>	1	<i>Pcs</i>	
22	<i>Handle Bar's Bike</i>	1	<i>Pcs</i>	
23	<i>Seat Kit's Bike</i>	1	<i>Pcs</i>	
24	<i>Pedal's Bike</i>	2	<i>Pcs</i>	
25	<i>Gear's Bike</i>	1	<i>Pcs</i>	
26	<i>Front and Rear Brake's Bike</i>	1	<i>Pcs</i>	

Daftar material yang dijabarkan pada tabel 5.4. menjelaskan material-material yang dibutuhkan untuk memproduksi barang jadi yang dibutuhkan pada perusahaan ini.

5.2.1.5 Sumber Daya

Pada perusahaan ini memiliki beberapa sumber daya yang terdiri atas beberapa pegawai dan beberapa mesin yang terpisah menjadi 2 (dua) baris. Terdapat 150 orang sebagai *direct labour*, 32 orang sebagai *indirect labour*. Untuk mesin, terdapat *welding machine*, *molding machine*, *laser cutting machine*, *spray machine*, dan *testing machine* yang masing-masing mesin untuk 2 (dua) baris.

5.2.1.6 Perencanaan keuangan

Perencanaan keuangan dalam perusahaan ini digunakan untuk merincikan keuangan yang digunakan untuk pengelolaan keuangan dalam perusahaan. Perencanaan keuangan ditunjukkan pada Table 5.5.

Tabel 5.5. Perencanaan Keuangan Perusahaan

No.	Nama perencanaan keuangan	Jumlah (dalam Rp)
1	Total pembelian aset aktif	27.997.311.200,00
2	Total pembelian bahan baku	88.615.744.062,50
3	Biaya lain-lain	3,386,944,737.50

4	Peminjaman dari Bank	120.000.000.000,00
---	----------------------	--------------------

5.2.2 Pengujian Fungsionalitas Sistem

Pengujian fungsionalitas sistem dilakukan secara mandiri dengan menyiapkan sejumlah skenario sebagai tolak ukur keberhasilan pengujian. Pengujian fungsionalitas dilakukan dengan mengacu pada kasus penggunaan yang telah dijelaskan pada subbab 3.5. Pengujian pada kebutuhan fungsionalitas dijabarkan pada subbab berikut.

5.2.2.1 Pengujian Mengelola Rute Pengiriman

Pengujian ini terdiri dari pengujian mengelola data rute pengiriman. Pengujian mengelola data rute pengiriman yaitu menambah, menyunting, dan menghapus data rute pengiriman yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.6.

Tabel 5.6. Pengujian Mengelola Rute Pengiriman

ID	UJ.UC-001
Referensi	UC-001
Nama	Mengelola rute pengiriman
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data rute pengiriman
Skenario 1	Pengguna menambah data rute pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping</i>
Data Uji	Data rute pengiriman
Langkah Pengujian	Pengguna masuk ke halaman shipping dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail rute pengiriman

Skenario 2	Pengguna menyunting data rute pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping</i>
Data Uji	Data rute pengiriman
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data rute pengiriman yang telah ada, kemudian menyunting data tersebut pada halaman <i>shipping</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail rute pengiriman
Skenario 3	Pengguna menghapus data rute pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping</i>
Data Uji	Data rute pengiriman
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data rute pengiriman yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama rute pengiriman

5.2.2.2 Pengujian Mengelola Narahubung

Pengujian ini terdiri dari pengujian mengelola data narahubung. Pengujian mengelola data narahubung yaitu menambah, menyunting, dan menghapus data narahubung yang

sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.7.

Tabel 5.7. Pengujian Mengelola Narahubung

ID	UJ.UC-002
Referensi	UC-002
Nama	Mengelola narahubung
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data narahubung
Skenario 1	Pengguna menambah data narahubung
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Contact person</i>
Data Uji	Data narahubung
Langkah Pengujian	Pengguna masuk ke halaman <i>contact person</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail narahubung
Skenario 2	Pengguna menyunting data narahubung
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Contact person</i>
Data Uji	Data narahubung
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data narahubung yang telah ada, kemudian menyunting data tersebut pada halaman <i>contact person</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail narahubung
Skenario 3	Pengguna menghapus data narahubung

Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Contact person</i>
Data Uji	Data narahubung
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data narahubung yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama narahubung

5.2.2.3 Pengujian Mengelola Pelanggan

Pengujian ini terdiri dari pengujian mengelola data pelanggan. Pengujian mengelola data pelanggan yaitu menambah, menyunting, dan menghapus data pelanggan yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.8.

Tabel 5.8. Pengujian Mengelola Pelanggan

ID	UJ.UC-003
Referensi	UC-003
Nama	Mengelola pelanggan
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data pelanggan
Skenario 1	Pengguna menambah data pelanggan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Customer</i>
Data Uji	Data pelanggan
Langkah Pengujian	Pengguna masuk ke halaman customer dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data

Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail pelanggan
Skenario 2	Pengguna menyunting data pelanggan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Customer</i>
Data Uji	Data pelanggan
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data pelanggan yang telah ada, kemudian menyunting data tersebut pada halaman <i>customer</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail pelanggan
Skenario 3	Pengguna menghapus data pelanggan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Customer</i>
Data Uji	Data pelanggan
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data pelanggan yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama pelanggan

5.2.2.4 Pengujian Mengelola *Inquiry*

Pengujian ini terdiri dari pengujian mengelola data *inquiry*. Pengujian mengelola data *inquiry* yaitu menambah, menyunting, dan menghapus data *inquiry* yang sudah ada pada

sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.9.

Tabel 5.9. Pengujian Mengelola *Inquiry*

ID	UJ.UC-004
Referensi	UC-004
Nama	Mengelola <i>inquiry</i>
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data <i>inquiry</i>
Skenario 1	Pengguna menambah data <i>inquiry</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Inquiry</i>
Data Uji	Data <i>inquiry</i>
Langkah Pengujian	Pengguna masuk ke halaman <i>inquiry</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>inquiry</i>
Skenario 2	Pengguna menyunting data <i>inquiry</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Inquiry</i>
Data Uji	Data <i>inquiry</i>
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data <i>inquiry</i> yang telah ada, kemudian menyunting data tersebut pada halaman <i>inquiry</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>inquiry</i>
Skenario 3	Pengguna menghapus data <i>inquiry</i>

Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Inquiry</i>
Data Uji	Data <i>inquiry</i>
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data <i>inquiry</i> yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama <i>inquiry</i>

5.2.2.5 Pengujian Mengelola *Quotation*

Pengujian ini terdiri dari pengujian mengelola data *quotation*. Pengujian mengelola data *quotation* yaitu menambah, menyunting, dan menghapus data *quotation* yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.10.

Tabel 5.10. Pengujian Mengelola *Quotation*

ID	UJ.UC-005
Referensi	UC-005
Nama	Mengelola <i>quotation</i>
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data <i>quotation</i>
Skenario 1	Pengguna menambah data <i>quotation</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Quotation</i>
Data Uji	Data <i>quotation</i>
Langkah Pengujian	Pengguna masuk ke halaman <i>quotation</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil

Kondisi Akhir	Pengguna berada pada halaman detail <i>quotation</i>
Skenario 2	Pengguna menyunting data <i>quotation</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Quotation</i>
Data Uji	Data <i>quotation</i>
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data <i>quotation</i> yang telah ada, kemudian menyunting data tersebut pada halaman <i>quotation</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>quotation</i>
Skenario 3	Pengguna menghapus data <i>quotation</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Quotation</i>
Data Uji	Data <i>quotation</i>
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data <i>quotation</i> yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama <i>quotation</i>

5.2.2.6 Pengujian Mengelola *Sales Order*

Pengujian ini terdiri dari pengujian mengelola data *sales order*. Pengujian mengelola data *sales order* yaitu menambah, menyunting, dan menghapus data *sales order* yang sudah ada pada

sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.11.

Tabel 5.11. Pengujian Mengelola *Sales Order*

ID	UJ.UC-006
Referensi	UC-006
Nama	Mengelola <i>sales order</i>
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data <i>sales order</i>
Skenario 1	Pengguna menambah data <i>sales order</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Sales order</i>
Data Uji	Data <i>sales order</i>
Langkah Pengujian	Pengguna masuk ke halaman <i>sales order</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>sales order</i>
Skenario 2	Pengguna menyunting data <i>sales order</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Sales order</i>
Data Uji	Data <i>sales order</i>
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data <i>sales order</i> yang telah ada, kemudian menyunting data tersebut pada halaman <i>sales order</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>sales order</i>
Skenario 3	Pengguna menghapus data <i>sales order</i>

Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Sales order</i>
Data Uji	Data <i>sales order</i>
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data <i>sales order</i> yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama <i>sales order</i>

5.2.2.7 Pengujian Mengelola *Return Order*

Pengujian ini terdiri dari pengujian mengelola data *return order*. Pengujian mengelola data *return order* yaitu menambah, menyunting, dan menghapus data *return order* yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.12.

Tabel 5.12. Pengujian Mengelola *Return Order*

ID	UJ.UC-007
Referensi	UC-007
Nama	Mengelola <i>return order</i>
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data <i>return order</i>
Skenario 1	Pengguna menambah data <i>return order</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Return order</i>
Data Uji	Data <i>return order</i>
Langkah Pengujian	Pengguna masuk ke halaman <i>return order</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data

Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>return order</i>
Skenario 2	Pengguna menyunting data <i>return order</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Return order</i>
Data Uji	Data <i>return order</i>
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data <i>return order</i> yang telah ada, kemudian menyunting data tersebut pada halaman <i>return order</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>return order</i>
Skenario 3	Pengguna menghapus data <i>return order</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Return order</i>
Data Uji	Data <i>return order</i>
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data <i>return order</i> yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama <i>return order</i>

5.2.2.8 Pengujian Mengelola Pengiriman

Pengujian ini terdiri dari pengujian mengelola data pengiriman. Pengujian mengelola data pengiriman yaitu

menambah, menyunting, dan menghapus data pengiriman yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.13.

Tabel 5.13. Pengujian Mengelola Pengiriman

ID	UJ.UC-008
Referensi	UC-008
Nama	Mengelola pengiriman
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data pengiriman
Skenario 1	Pengguna menambah data pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping and delivery</i>
Data Uji	Data pengiriman
Langkah Pengujian	Pengguna masuk ke halaman <i>shipping and delivery</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail pengiriman
Skenario 2	Pengguna menyunting data pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping and delivery</i>
Data Uji	Data pengiriman
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data pengiriman yang telah ada, kemudian menyunting data tersebut pada halaman <i>shipping and delivery</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil

Kondisi Akhir	Pengguna berada pada halaman detail pengiriman
Skenario 3	Pengguna menghapus data pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping and delivery</i>
Data Uji	Data pengiriman
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data pengiriman yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama pengiriman

5.2.2.9 Pengujian Mengelola Peramalan penjualan

Pengujian ini terdiri dari pengujian mengelola data peramalan penjualan. Pengujian mengelola data peramalan penjualan yaitu menambah, menyunting, dan menghapus data peramalan penjualan yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.14.

Tabel 5.14. Pengujian Mengelola Peramalan penjualan

ID	UJ.UC-009
Referensi	UC-009
Nama	Mengelola peramalan penjualan
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola data peramalan penjualan
Skenario 1	Pengguna menambah data peramalan penjualan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Forecast</i>
Data Uji	Data peramalan penjualan

Langkah Pengujian	Pengguna masuk ke halaman <i>forecast</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail peramalan penjualan
Skenario 2	Pengguna menyunting data peramalan penjualan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Forecast</i>
Data Uji	Data peramalan penjualan
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data peramalan penjualan yang telah ada, kemudian menyunting data tersebut pada halaman <i>forecast</i> dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail peramalan penjualan
Skenario 3	Pengguna menghapus data peramalan penjualan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Forecast</i>
Data Uji	Data peramalan penjualan
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data peramalan penjualan yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data

Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama peramalan penjualan

5.2.2.10 Pengujian Melihat Rute Pengiriman

Pengujian ini terdiri dari pengujian melihat data rute pengiriman. Pengujian melihat rute pengiriman yaitu melihat data rute pengiriman yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.15.

Tabel 5.15. Pengujian Melihat Rute Pengiriman

ID	UJ.UC-010
Referensi	UC-010
Nama	Melihat detail rute pengiriman
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data rute pengiriman
Skenario 1	Pengguna menambah data rute pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping</i>
Data Uji	Data rute pengiriman
Langkah Pengujian	Pengguna masuk ke halaman detail rute pengiriman dari data rute pengiriman yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail rute pengiriman
Hasil Yang Didapat	Data tertampilkan di halaman detail rute pengiriman
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail rute pengiriman

5.2.2.11 Pengujian Melihat Narahubung

Pengujian ini terdiri dari pengujian melihat data narahubung. Pengujian melihat narahubung yaitu melihat data narahubung yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.16.

Tabel 5.16. Pengujian Melihat Narahubung

ID	UJ.UC-011
Referensi	UC-011
Nama	Melihat detail narahubung
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data narahubung
Skenario 1	Pengguna menambah data narahubung
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Contact person</i>
Data Uji	Data narahubung
Langkah Pengujian	Pengguna masuk ke halaman detail narahubung dari data narahubung yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail narahubung
Hasil Yang Didapat	Data tertampilkan di halaman detail narahubung
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail narahubung

5.2.2.12 Pengujian Melihat Pelanggan

Pengujian ini terdiri dari pengujian melihat data pelanggan. Pengujian melihat pelanggan yaitu melihat data pelanggan yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.17.

Tabel 5.17. Pengujian Melihat Pelanggan

ID	UJ.UC-012
Referensi	UC-012
Nama	Melihat detail pelanggan
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data pelanggan
Skenario 1	Pengguna menambah data pelanggan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Customer</i>
Data Uji	Data pelanggan

Langkah Pengujian	Pengguna masuk ke halaman detail pelanggan dari data pelanggan yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail pelanggan
Hasil Yang Didapat	Data tertampilkan di halaman detail pelanggan
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail pelanggan

5.2.2.13 Pengujian Melihat *Inquiry*

Pengujian ini terdiri dari pengujian melihat data *inquiry*. Pengujian melihat *inquiry* yaitu melihat data *inquiry* yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.18.

Tabel 5.18. Pengujian Melihat *Inquiry*

ID	UJ.UC-013
Referensi	UC-013
Nama	Melihat detail <i>inquiry</i>
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data <i>inquiry</i>
Skenario 1	Pengguna menambah data <i>inquiry</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Inquiry</i>
Data Uji	Data <i>inquiry</i>
Langkah Pengujian	Pengguna masuk ke halaman detail <i>inquiry</i> dari data <i>inquiry</i> yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail <i>inquiry</i>
Hasil Yang Didapat	Data tertampilkan di halaman detail <i>inquiry</i>
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>inquiry</i>

5.2.2.14 Pengujian Melihat *Quotation*

Pengujian ini terdiri dari pengujian melihat data *quotation*. Pengujian melihat *quotation* yaitu melihat data *quotation* yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.19.

Tabel 5.19. Pengujian Melihat *Quotation*

ID	UJ.UC-014
Referensi	UC-014
Nama	Melihat detail <i>quotation</i>
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data <i>quotation</i>
Skenario 1	Pengguna menambah data <i>quotation</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Quotation</i>
Data Uji	Data <i>quotation</i>
Langkah Pengujian	Pengguna masuk ke halaman detail <i>quotation</i> dari data <i>quotation</i> yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail <i>quotation</i>
Hasil Yang Didapat	Data tertampilkan di halaman detail <i>quotation</i>
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>quotation</i>

5.2.2.15 Pengujian Melihat *Sales Order*

Pengujian ini terdiri dari pengujian melihat data *sales order*. Pengujian melihat *sales order* yaitu melihat data *sales order* yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.20.

Tabel 5.20. Pengujian Melihat *Sales Order*

ID	UJ.UC-015
Referensi	UC-015
Nama	Melihat detail <i>sales order</i>

Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data <i>sales order</i>
Skenario 1	Pengguna menambah data <i>sales order</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Sales order</i>
Data Uji	Data <i>sales order</i>
Langkah Pengujian	Pengguna masuk ke halaman detail <i>sales order</i> dari data <i>sales order</i> yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail <i>sales order</i>
Hasil Yang Didapat	Data tertampilkan di halaman detail <i>sales order</i>
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>sales order</i>

5.2.2.16 Pengujian Melihat *Return Order*

Pengujian ini terdiri dari pengujian melihat data *return order*. Pengujian melihat *return order* yaitu melihat data *return order* yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.21.

Tabel 5.21. Pengujian Melihat *Return Order*

ID	UJ.UC-016
Referensi	UC-016
Nama	Melihat detail <i>return order</i>
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data <i>return order</i>
Skenario 1	Pengguna menambah data <i>return order</i>
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Return order</i>
Data Uji	Data <i>return order</i>
Langkah Pengujian	Pengguna masuk ke halaman detail <i>return order</i> dari data <i>return order</i> yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail <i>return order</i>

Hasil Yang Didapat	Data tertampilkan di halaman detail <i>return order</i>
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail <i>return order</i>

5.2.2.17 Pengujian Melihat Pengiriman

Pengujian ini terdiri dari pengujian melihat data pengiriman. Pengujian melihat pengiriman yaitu melihat data pengiriman yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.22.

Tabel 5.22. Pengujian Melihat Pengiriman

ID	UJ.UC-017
Referensi	UC-017
Nama	Melihat detail pengiriman
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data pengiriman
Skenario 1	Pengguna menambah data pengiriman
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Shipping and delivery</i>
Data Uji	Data pengiriman
Langkah Pengujian	Pengguna masuk ke halaman detail pengiriman dari data pengiriman yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail pengiriman
Hasil Yang Didapat	Data tertampilkan di halaman detail pengiriman
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail pengiriman

5.2.2.18 Pengujian Melihat Peramalan Penjualan

Pengujian ini terdiri dari pengujian melihat data peramalan penjualan. Pengujian melihat peramalan penjualan yaitu melihat data peramalan penjualan yang sudah ada pada sistem.

Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.23.

Tabel 5.23. Pengujian Melihat Peramalan penjualan

ID	UJ.UC-018
Referensi	UC-018
Nama	Melihat detail peramalan penjualan
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data peramalan penjualan
Skenario 1	Pengguna menambah data peramalan penjualan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu</i> Forecast
Data Uji	Data peramalan penjualan
Langkah Pengujian	Pengguna masuk ke halaman detail peramalan penjualan dari data peramalan penjualan yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail peramalan penjualan
Hasil Yang Didapat	Data tertampilkan di halaman detail peramalan penjualan
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail peramalan penjualan

5.2.2.19 Pengujian Melihat Laporan Penjualan

Pengujian ini terdiri dari pengujian melihat data laporan penjualan. Pengujian melihat laporan penjualan yaitu melihat data laporan penjualan yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.24.

Tabel 5.24. Pengujian Melihat Laporan Penjualan

ID	UJ.UC-019
Referensi	UC-019
Nama	Melihat detail laporan penjualan
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data laporan penjualan

Skenario 1	Pengguna menambah data laporan penjualan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Sales and Distribution</i>
Data Uji	Data laporan penjualan
Langkah Pengujian	Pengguna masuk ke halaman detail laporan penjualan dari data laporan penjualan yang dipilih
Hasil Yang Diharapkan	Data ditampilkan di halaman detail laporan penjualan
Hasil Yang Didapat	Data tertampilkan di halaman detail laporan penjualan
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail laporan penjualan

5.2.2.20 Pengujian Melihat Laporan Pelanggan

Pengujian ini terdiri dari pengujian melihat data laporan pelanggan. Pengujian melihat laporan pelanggan yaitu melihat data laporan pelanggan yang sudah ada pada sistem. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada Tabel 5.25.

Tabel 5.25. Pengujian Mengelola Laporan Pelanggan

ID	UJ.UC-020
Referensi	UC-020
Nama	Melihat detail laporan pelanggan
Tujuan Pengujian	Menguji kemampuan sistem dalam melihat data laporan pelanggan
Skenario 1	Pengguna menambah data laporan pelanggan
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Customer report</i>
Data Uji	Data laporan pelanggan
Langkah Pengujian	Pengguna masuk ke halaman detail laporan pelanggan dari data laporan pelanggan yang dipilih

Hasil Yang Diharapkan	Data ditampilkan di halaman detail laporan pelanggan
Hasil Yang Didapat	Data tertampilkan di halaman detail laporan pelanggan
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail laporan pelanggan

5.2.3 Pengujian Fungsional *Role Based Access Control* (RBAC)

Pengujian ini terdiri dari pengujian mengelola *Role Based Access Control* (RBAC). Rincian skenario pengujian pada fitur *Role Based Access Control* (RBAC) dapat dilihat pada Tabel 5.26.

Tabel 5.26. Pengujian Fitur *Role Based Access Control* (RBAC)

ID	UJ.RBAC
Referensi	RBAC
Nama	Mengelola RBAC
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola RBAC
Skenario 1	Pengguna menambah pengguna baru
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Admin</i>
Data Uji	Data pengguna
Langkah Pengujian	Pengguna masuk ke halaman <i>admin</i> dan memasukkan data uji ke dalam basis data
Hasil Yang Diharapkan	Data yang dimasukkan tersimpan di basis data
Hasil Yang Didapat	Data tersimpan di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman <i>admin</i>
Skenario 2	Pengguna menyunting data pengguna

Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Admin</i>
Data Uji	Data pengguna
Langkah Pengujian	Pengguna memilih tombol <i>update</i> dari salah satu data pengguna yang telah ada, kemudian menyunting data tersebut pada halaman admin dan menyimpan data yang baru ke dalam basis data
Hasil Yang Diharapkan	Data baru yang dimasukkan, menggantikan data lama yang ada di dalam basis data
Hasil Yang Didapat	Data baru tersimpan di dalam basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman detail pengguna
Skenario 3	Pengguna menghapus data pengguna
Kondisi Awal	Pengguna berada pada <i>menu</i> utama dan memilih <i>menu Admin</i>
Data Uji	Data pengguna
Langkah Pengujian	Pengguna memilih tombol <i>delete</i> dari salah satu data pengguna yang telah ada, kemudian melakukan konfirmasi penghapusan data dan data akan dihapus dari basis data
Hasil Yang Diharapkan	Data yang dihapus akan terhapus dari basis data
Hasil Yang Didapat	Data terhapus di basis data
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman admin

Hasil pengujian fitur RBAC dapat dilihat pada Gambar 5.1, Gambar 5.2, Gambar 5.3, Gambar 5.4, dan Gambar 5.5.

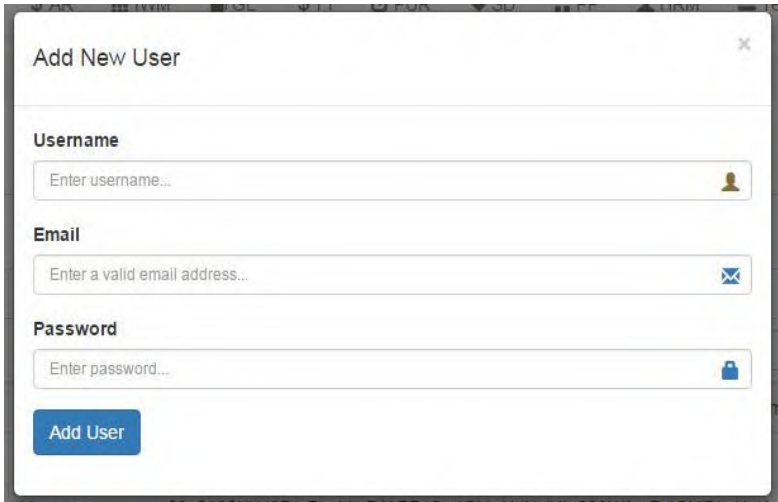
Gambar 5.1. Proses *Login User Admin*

Gambar 5.1 *menunjukkan* hasil pengujian untuk proses *login* oleh *user admin*.



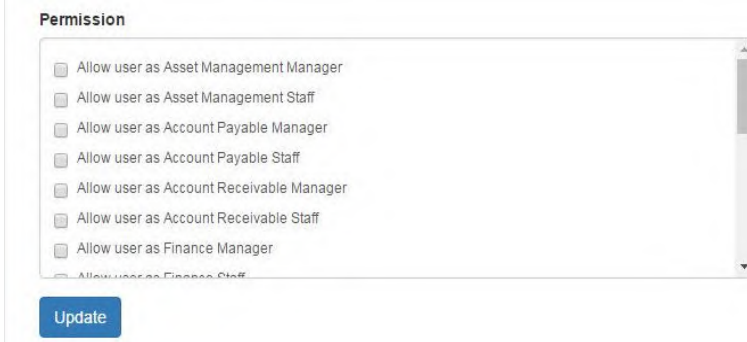
Gambar 5.2. Tampilan Awal Setelah *Login* Berhasil Dilakukan

Gambar 5.2 menunjukkan hasil pengujian berupa tampilan awal setelah proses *login* oleh *user admin*.



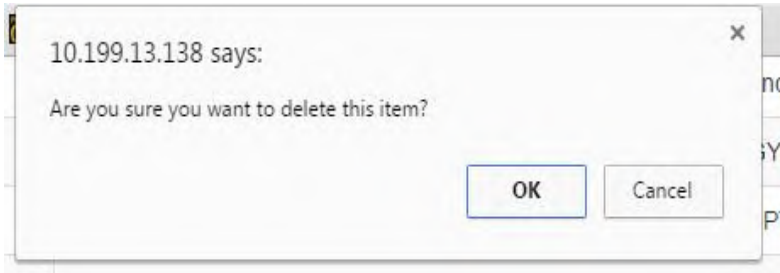
Gambar 5.3. Admin Menambahkan User Baru

Gambar 5.3 menunjukkan proses penginputan data *user* untuk skenario 1 dan hasil pengujiannya, dimana dapat dilihat penambahan pada halaman daftar *user* yang baru.



Gambar 5.4. Proses User Admin Menyunting Data User

Gambar 5.4 menunjukkan proses penyuntingan data *user* untuk skenario 2.



Gambar 5.5. Proses User Admin Menghapus Data User

Gambar 5.5 dapat dilihat proses penghapusan *user* untuk skenario 3. Dengan melihat hasil pengujian pada ketiga skenario diatas, bisa disimpulkan bahwa Kasus Penggunaan RBAC telah bekerja dengan baik seperti yang diharapkan.

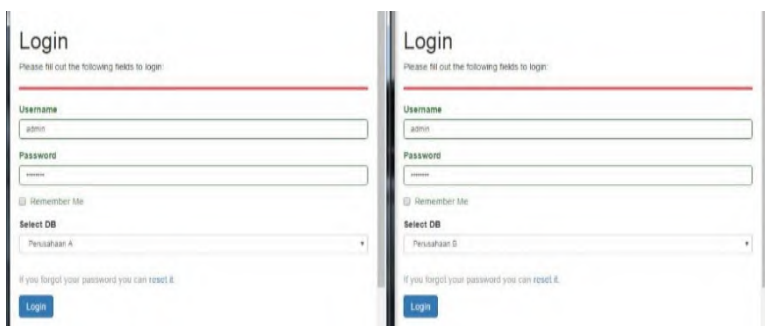
5.2.4 Pengujian Fungsional *Multitenancy*

Pengujian ini terdiri dari pengujian mengelola *multitenancy*. Pada pengujian *multitenancy*, *tenant* baru yang akan mendaftarkan *tenant*-nya untuk dapat mengakses sistem memasukkan data berupa nama basis data yang akan dibuat. Apabila basis data berhasil dibuat, maka *tenant* dapat mulai mengakses dengan pengguna awal sebagai administrator. Rincian skenario pengujian pada fitur *multitenancy* dapat dilihat pada Tabel 5.27, Gambar 5.6, dan Gambar 5.7.

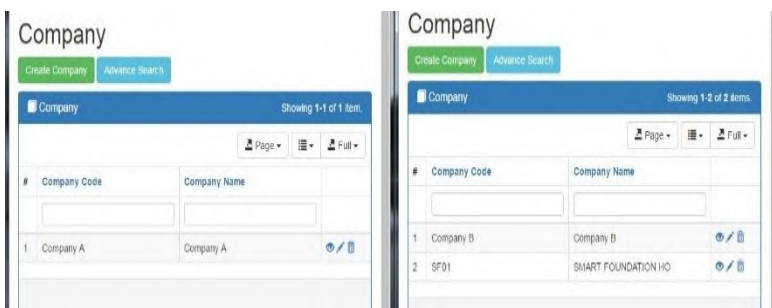
Tabel 5.27. Pengujian Fitur *Multitenancy*

ID	UJ.MTC
Referensi	Multitenancy
Nama	Mengelola Multitenancy
Tujuan Pengujian	Menguji kemampuan sistem dalam mengelola Multitenancy
Skenario 1	Pengguna mengakses Tenant
Kondisi Awal	Pengguna berada pada halaman <i>login</i>

Data Uji	Data <i>Tenant</i>
Langkah Pengujian	Pengguna memasukkan data uji ke form yang tersedia dan memilih tombol login
Hasil Yang Diharapkan	Pengguna berada di halaman utama dengan data <i>tenant</i> -nya
Hasil Yang Didapat	Pengguna berada di halaman utama dengan data <i>tenant</i> -nya
Hasil Pengujian	Berhasil
Kondisi Akhir	Pengguna berada pada halaman utama



Gambar 5.6. Pengujian *Multitenancy* Berada di Halaman *Login*



Gambar 5.7. Hasil Pengujian *Multitenancy*

5.2.5 Pengujian Fungsionalitas Basis Data Terdistribusi

Pengujian Basis Data Terdistribusi dilakukan untuk menguji 2 (dua) aspek utama dalam konsep Basis Data

Terdistribusi, yaitu replikasi dan *high availability*. Sistem Basis Data Terdistribusi menggunakan 4 (empat) *server* fisik. Dengan rincian sebagai berikut:

1. *Server* aplikasi, adalah *server* yang di dalamnya terdapat *web server* beserta file aplikasi ERP.
2. *Management node*, adalah *server* yang berfungsi sebagai pusat pengaturan sistem Basis Data Terdistribusi.
3. *Data node*, adalah 2 (dua) buah *server* yang berfungsi sebagai penyimpan data.

5.2.5.1 Replikasi

Replikasi adalah penyalinan tabel-tabel pada basis data ke beberapa *node* fisik yang tersebar. Operasi basis data yang terjadi pada sistem akan didistribusikan ke seluruh *node* data yang terlibat. Sehingga setiap *node* data memiliki struktur tabel dan data yang identik. Berikut langkah-langkah pengujian mekanisme replikasi:

1. Memastikan seluruh *server* dan sistem Basis Data Terdistribusi dalam keadaan *online*. Pada Gambar 5.8, ditunjukkan *tenant* yang sedang *online*.

Warehouse 3

ID	3
Company	EZTENANT BIKE INDONESIA
Warehouse Name	EZERP Testing BDT

Gambar 5.8. Pengujian Fitur Replikasi pada Sistem

2. Melakukan penambahan, perubahan, dan penghapusan data melalui aplikasi ERP yang ditunjukkan pada Gambar 5.9.

```

master@master-Aspire-M3970: ~
mysql> use test
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from lwm_warehouse where 1;
+-----+
-----+
| id | company_id | ws_name                | created_at      | updated_at      |
+-----+
| 3  | 1          | EZERP Testing BDT     | NULL            | NULL            |
| 1  | 1          | Maju                  | 2016-03-22 21:08:14 | 2016-03-22 21:08:14 |
| 2  | 1          | EZERP BIKE INDONESIA 2 | NULL            | NULL            |
+-----+
3 rows in set (0.01 sec)

mysql>

```

Gambar 5.9. Pengujian Replikasi pada Database Server1

3. Melakukan pengecekan terhadap hasil penambahan, perubahan, dan penghapusan data pada seluruh data *node* yang ditampilkan pada Gambar 5.10.

```

root@node2-Aspire-M3970: /home/node2
mysql> use test;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from lwm_warehouse where 1;
+-----+
-----+
| id | company_id | ws_name                | created_at      | updated_at      |
+-----+
| 3  | 1          | EZERP Testing BDT     | NULL            | NULL            |
| 1  | 1          | Maju                  | 2016-03-22 21:08:14 | 2016-03-22 21:08:14 |
| 2  | 1          | EZERP BIKE INDONESIA 2 | NULL            | NULL            |
+-----+
3 rows in set (0.00 sec)

mysql>

```

Gambar 5.10. Pengujian Fitur Replikasi pada Database Server2

5.2.5.2 *High Availability*

Aspek lain dari Basis Data Terdistribusi adalah *high availability* adalah kemampuan sistem basis data untuk tetap berjalan normal pada kondisi sebuah *node data* tidak aktif. Operasi basis data yang terjadi akan dieksekusi pada *node data* yang aktif. Di lain sisi, *management node* akan menciptakan *log* operasi-operasi yang terjadi. Sehingga operasi-operasi basis data dapat dieksekusi pada *node data* yang kembali aktif. Berikut langkah-langkah pengujian aspek *high availability*:

1. Mematikan *server* aplikasi, *management server* dan sistem Basis Data Terdistribusi pada salah satu *node*. Ditampilkan pada Gambar 5.11.

Warehouse 3

ID	3
Company	EZTENANT BIKE INDONESIA
Warehouse Name	EZERP Testing BDT HA

Gambar 5.11. Pengujian Fitur *High-Availability*

2. Melakukan penambahan, perubahan, dan penghapusan data melalui aplikasi yang ditunjukkan pada Gambar 5.12 dan Gambar 5.13.

```

root@node2-Aspire-M3970: /home/node2
id=3 @10.151.64.203 (mysql-5.1.73 ndb-7.1.34, Nodegroup: 0, *)
[ndb_mgmd(MGM)] 1 node(s)
id=1 @10.151.64.181 (mysql-5.1.73 ndb-7.1.34)

[mysqlld(API)] 2 node(s)
id=4 @10.151.64.182 (mysql-5.1.73 ndb-7.1.34)
id=5 @10.151.64.203 (mysql-5.1.73 ndb-7.1.34)

ndb_mgm> show
Cluster Configuration
-----
[ndbd(NDB)] 2 node(s)
id=2 (not connected, accepting connect from 10.151.64.182)
id=3 @10.151.64.203 (mysql-5.1.73 ndb-7.1.34, Nodegroup: 0, *)

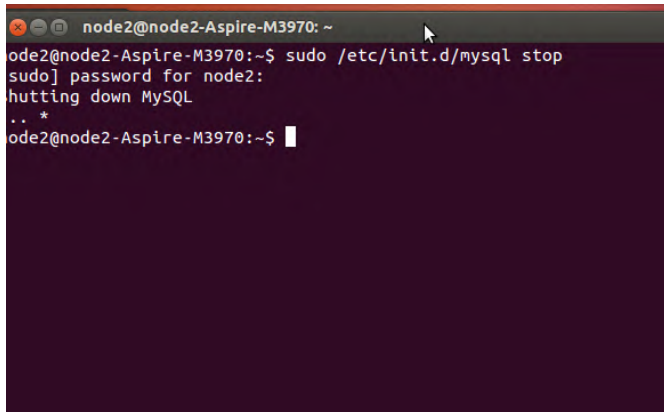
[ndb_mgmd(MGM)] 1 node(s)
id=1 @10.151.64.181 (mysql-5.1.73 ndb-7.1.34)

[mysqlld(API)] 2 node(s)
id=4 (not connected, accepting connect from 10.151.64.182)
id=5 @10.151.64.203 (mysql-5.1.73 ndb-7.1.34)

ndb_mgm>

```

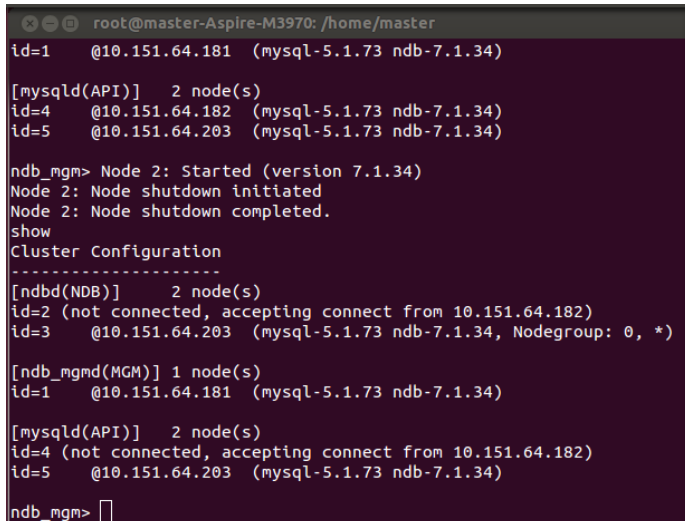
Gambar 5.12. Pengecekan *Availability* pada *Node Server*



```
node2@node2-Aspire-M3970: ~
ode2@node2-Aspire-M3970:~$ sudo /etc/init.d/mysql stop
sudo] password for node2:
hutting down MySQL
..*
ode2@node2-Aspire-M3970:~$
```

Gambar 5.13. Pengecekan Basis Data pada Node1

3. Melakukan pengecekan terhadap hasil pemrosesan data pada *node* yang masih aktif yang ditampilkan pada Gambar 5.14 dan Gambar 5.15.



```
root@master-Aspire-M3970: /home/master
id=1 @10.151.64.181 (mysql-5.1.73 ndb-7.1.34)

[mysqld(API)] 2 node(s)
id=4 @10.151.64.182 (mysql-5.1.73 ndb-7.1.34)
id=5 @10.151.64.203 (mysql-5.1.73 ndb-7.1.34)

ndb_mgm> Node 2: Started (version 7.1.34)
Node 2: Node shutdown initiated
Node 2: Node shutdown completed.
show
Cluster Configuration
-----
[ndbd(NDB)] 2 node(s)
id=2 (not connected, accepting connect from 10.151.64.182)
id=3 @10.151.64.203 (mysql-5.1.73 ndb-7.1.34, Nodegroup: 0, *)

[ndb_mgmd(MGM)] 1 node(s)
id=1 @10.151.64.181 (mysql-5.1.73 ndb-7.1.34)

[mysqld(API)] 2 node(s)
id=4 (not connected, accepting connect from 10.151.64.182)
id=5 @10.151.64.203 (mysql-5.1.73 ndb-7.1.34)

ndb_mgm>
```

Gambar 5.14. Pengecekan Availability pada Node Server

```

master@master-Aspire-M3970: ~
| tax
| uon
| user
+-----+
228 rows in set (0.01 sec)

mysql> select * from iwm_warehouse where 1;
+-----+-----+-----+-----+-----+
| id | company_id | ws_name          | created_at          | updated_at          |
+-----+-----+-----+-----+-----+
| 3 | 1 | EZERP Testing BDT HA | NULL                | NULL                |
+-----+-----+-----+-----+-----+
| 1 | 1 | Maju                | 2016-03-22 21:08:14 | 2016-03-22 21:08:14 |
+-----+-----+-----+-----+-----+
| 2 | 1 | EZERP BIKE INDONESIA 2 | NULL                | NULL                |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>

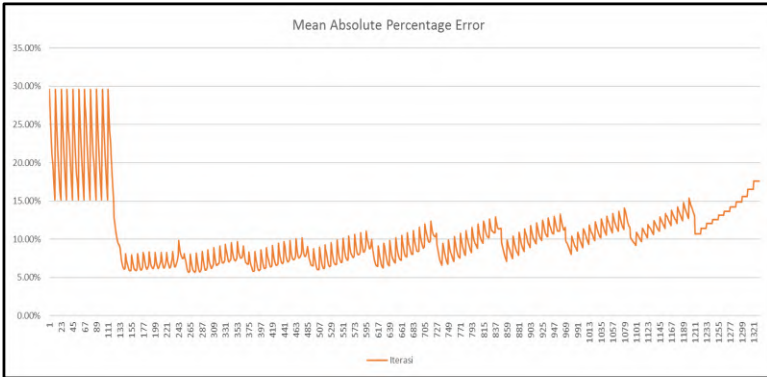
```

Gambar 5.15. Mematikan Basis Data Server pada Node2

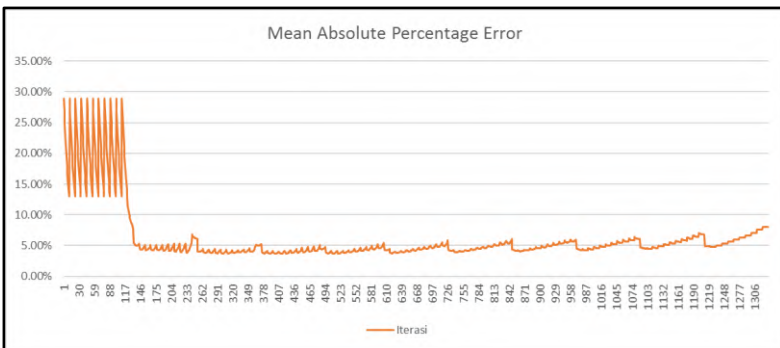
5.2.6 Pengujian Peramalan Penjualan

Skenario uji coba metode *Holt-Winters Additive* dan *Holt-Winters Multiplicative* menggunakan data masukan yang berada di Lampiran B.1. Data yang digunakan adalah 60 data yang terdiri dari 48 data latih dan 12 data uji. Hasil skenario pengujian nilai *error* yang ditampilkan menggunakan metode *Holt-Winters Additive* ditunjukkan pada lampiran D.1 dan hasil skenario pengujian nilai *error* menggunakan metode *Holt-Winters Multiplicative* ditunjukkan pada lampiran D.2.

Dari hasil uji coba menggunakan metode *Holt-Winters Multiplicative* dan *Holt-Winters Additive* menggunakan 1331 iterasi yang dilakukan dari masing-masing nilai α , β , dan γ . Dengan menggunakan *Holt-Winter Multiplicative* ditemukan MAPE minimum pada iterasi ke-262 dengan nilai α 0.2, β 0.1, dan γ 0.8, dengan nilai MAPE 5.69%. Dengan menggunakan *Holt-Winter Additive* ditemukan MAPE minimum pada iterasi ke-411 dengan nilai α 0.3, β 0.4, dan γ 0.3, dengan nilai MAPE 3.66%. Grafik hasil pengujian masing-masing metode ditunjukkan pada Gambar 5.16 dan Gambar 5.17.



Gambar 5.16. Hasil Uji Coba Menggunakan Metode *Holt-Winters Additive*



Gambar 5.17. Hasil Uji Coba Menggunakan Metode *Holt-Winters Multiplicative*

5.3 Evaluasi Pengujian

Pada subbab ini akan diberikan hasil evaluasi dari pengujian-pengujian yang telah dilakukan. Evaluasi yang diberikan meliputi evaluasi pengujian kebutuhan fungsional dan evaluasi pengujian kegunaan.

5.3.1 Evaluasi Pengujian Fungsionalitas Sistem

Rangkuman mengenai hasil pengujian fungsionalitas sistem dapat dilihat pada Tabel 5.28. Berdasarkan data pada tabel tersebut, semua skenario pengujian berhasil dan program berjalan dengan baik. Sehingga bisa ditarik disimpulkan bahwa fungsionalitas dari program telah bisa bekerja sesuai dengan yang diharapkan.

Tabel 5.28. Evaluasi Pengujian Fungsionalitas Sistem

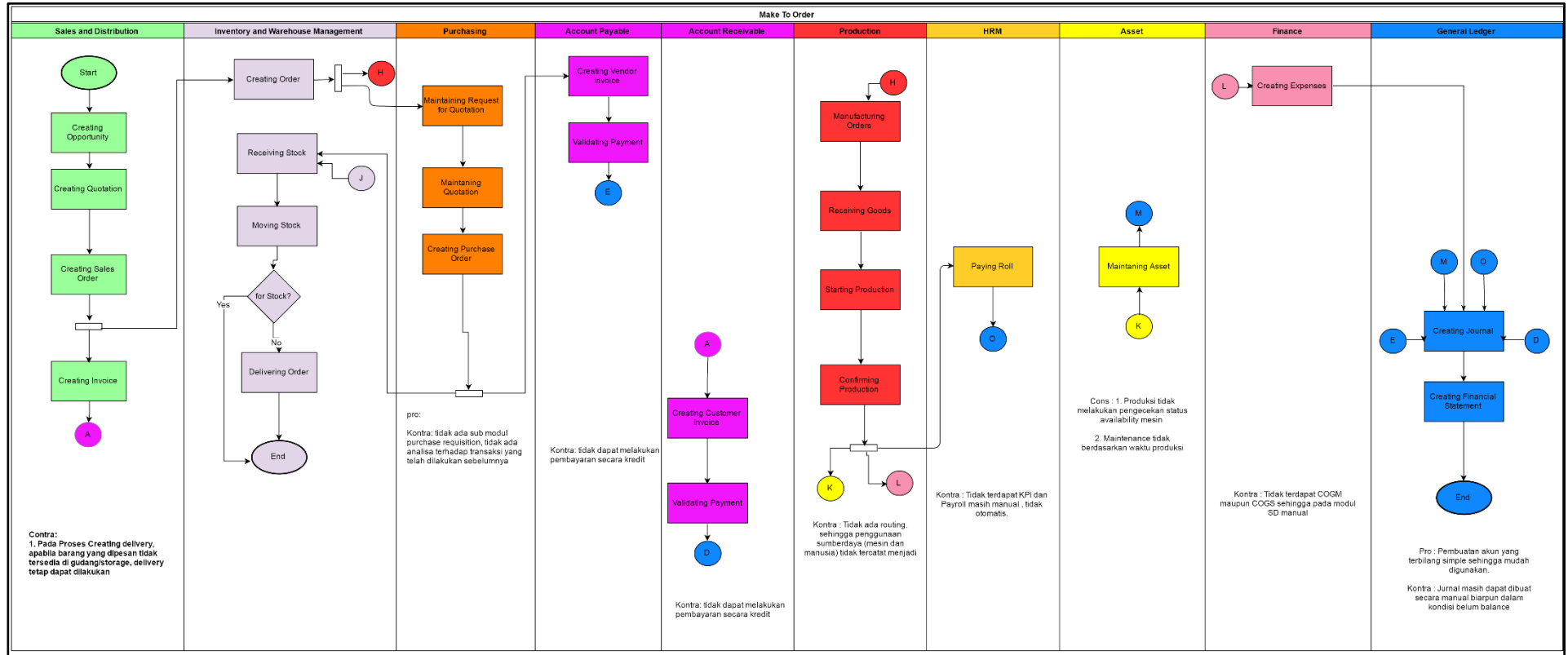
No.	ID	Nama Kasus Penggunaan	Skenario	Hasil
1	UJ.UC-001	Mengelola rute pengiriman	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
2	UJ.UC-002	Mengelola narahubung	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
3	UJ.UC-003	Mengelola pelanggan	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
4	UJ.UC-004	Mengelola <i>inquiry</i>	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
5	UJ.UC-005	Mengelola <i>quotation</i>	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
6	UJ.UC-006	Mengelola <i>sales order</i>	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
7	UJ.UC-007	Mengelola <i>return order</i>	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
8	UJ.UC-008	Mengelola pengiriman	Skenario 1	Berhasil
			Skenario 2	Berhasil

			Skenario 3	Berhasil
9	UJ.UC-009	Mengelola peramalan penjualan	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
10	UJ.UC-010	Melihat rute pengiriman	Skenario 1	Berhasil
11	UJ.UC-011	Melihat narahubung	Skenario 1	Berhasil
12	UJ.UC-012	Melihat pelanggan	Skenario 1	Berhasil
13	UJ.UC-013	Melihat <i>inquiry</i>	Skenario 1	Berhasil
14	UJ.UC-014	Melihat <i>quotation</i>	Skenario 1	Berhasil
15	UJ.UC-015	Melihat <i>sales order</i>	Skenario 1	Berhasil
16	UJ.UC-016	Melihat <i>return order</i>	Skenario 1	Berhasil
17	UJ.UC-017	Melihat pengiriman	Skenario 1	Berhasil
18	UJ.UC-018	Melihat permalan penjualan	Skenario 1	Berhasil
19	UJ.UC-019	Melihat laporan penjualan	Skenario 1	Berhasil
20	UJ.UC-020	Melihat laporan pelanggan	Skenario 1	Berhasil
21	UJ.RBAC	Mengelola RBAC	Skenario 1	Berhasil
			Skenario 2	Berhasil
			Skenario 3	Berhasil
22	UJ.MTC	Mengelola Multitenancy	Skenario 1	Berhasil

[Halaman ini sengaja dikosongkan]

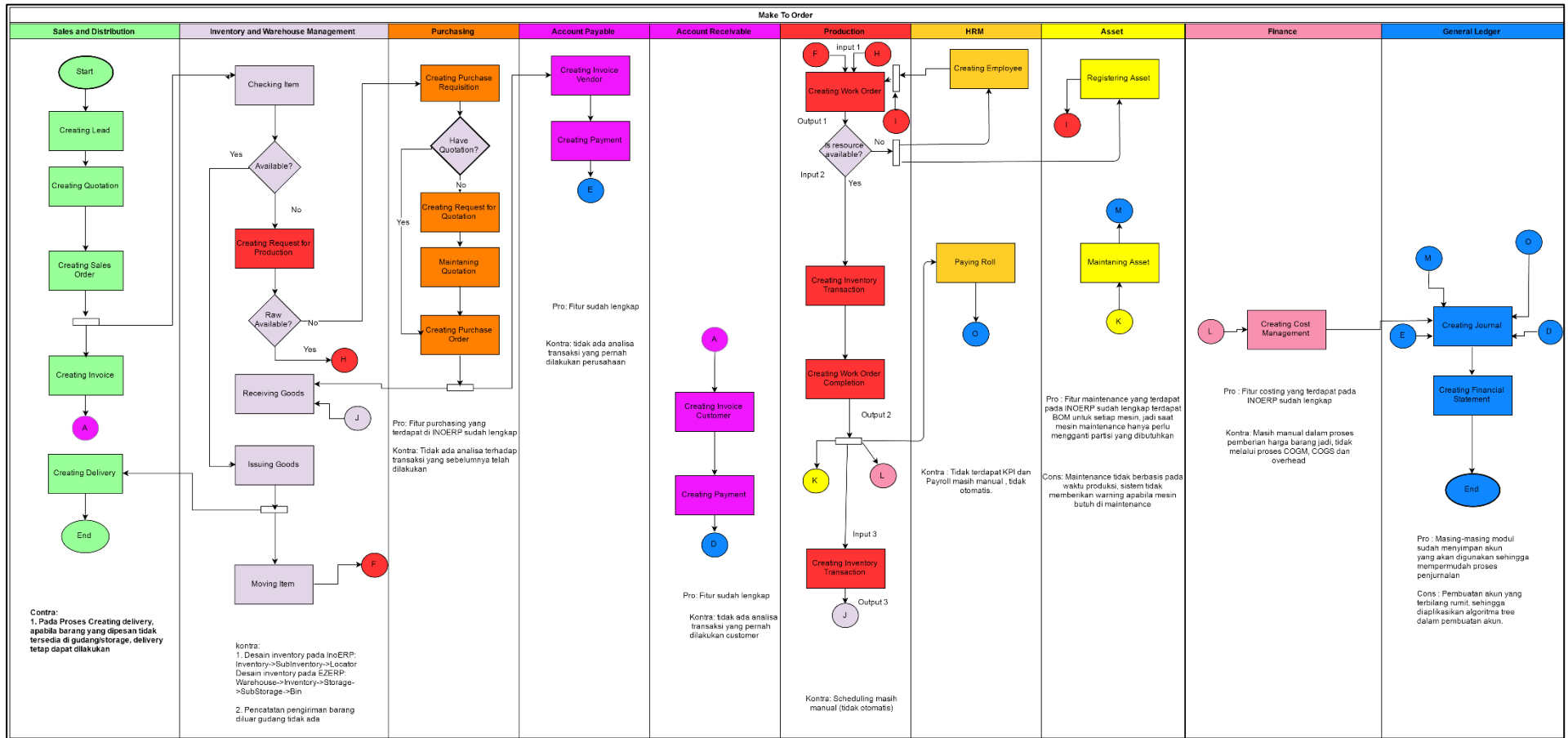
LAMPIRAN A – PROSES BISNIS

A.1 Proses Bisnis Odoo



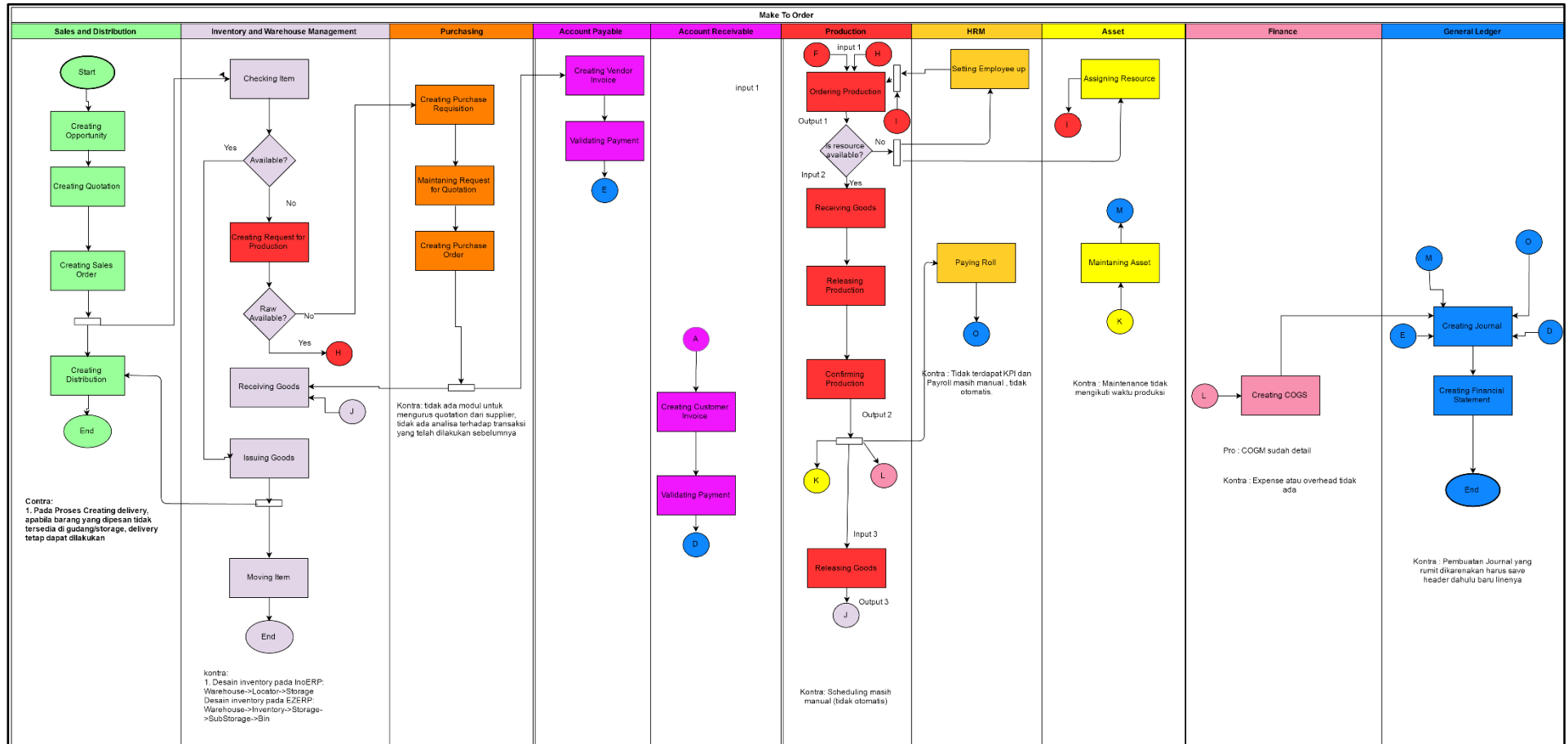
Gambar A.1. Proses Bisnis Odoo

A.2 Proses Bisnis InoERP



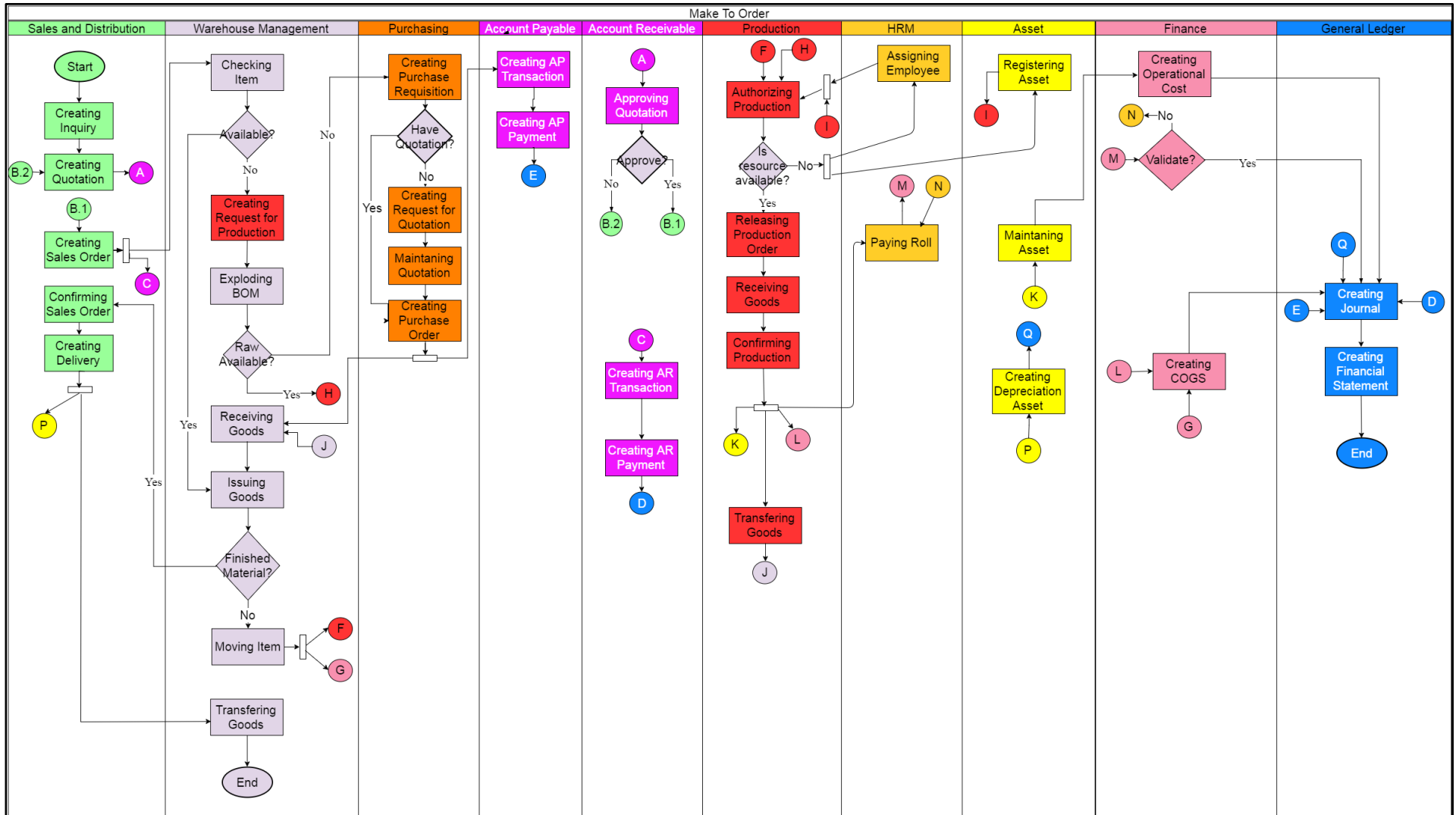
Gambar A.2. Proses Bisnis InoERP

A.3 Proses Bisnis Adempiere



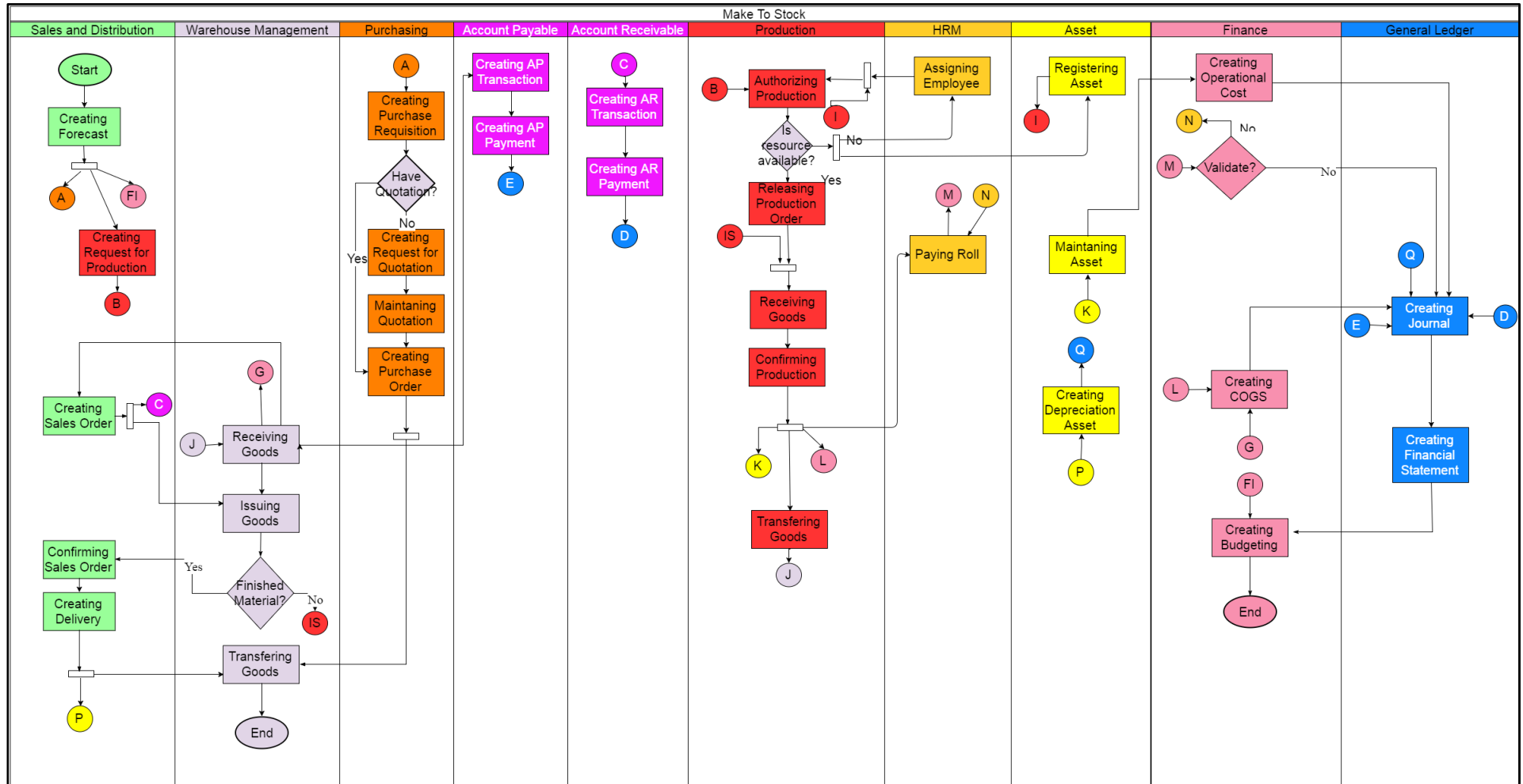
Gambar A.3. Proses Bisnis Adempiere

A.4 Proses Bisnis Make-To-Order ERP 2016



Gambar A.4. Proses Bisnis Make-To-Order ERP 2016

A.5 Proses Bisnis Make-To-Stock ERP 2016



Gambar A.5. Proses Bisnis Make-To-Stock ERP 2016

[Halaman ini sengaja dikosongkan]

LAMPIRAN B – DATA PERAMALAN PENJUALAN

B.1 Data Masukkan Peramalan Penjualan

Tabel B.1. Data Latih

Tahun	Bulan	Permintaan
2011	Juli	14629
2011	Agustus	14405
2011	September	15661
2011	Oktober	14700
2011	November	14563
2011	Desember	16365
2012	Januari	13968
2012	Februari	14762
2012	Maret	12903
2012	April	14537
2012	Mei	14634
2012	Juni	16825
2012	Juli	13414
2012	Agustus	14023
2012	September	15342
2012	Oktober	14081
2012	November	14128
2012	Desember	15851
2013	Januari	13245
2013	Februari	14636
2013	Maret	11913
2013	April	13035
2013	Mei	14084
2013	Juni	16773
2013	Juli	14341
2013	Agustus	13006
2013	September	15646
2013	Oktober	14036
2013	November	14276
2013	Desember	15360

2014	Januari	14706
2014	Februari	13936
2014	Maret	12186
2014	April	13699
2014	Mei	14713
2014	Juni	16975
2014	Juli	14222
2014	Agustus	14790
2014	September	16121
2014	Oktober	13790
2014	November	14429
2014	Desember	16108
2015	Januari	13827
2015	Februari	14464
2015	Maret	11415
2015	April	14173
2015	Mei	14963
2015	Juni	16417

Tabel B.2. Data Uji

Tahun	Bulan	Permintaan
2015	Juli	13179
2015	Agustus	14680
2015	September	15311
2015	Oktober	14723
2015	November	14102
2015	Desember	15593
2016	Januari	14925
2016	Februari	14331
2016	Maret	11127
2016	April	13109
2016	Mei	14565
2016	Juni	15528

LAMPIRAN C – KODE SUMBER

C.1 Kode Sumber Halaman Utama Modul

Kode sumber halaman utama modul berfungsi untuk menampilkan ringkasan informasi mengenai penjualan dan pelanggan yang terdapat pada modul *Sales and Distribution*. Pengaturan halaman utama *Sales and Distribution* diatur pada fungsi `actionIndex()` yang ditunjukkan pada Kode Sumber C.1.

```
1 public function actionIndex()
2 {
3     $customer = SdCustomer::find()->count();
4
5     $net = SdSale::find()->where('sd_sale.sd_
6     salestype_id = 1 AND MONTH(created_at)
7     ='. $thismonth->sum('net_sale');
8
9     $sale = SdSale::find()->where('sd_salestype_id
10    = 1')->count();
11
12    $return = SdSale::find()-
13    >where('sd_salestype_id = 2')->count();
14
15    return $this->render('index', [
16        'customer' => $customer,
17        'net' => $net,
18        'sale' => $sale,
19        'return' => $return,
20    ]);
21 }
```

Kode Sumber C.1. Halaman Utama Modul

Pada Kode Sumber C.1, ditunjukkan potongan kode sumber untuk halaman utama modul *Sales and Distribution*. `actionIndex()` adalah fungsi yang digunakan untuk menampilkan data-data yang dibutuhkan yang tersimpan di dalam basis data untuk ditampilkan ke halaman utama modul *Sales and Distribution*.

C.2 Kode Sumber Melihat Daftar Data Submodul

Kode sumber melihat daftar data submodul berfungsi untuk menampilkan daftar data submodul yang terdapat pada modul *Sales and Distribution*. Pengaturan melihat daftar data submodul diatur pada fungsi `actionIndex()` yang ditunjukkan pada Kode Sumber C.2.

```

1 public function actionIndex()
2 {
3     $searchModel = new SdCpSearch();
4     $dataProvider = $searchModel->search(Yii::$app->request->queryParams);
5
6     return $this->render('index', [
7         'searchModel' => $searchModel,
8         'dataProvider' => $dataProvider,
9     ]);
10 }

```

Kode Sumber C.2. Melihat Daftar Data Submodul

Pada Kode Sumber C.2, ditunjukkan potongan kode sumber untuk melihat daftar data submodul yang terdapat pada modul *Sales and Distribution*. `actionIndex()` adalah fungsi yang digunakan untuk menampilkan daftar data submodul yang dibutuhkan yang tersimpan di dalam basis data untuk ditampilkan ke halaman daftar data submodul.

C.3 Kode Sumber Melihat Detail Data Submodul

Kode sumber melihat detail data submodul berfungsi untuk menampilkan detail data submodul yang terdapat pada modul *Sales and Distribution*. Pengaturan melihat daftar data submodul diatur pada fungsi `actionView()` yang ditunjukkan pada Kode Sumber C.3.

```

1 public function actionView($id)
2 {
3     $model = $this->findModel($id);
4     return $this->render('view', [
5         'model' => $this->findModel($id),
6     ]);

```

7	}
---	---

Kode Sumber C.3. Melihat Detail Data Submodul

Pada Kode Sumber C.3, ditunjukkan potongan kode sumber untuk melihat detail data submodul yang terdapat pada modul *Sales and Distribution*. `actionView()` adalah fungsi yang digunakan untuk menampilkan detail data submodul yang dibutuhkan yang tersimpan di dalam basis data untuk ditampilkan ke halaman detail data submodul.

C.4 Kode Sumber Menambah dan Menyunting Data Submodul

Kode sumber menambah dan menyunting data submodul berfungsi untuk menambah dan menyunting data submodul yang terdapat pada modul *Sales and Distribution*. Pengaturan menambah data submodul diatur pada fungsi `actionCreate()` dan menyunting data submodul diatur pada fungsi `actionUpdate()` yang ditunjukkan pada Kode Sumber C.4.

1	<code>public function actionCreate()</code>
2	<code>{</code>
3	<code> \$model = new SdCp();</code>
4	
5	<code> if (\$model->loadAll(Yii::\$app->request->post())</code>
	<code> && \$model->saveAll()) {</code>
6	<code> return \$this->redirect(['view', 'id' =></code>
	<code> \$model->id]);</code>
7	<code> } else {</code>
8	<code> return \$this->render('create', [</code>
9	<code> 'model' => \$model,</code>
10	<code>]);</code>
11	<code> }</code>
12	<code>}</code>
13	
14	<code>public function actionUpdate(\$id)</code>
15	<code>{</code>
16	<code> \$model = \$this->findModel(\$id);</code>
17	
18	<code> if (\$model->loadAll(Yii::\$app->request->post())</code>
	<code> && \$model->saveAll()) {</code>
19	<code> return \$this->redirect(['view', 'id' =></code>
	<code> \$model->id]);</code>

```

20         } else {
21             return $this->render('update', [
22                 'model' => $model,
23             ]);
24         }
25     }

```

Kode Sumber C.4. Menambah dan Menyunting Data Submodul

Pada Kode Sumber C.4, ditunjukkan potongan kode sumber untuk menambah dan menyunting data submodul yang terdapat pada modul *Sales and Distribution*. `actionCreate()` adalah fungsi yang digunakan untuk menambahkan data submodul ke dalam basis data, yang kemudian ditampilkan ke halaman detail data submodul. `actionUpdate()` adalah fungsi yang digunakan untuk menyunting data submodul yang ada di dalam basis data, yang kemudian ditampilkan ke halaman detail data submodul.

C.5 Kode Sumber Menghapus Data Submodul

Kode sumber menghapus data submodul berfungsi untuk menghapus data submodul yang terdapat pada modul *Sales and Distribution*. Pengaturan melihat daftar data submodul diatur pada fungsi `actionDelete()` yang ditunjukkan pada Kode Sumber C.5.

```

1 public function actionDelete($id)
2 {
3     $this->findModel($id)->deleteWithRelated();
4     return $this->redirect(['index']);
5 }

```

Kode Sumber C.5. Menghapus Data Submodul

Pada Kode Sumber C.5, ditunjukkan potongan kode sumber untuk menghapus data submodul yang terdapat pada modul *Sales and Distribution*. `actionDelete()` adalah fungsi yang digunakan untuk menghapus data submodul yang tersimpan di dalam basis data untuk ditampilkan ke halaman daftar data submodul.

C.6 Kode Sumber Menambah *Forecast*

Kode sumber menambah *forecast* berfungsi untuk menambah peramalan penjualan dan menghitung hasil peramalan penjualan. Pengaturan menambah *forecast* diatur pada fungsi `actionCreate()`, `actionInitLevel()`, `actionInitTrend()`, `actionInitSeason()`, `actionBuild()`, `actionError()`, `actionForecast()` yang ditunjukkan pada Kode Sumber C.6.

```

1 public function actionCreate()
2 {
3     $model = new SdForecast();
4
5     if ($model->loadAll(Yii::$app->request->post())
6     && $model->saveAll()) {
7         $count = 1;
8         for ($i = count($data); $i < count($data) +
9         $period; $i++) {
10            $forecast = $this->actionForecast($i,
11            $data, $period, $build);
12            $count++;
13        }
14        return $this->redirect(['view', 'id' => $model-
15        >id]);
16    } else {
17        return $this->render('create', [
18            'model' => $model,
19        ]);
20    }
21 }
22
23 private function actionInitLevel()
24 {
25     $levels = array();
26     $sum = 0;
27     for ($i = 0; $i < $period; $i++) {
28         $levels[] = null;
29         $sum += $series[$i]['demand_forecastli
30         ne'];
31     }
32     $levels[$period-1] = $sum / $period;
33     return $levels;
34 }
35
36 private function actionInitTrend()
37 {

```

```

33     $trends = array();
34     for ($i = 0; $i < $period; $i++) {
35         $trends[] = null;
36     }
37     $trends[$period-1] = ($series[$period -
1] ['demand_forecastline']
-
$series[0] ['demand_forecastline']) / ($period -
1);
38     return $trends;
39 }
40
41 private function actionInitSeason()
42 {
43     $seasonals = array();
44     for ($i = 0; $i < $period; $i++) {
45         $seasonals[] =
$series[$i] ['demand_forecastline'] /
$level[$period-1];
46     }
47     return $seasonals;
48 }
49
50 public function actionBuild()
51 {
52     for ($i = $period - 1; $i < count($series) - 1;
53         $i++) {
54         $l = $alpha * ($xi - $s0) + (1 - $alpha)
* ($l0 + $t0);
55         $t = $beta * ($l - $l0) + (1 - $beta) *
$t0;
56         $s = $gamma * ($xi - $l) + (1 - $gamma)
* $s0;
57     }
58     return $result;
59 }
60 public function actionError()
61 {
62     $mape = $total_error / (count($series) -
$period);
63
64     return min($mape);
65 }
66
67 public function actionForecast()
68 {
69     $m = $t - count($data) + 1;
70
71     $i = count($data) - 1;
72     $j = $i - $period + (($m - 1) % $period) + 1;

```


73	
74	<code>\$forecast = (\$build[\$i-1]['new_level'] + \$m * \$build[\$i-1]['new_trend']) + \$build[\$j-1]['new_seasonal'];</code>
75	<code>return \$forecast;</code>
76	<code>}</code>

Kode Sumber C.6. Menambah Forecast

Pada Kode Sumber C.6, ditunjukkan potongan kode sumber untuk menambahkan dan menghitung peramalan penjualan. `actionCreate()` adalah fungsi yang digunakan untuk menyimpan data ke dalam basis data dan memanggil fungsi-fungsi lain untuk melakukan perhitungan peramalan penjualan. `actionInitLevel()` adalah fungsi yang digunakan untuk menghitung nilai awal pemulusan dalam metode *Holt-Winters* yang ditunjukkan pada Persamaan 2.2 dan Persamaan 2.6. `actionInitTrend()` adalah fungsi yang digunakan untuk menghitung nilai awal *trend* dalam metode *Holt-Winters* yang ditunjukkan pada Persamaan 2.3 dan Persamaan 2.7. `actionInitSeason()` adalah fungsi yang digunakan untuk menghitung nilai awal musiman dalam metode *Holt-Winters* yang ditunjukkan pada Persamaan 2.4 dan Persamaan 2.8. `actionBuild()` adalah fungsi yang digunakan untuk menghitung peramalan untuk beberapa iterasi sesuai dengan jumlah periode peramalan. `actionError()` adalah fungsi yang digunakan untuk menghitung nilai error pada peramalan dan berfungsi untuk mencari nilai error terkecil. `actionForecast()` adalah fungsi yang digunakan untuk menghitung hasil akhir peramalan yang ditunjukkan pada Persamaan 2.1 dan Persamaan 2.5.

[Halaman ini sengaja dikosongkan]

LAMPIRAN D – HASIL UJI PERAMALAN

D.1 Hasil Uji Peramalan Metode *Holt-Winters Additive*

Tabel D.1. Hasil Uji Metode *Holt-Winter Additive*

iterasi	alpha	beta	gamma	error
1	0.0	0.0	0.0	29.56%
2	0.0	0.0	0.1	27.70%
3	0.0	0.0	0.2	25.94%
4	0.0	0.0	0.3	24.27%
5	0.0	0.0	0.4	22.69%
6	0.0	0.0	0.5	21.20%
7	0.0	0.0	0.6	19.81%
8	0.0	0.0	0.7	18.51%
9	0.0	0.0	0.8	17.31%
10	0.0	0.0	0.9	16.20%
...				
260	0.2	0.1	0.6	5.84%
261	0.2	0.1	0.7	5.71%
262	0.2	0.1	0.8	5.69%
263	0.2	0.1	0.9	5.70%
264	0.2	0.1	1.0	5.81%
...				
1328	1.0	1.0	0.7	17.57%
1329	1.0	1.0	0.8	17.57%
1330	1.0	1.0	0.9	17.57%
1331	1.0	1.0	1.0	17.57%

D.2 Hasil Uji Peramalan Metode *Holt-Winters Multiplicative*

Tabel D.2. Hasil Uji Metode Holt-Winters Multiplicative

iterasi	alpha	beta	gamma	error
1	0.0	0.0	0.0	28.87%
2	0.0	0.0	0.1	26.76%
3	0.0	0.0	0.2	24.78%
4	0.0	0.0	0.3	22.90%
5	0.0	0.0	0.4	21.15%
6	0.0	0.0	0.5	19.51%
7	0.0	0.0	0.6	17.98%
8	0.0	0.0	0.7	16.57%
9	0.0	0.0	0.8	15.28%
10	0.0	0.0	0.9	14.10%
...				
409	0.3	0.4	0.1	3.75%
410	0.3	0.4	0.2	3.69%
411	0.3	0.4	0.3	3.66%
412	0.3	0.4	0.4	3.68%
413	0.3	0.4	0.5	3.70%
...				
1328	1.0	1.0	0.7	8.07%
1329	1.0	1.0	0.8	8.07%
1330	1.0	1.0	0.9	8.07%
1331	1.0	1.0	1.0	8.07%

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini akan diberikan kesimpulan yang diambil selama pengerjaan tugas akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap tugas akhir ini di masa yang akan datang.

6.1 Kesimpulan

Dari hasil pengamatan selama proses perancangan, implementasi, dan pengujian perangkat lunak yang dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Pengembangan sistem ERP pada modul *Sales and Distribution* dapat mempermudah proses penjualan barang dalam hal efisiensi dan *monitoring*, serta membantu dalam menangani pelanggan yang melakukan transaksi dengan perusahaan.
2. Peramalan penjualan dalam sebuah perusahaan dapat berguna sebagai penentuan kedepan akan permintaan barang oleh pelanggan. Peramalan penjualan dapat diimplementasikan menggunakan metode *Holt-Winters* yang menggunakan *exponential smoothing* dan dapat menangkap musiman pada penjualan.
3. *Multitenancy* memungkinkan pengguna untuk bisa mengakses aplikasi yang sama akan tetapi dengan data dari masing-masing *tenant*, Dalam pengaplikasian *multitenancy* dibantu dengan adanya *Role Based Access Control (RBAC)* untuk dapat membedakan kepemilikan data.
4. Basis data terdistribusi digunakan sebagai solusi untuk dapat menyelesaikan masalah ketika salah satu *tenant* mengalami *fail-over* dan *tenant* lain tetap dapat mengakses aplikasi, dengan menggunakan replikasi dan fragmentasi yang disediakan oleh *MySQL Cluster*.

6.2 Saran

Berikut merupakan beberapa saran untuk pengembangan sistem di masa yang akan datang. Saran-saran ini didasarkan pada hasil perancangan, implementasi dan pengujian yang telah dilakukan sebagai berikut:

1. Pengembangan peramalan penjualan yang dapat dipengaruhi oleh faktor-faktor eksternal seperti, kondisi pasar, naik-turun nilai mata uang, dan pengaruh ekonomi lainnya.

DAFTAR PUSTAKA

- [1] "Icmrindia," 2 6 2016. [Online]. Available: <http://www.icmrindia.org/courseware/Sales-and-Distribution-Management1/SDM-Contents.htm>.
- [2] T. K. Panda and S. Sahadev, *Sales and Distribution Management*, Oxford: Oxford University Press, 2005.
- [3] A. V. Allorerung, *Rancang Bangun Aplikasi Berorientasi Arsitektur Service (SOA) dengan Pendekatan Workflow pada Domain Customer Relationship Management (CRM) untuk Sistem Enterprise Resource Planning*, Surabaya: Institut Teknologi Sepuluh Nopember, 2013.
- [4] S. R. Magal and J. Word, *Integrated Business Processes with ERP Systems*, New Jersey: Wiley and Sons, Inc., 2012.
- [5] G. Williams, *Implementing SAP ERP Sales & Distribution*, McGraw-Hill Education, 2008.
- [6] "Pi Accounting," 12 6 2016. [Online]. Available: <http://sistem-akuntansi.com/sistem-akuntansi/sales-order/>.
- [7] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*, otext.com, 2013.
- [8] K. Roebuck, *Multitenancy: High-impact Strategies - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*, Queensland: Emereo Publishing, 2012.
- [9] D. Ferraiolo, D. R. Kuhn and R. Chandramouli, *Role-based Access Control*, Artech House, 2003.
- [10] H. Grubb and A. Mason, "Long lead-time forecasting of UK air passengers by Holt–Winters methods with damped trend," *International Journal of Forecasting*, vol. 17, no. 1, pp. 71-82, 2001.
- [11] M. T. Özsu and P. Valduriez, *Principles of Distributed Database Systems*, New York: Springer, 2011.

BIODATA PENULIS



Vicky Chrystian Sugiarto atau yang biasa disapa dengan nama Vicky, lahir di Kota Malang pada tanggal 5 Juni 1994 sebagai anak ketiga dari pasangan Andri Agus dan Nataliliana. Setelah menempuh pendidikan formal di SD Katolik St. Fransiskus Lawang, SMP Katolik Budi Mulia Lawang dan SMA Negeri 1 Lawang, penulis melanjutkan pendidikan tinggi di S1 Teknik Informatika Fakultas Teknologi Informasi - Institut Teknologi Sepuluh Nopember pada tahun 2012. Selama kuliah, penulis aktif dalam kegiatan robotika dan bergabung ke dalam tim robotika ITS.

Dalam menyelesaikan pendidikan S1, penulis mengambil bidang minat Manajemen Informasi (MI) dan memiliki ketertarikan di bidang robotika, *web development*, dan analisis bisnis. Penulis dapat dihubungi melalui email: vickychrystian@gmail.com.