

**RANCANG BANGUN SISTEM INFORMASI
PENYIMPANAN BARANG JADI PADA DIVISI
WAREHOUSE FINISHED GOODS MENGGUNAKAN
PHP 5.6.8 DAN MYSQL 5.6.24 PADA PT REKADAYA
MULTI ADIPRIMA**

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Penyelesaian Jenjang Diploma
Empat (D-4) Program Studi Sistem Informasi Industri Otomotif pada Politeknik
STMI Jakarta

OLEH

RISKA FITRI YANTI

1313014



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA**

2017

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas limpahan nikmat, rahmat serta karunia yang senantiasa diberikan sehingga penulis dapat menyelesaikan Laporan Tugas Akhir dengan judul “RANCANG BANGUN SISTEM INFORMASI PENYIMPANAN BARANG JADI PADA DIVISI *WAREHOUSE FINISHED GOODS* MENGGUNAKAN PHP 5.6.8 DAN MYSQL 5.6.24 PADA PT REKADAYA MULTI ADIPRIMA”. Laporan Tugas Akhir ini diajukan untuk memenuhi salah satu syarat penyelesaian Jenjang D-4 Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

Laporan Tugas Akhir ini tidak akan terwujud tanpa adanya bantuan, bimbingan, dukungan, doa dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Orang tua, Bapak Haryanto S dan Ibu Mutirah tercinta yang selalu mendoakan dan selalu memberikan dukungan baik moril maupun materil serta sebagai inspirasi selama penulisan Tugas Akhir.
2. Dr. Mustofa, ST, MT, selaku Direktur Politeknik Sekolah Tinggi Manajemen Industri.
3. Bapak Drs. Jacob Saragih, MM dan Ibu Noveriza Yuliasari, S.Si, MT selaku Ketua dan Sekretaris Program Studi Sistem Informasi Industri Otomotif.
4. Ibu Fifi L. Hadianastuti, S.Kom, M.Kes selaku dosen Pembimbing yang telah membantu penyelesaian Laporan Tugas Akhir.
5. Ibu Triana Fatmawati, ST, MT, selaku asisten dosen Pembimbing yang telah membantu penyelesaian Laporan Tugas Akhir.
6. Ibu Rosalina Faried selaku Presiden Direktur PT Rekadaya Multi Adiprima yang telah mengizinkan penulis melakukan penelitian dan pengamatan di perusahaan tersebut.

7. Bapak Herry selaku *Supervisor Warehouse* di PT Rekadaya Multi Adiprima yang telah membimbing dan membantu dalam pengumpulan data selama menjalankan Praktik Kerja Lapangan dan penulisan Tugas Akhir.
8. Seluruh karyawan di PT Rekadaya Multi Adiprima atas bantuan yang telah diberikan selama berlangsungnya kegiatan Praktik Kerja Lapangan.
9. Seluruh dosen Politeknik Sekolah Tinggi Manajemen Industri yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
10. Kepada teman-teman “Karang Taruna” Anisa, Devi, Anggyta atas dukungan dan kebersamaannya selama masa perkuliahan di Politeknik STMI Jakarta
11. Seluruh rekan-rekan mahasiswa Politeknik Sekolah Tinggi Manajemen Industri Program Studi Sistem Informasi khususnya untuk Kelas SA01 dan Agam N K yang telah memberikan semangat dan motivasi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik
12. Serta semua pihak yang baik langsung maupun tidak langsung memberikan kritik, saran dan bantuan dalam pembuatan laporan ini yang tidak bisa penulis sebutkan satu per satu.

Penulis menyadari Laporan Tugas Akhir ini masih jauh dari sempurna, untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan di masa yang akan datang.

Jakarta, 13 September 2017

Penulis

ABSTRAK

PT Rekadaya Multi Adiprima (RMA) merupakan perusahaan manufaktur yang bergerak di bidang industri komponen otomotif yang fokus pada *stamping parts*, *insulator dash panel*, dan *furniture parts*. Komponen otomotif yang diproduksi oleh PT RMA dibuat sesuai dengan merek dan jenis kendaraan bermotor. PT RMA memiliki gudang yang menyimpan barang jadi hasil produksi atau disebut dengan gudang barang jadi (*warehouse finished goods*). Proses pencatatan data *finished goods* masuk dan keluar masih dilakukan dengan penulisan tangan yaitu pada kartu *stock*. Penyimpanan data masih manual yaitu menggunakan arsip yang diletakkan pada dinding rak gudang sehingga menyebabkan kesulitan dalam pencarian data dan dapat terjadi kehilangan/kerusakan dokumen yang bisa menghambat proses pengolahan data. Untuk mengatasi permasalahan tersebut maka perlu dibuat sebuah sistem informasi penyimpanan barang jadi pada bagian *warehouse finished goods* yang dapat memberikan informasi dengan lebih efektif dan akurat. Metodologi pengembangan sistem informasi yang digunakan pada penelitian ini adalah *evolutionary prototype*. Analisis dan perancangan sistem informasi penyimpanan barang jadi menggunakan analisis perancangan berorientasi objek dan menggunakan *tools* pemodelan *Unified Modelling Language* (UML). Sistem informasi ini dibangun dengan menggunakan PHP 5.6.8 sebagai bahasa pemrograman dan MySQL 5.6.24 sebagai perangkat lunak manajemen basis datanya. Sistem ini diharapkan dapat membantu proses memasukan data barang jadi yang masuk dan yang keluar dengan di *input* ke dalam sistem, sehingga dapat mengurangi penggunaan kertas. Selain itu, berfungsi juga sebagai tempat penyimpanan data untuk mencegah terjadinya kehilangan data.

Kata kunci: Sistem Informasi, Penyimpanan Barang Jadi, *Warehouse Finished Goods*, Kartu Stok, *Evolutionary Prototype*, *Unified Modeling Language*, PHP, MySQL.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN DENGAN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN DENGAN ASISTEN DOSEN PEMBIMBING	
LEMBAR PERNYATAAN KEASLIAN	
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiv
DAFTAR TABEL.....	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Pokok Permasalahan	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat Tugas Akhir	4
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI.....	7
2.1 Pengertian Sistem	7
2.1.1 Elemen Sistem	7
2.1.2 Karakteristik Sistem	8
2.2 Pengertian dan Konsep Dasar Informasi	10
2.2.1 Siklus Informasi.....	10

2.2.2	Kualitas Informasi	11
2.3	Pengertian Sistem Informasi.....	11
2.3.1	Komponen Dasar Sistem Informasi.....	12
2.3.2	Klasifikasi dan Tujuan Sistem Informasi	13
2.4	Pengertian Penyimpanan	13
2.4.1	Tujuan Penyimpanan	14
2.4.2	Konsep 6S.....	14
2.4.3	Metode Penyimpanan	15
2.5	Pengertian <i>Finished Good</i>	16
2.6	Pengertian <i>Warehouse</i>	16
2.6.1	Jenis-jenis Gudang.....	17
2.6.2	Fungsi Gudang.....	17
2.7	<i>System Development Life Cycle (SDLC)</i>	19
2.8	Model <i>Prototype</i>	21
2.8.1	<i>Prototype Evolusioner</i>	21
2.8.2	<i>Prototype Requirement</i>	22
2.9	<i>Flowchart</i>	23
2.10	<i>Unified Modelling Language (UML)</i>	25
2.10.1	<i>Use Case Diagram</i>	27
2.10.2	<i>Activity Diagram</i>	28
2.10.3	<i>Sequence Diagram</i>	29
2.10.4	<i>Class Diagram</i>	31
2.10.5	<i>Deployment Diagram</i>	32
2.11	Kamus Data	33
2.12	<i>Hierarchy plus Input-Process-Output (HIPO)</i>	34
2.13	PHP (<i>Hypertext Pre-Processor</i>)	37
2.13.1	Keunggulan PHP	37
2.14	MySQL	38
2.14.1	Keunggulan MySQL.....	38

2.14.2	Tipe Data MySQL	40
BAB III	METODOLOGI PENELITIAN	42
3.1	Metodologi Penelitian.....	42
3.2	Jenis dan Sumber Data.....	42
3.3	Metode Pengumpulan Data.....	43
3.4	Metode Pengembangan Sistem.....	44
3.5	Analisis dan Perancangan Sistem	44
3.5.1	Analisis dan Perancangan Berorientasi Objek.....	44
3.6	Kerangka Penelitian	45
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA.....	49
4.1	Sekilas Perusahaan.....	49
4.2	Profil Umum Perusahaan	49
4.3	Visi dan Misi Perusahaan	50
4.4	Struktur Organisasi Perusahaan	51
4.4.1	Tugas dan Wewenang setiap Jabatan di PT RMA	51
4.5	Struktur Organisasi pada Divisi <i>Warehouse</i>	56
4.5.1	Tugas dan Wewenang pada Divisi <i>Warehouse</i> pada PT RMA	57
4.6	Produk pada PT Rekadaya Multi Adiprima.....	59
4.7	Penyimpanan.....	62
4.7.1	Siklus Penyimpanan	62
4.8	Sistem Penyimpanan Hasil Produksi yang Berjalan	63
4.9	Dokumen pada Sistem Penyimpanan Barang Jadi yang Berjalan	63
4.10	Prosedur Penyimpanan Barang Jadi	67
4.11	Penggambaran Sistem Penyimpanan Barang Jadi dengan UML (<i>Unified Modeling Language</i>).....	70
4.12	Analisis Sistem Informasi Penyimpanan Barang Jadi yang	

	Berjalan	74
BAB V	ANALISIS DAN PEMBAHASAN	75
5.1	Analisis Sistem Informasi Berjalan	75
5.2	Analisis Kebutuhan Sistem.....	75
5.3	Analisis Kebutuhan Rinci Sistem	76
5.4	Perancangan <i>Flowmap</i> Sistem Usulan.....	78
5.5	Analisis Sistem Informasi Usulan	81
5.5.1	<i>Use Case Diagram</i>	81
5.5.2	<i>Activity Diagram</i>	93
5.5.3	<i>Sequence Diagram</i>	103
5.5.4	<i>Class Diagram</i>	113
5.5.5	Kamus Data	114
5.5.6	<i>Deployment Diagram</i>	120
5.5.7	HIPO (<i>Hierarchy plus Input-Process-Output</i>)	121
5.5.8	<i>Flowchart</i> Aplikasi Usulan.....	122
5.5.9	Perancangan <i>Interface</i> Aplikasi Usulan	124
5.5.10	Implementasi Sistem.....	134
BAB VI	KESIMPULAN DAN SARAN	135
6.1	Kesimpulan	135
6.2	Saran	135
	DAFTAR PUSTAKA	136
	LAMPIRAN.....	L-1

DAFTAR TABEL

		Halaman
Tabel II.1	Simbol <i>Flow Direction</i>	24
Tabel II.2	Simbol Proses.....	24
Tabel II.3	Simbol <i>Input dan Output</i>	25
Tabel II.4	Simbol-simbol <i>Use Case Diagram</i>	27
Tabel II.5	Simbol-simbol <i>Activity Diagram</i>	28
Tabel II.6	Simbol-simbol <i>Sequence Diagram</i>	29
Tabel II.7	Simbol-simbol <i>Class Diagram</i>	31
Tabel II.8	Tipe <i>Multiplicity</i>	32
Tabel II.9	Simbol-simbol <i>Deployment Diagram</i>	33
Tabel II.10	Contoh Kamus Data Untuk Tabel Pemasok	34
Tabel II.11	Jenis Data pada MySQL.....	40
Tabel IV.1	Definisi Aktor Penyimpanan Barang Jadi.....	70
Tabel IV.2	Definisi <i>Use Case</i> Sistem Informasi Penyimpanan Barang Jadi	71
Tabel V.1	Kebutuhan Sistem Informasi Usulan	75
Tabel V.2	Kebutuhan Rinci Sistem.....	77
Tabel V.3	Definisi Aktor <i>Use Case</i>	83
Tabel V.4	Definisi <i>Use Case Diagram</i> Mengelola Data <i>Master</i>	84
Tabel V.5	Definisi <i>Use Case Login</i>	85
Tabel V.6	Definisi <i>Use Case</i> Memasukkan Data Rekap PO	86
Tabel V.7	Definisi <i>Use Case</i> Memasukkan Data Permintaan Produksi	87
Tabel V.8	Definisi <i>Use Case</i> Mengisi Data Aktual Permintaan Produksi	87
Tabel V.9	Definisi <i>Use Case</i> Mengisi Data <i>Check Sheet Temporary</i>	88
Tabel V.10	Definisi <i>Use Case</i> Mengisi Data Barang Jadi Masuk	89
Tabel V.11	Definisi <i>Use Case</i> Mengisi Data Barang Jadi Keluar	89
Tabel V.12	Definisi <i>Use Case</i> Meng- <i>Update</i> Data Transaksi Barang Jadi	91
Tabel V.13	Definisi <i>Use Case</i> Mencetak Laporan Rekapitulasi Data Stok Barang Jadi.....	92

Tabel V.14	Tabel <i>User</i>	114
Tabel V.15	Tabel <i>Customer</i>	115
Tabel V.16	Tabel Barang Jadi.....	115
Tabel V.17	Tabel Rekap PO	116
Tabel V.18	Tabel Rekap PO Detail.....	116
Tabel V.19	Tabel Permintaan Produksi	117
Tabel V.20	Tabel Permintaan Produksi Detail	117
Tabel V.21	Tabel <i>Check Sheet Temporary</i>	118
Tabel V.22	Tabel <i>Check Sheet Temporary</i> Detail	118
Tabel V.23	Tabel Transaksi Masuk	119
Tabel V.24	Tabel Transaksi Keluar	119

DAFTAR GAMBAR

	Halaman
Gambar II.1 Karakteristik Sistem	9
Gambar II.2 Siklus Informasi	10
Gambar II.3 Interaksi Blok Sistem Informasi.....	12
Gambar II.4 Pengembangan <i>Prototype Evolutioner</i>	22
Gambar II.5 Pengembangan <i>Prototype Requirement</i>	23
Gambar II.6 Diagram UML.....	26
Gambar II.7 <i>Visual Table of Contents</i>	36
Gambar II.8 <i>Overview Diagram</i>	36
Gambar III.1 Kerangka Penelitian	48
Gambar IV.1 Struktur Organisasi Perusahaan	51
Gambar IV.2 Struktur Organisasi Divisi <i>Warehouse</i>	57
Gambar IV.3 <i>Silincer Front Floor No 1</i>	59
Gambar IV.4 <i>Pad Roof Silincer Rear</i>	60
Gambar IV.5 <i>Pad RR Wheel Opening Ext No 1</i>	60
Gambar IV.6 <i>Pad RR Wheel Opening Ext No 2</i>	61
Gambar IV.7 <i>Silincer Front Floor No 3</i>	61
Gambar IV.8 Rekap PO	64
Gambar IV.9 Form Permintaan Produksi.....	65
Gambar IV.10 <i>Check Sheet Temporary</i>	65
Gambar IV.11 Form Kartu Stok.....	66
Gambar IV.12 Data <i>Inventory</i> Barang Jadi.....	67
Gambar IV.13 <i>Flowmap</i> Sistem Informasi Penyimpanan Barang Jadi yang Berjalan	69
Gambar IV.14 <i>Use Case Diagram</i> Sistem Informasi Penyimpanan Barang Jadi yang Berjalan	73
Gambar V.1 <i>Flowmap</i> Sistem Informasi Penyimpanan Barang Jadi Usulan	80

Gambar V.2	<i>Use Case Diagram</i> Sistem Informasi Penyimpanan Barang Jadi Usulan	82
Gambar V.3	<i>Activity Diagram</i> Proses Login	94
Gambar V.4	<i>Activity Diagram</i> Proses Mengelola Data <i>Master</i>	95
Gambar V.5	<i>Activity Diagram</i> Proses Memasukkan Data Rekap PO	96
Gambar V.6	<i>Activity Diagram</i> Proses Memasukkan Data Permintaan Produksi	97
Gambar V.7	<i>Activity Diagram</i> Proses Mengisi Data Aktual Permintaan Produksi	98
Gambar V.8	<i>Activity Diagram</i> Proses Mengisi Data CST.....	99
Gambar V.9	<i>Activity Diagram</i> Proses Mengisi Data Barang Jadi Masuk	100
Gambar V.10	<i>Activity Diagram</i> Proses Mengisi Data Barang Jadi Keluar	101
Gambar V.11	<i>Activity Diagram</i> Proses Meng- <i>Update</i> Data Stok Barang Jadi ..	102
Gambar V.12	<i>Activity Diagram</i> Proses Mencetak Laporan Rekapitulasi Data Stok Barang Jadi	103
Gambar V.13	<i>Sequence Diagram</i> Proses Login	104
Gambar V.14	<i>Sequence Diagram</i> Proses Mengelola Data <i>Master</i>	105
Gambar V.15	<i>Sequence Diagram</i> Proses Memasukkan Data Rekap PO	106
Gambar V.16	<i>Sequence Diagram</i> Proses Memasukkan Data Permintaan Produksi	107
Gambar V.17	<i>Sequence Diagram</i> Proses Mengisi Data Aktual Permintaan Produksi	108
Gambar V.18	<i>Sequence Diagram</i> Proses Mengisi Data CST.....	109
Gambar V.19	<i>Sequence Diagram</i> Proses Mengisi Data Barang Jadi Masuk	110
Gambar V.20	<i>Sequence Diagram</i> Proses Mengisi Data Barang Jadi Keluar	111
Gambar V.21	<i>Sequence Diagram</i> Proses Meng- <i>Update</i> Data Stok Barang Jadi	112
Gambar V.22	<i>Sequence Diagram</i> Proses Mencetak Laporan Data Stok Barang Jadi.....	112
Gambar V.23	<i>Class Diagram</i> Usulan	113
Gambar V.24	<i>Deployment Diagram</i> Sistem Informasi Penyimpanan Barang	

Jadi Usulan	120
Gambar V.25 HIPO Sistem Informasi Penyimpanan Barang Jadi Usulan	121
Gambar V.26 <i>Flowchart</i> Sistem Informasi Penyimpanan Barang Jadi Usulan..	123
Gambar V.27 Rancangan <i>Form Login</i>	124
Gambar V.28 Rancangan Tampilan Menu Utama.....	125
Gambar V.29 Rancangan Tampilan Menu Data <i>User</i>	125
Gambar V.30 Rancangan Tampilan Menu Data Rekap PO	126
Gambar V.31 Rancangan Tampilan Menu Data Rekap PO Detail.....	127
Gambar V.32 Rancangan Tampilan Menu Data Permintaan Produksi	128
Gambar V.33 Rancangan Tampilan Menu Data Permintaan Produksi Detail....	129
Gambar V.34 Rancangan Tampilan Menu Data <i>Check Sheet Temporary</i>	130
Gambar V.35 Rancangan Tampilan Menu Data CST Detail.....	131
Gambar V.36 Rancangan Tampilan Menu Data Transaksi Masuk	132
Gambar V.37 Rancangan Tampilan Menu Data Transaksi Keluar	133
Gambar V.38 Rancangan Tampilan Menu Laporan Data Transaksi Barang Jadi	134

BAB I

PENDAHULUAN

1.1 Latar Belakang

Suatu perusahaan memiliki sistem yang dapat mengelola segala aktivitas guna menjalankan organisasinya, salah satunya adalah sistem informasi. Perusahaan memerlukan sistem informasi yang mampu mengkoordinasikan informasi internal dan informasi eksternal perusahaan. Sebagai contoh dalam perusahaan seperti pada Divisi *Warehouse Finished Good* memberikan informasi berapa jumlah stok yang tersedia pada gudang dan berapa kapasitas gudang yang masih bisa digunakan untuk menyimpan barang. Informasi-informasi tersebut diharapkan dapat disampaikan kepada pihak luar Divisi *Warehouse Finished Good* itu sendiri seperti Bagian Produksi maupun Bagian Pengiriman.

Penyimpanan suatu barang merupakan suatu hal penting pada suatu perusahaan. Kegiatan penyimpanan suatu barang memiliki dampak positif berupa kondisi barang yang tetap dalam keadaan baik ketika disimpan sementara. Keadaan barang yang baik tersebut dapat meningkatkan kepuasan para *customer*, sehingga permintaan produk tersebut meningkat yang berdampak pada peningkatan pendapatan perusahaan.

Setiap perusahaan memiliki kebijakan sendiri atas fasilitas dalam penyimpanan suatu barang hasil produksi atau barang jadi berupa gudang. Gudang disini berperan penting sebagai media penyimpanan guna menyimpan barang jadi untuk menjaga kualitas barang ketika disimpan. Divisi yang bertanggung jawab terhadap hal tersebut adalah Divisi *Warehouse Finished Good*. Divisi *Warehouse Finished Good* bertugas untuk memperhatikan kualitas barang jadi ketika disimpan, serta mencatat jumlah barang masuk dan keluar. Dalam mengerjakan tugasnya Divisi *Warehouse Finished Good* memiliki sistem yang dapat memudahkan kinerja yaitu mencatat setiap barang

masuk dan barang keluar melalui *form* kartu stok. *Form* ini kemudian digunakan untuk memonitor penyimpanan suatu barang.

PT Rekadaya Multi Adiprima merupakan perusahaan yang bergerak di bidang industri komponen otomotif. Tiga diantara beberapa konsumennya adalah PT Daihatsu Astra Motor, Toyota, dan Yamaha. Dalam menjalankan berbagai bisnisnya PT Rekadaya Multi Adiprima membagi Divisi yang mengerjakan produknya, yaitu *metal division, plastic division, interior division, dan non-woven division*. Untuk tetap dapat bersaing dan bertahan, PT Rekadaya Multi Adiprima sangat memperhatikan sistem penyimpanan barang yang sedang berjalan. Sistem penyimpanan barang yang berjalan dengan baik akan menjaga kualitas barang hingga sampai ke tangan konsumen dan tentu akan mempertahankan kepercayaan konsumen terhadap perusahaan.

Pada saat ini sistem pengolahan data untuk penyimpanan barang jadi di PT Rekadaya Multi Adiprima masih dilakukan secara manual. Seperti pencatatan barang jadi yang masuk dan keluar akan dicatat dengan penulisan tangan kedalam *Form* kartu stok yang dimana dapat menimbulkan kesalahan dalam penulisan.

Selain itu masalah pada pengolahan data di kartu stok seperti tanggal masuk dan tanggal keluar barang yang dicatat ke dalam kartu stok serta jumlah barang masuk dan barang keluar dalam kurun waktu hanya sebulan untuk satu kartu stok. Hal ini menyebabkan informasi pada kartu stok tersebut terbatas hanya untuk satu bulan. Sehingga terdapat kesulitan mengakses data stok untuk bulan-bulan berikutnya.

Karena belum adanya aplikasi terkomputerisasi yang dapat membantu penyimpanan arsip dapat menyebabkan kartu stok hilang atau terselip yang tentu akan mengganggu pengaksesan informasi ketika dibutuhkan. Selain itu penyimpanan data masih menggunakan arsip yang diletakkan pada dinding rak gudang sehingga menyebabkan kesulitan dalam pencarian data dan dapat terjadi kehilangan/kerusakan dokumen yang bisa menghambat proses pengolahan data.

Berdasarkan uraian masalah sistem pengolahan data penyimpanan pada Divisi *Warehouse Finished Good*, diperlukan aplikasi untuk membantu perusahaan dalam mengatasi masalah tersebut. Penelitian Tugas Akhir ini akan menganalisis, merancang

dan membangun sistem untuk perusahaan. Adapun judul Tugas Akhir ini adalah “Rancang Bangun Sistem Informasi Penyimpanan Barang Jadi pada Divisi *Warehouse Finished Good* Menggunakan PHP 5.6.8 dan MySQL 5.6.24 pada PT Rekadaya Multi Adiprima”.

1.2 Pokok Permasalahan

Permasalahan yang terjadi pada Divisi *Warehouse Finished Good* pada PT Rekadaya Multi Adiprima adalah sebagai berikut:

1. Pencatatan barang masuk dan barang keluar pada *form* kartu stok masih dilakukan dengan penulisan tangan sehingga pembaharuan data berjalan lambat.
2. Penyimpanan data masih manual yaitu menggunakan arsip yang diletakan pada dinding rak gudang sehingga menyebabkan kesulitan dalam pencarian data dan dapat terjadi kehilangan / kerusakan dokumen yang bisa menghambat proses pengolahan data.

1.3 Tujuan Penelitian

Adapun tujuan penelitian Tugas Akhir ini adalah sebagai berikut:

1. Membuat sistem informasi penyimpanan barang jadi dimana proses pengisian *form* kartu stok menjadi terkomputerisasi untuk mempermudah proses pencatatan barang jadi masuk dan barang jadi keluar serta pembaharuan data seperti mengetahui jumlah stok.
2. Membuat sistem basis data penyimpanan barang jadi sehingga memudahkan dalam penyimpanan serta pencarian data dan mencegah terjadinya kerusakan atau kehilangan data dengan menggunakan *database* MySQL.

1.4 Batasan Masalah

Agar dalam penulisan Tugas Akhir ini lebih fokus dan terarah, maka perlu diadakan batasan-batasan sebagai berikut:

1. Penelitian dilakukan pada Divisi *Warehouse Finished Good* selama satu bulan, dari tanggal 03 Agustus – 03 September 2016.
2. Analisis dan penelitian yang dilakukan hanya sebatas menangani masalah proses penyimpanan barang jadi pada Divisi *Warehouse Finished Good*. (Tidak membahas proses Produksi dan proses Pengecekan Kualitas secara detail).

1.5 Manfaat Tugas Akhir

Adapun manfaat yang dapat diambil dari penelitian pada Tugas Akhir ini adalah sebagai berikut:

1. Bagi perusahaan
Hasil penelitian ini agar dapat diimplementasikan di perusahaan untuk membantu kinerja perusahaan dalam melakukan pengolahan data penyimpanan barang jadi yang dilakukan
2. Bagi mahasiswa
 - a. Memberikan kemampuan dalam mengaplikasikan teori secara jelas terhadap masalah yang diamati
 - b. Memberikan pembelajaran untuk menerapkan ilmu yang diperoleh selama perkuliahan serta menambah wawasan dan pengetahuan penulis, khususnya dalam analisis dan perancangan sistem informasi.
3. Bagi pihak lain
Hasil penelitian ini diharapkan dapat memberikan sumbangan pemikiran dan sebagai referensi bagi peneliti lain yang melakukan penelitian serupa.

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada di dalamnya. Agar lebih mempermudah perumusan yang akan dibahas pada Tugas Akhir ini, maka diuraikan tahapan-tahapan dalam penyusunan Tugas Akhir ini sebagai berikut:

BAB I PENDAHULUAN

Bab ini memuat latar belakang, pokok permasalahan, tujuan penelitian, batasan masalah, manfaat Tugas Akhir, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menjelaskan tentang berbagai teori yang diperoleh dari buku-buku literatur ataupun berbagai macam referensi. Teori-teori yang dipaparkan pada laporan ini adalah seputar pengertian sistem, jenis sistem, pengertian informasi, jenis-jenis informasi, pengertian penyimpanan, metodologi pengembangan sistem, *flowmap*, *Unified Modelling Language (UML)*, *Hypertext Preprocessor (PHP)*, dan *MYSQL*.

BAB III METODOLOGI PENELITIAN

Pada bab ini membahas tentang metode pengumpulan data, serta langkah-langkah yang akan dilakukan dalam perumusan dan pemecahan masalah termasuk metodologi pengembangan sistem yang digunakan.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini akan membahas mengenai data-data yang telah diperoleh berdasarkan hasil pengamatan pada objek penelitian yaitu PT Rekadaya Multi Adiprima.

BAB V ANALISIS DAN PEMBAHASAN

Pada bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis sistem yang meliputi diagram alir sistem yang berjalan, perancangan basis data, perancangan sistem dengan diagram UML, perancangan tampilan layar dan perancangan hierarki menu.

BAB V PENUTUP

Dalam bab penutup ini dikemukakan kesimpulan-kesimpulan dari hasil penelitian dan saran-saran untuk pihak perusahaan dalam berbagai hal yang berhubungan dengan sistem informasi dan saran untuk pengembangan penelitian selanjutnya pada Divisi *Warehouse Finished Good*.

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Pengertian sistem menurut beberapa ahli yaitu, menurut Sutabri (2012) pada buku Analisis Sistem Informasi, pada dasarnya sistem adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Selanjutnya menurut McLeod dikutip oleh Yakub (2012) dalam buku Pengantar Sistem Informasi mendefinisikan sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan. Sistem juga merupakan suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu.

Sedangkan menurut Jogiyanto (2005) pada buku Analisis dan Desain Sistem Informasi mendefinisikan sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

2.1.1 Elemen Sistem

Di dalam suatu sistem terdapat beberapa elemen atau unsur yang membentuk sebuah sistem, yaitu (Kadir, 2014):

1. Tujuan (*Goal*)

Setiap sistem memiliki tujuan. Tujuan bisa lebih dari satu. Tujuan inilah yang menjadi pendorong atau motivasi yang mengarahkan ke arah mana sistem bergerak. Tujuan antara satu sistem dengan sistem yang lain berbeda termasuk

berlaku juga dalam sistem informasi. Sekalipun dari setiap sistem berbeda, secara umum ada tiga macam tujuan utama yaitu (Kadir, 2014):

- a. Untuk mendukung fungsi kepengurusan manajemen.
- b. Untuk mendukung pengambilan keputusan manajemen.
- c. Untuk mendukung kegiatan operasi perusahaan.

Secara khusus, tujuan sistem informasi bergantung pada kegiatan yang ditangani. Namun, kecenderungan penggunaan sistem informasi lebih ditunjukkan pada usaha menuju keunggulan kompetitif, yaitu mampu bersaing mengungguli pesaing.

2. Masukan (*Input*)

Masukan adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses.

3. Proses (*Process*)

Merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna.

4. Keluaran (*Output*)

Merupakan hasil dari pemrosesan yang bisa berupa suatu informasi, sasaran cetakan laporan dan lain-lain.

5. Mekanisme Pengendalian dan Umpan Balik

Mekanisme pengendalian dilakukan dengan menggunakan umpan balik (*feedback*) dari keluaran.

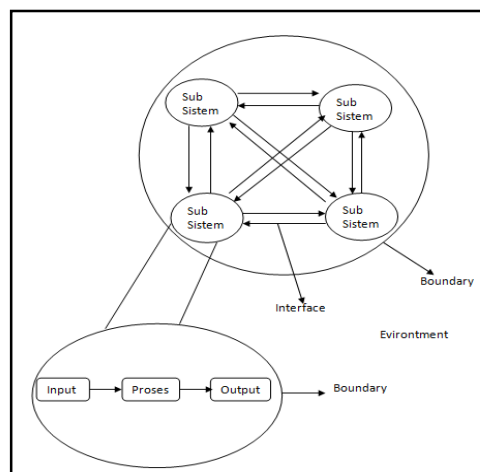
6. Batas (*Boundary*)

Batas adalah pemisah antara sistem dan daerah di luar sistem (lingkungan). Batas sistem menentukan konfigurasi, ruang lingkungan atau kemampuan sistem.

2.1.2 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, antara lain (Jogiyanto, 2005):

1. Suatu sistem mempunyai komponen-komponen sistem (*components*) atau sub-sistem-sub-sistem. Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerjasama dalam membentuk suatu kesatuan. Komponen sistem tersebut dapat berupa suatu bentuk sub-sistem.
 2. Suatu sistem mempunyai batas sistem (*boundary*). Batasan sistem membatasi antara sistem yang satu dengan yang lainnya atau sistem dengan lingkungan luarnya.
 3. Suatu sistem mempunyai lingkungan luar (*environment*). Lingkungan luar sistem adalah suatu bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut.
 4. Suatu sistem mempunyai penghubung (*interface*). Penghubung sistem merupakan media yang menghubungkan sistem dengan sub-sistem yang lain, dengan demikian dapat terjadi suatu integrasi sistem yang membentuk suatu kesatuan.
 5. Suatu sistem mempunyai tujuan (*goal*) atau sasaran sistem (*objective*). Sebuah sistem dikatakan berhasil apabila mengenai sasaran atau tujuannya, jika suatu sistem tidak mempunyai tujuan maka operasi sistem tidak akan ada gunanya.
- Untuk lebih jelasnya, karakteristik sistem dapat dilihat pada Gambar II.1



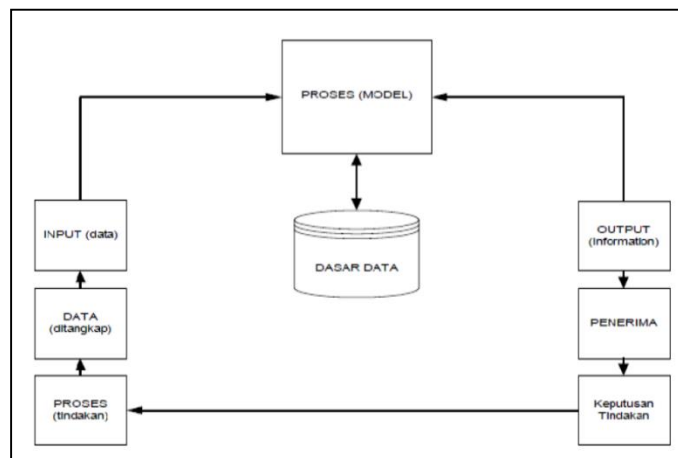
Gambar II.1. Karakteristik Sistem
(Sumber: Jogiyanto, 2005)

2.2 Pengertian dan Konsep Dasar Informasi

Menurut McLeod dikutip oleh Yakub (2012) pada buku Pengantar Sistem Informasi, informasi adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya. Sedangkan menurut Sutabri (2012) pada buku Analisis Sistem Informasi, informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

2.2.1 Siklus Informasi

Siklus informasi (*information cycle*) atau siklus pengolahan data (*data processing cycle*) adalah gambaran secara umum mengenai proses terhadap data sehingga menjadi informasi yang bermanfaat bagi pengguna. Informasi yang menghasilkan informasi berikutnya, demikian seterusnya proses pengolahan data menjadi informasi. Untuk memperoleh informasi yang bermanfaat bagi penerimanya, perlu untuk dijelaskan bagaimana siklus yang terjadi atau dibutuhkan dalam menghasilkan informasi. Siklus informasi digambarkan pada Gambar II.2 sebagai berikut (Yakub, 2012).



Gambar II.2 Siklus Informasi

Sumber: Yakub (2012)

2.2.2 Kualitas Informasi

Menurut Sutabri (2012) pada buku Analisis Sistem Informasi, kualitas dari suatu informasi tergantung dari 3 hal, yaitu informasi harus akurat (*accurate*), tepat waktu (*timeliness*) dan relevan (*relevance*).

1. Akurat (*Accuracy*)

Informasi harus bebas dari kesalahan–kesalahan dan tidak menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya.

2. Tepat Waktu (*Time Lines*)

Informasi yang datang kepada penerima tidak boleh terlambat. Informasi yang sudah usang tidak mempunyai nilai lagi, karena informasi merupakan suatu landasan dalam mengambil sebuah keputusan dimana bila pengambilan keputusan terlambat maka akan berakibat fatal untuk organisasi.

3. Relevan (*Relevance*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk setiap orang berbeda. Menyampaikan informasi tentang penyebab kerusakan mesin produksi kepada akuntan perusahaan tentunya kurang relevan. Akan lebih relevan bila ditujukan kepada ahli teknik perusahaan.

2.3 Pengertian Sistem Informasi

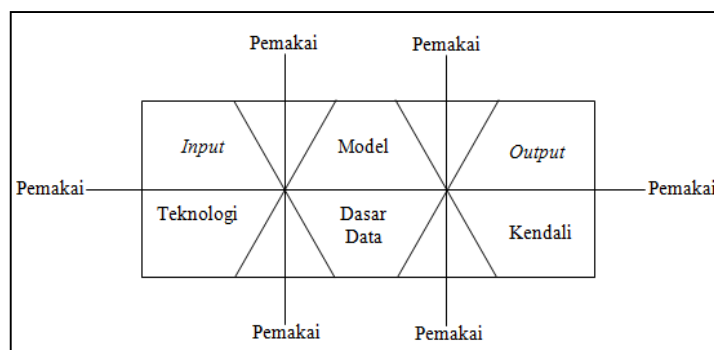
Menurut O'Brian dikutip oleh Yakub (2012) pada buku Pengantar Sistem Informasi, sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras, perangkat lunak, jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, menyebarkan informasi dalam sebuah organisasi.

Selain itu menurut Sutabri (2012) sistem informasi merupakan suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi.

2.3.1 Komponen Sistem Informasi

Menurut Yakub (2012) sistem informasi merupakan sebuah susunan yang terdiri dari beberapa komponen atau elemen. Komponen-komponen dari sistem informasi ini dapat digambarkan sebagai berikut ini:

1. Blok Masukan (*Input Block*), *input* memiliki data yang masuk ke dalam sistem informasi, juga metode-metode untuk menangkap data yang dimasukkan.
2. Blok Model (*Model Block*), blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data.
3. Blok Keluaran (*Output Block*), produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Blok Teknologi (*Technology Block*), blok teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu: teknisi (*brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*).
5. Basis Data (*Database Block*), basis data merupakan kumpulan dari data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.



Gambar II.3 Interaksi Blok Sistem Informasi
(Sumber: Yakub, 2012)

2.3.2 Klasifikasi dan Tujuan Sistem Informasi

Ada beberapa cara untuk mengelompokkan sistem informasi. Klasifikasi yang umum dipakai antara lain didasarkan pada (Sutabri, 2012):

1. Level Organisasi, berdasarkan level organisasi sistem informasi dikelompokkan menjadi sistem informasi departemen, sistem informasi perusahaan, dan sistem informasi antar organisasi.
2. Area Fungsional, sebagaimana diketahui bahwa dalam sebuah organisasi memiliki sejumlah bidang fungsional seperti akuntansi, pemasaran, produksi, dan sebagainya.

Tujuan dari sistem informasi adalah (Sutabri, 2012):

1. Menyediakan informasi untuk membantu pengambilan keputusan manajemen.
2. Membantu petugas di dalam melaksanakan operasi perusahaan dari hari ke hari.
3. Menyediakan informasi yang layak untuk pemakai pihak luar perusahaan.

2.4 Pengertian Penyimpanan

Menurut Amsyah (2003) penyimpanan merupakan rangkaian tata cara dan langkah-langkah yang harus dilaksanakan dalam menyimpan barang-barang, sehingga jika barang diperlukan lagi dapat ditemukan secara cepat.

Adapun kegiatan penyimpanan antara lain (Amsyah, 2003):

1. Menerima, menyimpan dan mengatur dan menjaga keutuhan barang dalam gudang atau ruang penyimpanan agar dapat dipergunakan sesuai dengan rencana secara tertib, rapi dan aman
2. Menyelenggarakan administrasi penyimpanan atau pergudangan atas semua barang yang ada dalam gudang
3. Melakukan *stock opname* secara berkala ataupun insidental terhadap barang persediaan yang ada di dalam gudang agar persediaan selalu dapat memenuhi kebutuhan
4. Membuat laporan secara berkala atas persediaan barang yang ada di gudang.

2.4.1 Tujuan Penyimpanan

Menurut Martono (2015) Tujuan penyimpanan *inventory* adalah menjamin:

1. Kecukupan kapasitas dan efisiensi pemakaian tempat penyimpanan
2. Pengendalian kualitas dan kuantitas material selama penyimpanan (*audit*)
3. Pasokan kebutuhan material untuk pemakai
4. Kerapihan dan perawatan tempat dan alat penyimpanan
5. Keselamatan orang dan lingkungan di sekitar tempat penyimpanan.

2.4.2 Konsep 6S

Konsep 6S (*Speed, Space, Spesification, Safety, Security and Situation*) memiliki tujuan menjalankan penyimpanan yang baik (Martono, 2015).

1. *Speed* (Kecepatan Pelayanan)

Dipengaruhi oleh frekuensi penggunaan barang, yaitu *fast moving* dan *slow moving*. Penyimpanan yang *fast moving* adalah penyimpanan yang memiliki tingkat frekuensi penerimaan dan pengiriman tinggi. Sedangkan penyimpanan *slow moving* tingkat frekuensi penerimaan dan pengiriman tidak terlalu tinggi

2. *Space* (Ruang simpan dan gerak)

Terdiri dari ruang simpan dan gerak untuk barang yang disimpan (kapasitas dan akses memadai), orang yang bertugas (keamanan dan kenyamanan), dan alat penyimpanan misal *forklift*.

3. *Spesification* (Sifat fisik)

Beberapa perusahaan tertentu barang yang disimpan di gudang memiliki sifat fisika ataupun sifat kimia yang harus dijaga baik kondisinya

4. *Safety* (Keselamatan)

Keselamatan ini mencakup barang (tidak hilang, terkontaminasi, pengecekan berkala, dan sebagainya), orang (aman dari bahaya luka), dan alat (tidak rusak akibat barang berlebih)

5. *Security* (Keamanan)

Gudang harus dirancang agar aman bagi orang yang bertugas (aman dalam menjalankan tugasnya di dalam gudang sehingga harus jelas tanggung jawab setiap orang), aman bagi barang yang disimpan sehingga terjaga dari risiko hilang, dan aman bagi peralatan yang digunakan sehingga terjaga dari risiko kehilangan

6. *Situation* (Keadaan lingkungan)

Selain dari sisi efektivitas kegunaan gudang, perusahaan perlu memperhatikan faktor-faktor lingkungan di sekitar gudang.

2.4.3 Metode Penyimpanan

Menurut Warman (2012) dalam penyimpanan barang di gudang agar nantinya barang yang disimpan tersebut tidak mengalami kerusakan dan memudahkan dalam penanganan sangatlah dibutuhkan suatu metode, cara maupun prosedur tertentu. Setidaknya dalam penyimpanan barang di gudang kita mengenal adanya dua sistem:

1. LIFO (*Last in First Out*)

LIFO adalah suatu sistem atau cara penyimpanan barang dalam gudang yaitu barang yang datang terakhir digunakan terlebih dahulu. Sistem ini biasanya digunakan untuk barang-barang yang dapat bertahan lama atau barang yang jika disimpan lebih lama kualitasnya akan lebih baik.

2. FIFO (*First in First Out*)

FIFO adalah suatu sistem penyimpanan barang yaitu barang yang masuk terlebih dahulu juga dikeluarkan terlebih dahulu. Sehingga keluar masuknya barang secara berurutan atau sesuai kronologis. Sistem ini biasanya digunakan untuk barang-barang yang kurang bisa tahan lama.

Kelebihan dari sistem *First In First Out* (FIFO), antara lain:

1. Barang akan terjaga kualitasnya
2. Pengendalian harga lebih terjamin
3. Pencatatan lebih sistematis

2.5 Pengertian *Finished Goods*

Finished Goods atau barang jadi barang yang telah diproses dengan suatu cara. Barang jenis ini adalah lawan dari bahan mentah, dan dapat terdiri dari barang setengah jadi ataupun barang jadi (Wikipedia, 2013).

Setiap perusahaan mempunyai kebijaksanaan yang berbeda-beda dalam menentukan tingkat persediaan produk barang jadi. Tujuan adanya persediaan barang jadi adalah untuk meredam fluktuasi permintaan. Persediaan dapat difungsikan untuk memenuhi kekurangan pasokan produk barang jadi di pasaran sebagai akibat permintaan yang disimpan perusahaan. Oleh karena itu tingkat persediaan produk barang jadi yang ditetapkan manajemen perusahaan memegang peran yang sangat penting dalam menjaga kestabilan pemasokan produk ke pelanggan.

2.6 Pengertian *Warehouse*

Warehouse atau Gudang merupakan suatu fasilitas yang berfungsi sebagai lokasi penyaluran barang dari *supplier* (pemasok), sampai ke *end user* (pengguna). Dalam praktik operasional setiap perusahaan cenderung memiliki suatu ketidakpastian akan permintaan. Hal ini mendorong timbulnya kebijakan dari perusahaan untuk melakukan sistem persediaan (*inventory*) agar permintaan dapat diantisipasi dengan cermat. Dengan adanya kebijakan mengenai *inventory* ini mendorong perusahaan untuk menyediakan fasilitas gudang sebagai tempat untuk menyimpan barang *inventory* (Riadi, 2016).

Siahaya (2016) menerangkan bahwa gudang adalah suatu tempat atau bangunan yang digunakan untuk menimbun, menyimpan barang baik berupa bahan baku (*raw material*), barang setengah jadi (*work-in process*) ataupun barang jadi (*finished goods*).

Menurut Martono (2015) gudang adalah tempat penyimpanan sementara dan pengambilan *inventory* untuk mendukung kegiatan operasi bagi proses operasi berikutnya, ke lokasi distribusi, atau kepada konsumen akhir. Keuntungan adanya gudang bagi *inventory* adalah:

1. Menyediakan tempat untuk meletakkan dan melindungi.
2. Menyediakan tepat waktu sesuai pesanan (menjamin *service level*).
3. Memonitor status.
4. Sebagai alat komunikasi dengan kosumen.
5. Mengurangi biaya transportasi

2.6.1 Jenis-jenis Gudang

Menurut Siahaya (2016), gudang terdiri dari 4 (empat) jenis, yaitu gudang operasional, gudang perlengkapan, gudang distribusi, dan gudang musiman.

1. Gudang Operasional

Gudang operasional digunakan untuk menyimpan bahan baku (*raw material*) yang diperlukan untuk proses dan menunjang produksi serta barang setengah jadi (*work in process*).

2. Gudang Perlengkapan

Gudang perlengkapan digunakan untuk menyimpan perlengkapan, yaitu untuk menunjang dan memperlancar proses produksi yang terletak didekat *line* produksi.

3. Gudang Distribusi

Gudang distribusi digunakan untuk menyimpan barang jadi (*finished goods*) yang siap dikirim ke distributor atau *reseller*.

4. Gudang Musiman

Gudang musiman adalah gudang yang bersifat insidental pada saat gudang operasional dan gudang distribusi penuh.

2.6.2 Fungsi Gudang

Manfaat gudang dalam logistik dapat dilihat dari dua sudut pandang, yaitu dari segi ekonomi dan pelayanan (Trisna, 2012):

1. Manfaat ekonomi

Manfaat gudang dari segi ekonomi yaitu apabila keseluruhan biaya logistik mengalami penurunan dengan adanya pemanfaatan satu atau beberapa fasilitas gudang. Empat jenis manfaat ekonomi dari pemanfaatan gudang, yaitu:

a. *Consolidation*

Pemanfaatan gudang sebagai *consolidation* yaitu gudang berfungsi sebagai tempat penerimaan dan konsolidasi material dari beberapa manufaktur (produsen). Sebelum selanjutnya akan didistribusikan kepada konsumen dalam transportasi tunggal.

b. *Break bulk and crossdock*

Pemanfaatan gudang sebagai *Break bulk* dan *crossdock* sebenarnya hampir dengan pemanfaatan gudang sebagai *consolidation*. Hanya dalam *break bulk* dan *crossdock* tidak dilakukan proses penyimpanan.

c. *Processing/Postponement*

Pemanfaatan gudang sebagai *Processing* atau *Postponement* yaitu pada gudang tersebut dilakukan proses penundaan dengan melakukan proses manufaktur ringan terhadap produk atau material.

d. *Stockpiling*

Gudang berfungsi untuk menyimpan produk yang sifatnya musiman. Ketika tiba musim dimana jumlah permintaan akan produk tinggi, dengan adanya gudang maka produk dapat didistribusikan kepada konsumen

2. Manfaat Pelayanan

Manfaat pelayanan yang diperoleh dengan pemanfaatan yang diperoleh dengan pemanfaatan gudang dalam logistik mungkin tidak dapat mengurangi biaya, namun dapat memperbaiki pelayanan dengan mengurangi waktu pengiriman dan kapabilitas tempat. Berikut ini adalah penjelasan dari manfaat pelayanan:

a. *Spot Stock*

Digunakan untuk saluran distribusi, khususnya untuk produk personal. Perusahaan lebih memilih untuk mendapatkan produknya di gudang agar dapat mengurangi waktu pengiriman produk ke konsumen

b. *Assortment*

Gudang berfungsi sebagai tempat penyimpanan kombinasi produk sebagai antisipasi terhadap permintaan konsumen.

2.7 *System Development Life Cycle (SDLC)*

System Development Life Cycle (SDLC) atau sering disebut juga *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Rosa dan Shalahuddin, 2014).

Tahapan yang ada pada SDLC secara global adalah sebagai berikut (Rosa dan Shalahudin, 2014):

1. Inisiasi (*initiation*)

Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

2. Pengembangan konsep sistem (*system concept development*)

Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis manfaat biaya, manajemen rencana dan pembelajaran kemudahan sistem.

3. Perencanaan (*planning*)

Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.

4. Analisis kebutuhan (*requirements analysis*)

Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.

5. Desain (*design*)

Mentransformasikan kebutuhan *detail* menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

6. Pengembangan (*development*)

Mengkonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan, membuat basis data dan mempersiapkan prosedur kasus pengujian, mempersiapkan berkas atau *file* pengujian, pengkodean, pengkompilasian, memperbaiki dan membersihkan program serta peninjauan pengujian.

7. Integrasi dan pengujian (*integration and test*)

Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user* sehingga menghasilkan laporan analisis pengujian.

8. Implementasi (*implementation*)

Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan luar *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9. Operasi dan pemeliharaan (*operations and maintenance*)

Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

10. Disposisi (*disposition*)

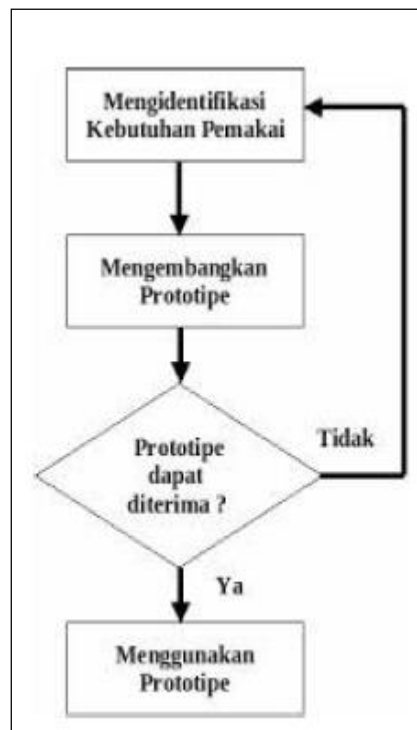
Mendeskripsikan aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

2.8 Model *Prototype*

Menurut McLeod (2011) prototipe adalah suatu versi sistem potensial yang disediakan bagi pengembang dan calon pengguna yang dapat memberikan gambaran bagaimana kira-kira sistem tersebut akan berfungsi bila telah disusun dalam bentuk yang lengkap. Proses dalam memproduksi suatu prototipe disebut *prototyping*. Tujuannya adalah menghasilkan prototipe secepat mungkin dan memperoleh umpan balik dari pengguna yang akan memungkinkan prototipe untuk ditingkatkan sampai sistem dianggap sempurna. Adapun jenis *prototype* terdapat dua jenis prototipe yaitu prototipe evolusioner (*evolutionary prototype*) dan prototipe *requirement* (*requirement prototype*).

2.8.1 *Prototype Evolusioner*

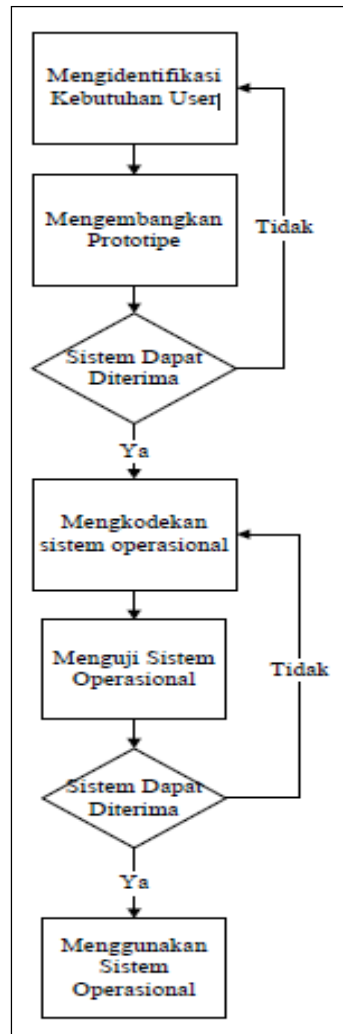
Prototype evolusioner adalah prototipe yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi. Menurut McLeod (2011) ada empat langkah yang diambil dalam mengembangkan suatu *prototype evolusioner* yaitu identifikasi kebutuhan pengguna, mengembangkan prototipe, menentukan prototipe dapat diterima atau tidak, dan penggunaan prototipe. Pengembangan Prototipe Evolusioner dapat dilihat pada Gambar II.4



Gambar II.4 Pengembangan *Prototype Evolusioner*
(Sumber: McLeod, 2011)

2.8.2 *Prototype Requirement*

Prototype requirement adalah suatu pengembangan untuk menentukan kebutuhan fungsional dari sistem baru pada saat para pengguna tidak mampu mengungkapkan dengan tepat apa yang mereka butuhkan. Saat kebutuhan telah ditentukan prototipe *requirement* dapat mulai dikerjakan dan proyek siap untuk mengembangkan suatu sistem yang baru. Pengembangan *Prototype Requirement* dapat dilihat pada Gambar II.5



Gambar II.5 Pengembangan *Prototype Requirement*
(Sumber: McLeod, 2011)

2.9 *Flowchart*

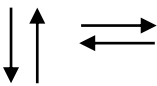


Flowchart merupakan penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program. *Flowchart* menolong analisis dan programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. (Sitorus, 2015).

Simbol-simbol *flowchart* yang digunakan merupakan simbol-simbol *flowchart* standar yang dikeluarkan oleh ANSI dan ISO. Berikut simbol-simbol standar yang digunakan untuk menggambarkan diagram alir sesuai kegunaan simbol. (Sitorus, 2015):

1. *Flow Direction Symbols*

Simbol yang dipakai untuk menghubungkan antara simbol yang satu dengan simbol lainnya atau disebut juga *connecting line*. Tabel II.1 menjelaskan tentang simbol-simbol penghubung dalam menggambarkan diagram alir.

Tabel II.1 Simbol *Flow Direction*



Simbol	Nama	Keterangan
	<i>Arus/Flow</i>	Penghubung antara prosedur/proses.
	<i>Connector</i>	Simbol keluar/masuk prosedur atau proses dalam lembar/halaman yang sama.
	<i>Off-line Connector</i>	Simbol keluar/masuk prosedur atau proses dalam lembar/halaman yang lain.

Sumber: Sitorus(2015)




2. Simbol Proses

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu prosedur. Tabel II.2 menjelaskan tentang simbol-simbol proses dalam menggambarkan diagram alir.

Tabel II.2 Simbol Proses

Simbol	Nama	Keterangan
	<i>Process</i>	Simbol yang menunjukkan pengolahan yang dilakukan komputer.
	<i>Decision</i>	Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban/aksi.

Tabel II.2 Simbol Proses (lanjutan)




	<i>Predafined Process</i>	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam <i>storage</i> .
	<i>Terminal</i>	Simbol untuk permulaan atau akhir dari suatu program.
	<i>Manual Input</i>	Simbol untuk pemasukan data secara manual <i>on-line keyboard</i> .

Sumber: Sitorus (2015)

3. Simbol *Input* dan *Output*

Simbol yang dipakai untuk menyatakan jenis peralatan yang digunakan sebagai media *input* atau *output*. Tabel II.3 menjelaskan tentang simbol-simbol *input* dan *output* dalam menggambarkan diagram alir.

Tabel II.3 Simbol *Input* dan *Output*

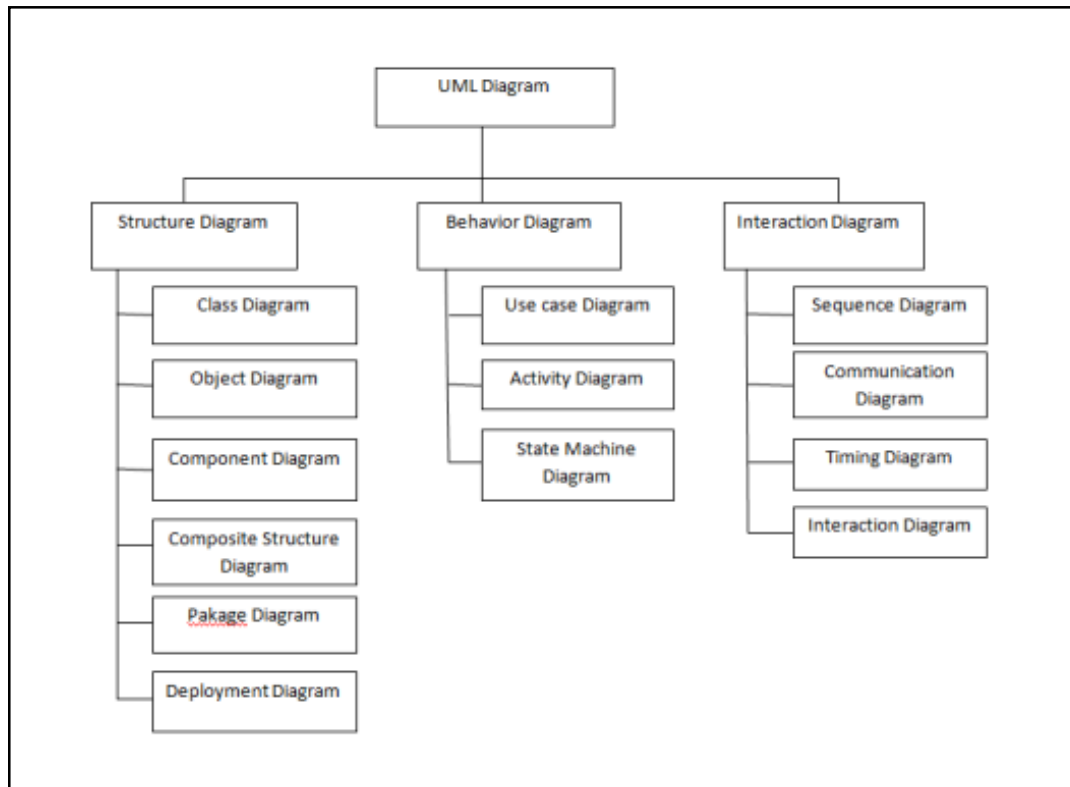
Simbol	Nama	Keterangan
	<i>Input-Output</i>	Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
	<i>Document</i>	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak di kertas.
	<i>Disk and On-line Storage</i>	Simbol untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .

Sumber: Sitorus(2015)

2.10 *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks

tertentu. Pada Gambar II.6 dijelaskan bahwa UML terdiri dari 13 macam diagram yang dikelompokkan dalam tiga kategori (Rosa dan Shalahuddin, 2014).



Gambar II.6 Diagram UML
(Sumber: Rosa dan Shalahudin, 2014)

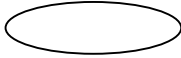
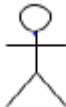


Berikut ini penjelasan singkat dari pembagian kategori UML tersebut (Rosa dan Shalahuddin, 2014):

1. *Structure diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan proses sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.



2.10.1 Use Case Diagram

Use case adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Tabel II.4 Simbol-Simbol *Use Case Diagram*

No.	Simbol	Deskripsi
1.	<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2.	<p><i>Actor</i></p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3.	<p><i>Association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	<p><i>Extend</i></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

Tabel II.4 Simbol-Simbol *Use Case Diagram* (lanjutan)






No.	Simbol	Deskripsi
5.	<i>Generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6.	<i>Include</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

Sumber: Rosa dan Shalahuddin (2014)

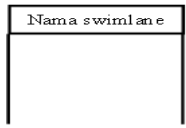
2.10.2 Activity Diagram

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Tabel II.5 Simbol-Simbol *Activity Diagram*

No	Simbol	Deskripsi
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan/decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan/join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Tabel II.5 Simbol-Simbol *Activity Diagram* (lanjutan)

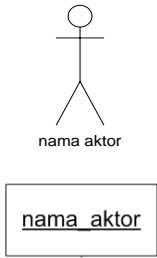

No.	Simbol	Deskripsi
6.	<p>Swimlane</p> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Sumber: Rosa dan Shalahuddin (2014)

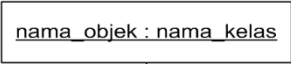

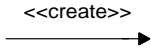
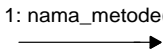
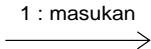
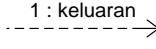
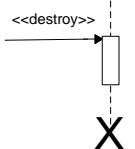
2.10.3 *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat scenario yang ada pada *use case*.

Tabel II.6 Simbol-Simbol *Sequence Diagram*

No.	Simbol	Deskripsi
1.	<p>Aktor</p> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2.	<p>Garis Hidup/<i>lifeline</i></p> 	Menyatakan kehidupan suatu objek.

Tabel II.6 Simbol-Simbol *Sequence Diagram* (lanjutan)

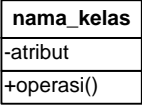




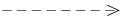

No.	Simbol	Deksripsi
3.	Objek 	Menyatakan objek yang berinteraksi pesan.
4.	Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.	Pesan tipe create 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
6.	Pesan tipe call 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode.
7.	Pesan tipe send 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8.	Pesan tipe return 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9.	Pesan Tipe Destroy 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.

Sumber: Rosa dan Shalahuddin(2014)

2.10.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. (Rosa dan Shalahudin, 2014).

Tabel II.7 Simbol-Simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	<p>Kelas</p> 	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2.	<p>Antarmuka/interface</p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	<p>Asosiasi/association</p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.	<p>Asosiasi berarah</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.	<p>Kebergantungan</p> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
7.	<p>Agregasi/aggregation</p> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

Sumber: Rosa dan Shalahuddin (2014)

Dalam kelas terdapat beberapa *keys* yang berfungsi untuk membedakan semua basis data dalam tabel secara unik. Pengertian *keys* menurut Yanto (2016) yaitu sebagai berikut:

1. *Candidate key*

Candidate key adalah sejumlah kecil *attribute* yang secara unik mengidentifikasi setiap kejadian dari setiap tipe *entity*.

2. *Primary key*

Primary key adalah *candidate key* yang terpilih untuk mendefinisikan secara unik pada setiap kejadian dari sebuah tipe *entity*.

3. *Composite key*

Composite key adalah sebuah *candidate key* yang terdiri dari dua atau banyak *attribute*.

4. *Foreign key*

Foreign key adalah himpunan *attribute* dalam data *relationship* yang cocok dengan *candidate key* dari beberapa *relationship* lainnya.

Pada relasi *class diagram* terdapat suatu penanda yang disebut *multiplicity*. *Multiplicity* ini akan mengindikasikan berapa banyak obyek dari suatu kelas terelasi ke obyek lain. Berikut Tabel II.8 tentang notasi UML untuk *multiplicity*:

Tabel II.8 Tipe *Multiplicity*

<i>Multiplicity</i>	Keterangan
*	Banyak
0	Nol
1	Satu
0..*	Antara nol sampai banyak
1..*	Antara satu sampai banyak
0..1	Nol atau satu
1..1	Tepat satu

(Sumber: Britton dan Doake, 2005)

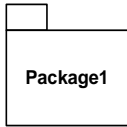
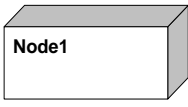
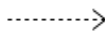

2.10.5 *Deployment Diagram*

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* juga dapat digunakan untuk memodelkan hal-hal berikut (Rosa dan Shalahuddin, 2014):

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node* dan *hardware*.
2. Sistem *client/server*.

3. Sistem terdistribusi murni.
4. Rekayasa ulang aplikasi.

Tabel II.9 Simbol-simbol *Deployment Diagram*

No.	Simbol	Deskripsi
1.	<p style="text-align: center;"><i>Package</i></p> 	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih <i>node</i>
2.	<p style="text-align: center;"><i>Node</i></p> 	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka, komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.
3.	<p style="text-align: center;"><i>Dependency</i></p> 	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai
4.	<p style="text-align: center;"><i>Link</i></p> 	Relasi antar <i>node</i>

Sumber: Rosa dan Shalahuddin (2014)

2.11 Kamus Data

Menurut Jogiyanto (2005), kamus data (*data dictionary*) adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data diharapkan, analisis sistem dapat mendefinisikan data yang mengalir dalam sistem dengan lengkap. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem.

Kamus data dapat berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan di dalam

sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses.

Keuntungan penggunaan kamus data adalah (Sommerville, 2003):

1. Kamus data merupakan mekanisme untuk manajemen nama. Banyak orang yang harus menciptakan nama untuk entitas dan relasi ketika mengembangkan model sistem yang besar. Nama-nama ini harus dipakai secara konsisten dan tidak boleh bentrok. Kamus data dapat memeriksa keunikan nama dan memberitahu analis persyaratan sekiranya terjadi duplikasi nama.
2. Kamus data sebagai tempat penyimpanan informasi yang dapat menghubungkan analis, desain, implementasi dan evolusi. Sementara sistem dikembangkan, informasi diambil untuk memberitahu perkembangan informasi baru ditambahkan pada sistem. Berikut adalah contoh penulisan kamus data:

Spesifikasi Tabel pemasok

Nama Tabel : Pemasok

Tipe : File master

Tabel II.10 Contoh Kamus Data Untuk Tabel Pemasok

No.	Nama Elemen	Akronim	Tipe	Width	Keterangan
1.	ID pemasok	ID_pemasok	Varchar	10	<i>Primary Key</i>
2.	Nama pemasok	Nama_pemasok	Char	40	
3.	Alamat pemasok	Alamat	Varchar	100	
4.	Nomor telepon	Telepon	Varchar	12	

(Sumber: Jogiyanto, 2005)

2.12 HIPO (*Hierarchy plus Input-Process-Output*)

HIPO (*Hierarchy plus Input-Process-Output*) aslinya dibuat oleh IBM sebagai alat untuk mendokumentasikan program. Bagan HIPO merupakan bagan yang memperagakan apa yang dikerjakan suatu program, data apa yang digunakan, dan keluaran yang dihasilkannya (Jogiyanto, 2005).

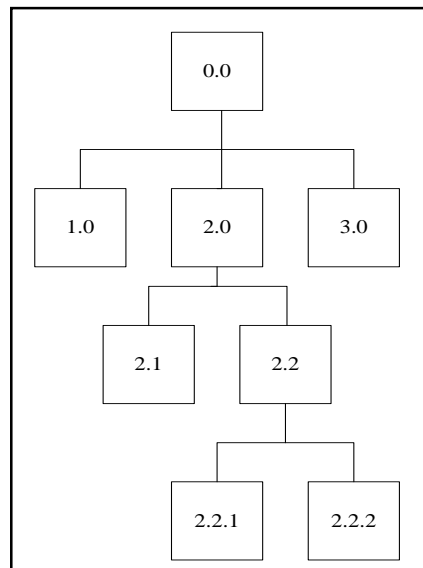
HIPO (*Hierarchy plus Input-Process-Output*) mempunyai sasaran utama sebagai berikut (Jogiyanto, 2005):

1. Untuk menyediakan suatu struktur guna memahami fungsi-fungsi dari sistem.
2. Untuk lebih menekankan fungsi-fungsi yang harus diselesaikan oleh program, bukannya menunjukkan statemen-statemen program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan yang jelas dari *input* yang harus digunakan dan *output* yang harus dihasilkan oleh masing-masing fungsi pada tiap-tiap tingkatan dari diagram-diagram HIPO.
4. Untuk menyediakan *output* yang tepat dan sesuai dengan kebutuhan-kebutuhan pemakai.

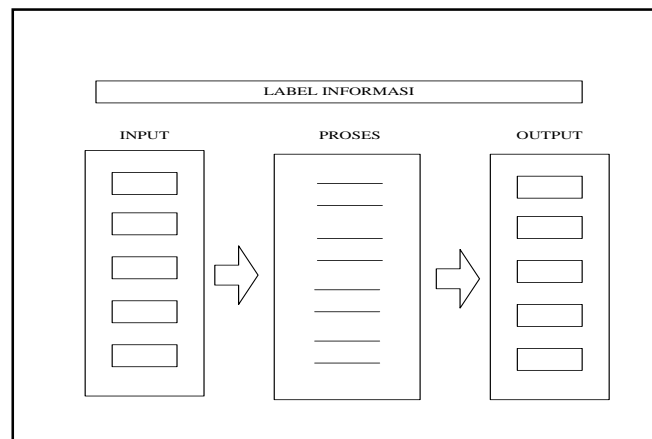
Menurut Jogiyanto (2005), HIPO dapat digunakan sebagai alat pengembangan sistem dan teknik dokumentasi program, fungsi-fungsi dari sistem digambarkan oleh HIPO dalam tiga tingkatan. Untuk masing-masing tingkatan digambarkan dalam bentuk diagram tersendiri, dengan demikian HIPO menggunakan tiga macam diagram untuk masing-masing tingkatannya, yaitu sebagai berikut:

1. *Visual Table Of Contents (VTOC)*
Visual table of contents menggambarkan hubungan fungsi-fungsi di sistem secara berjenjang, *VTOC* menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur. Pada diagram ini nama dan nomor dari program HIPO, struktur paket diagram dan hubungan fungsi diidentifikasi dalam bentuk hirarki. Contoh *Visual table of contents* dapat dilihat pada Gambar II.7
2. *Overview Diagram*
Overview Diagram menunjukkan secara garis besar hubungan dari *input*, proses dan *output*. Bagian *input* menunjukkan item-item data yang akan digunakan oleh bagian proses. Bagian proses berisi sejumlah langkah-langkah yang menggambarkan kerja dari fungsi. Bagian *output* berisi dengan item-item data

yang dihasilkan atau dimodifikasi oleh langkah-langkah proses. Contoh *Overview Diagram* dapat dilihat pada Gambar II.8



Gambar II.7 *Visual Table Of Contents*
Sumber: Jogiyanto (2005)



Gambar II.8 *Overview Diagram*
Sumber: Jogiyanto (2005)

3. *Detail Diagram*

Detail Diagram merupakan diagram tingkatan yang paling rendah di diagram HIPO. Diagram ini berisi elemen-elemen dasar dari paket yang menggambarkan secara rinci kerja dari fungsi.

2.13 PHP (*Hypertext Pre-Processor*)

PHP Hypertext Preprocessor (PHP) adalah bahasa *server-side scripting* yang digunakan untuk aplikasi web yang dinamis dan interaktif. Sebuah halaman PHP adalah sebuah halaman *HTML* yang memiliki *server-side scripts* yang ditempatkan dalam *server* dan diproses oleh web *server* sebelum dikirim ke *browser* pemakai (Welling dan Thomson, 2003).

Server-side scripts dijalankan ketika *browser* melakukan permintaan *file.php* dari *server*. PHP dipanggil oleh web *server*, dimana proses script perintah yang ada di suatu halaman dieksekusi mulai dari awal sampai akhir di dalam mesin PHP. Setelah *script* PHP tersebut diolah, hasilnya akan ditampilkan kepada *client* melalui web *browser* berupa tampilan *HTML*.

2.13.1 Keunggulan PHP

Menurut Welling dan Thomson (2003), beberapa keunggulan PHP adalah:

1. *High Performance*
PHP sangat efisien. Dengan menggunakan *server* tunggal yang tidak mahal, *user* dapat melakukan banyak pekerjaan setiap harinya.
2. *Database Integration*
PHP mempunyai sambungan ke banyak sistem basis data, antara lain MySQL, PostgreSQL, Oracle, Informix, dan *Sysbase databases*.
3. *Built-in-Libraries*
PHP dirancang khusus untuk web, dan mempunyai banyak *built-in-function* untuk menampilkan banyak fungsi di dalam web.
4. Harga yang murah
PHP adalah perangkat lunak gratis.

5. Mudah dalam pembelajaran dan penggunaan
Sintaks PHP berdasarkan bahasa pemrograman lainnya, terutama C dan Java.
6. *Portability*
PHP dapat digunakan di banyak sistem operasi yang berbeda.
7. Ketersediaan *Source Code*
Kode PHP dapat langsung diakses dan dimodifikasi secara bebas.

2.14 MySQL

My Structured Query Language (MySQL) adalah *Relational Database Management System* (RDBMS) yang sangat cepat dan kuat. Sebuah basis data memungkinkan *user* untuk melakukan penyimpanan yang efisien, pencarian dan pengurutan data. MySQL *server* memiliki kemampuan melakukan kontrol akses terhadap data untuk memastikan bahwa setiap *user* dapat bekerja dengan sesuai, menyediakan akses yang cepat, dan meyakinkan bahwa hanya *user* yang mempunyai hak akses yang dapat mengaksesnya. Oleh karena itu, MySQL merupakan *database server* yang yang dapat digunakan banyak *user* dan banyak pekerjaan (Welling dan Thomson, 2003).

MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam basis data sejak lama, yaitu SQL (*Structure Query Language*). SQL adalah sebuah konsep pengoperasian basis data terutama untuk proses seleksi, pemasukan, pengubahan, dan penghapusan data yang dimungkinkan dapat dikerjakan dengan mudah dan otomatis (Sutaji, 2012)

2.14.1 Keunggulan MySQL

Berikut keunggulan dari MySQL, di antaranya adalah (Sutaji, 2012):

1. *Portability*
Dapat berjalan stabil pada berbagai sistem operasi, di antaranya: Windows, Linux, FreeBSD, MacOS X Server, Solaris, Asigma.

2. *Open source*
Didistribusikan secara gratis di bawah lisensi dari *General Public License* (GPL), dimana setiap orang bebas untuk menggunakannya tetapi tidak boleh untuk dijadikan program induk turunan bersifat *close source* (komersial).
3. *Multi User*
Dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan.
4. *Performance tuning*
Memiliki kecepatan yang tinggi dalam menangani *query*.
5. *Column types*
Memiliki tipe data yang sangat kompleks, seperti: *signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year* dan *enum*.
6. *Command dan function*
Memiliki operator dan fungsi penuh yang mendukung *select* dan *where* dalam *query*.
7. *Security*
Memiliki beberapa lapisan sekuritas seperti tingkat *subnet mask, hostname, privilege user* dengan sistem perijinan yang mendetail serta *password* yang terenkripsi.
8. *Scalability dan limits*
Mampu menangani basis data dalam jumlah besar, dengan jumlah *field* lebih dari 50 juta, 60 ribu tabel, dan 5 miliar *record*. Batas indeks mencapai 32 buah per tabel.
9. *Localization*
Dapat mendeteksi pesan kesalahan (*error code*) pada klien dengan menggunakan lebih dari 20 bahasa.
10. *Connectivity*
Dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix Socket, Named Pipes*.

11. *Interface*

Memiliki antarmuka terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan API.

12. *Client dan tools*

Dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi basis data sekaligus dokumen petunjuk *online*.

13. Struktur tabel

Memiliki struktur tabel yang lebih fleksibel dalam menangani *alter* tabel dibandingkan dengan PostgreSQL dan Oracle.

2.14.2 Tipe Data MySQL

MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam merancang *table*. Beberapa jenis data yang tersedia pada MySQL dapat dilihat pada Tabel II.11 (Sutaji, 2012).

Tabel II.11 Jenis Data pada MySQL

Jenis Data	Keterangan
<i>CHAR</i>	Sebuah <i>string</i> dengan panjang tetap. Sisa Jumlah karakter yang belum terisi akan diisi dengan spasi, akan tetapi spasi ini dibuang jika data dipanggil. Jangkauan nilai M adalah 1-255 karakter
<i>VARCHAR</i>	<i>String</i> dengan panjang berupa variabel. M bisa mencapai 65535
<i>DATE</i>	Data berupa tanggal. Format tanggal dalam bentuk 'YYYY-MM-DD'
<i>TIME</i>	Data berupa waktu. Format waktu dalam bentuk 'HH:MM:SS'
<i>TINYINT</i>	Bilangan antara -128 sampai dengan +127
<i>SMALLINT</i>	Bilangan antara -32768 sampai dengan +32767
<i>INT</i>	Bilangan antara -2147683648 sampai dengan +2147683647

Tabel II.12 Jenis Data pada MySQL (lanjutan)

Jenis Data	Keterangan
<i>FLOAT</i>	Bilangan <i>floating point</i> yang kecil (presisi tunggal). Jangkauan nilai yang diizinkan adalah -3.402823466E+38 s/d -1.175494351e-38,0 dan 1.175494351E-38 s/d 3.402823466E+38
<i>DOUBLE</i>	Bilangan <i>floating point</i> dengan ukuran normal (presisi ganda). Jangkauan nilai yang diizinkan adalah -1.7976931348623157E+308 s/d -2.225073858507201E-308,0 dan 2.225073858507201E-308 s/d 1.7976931348623157E+308.
<i>ENUM</i>	Sebuah <i>enumeration</i> . Sebuah obyek string yang hanya boleh memiliki satu nilai, yang terambil dari 'value1','value2',...,NULL atau nilai spesial "" <i>error</i> . Sebuah <i>enum</i> dapat menampung 65535 pilihan nilai.
<i>TEXT, BLOB</i>	Sebuah <i>TEXT</i> atau <i>BLOB</i> dengan panjang karakter maksimum 65535 karakter.

Sumber: Sutaji (2012)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian atau *methodology of research* berasal dari kata “*metoda*” yang berarti cara atau teknik dan “*logos*” yang berarti ilmu. Menurut Soewardi (2012) metodologi penelitian adalah sebagai suatu usaha atau proses mencari jawaban atas suatu pertanyaan atau masalah dengan cara sabar, hati-hati, terencana, sistematis atas fakta atau prinsip-prinsip, mengembangkan dan menguji kebenaran ilmiah suatu pengetahuan

Metodologi penelitian yang digunakan dimulai dari mengidentifikasi masalah sampai dengan kesimpulan yang membentuk sebuah alur yang sistematis. Metodologi penelitian ini juga digunakan sebagai pedoman dalam pelaksanaan penelitian Tugas Akhir agar hasil yang dicapai tidak menyimpang dari tujuan yang telah ditentukan sebelumnya.

3.2 Jenis dan Sumber Data

Data yang diperlukan dalam penelitian ini ada dua jenis, yaitu data primer dan data sekunder.

1. Data Primer

Data yang diperoleh secara langsung dari perusahaan, dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data yang diperoleh pada penelitian ini berdasarkan pengamatan di *Warehouse Finished Goods Plant 3* terhadap sistem yang berjalan dan wawancara dengan karyawan. Dalam penelitian ini data primer hasil pengamatan tersebut berupa alur data proses penyimpanan barang jadi pada *Warehouse Finished Goods* dan dokumen yang terkait pada proses tersebut.

2. Data Sekunder

Data sekunder merupakan sumber data penelitian yang diperoleh secara tidak langsung melalui media perantara, kepustakaan, buku-buku, internet dan referensi. Dalam penelitian ini data sekunder tersebut berupa data umum perusahaan, profil perusahaan dan struktur organisasi perusahaan serta teori-teori mengenai *warehouse*, penyimpanan, pemrograman PHP, dan *database* MySQL.

3.3 Metode Pengumpulan Data

Menurut Sugiyono (2013) teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Penelitian ini menggunakan beberapa teknik pengumpulan data, yaitu:

1. Observasi

Observasi ini melakukan pengamatan langsung pada sistem yang berjalan di Bagian *Warehouse Finished Goods*. Pengamatan yang dilakukan mengenai alur proses penyimpanan barang jadi yang dilakukan mulai dari produksi hingga masuk gudang, dokumen yang terkait dengan sistem yang diteliti dan laporan yang digunakan *Warehouse Finished Goods*.

2. Wawancara

Wawancara yaitu metode pengumpulan data melalui tanya jawab secara langsung dengan pihak-pihak yang berkaitan dengan sistem penyimpanan barang jadi pada Divisi *Warehouse Finished Goods*. Pihak yang diwawancarai adalah karyawan pada Divisi *Warehouse Finished Goods Plant 3* di PT Rekadaya Multi Adiprima.

3. *Sampling*

Sampling yaitu bagian dari metodologi statistika yang berhubungan dengan pengambilan sebagian dari populasi. *Sampling* digunakan untuk pengambilan *sample* dokumen yang ada pada Divisi *Warehouse Finished Goods*.

3.4 Metode Pengembangan Sistem

Metode pengembangan sistem informasi penyimpanan barang jadi ini menggunakan model *Prototype* jenis *Evolutionary Prototype*. Model *Prototype* cocok digunakan untuk menjabarkan kebutuhan *user* secara lebih detail karena *user* sering kali kesulitan menyampaikan kebutuhannya secara *detail* tanpa melihat gambaran yang jelas. Tahapan-tahapan dalam *Evolutionary Prototype* adalah sebagai berikut (McLeod dan Schell, 2011):

1. Mengidentifikasi kebutuhan pengguna.
Pengembangan mewawancarai pengguna untuk mendapatkan ide mengenai apa yang diminta dari sistem.
2. Membuat satu *prototype*.
Pengembang merancang *prototype* yang sesuai dengan kebutuhan pengguna.
3. Menentukan apakah *prototype* dapat diterima
Menentukan apakah *prototype* kepada para pengguna untuk mengetahui apakah telah memberikan hasil yang memuaskan. Jika ya, langkah 4 akan diambil, jika tidak, *prototype* direvisi dengan mengulang kebutuhan pengguna.
4. Menggunakan *prototype*.
Prototype menjadi sistem yang siap diaplikasikan pada perusahaan.

3.5 Analisis dan Perancangan Sistem

Analisis dan Perancangan Sistem dapat didefinisikan sebagai proses organisasional kompleks dimana sistem informasi berbasis komputer diimplementasikan. Analisis dan Perancangan Sistem Informasi yang digunakan adalah analisis dan perancangan berbasis objek.

3.5.1 Analisis dan Perancangan Berorientasi Objek

Analisis dan Perancangan Berorientasi Objek atau *Object Oriented Analysis and Design* (OOAD) adalah metode analisis yang memeriksa *requirements* dari sudut pandang kelas-kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang

mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem atau sub-sistem. Ada beberapa notasi yang mewakili model ini, yaitu salah satunya *Unified Modelling Language* (UML).

3.6 Kerangka Penelitian

Dalam penelitian yang dibahas dalam tugas akhir ini, dilakukan langkah-langkah atau tahapan dalam penelitian yang ada pada Gambar III.1 sesuai dengan metodologi penelitian yang digunakan dalam penelitian tersebut. Penjelasan langkah-langkah atau tahapan dalam penelitian tersebut adalah sebagai berikut:

1. Melakukan studi pendahuluan dengan membaca penelitian yang telah dilakukan di *Warehouse Finished Goods* sebelumnya, melakukan pengamatan/observasi pada alur proses penyimpanan barang jadi serta membaca literatur-literatur yang berkaitan dengan aktivitas penyimpanan barang, terutama bahan barang jadi.
2. Mengidentifikasi dan merumuskan permasalahan yang ditemukan saat dilakukan penelitian pada sistem penyimpanan barang jadi yang sedang berjalan di *Warehouse Finished Goods Plant 3* pada PT Rekadaya Multi Adiprima
3. Melakukan studi literatur untuk mendapatkan landasan teori yang dapat mendukung pemecahan masalah yang sedang diteliti. Teori yang didapatkan merupakan langkah awal agar permasalahan lebih dipahami dengan benar sesuai dengan kerangka berpikir ilmiah.
4. Menentukan tujuan penelitian. Menentukan tujuan dari dilakukannya penelitian yaitu merancang bangun sistem informasi penyimpanan barang jadi yang dapat memudahkan dalam penyimpanan serta pencarian data stok barang jadi pada *Warehouse Finished Goods*.
5. Menentukan pendekatan pemrograman sistem, *tools* yang digunakan untuk pemodelan dan metodologi yang akan digunakan dalam pengembangan perangkat lunak. Pendekatan analisis dan perancangan pemrograman sistem yang dipilih untuk rancang bangun sistem informasi penyimpanan barang jadi pada *Warehouse Finished Goods* ini adalah pendekatan berorientasi objek

dengan *tools* untuk pemodelannya adalah UML (*Unified Modeling Language*) berupa *use case diagram*, *activity diagram*, *sequence diagram*. Sedangkan pemodelan data dengan menggunakan *class diagram* dan kamus data. Sementara untuk metodologi pengembangan perangkat lunak yang digunakan adalah *Evolutionary Prototype*.

6. Mengumpulkan data yang diperlukan dengan cara observasi langsung terhadap sistem yang berjalan, serta wawancara dengan pihak yang mempunyai hubungan dan pengetahuan terhadap sistem yang berjalan.
7. Untuk melakukan pengembangan sistem penyimpanan barang jadi pada *Warehouse Finished Goods* menggunakan *Evolutionary Prototype* dilakukan dengan tahapan-tahapan berikut:
 - a. Melakukan identifikasi kebutuhan pemakai

Dengan cara menganalisis terhadap data yang telah diperoleh sebelumnya untuk membuat aplikasi. Adapun identifikasi masalah sebagai berikut:

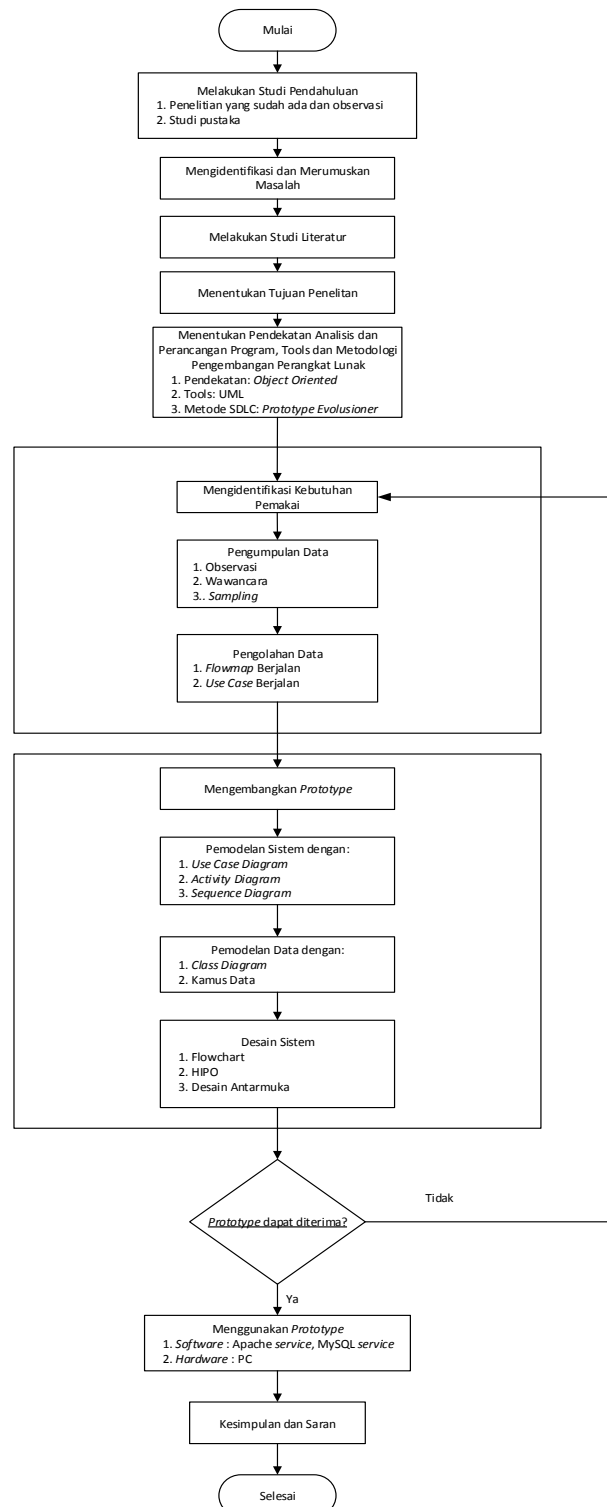
 - 1) Pengumpulan Data
 - a) Wawancara
 - b) Observasi
 - c) *Sampling*
 - 2) Pengolahan Data
 - a) *Flowmap* Berjalan
 - b) *Use Case* Berjalan
 - b. Mengembangkan *Prototype*
 - 1) Pemodelan Sistem menggunakan UML
 - 2) Pemodelan Data dengan *Class Diagram* dan Kamus Data
 - 3) Desain Sistem menggunakan *Flowchart*, HIPO, dan Desain Antarmuka.
 - c. Pada tahap berikutnya *prototype* yang diusulkan akan dilihat apakah sudah sesuai dengan keinginan *user* atau belum. Jika belum maka akan dilakukan identifikasi kebutuhan pemakai kembali, tetapi apabila sudah sesuai maka tahap selanjutnya adalah menggunakan *prototype* tersebut.

d. Menggunakan *Prototype*

Penggunaan dilakukan setelah sistem spesifikasi sudah terpenuhi yaitu *notebook* atau *Personal Computer (PC)* sebagai perangkat keras yang akan digunakan dan perangkat lunak yaitu *Apache Service* dan *MySQL Service*.

8. Setelah *prototype* sistem usulan diterima maka tahap selanjutnya adalah menarik kesimpulan dari penelitian yang telah dilakukan di PT Rekadaya Multi Adiprima dan memberikan saran yang membangun bagi perusahaan dan bagi peneliti selanjutnya dalam penggunaan dan pengembangan sistem informasi penyimpanan barang jadi pada *Warehouse Finished Goods Plant 3*.

Berikut adalah Gambar III.1 *Flowchart* kerangka penelitian untuk menyelesaikan masalah dalam Tugas Akhir ini:



Gambar III.1 Kerangka Penelitian
Sumber: Hasil Pengolahan Data (2017)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Perusahaan

PT Rekadaya Multi Adiprima merupakan salah satu perusahaan yang bergerak dalam industri manufaktur. PT Rekadaya Multi Adiprima didirikan sejak tahun 1994 telah memproduksi komponen otomotif baik untuk kendaraan roda empat maupun roda dua yang merupakan produk orisinil atau *Original Equipment Manufactured* (OEM). Kini PT Rekadaya Multi Adiprima telah memiliki lokasi usaha yang tersebar di Cikeas Nagrak-Kabupaten Bogor sebanyak 9 *plant*. 1 *plant* di Sumurkondang-Klari, Kabupaten Karawang, dan 1 *plant* di Cikarang-Kabupaten Bekasi.

Seiring dengan pertumbuhan bisnis otomotif di Indonesia juga terjadinya peningkatan permintaan di luar negeri (ekspor), berimbas kepada industri komponen otomotif tersebut, PT Rekadaya Multi Adiprima berfokus pada produksi *stamping parts*, *insulator dush panel*, dan *furniture parts* baik untuk kendaraan roda empat maupun roda dua.

Di Indonesia khususnya daerah Jawa Barat merupakan tempat dimana tersebar bagian dari Rekadaya Group. Adapun anggota Rekadaya Group adalah sebagai berikut:

1. PT Rekadaya Multi Adiprima (PT RMA)
2. PT Rekadaya Kreasi Indonesia (PT RKI)
3. PT Rekadaya Prima Indonesia (PT RPI)
4. PT Rekadaya Global Logistik (PT RGL)
5. PT Rekadaya Nonwoven Indonesia (PT RNI)

4.2 Profil Umum Perusahaan

Adapun profil umum PT Rekadaya Multi Adiprima sebagai berikut:

Nama perusahaan : PT Rekadaya Multi Adiprima

Alamat kantor pusat : Alt. Cibubur-Cileungsi, Ciangsana Raya, No.55

	Nagrak-Cikeas, Gunung Putri, Bogor-Jawa Barat
Telepon	: 021 – 8232888
Fax	: 021 – 8231774
Email	: rma@rekadaya.com
Website	: www.rekadaya.com
Produksi	: <i>Part Plastic Service Hole, Felt, Insulator Dush Panel</i>
Status	: Perseroan Terbatas
Luas Tanah	: <i>Plant 1 1000m²</i> <i>Plant 2 1200m²</i> <i>Plant 3 2000m²</i> <i>Plant 4 4000m²</i> <i>Plant 5 900m²</i> <i>Plant 6 4000m²</i> <i>Plant 7 2700m²</i> <i>Plant 8 1300 m²</i> <i>Plant 9 3000m²</i> Cikarang 30000m ² Karawang 15800m ²

4.3 Visi dan Misi Perusahaan

PT Rekadaya Multi Adiprima perusahaan yang bergerak di bidang industri manufaktur komponen otomotif seperti *Part Plastic Service Hole, Felt, Insulator Dush Panel* serta komponen otomotif lainnya mempunyai visi dan misi sebagai berikut:

Visi:

Menjadi perusahaan manufaktur komponen otomotif dan industri pendukung terbesar di Indonesia.

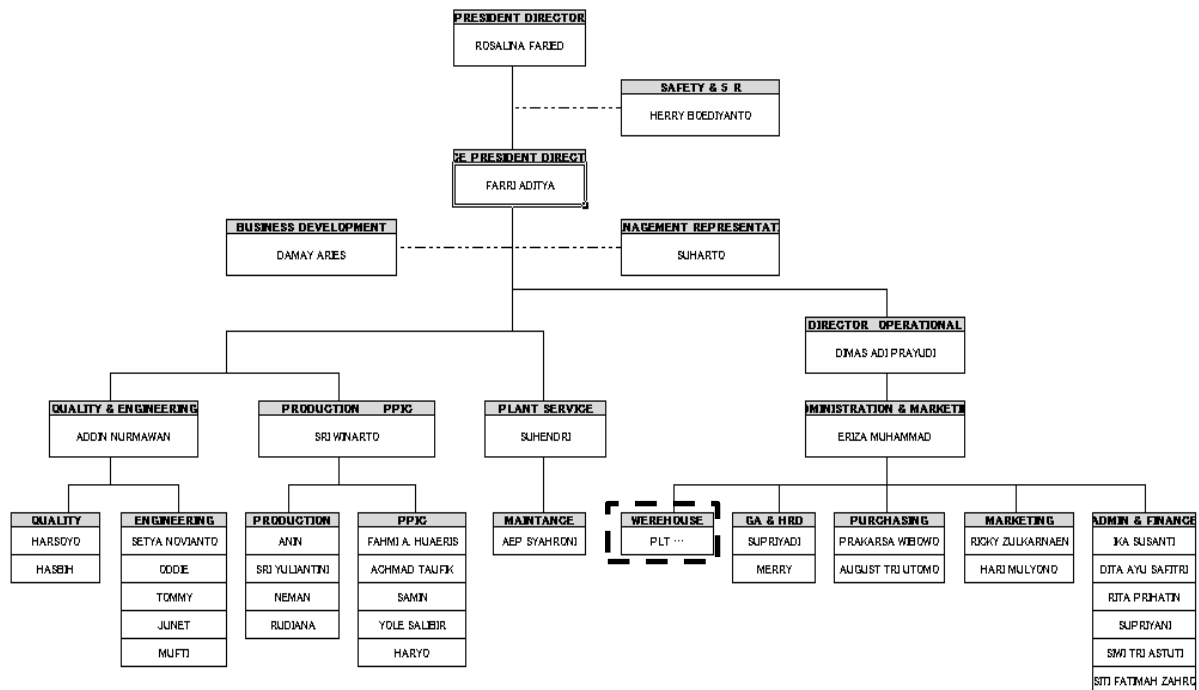
Misi:

1. Mewujudkan perusahaan yang maju berdasarkan Sistem Manajemen berstandar Internasional.


2. Mewujudkan produk perusahaan yang bermutu dan berwawasan lingkungan, keselamatan dan kesehatan kerja untuk memenuhi kebutuhan pelanggan dalam hal *Quality, Cost, Delivery, and Moral*.

4.4 Struktur Organisasi Perusahaan

Struktur organisasi yang tersusun dengan baik akan memudahkan koordinasi, integrasi, serta meningkatkan efektivitas dan efisiensi suatu perusahaan didalam mencapai tujuannya. Berikut merupakan struktur organisasi PT Rekadaya Multi Adiprima secara keseluruhan yang dapat dilihat pada gambar berikut.



Gambar IV.1 Struktur Organisasi PT Rekadaya Multi Adiprima
(Sumber: PT Rekadaya Multi Adiprima, 2017)

 = lokasi penelitian

4.4.1 Tugas dan Wewenang setiap Jabatan di PT Rekadaya Multi Adiprima

Dengan adanya suatu organisasi yang baik, maka memungkinkan terselenggaranya suatu kelancaran dalam proses usaha yang dilakukan oleh perusahaan dikarenakan dalam organisasi tersebut sudah ditetapkan dengan jelas

mengenai wewenang, tugas, dan tanggung jawab seseorang. Pada PT Rekadaya Multi Adiprima dikenal dengan *Job Description*. Berikut ini merupakan *Job Description* dan wewenang kegiatan dari masing-masing jabatan yang ada di PT Rekadaya Multi Adiprima adalah sebagai berikut:

1. Presiden Direktur

Presiden Direktur bertindak sebagai pimpinan Eksekutif perusahaan dan secara keseluruhan mempunyai tanggung jawab strategi dan manajemen sehari-hari terhadap aktivitas perseroan. Presiden direktur secara mendasar menetapkan arah, tujuan, dan strategi untuk dikerjakan oleh semua unit operasi dan pada akhirnya bertanggung jawab untuk kinerja keuangan perseroan.

2. Wakil Presiden Direktur

- a. Membuat dan menentukan garis kebijaksanaan perusahaan baik jangka pendek maupun jangka panjang
- b. Bertanggung jawab atas kegiatan operasi perusahaan dan kelancaran manajemen yang harus dilaporkan pada Presiden Direktur
- c. Melaksanakan pengawasan dan mengkoordinir tugas-tugas serta kegiatan-kegiatan dari perusahaan yang akan dilaksanakan oleh karyawan.

3. *Safety & 5 R*

- a. Melakukan pemeriksaan peralatan kerja, tenaga kerja, kesehatan kerja serta lingkungan kerja
- b. Meninjau dan mengarahkan karyawan bekerja sesuai kewajiban dan sesuai dengan sistem operasi perusahaan
- c. Mampu melakukan penanggulangan kecelakaan kerja dan melakukan penyelidikan penyebabnya

4. *Business Development*

- a. Mengobservasi kinerja *project* yang selama ini telah berjalan pada perusahaan
- b. Menganalisa potensi untuk melakukan efisiensi dari berbagai lini
- c. Melihat peluang pengembangan bisnis yang memungkinkan dengan mengacu pada *core business* yang sudah ada

5. *Management Representative*

- a. Mempromosikan kesadaran tentang persyaratan pelanggan

- b. Mempersiapkan tinjauan management jadwal pertemuan dan melakukan rapat *management review*
 - c. Membuat ISO atau kesadaran kualitas untuk rekan dengan pelatihan internal
6. *Director Operational*
- a. Mengelola dan meningkatkan efektivitas dan efisiensi operasi perusahaan
 - b. Mengawasi persediaan, distribusi barang dan tata letak fasilitas operasional
7. *Quality & Engineering*
- a. Membuat dan menetapkan prosedur yang berkaitan dengan *engineering*, seperti:
 - 1) Membuat *design tooling* proses dan design alat bantu.
 - 2) Membuat dan memonitor *schedule project-project* sesuai dengan waktu yang telah disubmit kepada *customer*.
 - 3) Mengontrol dan mendistribusikan dokumen-dokumen teknis seperti *Part Drawing, Technical Standart, Engineering Change Instruction, Estimasi Tooling, Mapping Process, Bill of Material*.
 - b. Mengontrol dan memonitor pelaksanaan *preventive maintenance* terhadap *dies*, jig-jig produksi.
 - c. Melakukan pengecekan kualitas hasil produksi sesuai kriteria kelayakan.
 - d. Mencatat hasil pengecekan kedalam *form check sheet temporary*.
8. PPIC (*Production Planning & Inventory Control*)
- a. Menjalankan kebijakan departemen yang tertuang pada sasaran mutu, seperti membuat *loading capacity*, membuat *budget material* dan *request material*, membuat *master plan schedule* dan *daily schedule* serta membuat *monthly schedule delivery*, dan membuat form permintaan produksi.
 - b. Melakukan kontrol terhadap *Inventory Raw Material*, produk *WIP*, dan *Finished Goods*.
 - c. Melakukan koordinasi dengan departemen terkait untuk lancarnya pergerakan produksi, seperti dengan departemen :
 - 1) Departemen *Purchasing* dalam pengadaan material produksi
 - 2) Departemen *Quality* dalam hal kualitas material dari *supplier*
 - 3) Departemen Produksi dalam hal pencapaian hasil produksi

4) Departemen *Marketing* dalam hal menerima Rekap PO dari *customer*.

9. *Production*

- a. Melaksanakan kegiatan produksi sesuai dengan data produksi yang diberikan oleh bagian PPIC.
- b. Melaksanakan kegiatan produksi dengan menggunakan bahan baku dan mesin secara efisien dan produktif.
- c. Meneliti, menganalisa dan membuat rekomendasi perbaikan, perubahan atau penambahan sarana yang diperlukan untuk menanggulangi hambatan produksi, dan meningkatkan kinerja secara berkesinambungan.
- d. Melaksanakan kegiatan produksi setiap produk yang dihasilkan dengan memanfaatkan tenaga kerja yang ada serta memberikan arahan dan pelatihan untuk mencapai produktifitas kerja yang tinggi dengan tingkat resiko kecelakaan kerja yang serendah-rendahnya.

10. *Plant Service*

- a. Menjamin bahwa break down mesin / mesin rusak pada saat produksi sesuai dengan target yang ditetapkan.
- b. Melakukan pengawasan terhadap pelaksanaan *preventive maintenance* terhadap mesin-mesin produksi secara berkala/reguler.
- c. Melakukan pengawasan terhadap perawatan infrastruktur termasuk *building* dan fasilitasnya.

11. Admin

- a. Membuat dan menetapkan prosedur yang berkaitan dengan Administrasi seperti surat antara *customer* dan *supplier*
- b. Mengatur administrasi umum dan membuat jurnal setiap kegiatan
- c. Membuat surat jalan yang berhubungan dengan pengiriman, menerima dan merekap surat jalan *delivery*, surat tugas keluar, dan pengadaan alat tulis kantor.

12. *Marketing*

- a. Menjalankan kebijakan terhadap pengembangan Bisnis Perusahaan yang tertuang pada sasaran mutu.

- b. Menetapkan strategi dalam pencapaian suatu *project / order* seperti meninjau dan menganalisa informasi *project / order* yang diterima, membuat *list control feasibility study* dan membuat *summary* hasil dari *progress marketing*
- c. Mengkoordinasi, menganalisa, melaksanakan dan memantau tindakan perbaikan atau penanganan terhadap keluhan pelanggan yang diterima.

13. *Warehouse*

Divisi *Warehouse* terdiri dari tiga bagian yaitu bagian *Inventory*, bagian Logistik dan bagian Gudang Teknik, tugas dan wewenang ketiga bagian pada Divisi *Warehouse* dijelaskan sebagai berikut:

- a. Bagian Inventori bahan baku berfungsi untuk mengatur dan melaksanakan penerimaan, penyimpanan, pengeluaran dan administrasi bahan baku.
- b. Bagian Inventori barang jadi berfungsi untuk mengatur dan melaksanakan penerimaan, penyimpanan, pengeluaran dan administrasi barang jadi.
- c. Menjaga kondisi bahan baku dan barang jadi terhadap kerusakan karena lingkungan penyimpanan dan kehilangan.

14. HRD (*Human Resources Division*)

Melaksanakan program Sumber Daya Manusia, termasuk pencarian tenaga kerja yang tepat, sistem upah dan tunjangan, pelatihan dan pengembangan karier, dan pengurusan masalah perizinan yang berkaitan dengan operasional usaha perusahaan.

15. *Purchasing*

- a. Membuat dan menetapkan prosedur yang berkaitan dengan *Purchasing* seperti:
 - 1) Melakukan evaluasi terhadap *budget* yang diajukan oleh PPIC
 - 2) Melaksanakan pengadaan atau pembelian material berdasarkan *budget* material untuk kebutuhan operasional dan permintaan dari departemen lain yang sudah disetujui atasan
- b. Mengkoordinir pelaksanaan audit *vendor* dan melakukan evaluasi dari hasil audit *vendor*.

16. *Finance*

- a. Membuat dan menetapkan prosedur yang berkaitan dengan *Finance*, seperti jurnal pembelian barang, mengontrol laporan posisi aktiva tetap, mengontrol laporan posisi biaya penyusutan aktiva tetap, baik laporan komersil atau laporan pajak.
- b. Mengendalikan transaksi pembelian, penjualan, dan *project* berdasarkan anggaran atau budget yang sudah disetujui/*approval*.

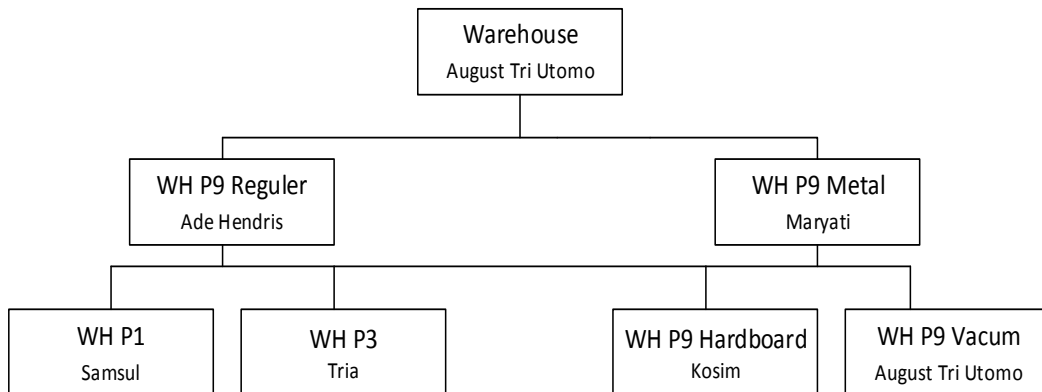
4.5 Struktur Organisasi pada Divisi *Warehouse*

Warehouse atau pergudangan berfungsi menyimpan barang untuk produksi atau hasil produksi dalam jumlah atau jangka waktu tertentu yang kemudian didistribusikan ke lokasi yang dituju.

Pada PT Rekadaya Multi Adiprima terdapat gudang di setiap *Plant*-nya yang mempunyai fungsi sama yaitu untuk menyimpan barang jadi atau bahan baku. Untuk gudang di *Plant 3* mempunyai fungsi menyimpan barang jadi hasil produksi yang berbahan baku *Needle*, *Dacron* dan *Felto* setiap harinya.

Pada gudang *Plant 3* ini terdapat pegawai gudang yang mempunyai tugas untuk mencatat setiap barang yang masuk dan barang yang keluar. Kemudian pegawai gudang juga harus memperhatikan kondisi gudang seperti merapikan posisi barang yang tidak teratur dan membantu mengeluarkan barang yang ingin didistribusikan ke *customer*. Selain itu pegawai *Plant 3* bertugas memperhatikan kondisi barang apabila terjadi kerusakan pada kemasan maka harus diberikan kepada Bagian *Packing* untuk dikemas ulang.

Berikut merupakan struktur organisasi pada Divisi *Warehouse* yang dapat dilihat pada Gambar IV.2



Gambar IV.2 Struktur Organisasi Divisi *Warehouse*
(Sumber: PT Rekadaya Multi Adiprima, 2016)

4.5.1 Tugas dan Wewenang pada Divisi *Warehouse* pada PT Rekadaya Multi Adiprima

Pada PT Rekadaya Multi Adiprima terdapat Divisi *Warehouse* yang memiliki bagian tersendiri. Dalam divisi *warehouse* itu juga memiliki berbagai peran yang berbeda pada setiap *Plant* nya. Berikut ini merupakan *Job Description* dari masing-masing gudang dari *Plant* yang ada:

1. *Warehouse*

Warehouse itu sendiri merupakan tingkat teratas yang terdapat pada struktur organisasi divisi *warehouse* atau biasa disebut kepala gudang. Kepala gudang disini memiliki peran sebagai pengawas alur keluar dan masuk barang pada setiap gudang. *Warehouse* juga memiliki tanggung jawab atas *problem* yang terjadi di gudang sewaktu-waktu.

2. *WH Plant 9 Reguler*

Gudang pada *plant* ini merupakan gudang yang digunakan untuk menyimpan bahan baku penunjang produksi. Adapun tugas dari pegawai gudang pada *Plant* ini:

- a. Mengatur masuknya barang dari *supplier*
- b. Mengatur keluar barang dan mencatat Form Bukti Order Barang (BOB)

- c. Meng-*input* stok barang ke *database Ms. Excel*
3. WH *Plant 9 Metal*

Gudang pada *plant* ini merupakan gudang yang digunakan untuk menyimpan bahan baku jenis metal. Adapun tugas dari pegawai gudang pada *Plant* ini:

 - a. Mengatur masuknya barang dari *supplier*
 - b. Mengatur keluar barang dan mencatat Form Bukti Order Barang (BOB)
 - c. Meng-*input* stok barang ke *database Ms.Excel*
4. WH *Plant 1*

Gudang pada *plant* ini merupakan gudang yang digunakan untuk menyimpan barang jadi jenis metal. Adapun tugas dari pegawai gudang pada *Plant* ini:

 - a. Mengatur masuk dan keluar barang hasil produksi
 - b. Menjaga kondisi barang ketika berada dalam gudang
5. WH *Plant 3*

Gudang pada *Plant* ini merupakan gudang yang digunakan untuk menyimpan barang jadi berbahan baku *Dacron, Needle* dan *Felto*. Adapun tugas dari pegawai gudang pada *Plant* ini:

 - a. Mengatur masuk dan keluar barang hasil produksi
 - b. Mencatat kartu stok
 - c. Mengatur tata letak barang
6. WH *Plant 9 Hardboard*

Gudang pada *plant* ini merupakan gudang yang digunakan untuk menyimpan barang jadi berbahan baku *hardboard*. Adapun tugas dari pegawai gudang pada *Plant* ini:

 - a. Mengatur masuk dan keluar barang hasil produksi
 - b. Mencatat kartu stok
 - c. Mengatur tata letak barang

7. *WH Plant 9 Vacuum*

Gudang pada *plant* ini merupakan gudang yang digunakan untuk menyimpan barang jadi dengan proses *vacum*. Adapun tugas dari pegawai gudang pada *Plant* ini:

- a. Mengatur masuk dan keluar barang hasil produksi
- b. Mengatur tata letak barang

4.6 Produk pada PT Rekadaya Multi Adiprima

PT Rekadaya Multi Adiprima menghasilkan berbagai macam jenis komponen otomotif. *Plant 3* merupakan tempat diproduksinya *interior* kendaraan roda empat dengan bahan baku *Needle, Dacron* dan *Felto*. Berikut ini merupakan sebagian produk yang dihasilkan di *Plant 3*:

1. *Silincer Front Floor No 1*

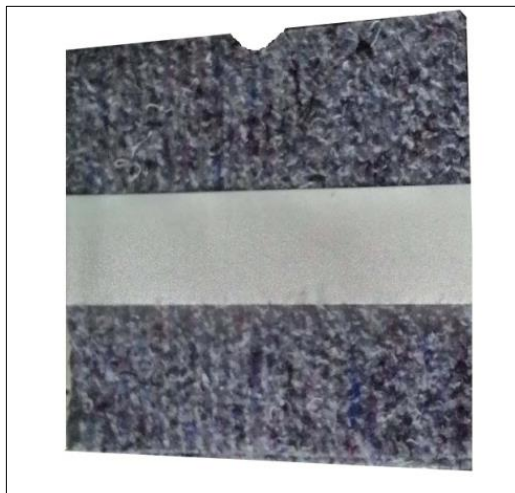
Untuk pembuatan *part silincer* menggunakan bahan baku *felto* dengan tahapan dimulai dari meng-*cutting felto* sesuai ukuran bentuk yang diminta, kemudian di-*assy double tip*, masuk ke proses *punching* atau *dipress* kemudian disusun. *Silincer Front Floor No 1* ini digunakan untuk komponen bawah pada bagasi kendaraan roda empat.



Gambar IV.3 *Silincer Front Floor No 1*
(Sumber: PT Rekadaya Multi Adiprima, 2016)

2. *Pad Roof Silincer Rear*

Untuk pembuatan *part silincer* menggunakan bahan baku *felto* dengan tahapan dimulai dari meng-*cutting felto* sesuai ukuran bentuk yang diminta, kemudian di-*assy double tip*, masuk ke proses *punching* atau *dipress* kemudian disusun. *Silincer Front Floor No 1* ini digunakan untuk komponen bawah pada bagasi kendaraan roda empat.



Gambar IV.4 *Pad Roof Silincer Rear*
(Sumber: PT Rekadaya Multi Adiprima, 2016)

3. *Pad RR Wheel Opening Ext No 1*

Untuk pembuatan *part silincer* menggunakan bahan baku *Needle* dengan tahapan dimulai dari meng-*cutting Needle* yang telah dijahit sesuai ukuran bentuk yang diminta, kemudian di-*assy double tip*, masuk ke proses *punching* atau *dipress* kemudian disusun. *Pad RR Wheel Opening Ext No 1* ini digunakan untuk komponen bawah pada bagasi kendaraan roda empat.



Gambar IV.5 *Pad RR Wheel Opening Ext No 1*
(Sumber: PT Rekadaya Multi Adiprima, 2016)

4. *Pad RR Wheel Opening Ext No 2*

Untuk pembuatan *part silincer* menggunakan bahan baku *Needle* dengan tahapan dimulai dari meng-*cutting Needle* yang telah dijahit sesuai ukuran bentuk yang diminta, kemudian di-*assy double tip*, masuk ke proses *punching* atau *dipress* kemudian disusun. Perbedaan antara *Pad RR Wheel Opening Ext No 2* dengan *Pad RR Wheel Opening Ext No 1* berada pada tingkat ketebalannya.



Gambar IV.6 *Pad RR Wheel Opening Ext No 2*
(Sumber: PT Rekadaya Multi Adiprima, 2016)

5. *Silincer Front Floor No 3*

Untuk pembuatan *part silencer* ini sama seperti *Silincer Front Floor No 1*. Perbedaan dengan *Silincer Front Floor No 1* terletak pada bentuknya dan ketebalannya.



Gambar IV.7 *Silincer Front Floor No 3*
(Sumber: PT Rekadaya Multi Adiprima, 2016)

4.7 Penyimpanan

Gudang berpengaruh besar terhadap perusahaan, tanpa adanya gudang maka barang jadi hasil produksi akan sulit diatur keberadaannya. Pengendalian dan pengaturan barang ini yang menjadi sebab mengapa manajemen pergudangan harus dilakukan dengan baik. Penyimpanan barang dalam perusahaan amatlah penting jika perusahaan berskala besar maupun menengah. Kegiatan penyimpanan adalah salah satu tugas Divisi *Warehouse* pada PT Rekadaya Multi Adiprima yang paling rutin dilakukan.

Proses penyimpanan dilakukan untuk menjaga barang jadi hasil produksi sebelum dikirimkan kepada pemesan. Selain itu penyimpanan yang baik dengan beberapa teknik penyimpanan salah satunya *First In First Out* (FIFO) diharapkan dapat mengurangi kerusakan barang.

4.7.1 Siklus Penyimpanan

Setiap perusahaan memiliki gudang dan ada satu Divisi yang bertugas untuk mengatur atau mengelola gudang tersebut. Namun Divisi ini tetap berhubungan dengan Divisi lain khususnya Divisi Produksi.

Proses penyimpanan harus dilakukan agar barang yang telah diproduksi tidak mengalami kerusakan dan dapat dikeluarkan sewaktu-waktu. Adapun prosedur penyimpanan hasil produksi secara umum sebagai berikut:

1. Adanya penerimaan hasil produksi di dalam gudang
2. Adanya pengurusan dan penyimpanan hasil produksi di dalam gudang
3. Adanya pengaturan keluar masuknya hasil produksi di dalam gudang
4. Adanya pengurusan administrasi penyimpanan hasil produksi di dalam gudang

PT Rekadaya Multi Adiprima memiliki suatu tim yang khusus mengelola pergudangan di perusahaan, yaitu Divisi *Warehouse*. Divisi *Warehouse* pada PT Rekadaya Multi Adiprima terdiri dari dua bagian yaitu bagian Gudang Bahan Baku dan Gudang Barang Jadi. Adapun tugas dan wewenang Divisi *Warehouse* pada PT Rekadaya Multi Adiprima, yaitu:

1. Bagian *Inventory* bahan baku berfungsi untuk mengatur dan melaksanakan penerimaan, penyimpanan, pengeluaran dan administrasi bahan baku.
2. Bagian *Inventory* barang jadi berfungsi untuk mengatur dan melaksanakan penerimaan, penyimpanan, pengeluaran dan administrasi barang jadi.

4.8 Sistem Penyimpanan Hasil Produksi yang Berjalan


Penyimpanan merupakan hal penting bagi setiap perusahaan baik perusahaan kecil atau besar. Hal itu juga berlaku pada PT Rekadaya Multi Adiprima. Pada perusahaan tersebut memiliki perbedaan pada tata letak gudang setiap *Plant* nya. Ada yang terpisah dan ada yang tergabung pada ruang produksi itu sendiri. Sebagai contoh seperti yang ada pada *Plant 3*, *Plant* ini merupakan *Plant* yang memproduksi material berbahan *Needle*, *Dacron* dan *Felto*.

Pada PT Rekadaya Multi Adiprima khususnya pada *Plant 3* dalam hal pencatatan hasil produksi barang jadi atau *finished goods* menggunakan kartu stok yang diletakkan pada dinding gudang. Pencatatan stok *finished goods* di gudang pada kartu stok dilakukan pada pagi hari dan sore hari oleh pegawai gudang serta proses penempatan barang jadi dikelompokkan berdasarkan *customer*.

4.9 Dokumen pada Sistem Penyimpanan Barang Jadi yang Berjalan

Sistem informasi penyimpanan barang jadi atau *Finished Goods Inventory* yang sedang berjalan pada PT Rekadaya Multi Adiprima masih menggunakan *Form-form* secara manual. Dokumen yang terlibat dalam sistem informasi penyimpanan barang jadi adalah sebagai berikut:

1. Rekap PO (*Purchase Order*)
Rekap PO merupakan rekap PO dari *Customer*. Rekap PO ini direkap oleh bagian Marketing yang nanti diserahkan pada Admin PPIC untuk dibuat Form Permintaan Produksi.

 PT. REKADAYA MULTIADIPRIMA Automotive Part & Manufactures				
PERIODE				
JULI 2016				
DIVISION				
INTERIOR DIVISION				
CUSTOMER ORDER INFORMATION				
No	CUSTOMER	PART NAME	TANGGAL KIRIM	QTY ORDER
1	PT IRC INOAC INDONESIA	CARPET ASSY FLOOR FR // (CARPET D40)	1/7/2016	1,000 pcs
		COVER COMP LUGGAGE FLOOR	1/7/2016	552 pcs
		CARPET COMP, LUGGAGE FLOOR	3/7/2016	574 pcs
		CARPET COMP FR RISER R Y9	3/7/2016	360 pcs
		CARPET COMP FR RISER L Y9	3/7/2016	352 pcs
		SILNCER FR FLOOR NO.1	1/7/2016	210 pcs
		SILNCER FR FLOOR NO.2	1/7/2016	210 pcs
		PAD ROOF SILNCER FR	1/7/2016	966 pcs
		PAD ROOF SILNCER RR	1/7/2016	728 pcs
		2	PT TOYOTA BOSHOKU INDONESIA	COVER ROOM PARTITION
PP CORR BOARD+LAMP CARPET	7/7/2016			330 pcs
SILNCER NO. 4 (0Q7)	29/7/2016			210 pcs
SILNCER RR CARPET (1Q6)	29/7/2016			510 pcs
PAD, P/Tray TRIM SILNCER	29/7/2016			900
SILNCER DOOR TRIM (4Q5)	29/7/2016			330
PAD RR DOOR TRIM SILNCER (8Q5)	29/7/2016			1,680
PAD ROOF SILNCER (SUBCONT)	1/7/2016			816
PAD ROOF SILNCER NO.2	1/7/2016			2,225
SILNCER ROOF	1/7/2016			992
3	PT ASTRA DAIHATSU MOTOR	PAD ROOF SILNCER NO.2 (D30D)	1/7/2016	540
		PAD ROOF SILNCER NO.6 (D30D)	1/7/2016	480
		PAD FR DOOR SILNCER (CB2)	3/7/2016	3,600
		PAD FR DOOR SILNCER (CA9)	3/7/2016	8,420
		PAD FR DOOR SILNCER (CB0)	3/7/2016	11,880
		PAD FR DOOR SILNCER (EQ7)	3/7/2016	12,020
		PAD FR DOOR SILNCER (EQ9)	3/7/2016	8,920
		PAD FR FLOOR SILNCER CTR (GA4)	3/7/2016	4,180
		PAD FR FLOOR SILNCER CTR (F22)	3/7/2016	4,780
		PAD FR FLOOR SILNCER CTR (F23)	3/7/2016	4,500
4	PT BONECOM TRICOM	FELT, REAR QUARTER TRIM	1/7/2016	1,560
		SILNCER, ROOF FR	1/7/2016	960
		SILNCER, ROOF CTR	1/7/2016	360
		CARPET COMP, FRONT FLOOR	20/7/2016	480
		CARPET COMP, REAR FLOOR REAR	20/7/2016	720
		FELT, REAR QUARTER TRIM	20/7/2016	360
		PAD ROOF SILNCER 1	2/7/2016	2,460
		PAD ROOF SILNCER 3	2/7/2016	2,460
		PAD COWL ASSY RH	7/7/2016	2,244
		PAD COWL ASSY LH	7/7/2016	2,100
7	PT USRA TAMPI	Felto t=10mm 780 X 390	1/7/2016	26,600
8	PT SARI TAKAGI ELOK PRODUK	FELT SEAT CUSHION X7A02	29/7/2016	4,350
		FELT SEAT CUSHION 79977	29/7/2016	2,340
9	PT SUGITY CREATIVES	PAD, QUARTER TRIM (500*160*T10)	1/7/2016	12,040

Gambar IV.8 Rekap PO
(Sumber: PT Rekadaya Multi Adiprima, 2017)

2. Form Permintaan Produksi

Form ini dibuat atau dikeluarkan oleh Admin PPIC berdasarkan acuan Rekap PO untuk diserahkan ke Bagian Produksi. Form Permintaan Produksi dapat dilihat pada Gambar IV.9

PT. REKADAYA MULTI ADIPRIMA
PRODUCTION DEPARTMENT

FORM PERMINTAAN PRODUKSI

WAREHOUSE AREA

NO.	PART NO.	PART NAME	TARGET	ACTUAL	BAL (+/-)	KETERANGAN
	5863-XV13	F22	400			Slap paking
	5863-XV14	F23	400	220	-180	belum ter packing
		E67	800	400		
		E68	800	400	-400	HBS.
		CB2	800	1000		belum ter packing
		F21	400			loading hbs
		F24	400			loading HBS
		F25	400	220	-180	mat HBS
		CB0	300	380	+80	

Gambar IV.9 Form Permintaan Produksi
Sumber: PT. Rekadaya Multi Adiprima (2016)

3. *Check Sheet Temporary*

Check sheet temporary digunakan oleh operator *Quality Control* untuk diisikan jumlah barang yang telah di cek kualitasnya oleh operator *Quality Control* itu sendiri.

CHECK SHEET TEMPORARY

PART NAME / NO.	TOTAL CHECK	OK	NG	REVISI	REVISI	REVISI	REVISI	REVISI	REVISI
F22	153	100	3	11					
F25	145	140	5						

Gambar IV.10 *Check Sheet Temporary*
(Sumber: PT Rekadaya Multi Adiprima, 2017)

4. *Form Kartu Stok*

Form kartu stok ini untuk mencatat berapa jumlah stok awal hasil produksi kemudian berapa jumlah barang yang dikeluarkan pada *Shift* 1 dan *Shift* 2 serta

berapa jumlah stok akhir barang jadi yang ada pada gudang. Form Kartu Stok dapat dilihat pada Gambar IV.11

PT REKADAYA MULTI ADIPRIMA
Automotive part and Manufacture

KARTU STOCK


PART NAME : T film comp TR 800E LH
 PART NUMBER : 21 - KTBK JM - P03 - 10
 CUSTOMER : PT AAM
 STANDAR STOCK :
 MIN : 100 MAX : 300
 BULAN : Agustus

TANGGAL	STOCK AWAL	SHIFT 1		SHIFT 2		STOCK AKHIR	KETERANGAN
		IN	OUT	IN	OUT		
1	80	60	20			80	
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							
25							
26							
27							
28							
29							
30							
31							

Gambar IV.11 Form Kartu Stok
(Sumber: PT Rekadaya Multi Adiprima, 2017)

5. Data *Inventory* Barang Jadi

Merupakan rekapitulasi data keseluruhan persediaan barang jadi yang didasarkan kepada data yang terdapat *form* kartu stok barang jadi. Data ini digunakan untuk mengetahui sisa stok barang jadi yang tersedia di gudang *finished goods*. Data *Inventory* Barang Jadi dapat dilihat pada Gambar IV.12

 PT. REKADAYA MULTI ADIPRIMA Automotive Parts and Manufacturers						
PERIODE : Jul-16						
ALL PLANT						
CUST	PLANT	No	NAMA PARTS	PART NUMBER	STOCK	
					AWAL (PCS)	AKHIR (PCS)
AAA	9	154	HRBD CJM-PU FDT R+MEDIUM	LH-MBCJM-HRBD-010	100	200
		155	HRBD CJM - PU FDT L+MEDIUM	LH-MBCJM-HRBD-020	100	200
		156	HRBD CJM-PU FDT R+MEDIUM	LH-MBCJM-HRBD-010	1428	1656
		157	HRBD CJM - PU FDT L+MEDIUM	LH-MBCJM-HRBD-020	1404	1560
		158	HRBD CJM-PU FDT R+MEDIUM	LH-MBCJM-HRBD-030	200	180
		159	HRBD CJM - PU FDT L+MEDIUM	LH-MBCJM-HRBD-040	250	280
		160	HRBD Y9B FDT R+MEDIUM	LH-MBY9B-HRBD-050	1340	1570
		161	HRBD Y9B FDT L+MEDIUM	LH-MBY9B-HRBD-060	1480	1810
		162	HRBD Y9B FDT R+MEDIUM	LH-MBY9B-HRBD-050	1346	1766
		163	HRBD Y9B FDT L+MEDIUM	LH-MBY9B-HRBD-060	1500	1994
		164	HRBD H/L FUSO + LATEX	LH-MBFMN-HRBD-020	0	0
		165	HRBD H/L FUSO + LATEX	LH-MBFMN-HRBD-020	0	0
		166	RIP WAFER H/L FUSO	LH-MBFMN-HRBD-030	3040	3560
		167	HRBD H/L 100P R/L	LH-MB100-HRBD-010	0	0
		168	HRBD H/L 100P R/L	LH-MB100-HRBD-010	0	0
		169	HRBD ST-100 FDT R	LH-MB100-HRBD-020/100	0	0
		170	HRBD ST-100 FDT L	LH-MB100-HRBD-030/110	0	0
		171	HRBD FE-200 FDT R	LH-MB200-HRBD-010	330	380
		172	HRBD FE-200 FDT L	LH-MB200-HRBD-020	360	410
		173	HRBD F61 FDT R	LH-MBF61-HRBD-010	0	0
		174	HRBD F61 FDT L	LH-MBF61-HRBD-020	0	0
		175	HRBD F61 QDT R	LH-MBF61-HRBD-030	0	0
		176	HRBD F61 QDT L	LH-MBF61-HRBD-040	0	0
		177	HRBD F69 BDT	LH-MBF61-HRBD-050	200	250
		178	HRBD FDT RH	LH-MBS91-HRBD-010	100	100
		179	HRBD FDT LH	LH-MBS91-HRBD-020	100	100
		180	HRBD D16 QTR R STD	LH-MBD16-HRBD-010	150	150
		181	HRBD D16 QTR L STD	LH-MBD16-HRBD-030	150	150
		182	HRBD D16 QTR R LUBANG	LH-MBD16-HRBD-030	0	0
		183	HRBD D16 QTR L LUBANG	LH-MBD16-HRBD-031	0	0
		184	HRBD QTR LBNG 2 RH	LH-MBD16-HRBD-050	0	0
		185	HRBD QTR LBNG 2 LH	LH-MBD16-HRBD-060	0	0
		186	HRBD SIDE BODY JIMNY RH	LH-MBSJ4-HRBD-010	0	0
		187	HRBD SIDE BODY JIMNY LH	LH-MBSJ4-HRBD-020	0	0
		188	HRBD FDT RH	LH-MBSJ4-HRBD-030	0	0
		189	HRBD FDT LH	LH-MBSJ4-HRBD-040	0	0
		190	HRBD SL415 RDT LH	LH-MBSL4-HRBD-010	0	0
		191	HRBD SL415 QTR RH	LH-MBSL4-HRBD-020	0	0
		192	HRBD SL415 QTR LH	LH-MBSL4-HRBD-030	0	0
		193	HRBD SL415 BDT	LH-MBSL4-HRBD-040	0	0
		194	HRBD SL415 QTR LH	LH-MBSL4-HRBD-050	0	0
		195	HRBD SL415 QTR RH (GRV)	LH-MBSL4-HRBD-060	0	0
		196	HRBD 100P TR PAD SPKR RH+MEDIUM	LH-MB100-HRBD-05A	0	0
		197	HRBD 100P TR PAD SPKR LH+MEDIUM	LH-MB100-HRBD-06A	0	0
		198	DOOR TRIM L300	MDF L300 POLOS	0	0
		199	DOOR TRIM L300	MDF L300 POLOS	0	0
		200	DOOR TRIM L300	L300. RH	2490	3350
201	DOOR TRIM L300	L300. LH	2490	3310		
202	DOOR TRIM L300	L300. RH	60	60		
203	DOOR TRIM L300	L300. LH	50	30		
204	PANEL A	PANEL A	1550	1580		
205	PANEL B	PANEL B	1920	2220		
206	DECKBOARD BZ 140 (A)		1100	1160		
207	DECKBOARD BZ 150 (B)		5460	6240		
TOTAL PT. AAA					28,698	34,266

Gambar IV.12 Data *Inventory* Barang Jadi
(Sumber: PT Rekadaya Multi Adiprima, 2017)

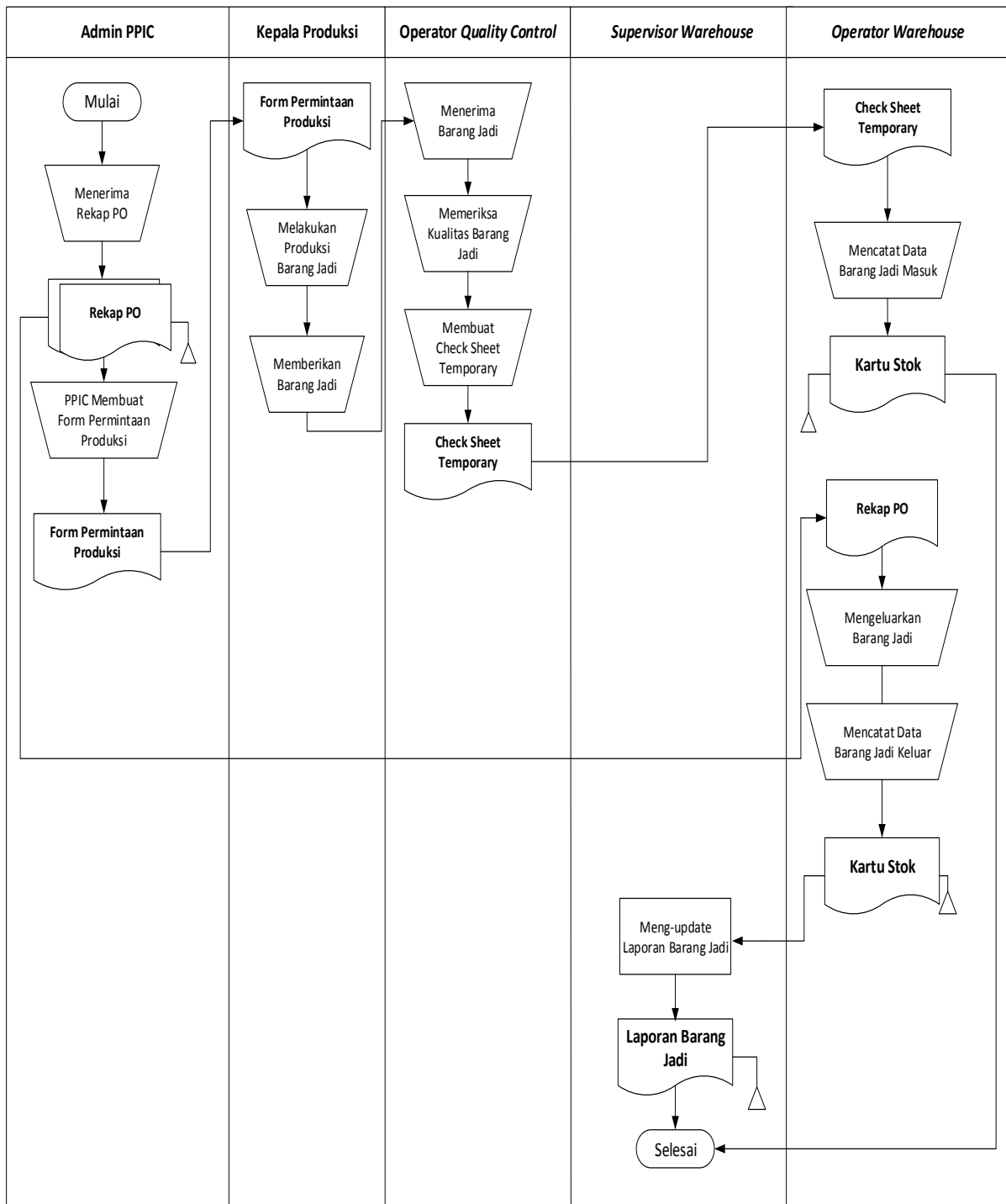
4.10 Prosedur Penyimpanan Barang Jadi

Adapun prosedur sistem informasi dalam penyimpanan barang jadi pada *Warehouse Finish Goods*:

1. Admin PPIC menerima Rekap PO dari Marketing dan membuat Form Permintaan Produksi.

2. Form Permintaan Produksi akan diserahkan ke Kepala Produksi.
3. Kepala produksi memproduksi barang jadi.
4. Setelah melakukan produksi maka kepala produksi akan menyerahkan barang jadi ke *Operator Quality Control*.
5. *Operator QC* menerima barang barang jadi.
6. *Operator QC* akan memeriksa kualitas dari barang jadi untuk diberikan kepada *Warehouse Finished Goods*.
7. *Operator QC* mencatat hasil pengecekan kualitas pada Form *Check Sheet Temporary*
8. *Operator Warehouse* menerima Form *Check Sheet Temporary* dari *Operator Quality Control* dan mencatat data barang masuk ke dalam Kartu Stok berdasarkan nama *part*, jumlah yang ada, dan tanggal barang itu masuk.
9. *Operator Warehouse* melihat Rekap PO untuk melihat tanggal keluar dan jumlah barang yang ingin dikeluarkan
10. *Operator Warehouse* akan menyiapkan barang jadi yang akan dikeluarkan dari gudang.
11. Selanjutnya pihak *operator warehouse* memasukan data barang jadi keluar pada Kartu Stok
12. Selanjutnya *Supervisor Warehouse* melakukan *update stock* dengan mengisi berapa jumlah barang jadi yang masuk dan yang keluar.

Berikut merupakan *flowmap* penyimpanan barang jadi yang berjalan pada *Warehouse Finished Goods*:



Gambar IV.13 *Flowmap* Sistem Informasi Penyimpanan Barang Jadi yang Berjalan
(Sumber: Hasil Analisis, 2017)

4.11 Penggambaran Sistem Penyimpanan Barang Jadi dengan UML (*Unified Modeling Language*)

Use case diagram sistem informasi penyimpanan barang jadi yang sedang berjalan pada Divisi *Warehouse Finished Goods* di PT Rekadaya Multi Adiprima adalah sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* penyimpanan barang jadi yang sedang berjalan pada divisi *Warehouse* PT Rekadaya Multi Adiprima dapat dilihat pada tabel IV.1 berikut:

Tabel IV.1. Definisi Aktor Penyimpanan Barang Jadi

No	Aktor	Definisi
1	Admin PPIC	PPIC bagian <i>Warehouse</i> menerima rekap PO dan membuat form permintaan produksi
2	Kepala Produksi	Kepala produksi menerima form permintaan produksi dan melakukan produksi serta menyerahkan barang jadi hasil produksi ke <i>operator Quality Control</i> .
3	<i>Operator Quality Control</i>	Menerima barang jadi dan memeriksa kualitas barang jadi untuk masuk ke gudang serta mencatat ke hasil pemeriksaan ke form <i>check sheet temporary</i>
4	<i>Supervisor Warehouse</i>	Meng- <i>update</i> data laporan barang jadi setelah ada transaksi barang masuk dan barang keluar
5	<i>Operator Warehouse</i>	Menerima barang dan menyiapkan pengeluaran barang jadi serta mencatat barang jadi yang masuk dan yang keluar kedalam kartu stok.

Sumber: Pengolahan Data (2017)

2. Definisi *Use case*

Pendefinisian *use case* pada *use case diagram* sistem informasi penyimpanan barang jadi pada *Warehouse Finished Goods* di PT Rekadaya Multi Adiprima dapat dilihat pada tabel IV.2 berikut:

Tabel IV.2 Definisi *Use Case* Sistem Informasi Penyimpanan Barang Jadi

No	<i>Use case</i>	Deskripsi
1	Menerima Rekap PO	Kegiatan ini dilakukan oleh Admin PPIC dalam menerima Rekap PO dari Marketing.
2	Membuat Form Permintaan Produksi	Kegiatan ini dilakukan oleh Admin PPIC untuk memberikan kepada Kepala Produksi berdasarkan Rekap PO Terdistribusi
3	Melakukan Produksi Barang Jadi	Kegiatan produksi yang dilakukan oleh bagian produksi
4	Memberikan Barang Jadi	Kegiatan ini dilakukan oleh bagian produksi untuk menyerahkan hasil produksi ke Bagian <i>Quality Control</i>
5	Menerima Barang Jadi	Kegiatan ini dilakukan oleh Bagian <i>Quality Control</i> dalam menerima barang jadi hasil produksi dari bagian Produksi
6	Memeriksa Kualitas Barang Jadi	Kegiatan ini dilakukan oleh bagian <i>Quality Control</i> untuk memeriksa kualitas barang jadi
7	Membuat Check Sheet Temporary	Kegiatan ini dilakukan oleh Bagian <i>Quality Control</i> dalam mengisi hasil pengecekan barang jadi.

Tabel IV.2 Definisi *Use Case* Sistem Informasi Penyimpanan Barang Jadi (lanjutan)

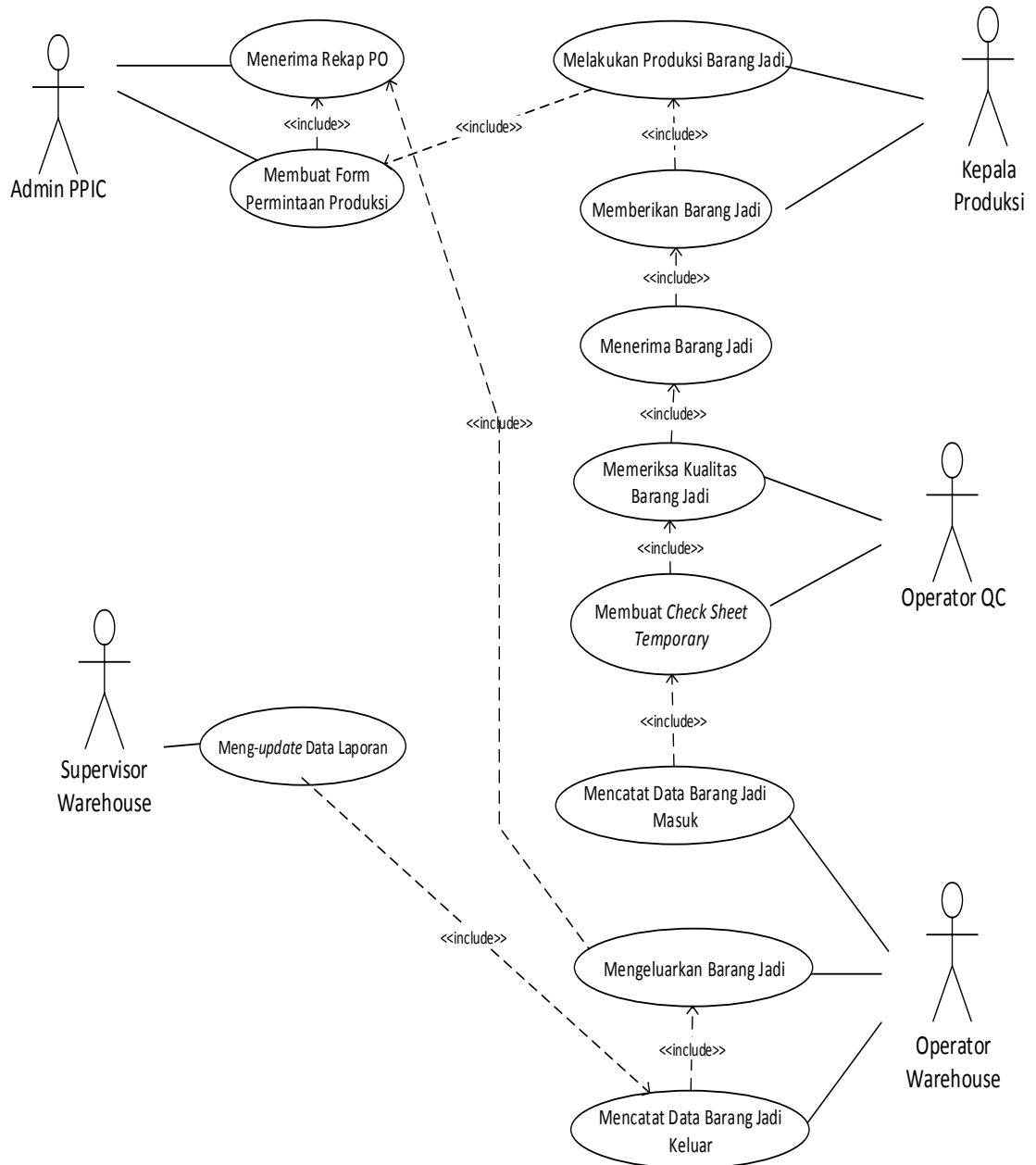
No	<i>Use Case</i>	Deskripsi
8	Mencatat Data Barang Jadi Masuk	Kegiatan ini dilakukan oleh <i>Operator Warehouse</i> untuk mencatat barang jadi yang masuk
9	Mengeluarkan Barang Jadi	Kegiatan ini dilakukan oleh <i>Operator Warehouse</i> dalam mengeluarkan barang jadi.
10	Mencatat Data Barang Jadi Keluar	Kegiatan ini dilakukan oleh <i>Operator Warehouse</i> untuk mencatat barang jadi yang keluar.
11	Meng-update Data Laporan	Kegiatan ini dilakukan oleh <i>Supervisor Warehouse</i> dalam meng-update data barang jadi di gudang.

Sumber: Pengolahan Data (2017)

3. *Use case diagram*

Use case diagram sistem informasi penyimpanan barang jadi yang sedang berjalan pada Divisi *Warehouse Finished Goods* terdiri dari 5 aktor sebagaimana digambarkan pada Gambar IV.17 di bawah ini:

Sistem Penyimpanan Barang Jadi pada *Warehouse Finished Good*



Gambar IV.14 Use Case Diagram Sistem Informasi Penyimpanan Barang Jadi yang Berjalan

(Sumber: Pengolahan Data, 2017)

4.12 Analisis Sistem Informasi Penyimpanan Barang Jadi yang Berjalan

Setelah melakukan analisis pada prosedur serta dokumen-dokumen yang terlibat pada sistem informasi penyimpanan barang jadi yang berjalan pada PT Rekadaya Multi Adiprima permasalahan yang muncul adalah untuk pencatatan dokumen yang masih manual menggunakan tangan sehingga pekerjaan berjalan lambat.

Kemudian permasalahan lain adalah masih digunakannya kartu stok yang bersifat fisik yang tentunya memiliki kekurangan diantaranya dokumen tersebut bisa hilang dan perlu dilakukan perhitungan manual pada saat *monitoring warehouse*. Belum tersedianya *database* penyimpanan barang jadi yang dilakukan dan aplikasi pembuatan laporannya serta penggunaan *file* data stok yang masih berbasis manual juga merupakan salah satu masalah yang muncul pada sistem tersebut.

Untuk membantu perusahaan dalam mengatasi masalah-masalah tersebut, bersama dengan pembuatan Tugas Akhir ini, penulis mengusulkan sebuah sistem penyimpanan barang jadi yang terintegrasi. Sistem tersebut nantinya akan menjadi sistem berbasis komputer yang memiliki fitur yang dapat mendukung proses penyimpanan pada PT Rekadaya Multi Adiprima menjadi lebih terkomputerisasi. Sebagai contoh penggunaan *database* sebagai pengganti kartu stok, pengolahan data penyimpanan barang jadi yang terkomputerisasi penyimpanan dan pembuatan laporan rekapitulasi penyimpanan barang jadi secara periodik.

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Sistem Informasi Berjalan

Berdasarkan hasil pengumpulan dan pengolahan data yang dilakukan pada sistem informasi yang sedang berjalan di *Warehouse Finished Goods* PT Rekadaya Multi Adiprima, dapat disimpulkan bahwa proses penyimpanan dan pengeluaran barang jadi masih bersifat manual. Hal ini dapat dilihat dengan masih digunakannya Form Kartu Stok sebagai dokumen fisik atas transaksi keluar masuknya barang jadi dan belum mempunyai *database*. Bagian PPIC juga masih menggunakan dokumen fisik Form Permintaan Produksi untuk memproduksi barang jadi kepada Bagian Produksi dan Form *Check Sheet Temporary* yang digunakan Bagian *Quality Control* dalam mencatat hasil pengecekan kualitas.

5.2 Analisis Kebutuhan Sistem

Berdasarkan hasil analisis sistem informasi berjalan, maka diajukan sistem informasi usulan yang khusus untuk menangani penyimpanan barang jadi yang ditujukan kepada *Warehouse Finished Goods*. Tabel V.1 menjelaskan daftar kebutuhan sistem informasi penyimpanan barang jadi di *Warehouse Finished Goods* PT Rekadaya Multi Adiprima.

Tabel V.1 Kebutuhan Sistem Informasi Usulan

<i>Project Name</i>	Sistem Informasi Penyimpanan Barang Jadi pada <i>Warehouse Finished Goods</i> .
---------------------	---

Tabel V.1 Kebutuhan Sistem Informasi Usulan (lanjutan)

<i>Business Recruitment</i>	Sistem informasi yang khusus untuk menangani proses penyimpanan barang jadi hingga proses pengeluaran barang jadi yang dapat membantu <i>Warehouse Finished Goods</i> .
<i>Business Needs</i>	<ol style="list-style-type: none"> 1. Terwujudnya sebuah aplikasi yang dapat membantu aktivitas penyimpanan barang jadi dan pengeluaran barang jadi seperti penginputan data barang jadi yang masuk atau keluar, sehingga lebih cepat dan tepat dalam proses pembuatannya. 2. Terwujudnya sebuah aplikasi yang dapat memberikan informasi yang mempermudah proses penyimpanan data dan dapat mencegah terjadinya kehilangan data karena adanya <i>database</i>.
<i>Business Value</i>	<ol style="list-style-type: none"> 1. Mengganti penggunaan kertas pada pengaplikasian Kartu Stok dengan <i>database</i> transaksi barang masuk dan keluar yang <i>real time</i>. 2. Mempermudah Bagian <i>Warehouse</i> dalam menerima informasi jadwal pengiriman barang jadi secara informatif 3. Mempermudah proses rekapitulasi data barang jadi masuk dan barang jadi keluar, serta meminimalisir permasalahan ketidaksesuaian data stok aktual dengan data stok yang tersimpan dalam <i>file</i> kartu stok
<i>Special Issues or Constrains</i>	-

Sumber: Pengolahan Data (2017)

5.3 Analisis Kebutuhan Rinci Sistem

Analisis kebutuhan rinci sistem untuk sistem informasi penyimpanan barang jadi pada *Warehouse Finished Goods* PT Rekadaya Multi Adiprima dijelaskan pada Tabel V.2

Tabel V.2 Kebutuhan Rinci Sistem

No.	Kebutuhan Rinci Sistem	Uraian
1	Pengelolaan data <i>master</i> barang jadi	Proses yang dapat dilakukan oleh <i>user</i> pada pengelolaan data <i>master</i> barang jadi yaitu: a. <i>Supervisor Warehouse</i> dapat menambah data <i>master</i> barang jadi b. <i>Supervisor Warehouse</i> dapat mengubah dan menghapus data <i>master</i> barang jadi
2	Pengelolaan data <i>master user</i>	Proses yang dapat dilakukan oleh <i>Supervisor Warehouse</i> pada mengelola data <i>master user</i> yaitu: a. <i>Supervisor Warehouse</i> dapat menambah <i>user</i> baru pada data <i>master user</i> . b. <i>Supervisor Warehouse</i> dapat mencari dan memilih data <i>user</i> tertentu, kemudian mengubah atau menghapus data <i>user</i> tersebut
3	Pengelolaan data <i>master customer</i>	Proses yang dapat dilakukan oleh <i>Supervisor Warehouse</i> pada mengelola data <i>master customer</i> yaitu: a. <i>Supervisor Warehouse</i> dapat menambah <i>customer</i> baru pada data <i>master customer</i> . b. <i>Supervisor Warehouse</i> dapat mencari dan memilih data <i>customer</i> tertentu, kemudian mengubah atau menghapus data <i>customer</i> tersebut

Tabel V.2 Kebutuhan Rinci Sistem (lanjutan)

No.	Kebutuhan Rinci Sistem	Uraian
		<i>Supervisor Warehouse</i> mengelola pembaharuan data stok barang jadi di gudang.
6	Membuat transaksi barang jadi masuk	Proses yang dapat dilakukan oleh <i>user</i> pad membuat transaksi barang jadi masuk yaitu: a. <i>Operator Warehouse</i> menginput data barang jadi sesuai dengan barang jadi yang masuk b. <i>Operator Warehouse</i> menyimpan data barang jadi masuk
7	Membuat transaksi barang jadi keluar	Proses yang dapat dilakukan oleh <i>user</i> pad membuat transaksi barang jadi keluar yaitu: a. <i>Operator warehouse</i> melihat jadwal pengiriman yang di- <i>input</i> oleh Admin PPIC pada data Rekap PO b. <i>Operator Warehouse</i> menginput data barang jadi keluar dan mencetak data barang jadi keluar.
8	Menampilkan dan Mencetak laporan data transaksi barang jadi.	<i>Supervisor Warehouse</i> dapat mencetak laporan dari data stok barang jadi yang telah diproses.

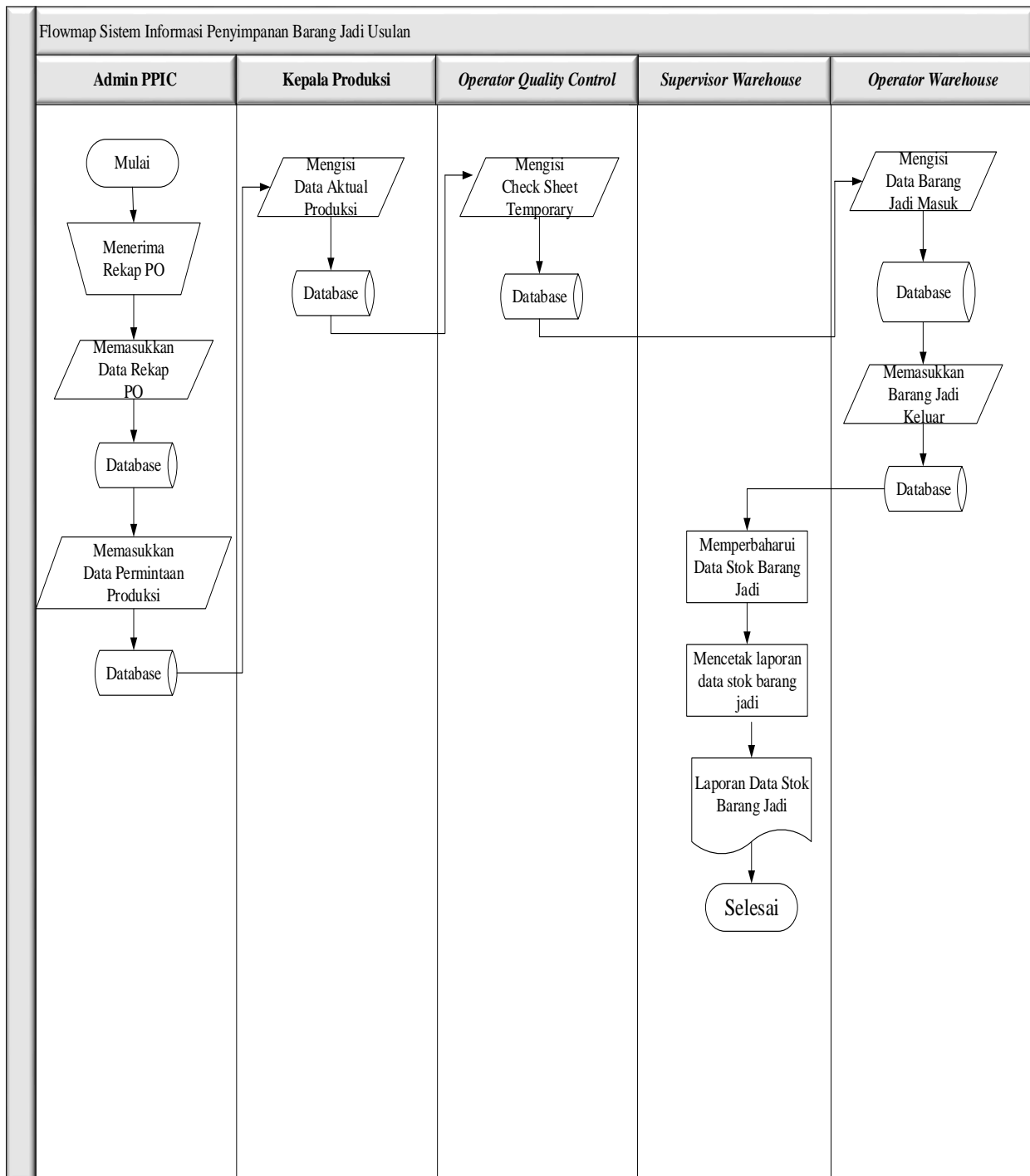
Sumber: Pengolahan Data (2017)

5.4 Perancangan *Flowmap* Sistem Usulan

Urutan prosedur dan *Flowmap* sistem informasi penyimpanan barang jadi yang diusulkan sebagai berikut:

1. Prosedur sistem informasi penyimpanan barang jadi yang diusulkan sebagai berikut:
 - a. Admin PPIC *Warehouse* menerima Rekap PO dari Marketing.
 - b. Admin PPIC *Warehouse* meng-*input* data Rekap PO ke dalam *database*.

- c. Admin PPIC *Warehouse* meng-*input* data Permintaan Produksi ke *database* untuk dilihat oleh Kepala Produksi.
 - d. Kepala Produksi melihat dan mengisi jumlah hasil produksi aktual ke data Permintaan Produksi yang telah di-*input* oleh Admin PPIC dan menyimpan ke *database*.
 - e. *Operator Quality Control* mengisi data hasil pengecekan kualitas ke dalam *Check Sheet Temporary*
 - f. *Operator Warehouse* memasukkan data barang jadi masuk ke *database*.
 - g. *Operator Warehouse* memasukkan data barang jadi keluar ke *database*.
 - h. *Supervisor Warehouse* melakukan pembaharuan data stok barang jadi.
 - i. *Supervisor Warehouse* mencetak laporan dari data stok barang jadi.
2. *Flowmap* sistem informasi penyimpanan barang jadi yang diusulkan sebagai berikut:



Gambar V.1 *Flowmap* Sistem Informasi Penyimpanan Barang Jadi Usulan
 Sumber: Hasil Analisis (2017)

5.5 Analisis Sistem Informasi Usulan

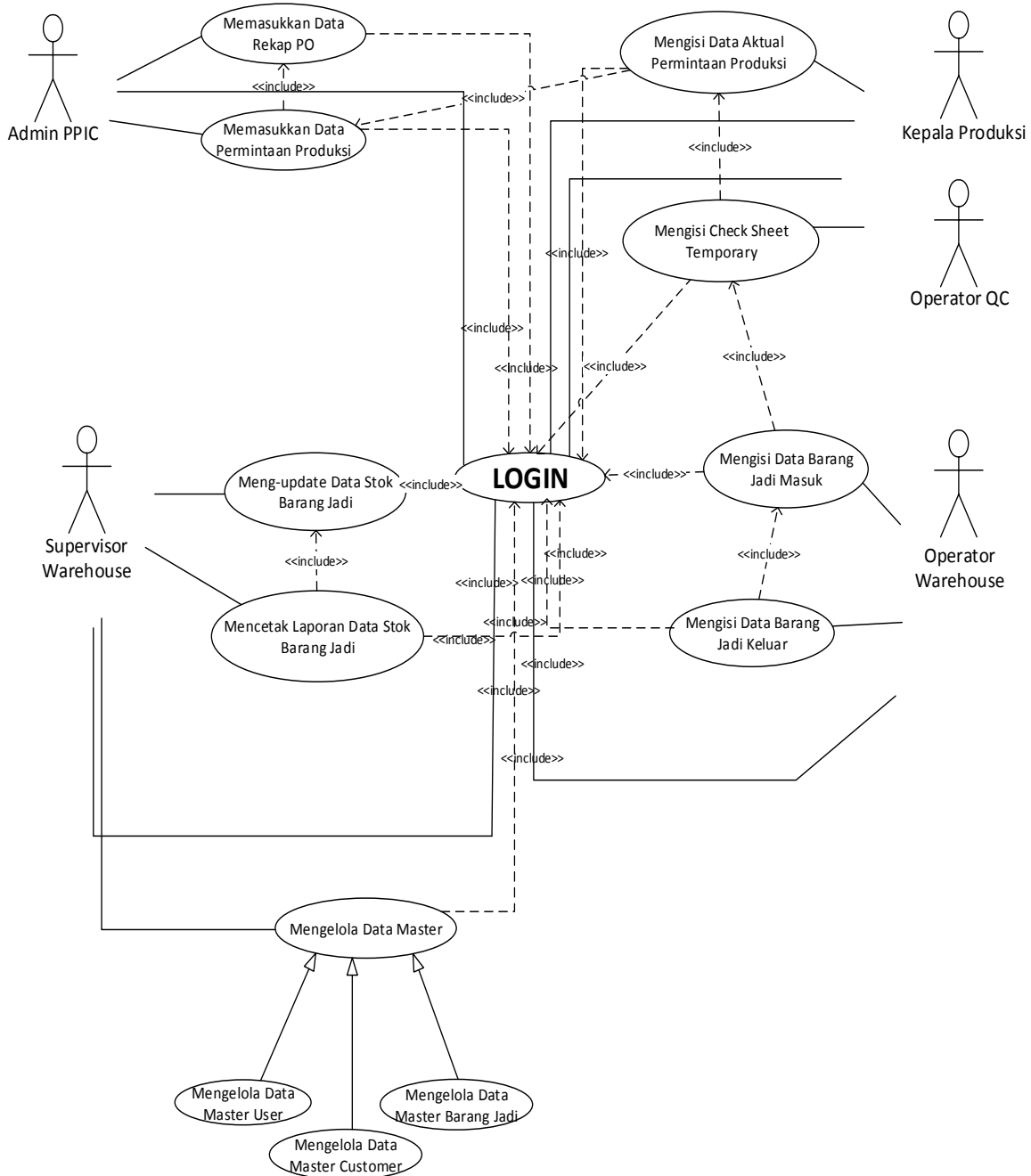
Analisis proses sistem informasi penyimpanan barang jadi pada *Warehouse Finished Goods* menggunakan *tools* pemodelan sistem UML (*Unified Modeling Language*) yang meliputi pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*, *deployment diagram* dan pembuatan kamus data. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun

5.5.1 Use Case Diagram

Use case diagram adalah deskripsi fungsi sebuah sistem dari perspektif pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem dipakai (Munawar, 2005)

Berdasarkan analisis yang telah dilakukan, maka *use case diagram* sistem informasi penyimpanan barang jadi usulan dapat dilihat pada Gambar V.2 sebagai berikut ini:

Use Case Diagram Sistem Informasi Penyimpanan Barang Jadi



Gambar V.2 Use Case Diagram Sistem Informasi Penyimpanan Barang Jadi Usulan
 Sumber: Hasil Analisis (2017)

Penjelasan *use case diagram* sistem informasi penyimpanan barang jadi di *warehouse finished goods* usulan dijelaskan sebagai berikut ini:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi penyimpanan barang jadi usulan dijelaskan pada Tabel V.3

Tabel V.3 Definisi Aktor *Use Case*

No	Aktor	Deskripsi
1	Admin PPIC	Admin PPIC merupakan aktor yang menerima dan meng- <i>input</i> Rekap PO serta meng- <i>input</i> data permintaan produksi ke produksi.
2	Kepala Produksi	Kepala Produksi merupakan aktor yang mengisi data aktual hasil produksi
3	Operator Quality Control	Operator <i>quality control</i> merupakan aktor yang mengecek kualitas barang jadi yang telah diproduksi serta mencatat hasil pengecekan ke <i>check sheet temporary</i>
4	Supervisor Warehouse	Supervisor Warehouse merupakan aktor yang melakukan pembaharuan data stok barang jadi serta mendapatkan akses untuk mencetak laporan data stok barang jadi.
5	Operator Warehouse	Operator Warehouse merupakan aktor yang meng- <i>input</i> data barang jadi masuk dan keluar.

Sumber: Hasil Analisis (2017)

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* sistem informasi penyimpanan barang jadi di *warehouse finished goods* usulan dijelaskan pada tabel berikut:

Tabel V.4 di bawah berisi deskripsi *use case* mengelola data *master*

Tabel V.4 Definisi *Use Case* Mengelola Data *Master*

<i>Use Case Name</i>	Mengelola data <i>master</i>
<i>Primary Actor</i>	Supervisor Warehouse
<i>Brief Description</i>	<p>Proses mengelola data <i>master</i> terdiri dari:</p> <ol style="list-style-type: none"> 1. Mengelola data <i>master user</i>, Supervisor Warehouse dapat membuat akun <i>user</i> baru serta mencari, mengubah, dan menghapus data <i>user</i> yang telah terdaftar sebelumnya. 2. Mengelola data <i>master customer</i>, Supervisor Warehouse dapat menambah, mencari, mengubah dan menghapus data <i>master customer</i>. 3. Mengelola data <i>master</i> barang jadi, Supervisor Warehouse dapat menambah, mencari, mengubah dan menghapus data <i>master</i> barang jadi.
<i>Relationship</i>	<p><i>Generalization</i>: Mengelola data <i>master user</i>, data <i>master customer</i> dan mengelola data <i>master</i> barang jadi.</p> <p><i>Include</i>: Login.</p>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Supervisor Warehouse telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. Supervisor Warehouse memilih menu Data Master. 4. Supervisor Warehouse memilih Submenu Data Master 5. Sistem menampilkan halaman data master yang dipilih. 6. Supervisor Warehouse dapat memilih aksi cari, simpan, edit dan hapus. 7. Jika memilih masukan kata kunci kemudian <i>user</i> klik cari. 8. Sistem menampilkan data sesuai kata kunci. 9. Jika memilih masukan data kemudian klik simpan maka sistem akan menyimpan data ke <i>database</i>. 10. Jika memilih edit, sistem akan menampilkan form ubah data

Sumber: Hasil Analisis (2017)

Tabel V.4 Definisi *Use Case* Mengelola Data *Master* (lanjutan)

	<p>11. Supervisor Warehouse akan mengubah data dan klik simpan.</p> <p>12. Jika memilih hapus, sistem akan menghapus data.</p> <p>13. Sistem akan memperbaharui basis data.</p> <p>14. Sistem akan menampilkan halaman data master.</p>
<i>Precondition</i>	Supervisor Warehouse telah melakukan <i>login</i> terlebih dahulu.

Tabel V.5 di bawah ini berisi deskripsi untuk *use case login*.

Tabel V.5 Definisi *Use Case Login*

<i>Use Case Name</i>	<i>Login</i>
<i>Primary Actor</i>	Admin PPIC, Kepala Produksi, Operator QC, Supervisor Warehouse, Operator Warehouse.
<i>Brief Description</i>	Proses yang dilakukan untuk menjaga akses sistem dari pihak yang tidak diizinkan untuk menggunakan sistem informasi ini. Proses <i>Login</i> merupakan proses <i>input username</i> dan <i>password</i> oleh <i>user</i> yang ingin masuk ke dalam sistem. Sistem nantinya akan memvalidasi data yang dimasukkan. Apabila <i>user</i> terdaftar maka <i>user</i> dapat menggunakan sistem sesuai dengan hak aksesnya.
<i>Relationship</i>	<i>Association</i> : Admin PPIC, Kepala Produksi, Operator Quality Control, Supervisor Warehouse, Operator Warehouse.
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> membuka aplikasi 2. Sistem menampilkan menu <i>login</i>. 3. <i>User</i> mengisi <i>username</i> dan <i>password</i> di menu <i>login</i>. 4. Sistem memvalidasi data yang di-input. Apabila data tidak <i>valid</i>, maka muncul pesan gagal <i>login</i> dan <i>user</i> tersebut tidak dapat menggunakan sistem. 5. Apabila data <i>valid</i>, maka sistem akan menampilkan menu utama sesuai hak akses.

Sumber: Hasil Analisis (2017)

Tabel V.6 di bawah ini berisi deskripsi untuk *use case* memasukkan data Rekap PO

Tabel V.6 Definisi *Use Case* Memasukkan Data Rekap PO

<i>Use Case Name</i>	Memasukkan data Rekap PO
<i>Primary Actor</i>	Admin PPIC
<i>Brief Description</i>	Proses memasukkan data Rekap PO dilakukan apabila Admin PPIC telah menerima Rekap PO dari Marketing.
<i>Relationship</i>	<i>Association:</i> Admin PPIC <i>Include:</i> Login
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin PPIC telah sukses melakukan login 2. Sistem menampilkan menu utama 3. Admin PPIC memilih Menu Data Transaksi. 4. Admin PPIC memilih Submenu Data Rekap PO 5. Sistem menampilkan halaman submenu Data Rekap PO. 6. Admin PPIC klik Tambah. 7. Sistem menampilkan Form Tambah Data Rekap PO. 8. Admin PPIC memasukkan Data Rekap PO, klik simpan. 9. Sistem menampilkan Form Detail Rekap PO 10. Admin PPIC dapat memilih aksi Simpan, Edit, Hapus. 11. Admin PPIC memasukan data detail PO 12. Jika memilih Simpan, sistem akan menyimpan ke <i>database</i>. 13. Jika memilih Edit, sistem akan menampilkan Form Ubah Data. 14. Jika memilih Hapus, sistem akan menghapus data. 15. Sistem memperbaharui data, klik Selesai. 16. Sistem menampilkan halaman Data Rekap PO.
<i>Precondition</i>	Admin PPIC telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.7 di bawah ini berisi deskripsi untuk *use case* memasukkan data form permintaan produksi.

Tabel V.7 Definisi *Use Case* Memasukkan Data Permintaan Produksi

<i>Use Case Name</i>	Memasukkan data permintaan produksi
<i>Primary Actor</i>	Admin PPIC
<i>Brief Description</i>	Proses memasukkan data permintaan produksi dilakukan apabila Admin PPIC telah meng- <i>input</i> data rekap PO.
<i>Relationship</i>	<i>Association:</i> Admin PPIC <i>Include:</i> Login, memasukkan data Rekap PO
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin PPIC telah sukses melakukan login 2. Sistem menampilkan menu utama 3. Admin PPIC memilih Menu Data Transaksi. 4. Admin PPIC memilih Submenu Data FPP 5. Sistem menampilkan halaman submenu Data FPP. 6. Admin PPIC klik Tambah. 7. Sistem menampilkan Form Tambah Data FPP. 8. Admin PPIC memasukkan Data FPP, klik simpan. 9. Sistem menampilkan Form Detail FPP. 10. Admin PPIC dapat memilih aksi Simpan, Edit, Hapus. 11. Admin PPIC memasukan data detail FPP 12. Jika memilih Simpan, sistem akan menyimpan ke <i>database</i>. 13. Jika memilih Edit, sistem akan menampilkan Form Ubah Data. 14. Jika memilih Hapus, sistem akan menghapus data. 15. Sistem memperbaharui data, klik Selesai. 16. Sistem menampilkan halaman Data Permintaan Produksi.
<i>Precondition</i>	Admin PPIC telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.8 di bawah ini berisi deskripsi untuk *use case* mengisi data aktual permintaan produksi

Tabel V.8 Definisi *Use Case* Mengisi Data Aktual Permintaan Produksi

<i>Use Case Name</i>	Mengisi Data Aktual Permintaan Produksi
<i>Primary Actor</i>	Kepala Produksi
<i>Brief Description</i>	Proses mengisi data permintaan produksi dilakukan oleh Kepala Produksi
<i>Relationship</i>	<i>Association:</i> Kepala Produksi <i>Include:</i> Login, Memasukkan data permintaan produksi
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Produksi telah sukses melakukan <i>login</i>. 2. Sistem menampilkan menu utama. 3. Kepala produksi memilih menu data transaksi. 4. Kepala Produksi memilih Submenu Data FPP. 5. Sistem menampilkan <i>form</i> data permintaan produksi. 6. Kepala produksi klik Detail, sistem menampilkan FPP. 7. Kepala Produksi klik View. 8. Kepala Produksi memasukkan data aktual permintaan produksi. 9. Kepala Produksi klik Simpan, sistem menampilkan FPP. 10. Kepala Produksi klik Selesai. 11. Sistem memperbaharui basis data. 12. Sistem menampilkan halaman Data Permintaan Produksi.
<i>Precondition</i>	Kepala produksi telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.9 di bawah ini berisi deskripsi untuk *use case* Memasukkan Data *Check Sheet Temporary*

Tabel V.9 Definisi *Use Case* Mengisi Data *Check Sheet Temporary*

<i>Use Case Name</i>	Mengisi Data <i>Check Sheet Temporary</i>
<i>Primary Actor</i>	Operator Quality Control

Tabel V.9 Definisi *Use Case* Mengisi Data *Check Sheet Temporary* (lanjutan)

<i>Brief Description</i>	Proses pengisian data <i>check sheet temporary</i> oleh Operator QC
<i>Relationship</i>	<i>Association:</i> Operator QC <i>Include:</i> Login, Mengisi Data Permintaan Produksi
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Operator QC telah sukses melakukan login 2. Sistem menampilkan menu utama 3. Operator QC memilih Menu Data Transaksi. 4. Operator QC memilih Submenu Data CST. 5. Sistem menampilkan halaman submenu Data CST. 6. Operator QC klik Tambah. 7. Sistem menampilkan Form Tambah Data CST. 8. Operator QC memasukkan Data CST, klik simpan. 9. Sistem menampilkan Form Detail CST. 10. Operator QC dapat memilih aksi Simpan, Edit, Hapus. 11. Operator QC memasukan data detail CST. 12. Jika memilih Simpan, sistem akan menyimpan ke <i>database</i>. 13. Jika memilih Edit, sistem akan menampilkan Form Ubah Data. 14. Jika memilih Hapus, sistem akan menghapus data. 15. Sistem memperbaharui data, klik Selesai. 16. Sistem menampilkan halaman <i>Check Sheet Temporary</i>.
<i>Precondition</i>	Operator QC telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.10 di bawah ini berisi deskripsi untuk *use case* mengisi data barang jadi masuk

Tabel V.10 Definisi *Use Case* Mengisi Data Barang Jadi Masuk

<i>Use Case Name</i>	Mengisi Data Barang Jadi Masuk
<i>Primary Actor</i>	Operator Warehouse

Tabel V.10 Definisi *Use Case* Mengisi Data Barang Jadi Masuk (lanjutan)

<i>Brief Description</i>	Proses pengisian data barang jadi masuk oleh <i>Operator Warehouse</i>
<i>Relationship</i>	<i>Association: Operator Warehouse</i> <i>Include: Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Operator Warehouse telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. Operator Warehouse memilih menu Data Transaksi. 4. Operator Warehouse memilih Submenu Data Transaksi Barang Jadi Masuk 5. Sistem menampilkan halaman Submenu Data Transaksi Barang Jadi Masuk. 6. Operator Warehouse memasukkan Data Barang Jadi Masuk, klik Simpan. 7. Jika Operator Warehouse memilih aksi Hapus, Sistem menghapus transaksi masuk. 8. Sistem memperbaharui basis data. 9. Sistem menampilkan halaman menu Data Transaksi Barang Jadi Masuk.
<i>Precondition</i>	Operator Warehouse telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.11 di bawah ini berisi deskripsi untuk *use case* mengisi data barang jadi keluar

Tabel V.11 Definisi *Use Case* Mengisi Data Barang Jadi Keluar

<i>Use Case Name</i>	Mengisi Data Barang Jadi Keluar
<i>Primary Actor</i>	Operator Warehouse
<i>Brief Description</i>	Proses mengisi data barang jadi keluar oleh Operator Warehouse
<i>Relationship</i>	<i>Association: Operator Warehouse</i> <i>Include: Login</i>

Tabel V.11 Definisi *Use Case* Mengisi Data Barang Jadi Keluar (lanjutan)

<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Operator Warehouse telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. Operator Warehouse memilih menu Data Transaksi. 4. Operator Warehouse memilih Submenu Data Transaksi Barang Jadi Keluar. 5. Sistem menampilkan halaman Submenu Data Transaksi Barang Jadi Keluar. 6. Operator Warehouse memasukkan Data Barang Jadi Keluar, klik Simpan. 7. Jika Operator Warehouse memilih aksi Hapus, Sistem menghapus transaksi keluar. 8. Sistem memperbaharui basis data. 9. Sistem menampilkan halaman menu Data Transaksi Barang Jadi Keluar.
<i>Precondition</i>	Operator Warehouse telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.12 di bawah ini berisi deskripsi untuk *use case* meng-*update* data transaksi barang jadi

Tabel V.12 Definisi *Use Case* Meng-*Update* Data Transaksi Barang Jadi

<i>Use Case Name</i>	Meng- <i>update</i> Data Transaksi Barang Jadi
<i>Primary Actor</i>	Supervisor Warehouse
<i>Brief Description</i>	Proses meng- <i>update</i> transaksi data barang jadi masuk dan keluar.
<i>Relationship</i>	<i>Association:</i> Supervisor Warehouse <i>Include:</i> Login.
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Supervisor Warehouse telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. Supervisor Warehouse memilih menu data laporan.

Tabel V.12 Definisi *Use Case* Meng-*Update* Data Transaksi Barang Jadi (lanjutan)

	<ol style="list-style-type: none"> 4. Supervisor Warehouse memilih Submenu Data Laporan Transaksi Barang Jadi. 5. Sistem menampilkan halaman Submenu Data Laporan Transaksi Barang Jadi. 6. Supervisor Warehouse memasukkan data yang ingin ditampilkan, klik Lihat Laporan. 7. Sistem menampilkan laporan barang jadi. 8. Supervisor Warehouse meng-<i>update</i> stok barang jadi. 9. Sistem menampilkan Data Laporan Transaksi Barang Jadi.
<i>Precondition</i>	Supervisor Warehouse telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.13 di bawah ini berisi deskripsi untuk *use case* mencetak laporan data stok barang jadi

Tabel V.13 Definisi *Use Case* Mencetak Laporan Data Stok Barang Jadi

<i>Use Case Name</i>	Mencetak Laporan Data Stok Barang Jadi
<i>Primary Actor</i>	Supervisor Warehouse
<i>Brief Description</i>	Proses mencetak laporan data stok barang jadi
<i>Relationship</i>	<i>Association:</i> Supervisor Warehouse <i>Include:</i> Login, Meng- <i>update</i> data stok barang jadi
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Supervisor Warehouse telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. Supervisor Warehouse memilih menu data laporan. 4. Supervisor Warehouse memilih Submenu Data Laporan Transaksi Barang Jadi. 5. Sistem menampilkan halaman Submenu Data Laporan Transaksi Barang Jadi. 6. Supervisor Warehouse memasukkan data yang ingin ditampilkan, klik Lihat Laporan.

	<p>7. Sistem menampilkan laporan barang jadi.</p> <p>8. Supervisor Warehouse klik Cetak untuk mencetak laporan data stok barang jadi.</p> <p>9. Sistem menampilkan Data Laporan Transaksi Barang Jadi.</p>
<i>Precondition</i>	Supervisor Warehouse telah melakukan <i>login</i> terlebih dahulu

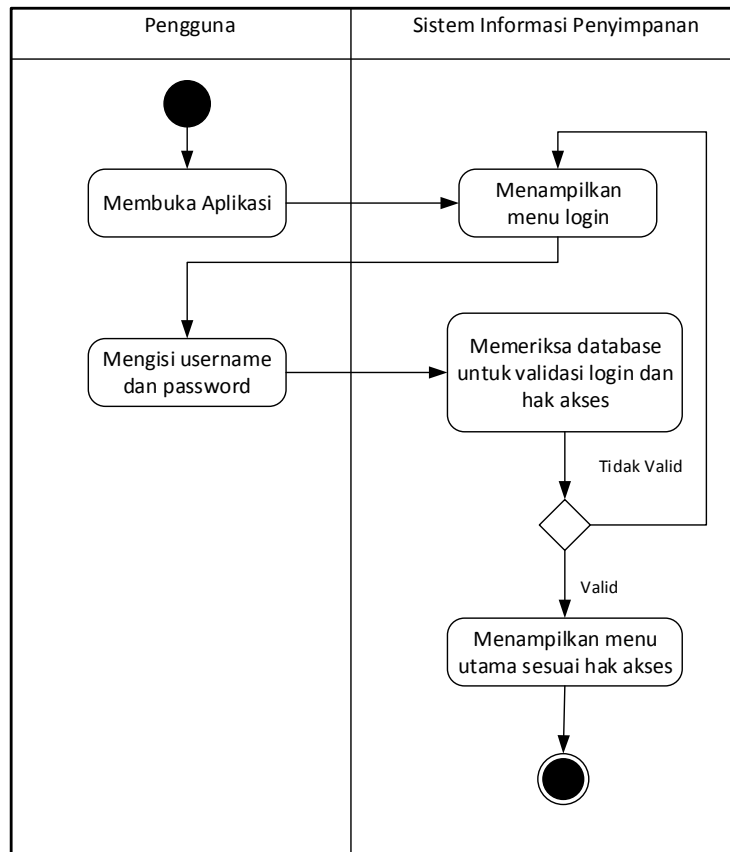
Sumber: Hasil Analisis (2017)

5.5.2 Activity Diagram

Activity Diagram digunakan untuk menggambarkan aliran tiap *use case* pada sistem informasi penyimpanan barang jadi. *Activity diagram* masing-masing *use case* digambarkan pada gambar-gambar berikut ini:

1. Activity Diagram Proses Login

Activity diagram berikut ini menggambarkan aktivitas saat melakukan proses *login*. Pengguna mengisi *username* dan *password* untuk dapat masuk ke sistem. *Activity diagram* proses *login* digambarkan pada Gambar V.3.



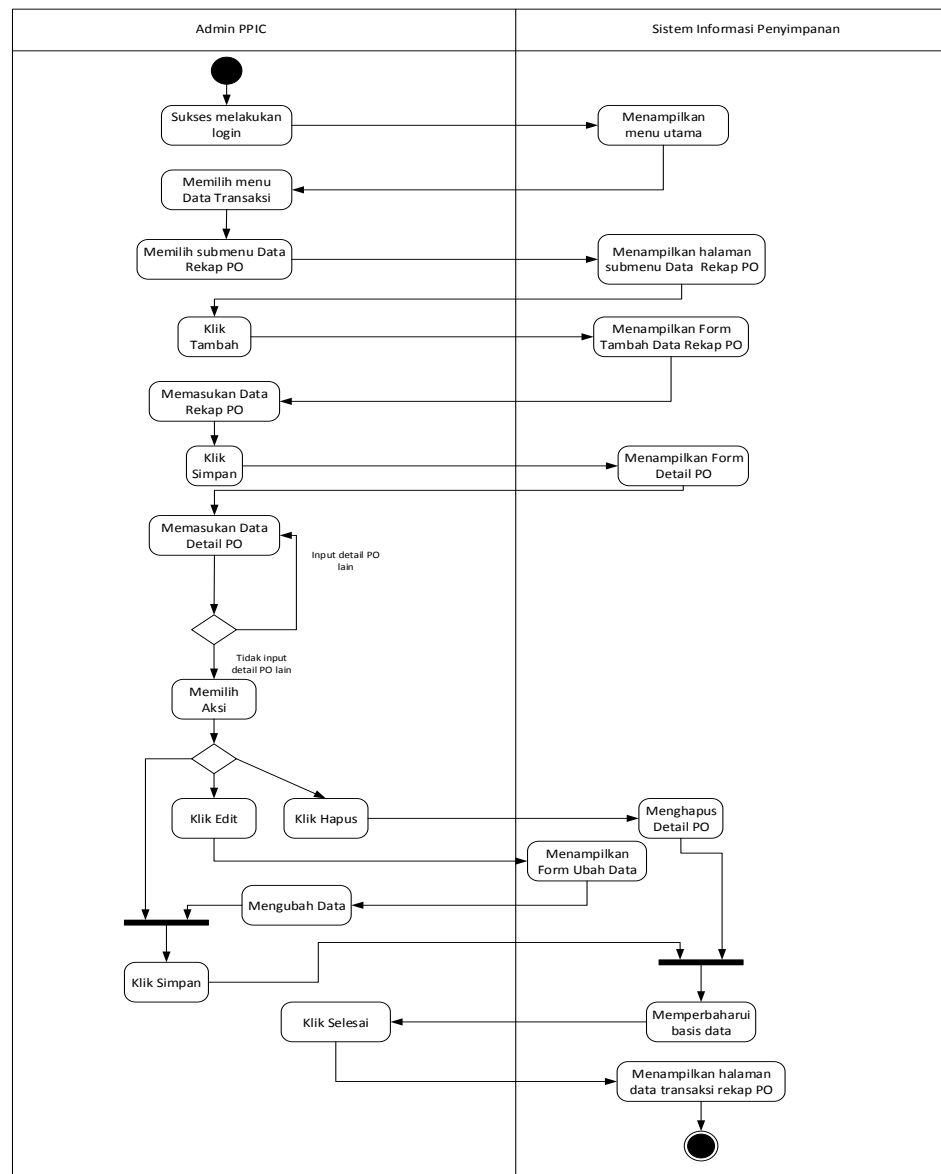
Gambar V.3 *Activity Diagram* Proses Login
Sumber: Hasil Analisis (2017)

2. *Activity Diagram* Proses Mengelola Data Master

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan melakukan mengelola data *master*, dimana Supervisor Warehouse dapat menambah, mencari, mengubah dan menghapus data *master*. *Activity diagram* proses mengelola data *master*. Digambarkan pada Gambar V.4

3. Activity Diagram Proses Memasukkan Data Rekap PO

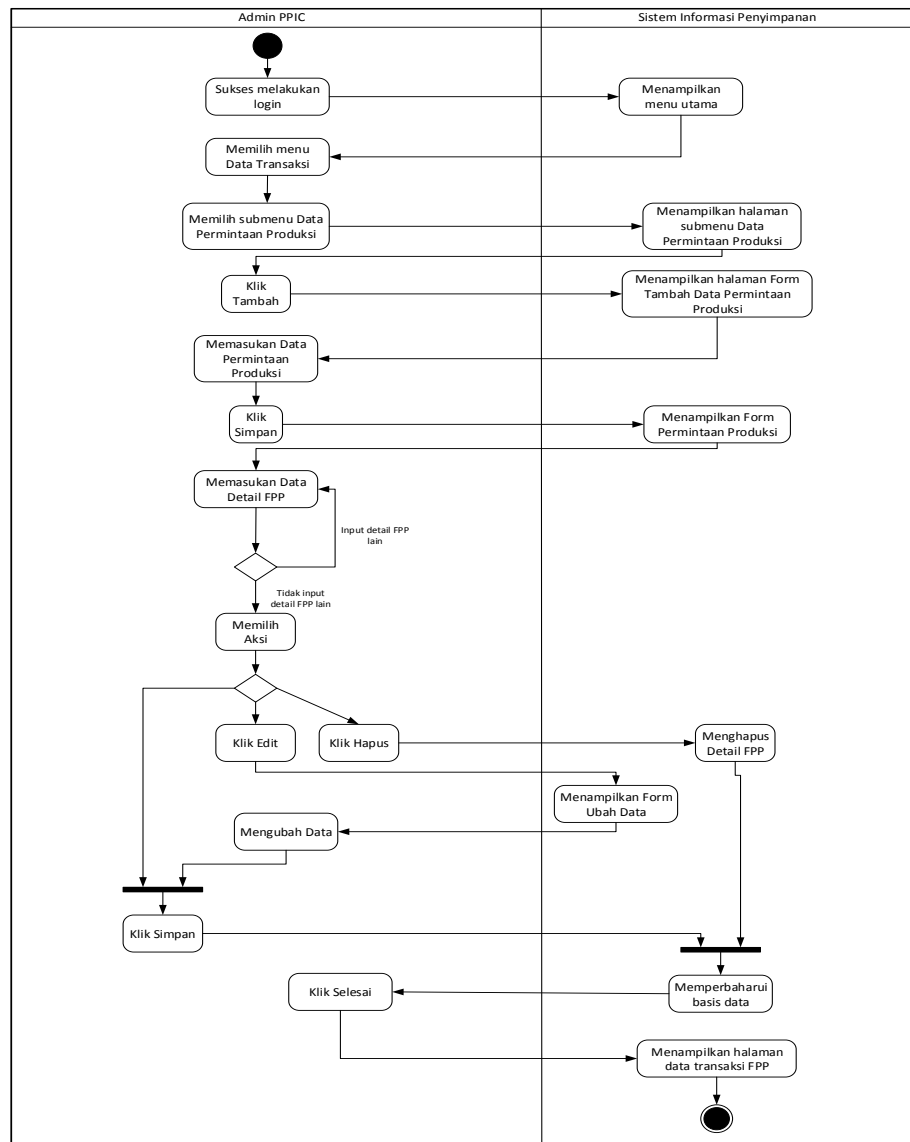
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan memasukkan data rekap PO, dimana Admin PPIC menerima rekap PO dari Marketing kemudian meng-input data PO yang masuk ke database untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses memasukkan data rekap PO digambarkan pada Gambar V.5.



Gambar V.5 Activity Diagram Proses Memasukkan Data Rekap PO
Sumber: Hasil Analisis (2017)

4. Activity Diagram Proses Memasukkan Data Permintaan Produksi

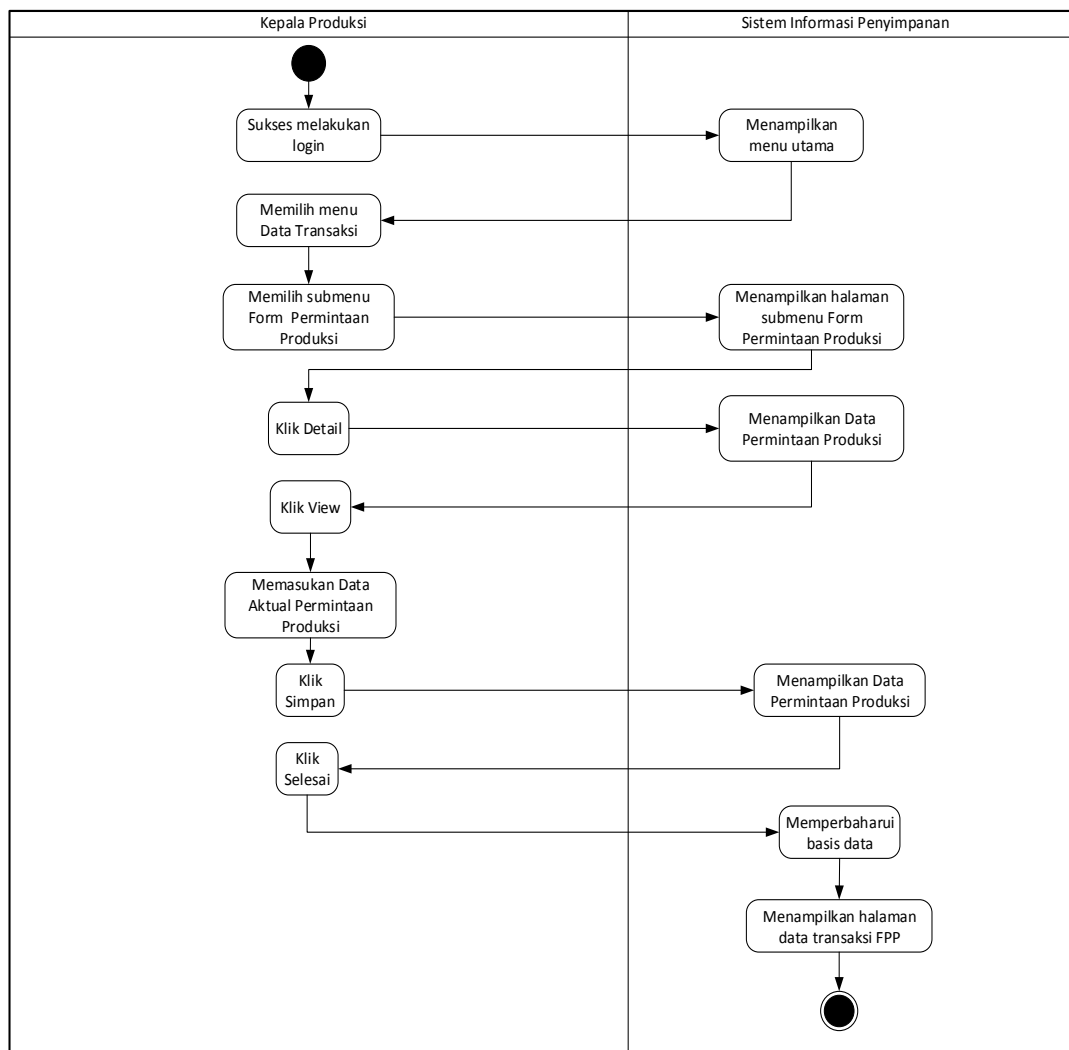
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan memasukkan data Permintaan Produksi, dimana Admin PPIC meng-input data Permintaan Produksi yang masuk ke database untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses memasukkan data Permintaan Produksi digambarkan pada Gambar V.6.



Gambar V.6 Activity Diagram Proses Memasukkan Data Permintaan Produksi
Sumber: Hasil Analisis (2017)

5. Activity Diagram Mengisi Data Aktual Permintaan Produksi

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mengisi data permintaan produksi, dimana Kepala Produksi melihat daftar permintaan produksi dan meng-*input* data permintaan produksi aktual kemudian disimpan ke database untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses mengisi data permintaan produksi digambarkan pada Gambar V.7.

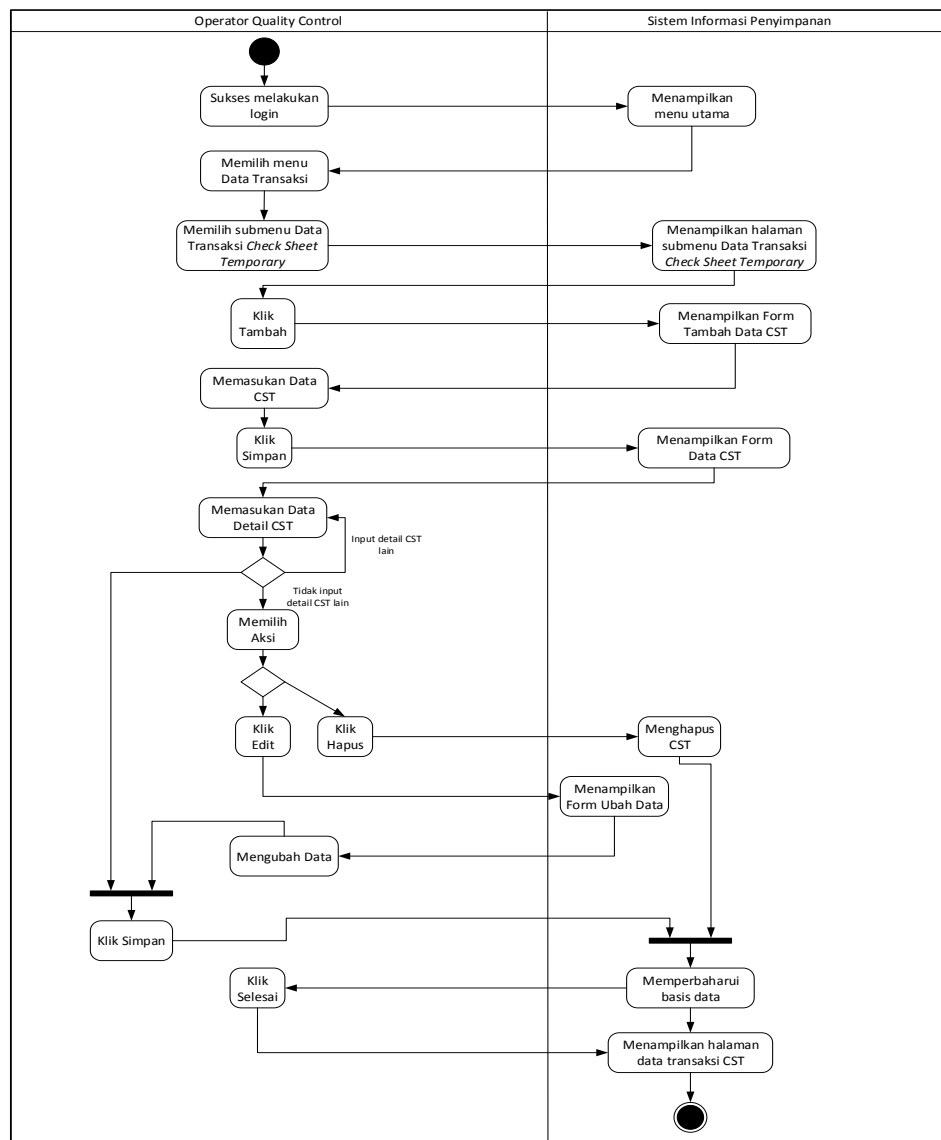


Gambar V.7 Activity Diagram Proses Mengisi Data Permintaan Produksi

Sumber: Hasil Analisis (2017)

6. Activity Diagram Mengisi Data *Check Sheet Temporary*

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mengisi data *check sheet temporary*, dimana Operator *Quality Control* meng-input data pengecekan kualitas barang jadi kemudian disimpan ke database untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses mengisi data *check sheet temporary* digambarkan pada Gambar V.8.

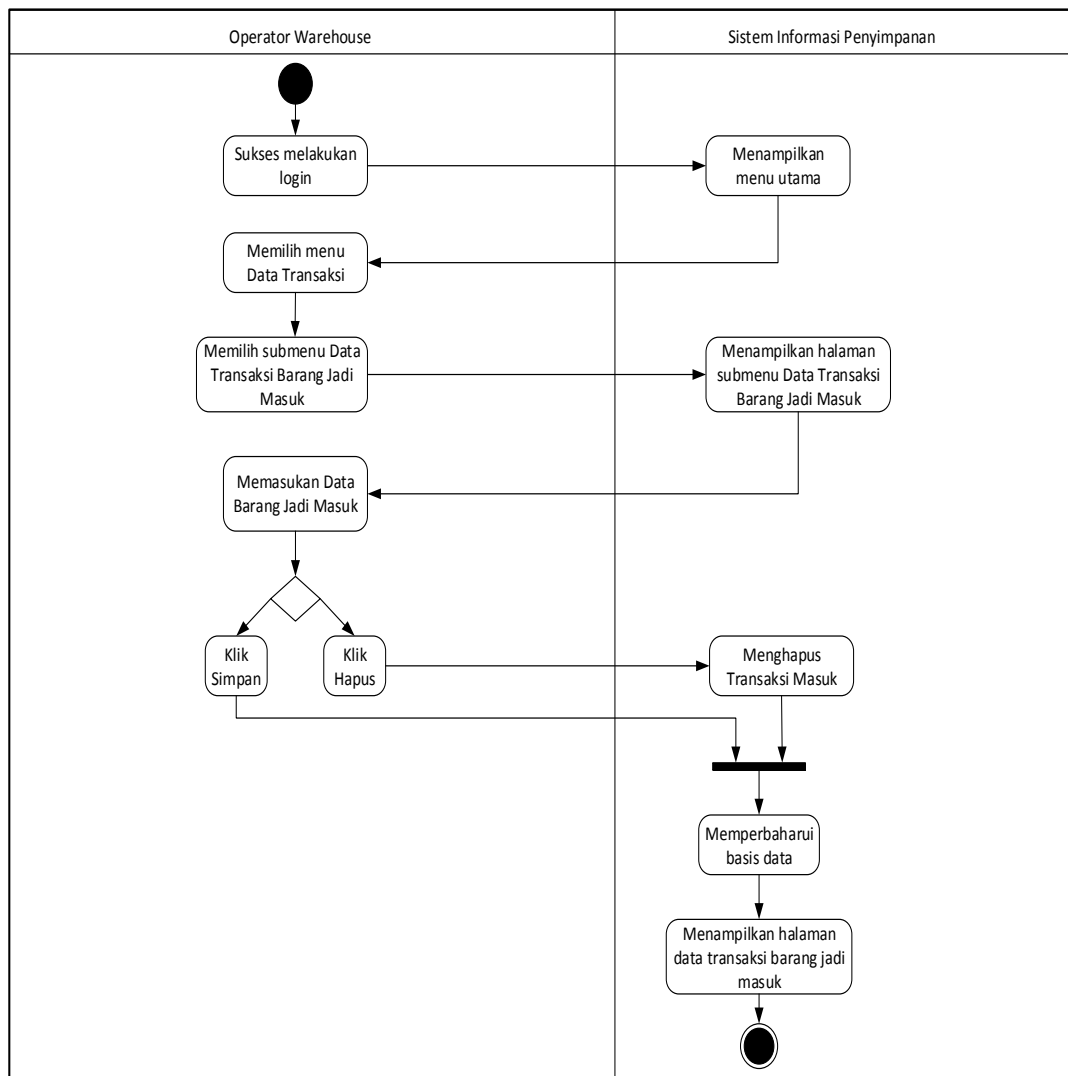


Gambar V.8 Activity Diagram Proses Mengisi Data *Check Sheet Temporary*

Sumber: Hasil Analisis (2017)

7. Activity Diagram Mengisi Data Barang Jadi Masuk

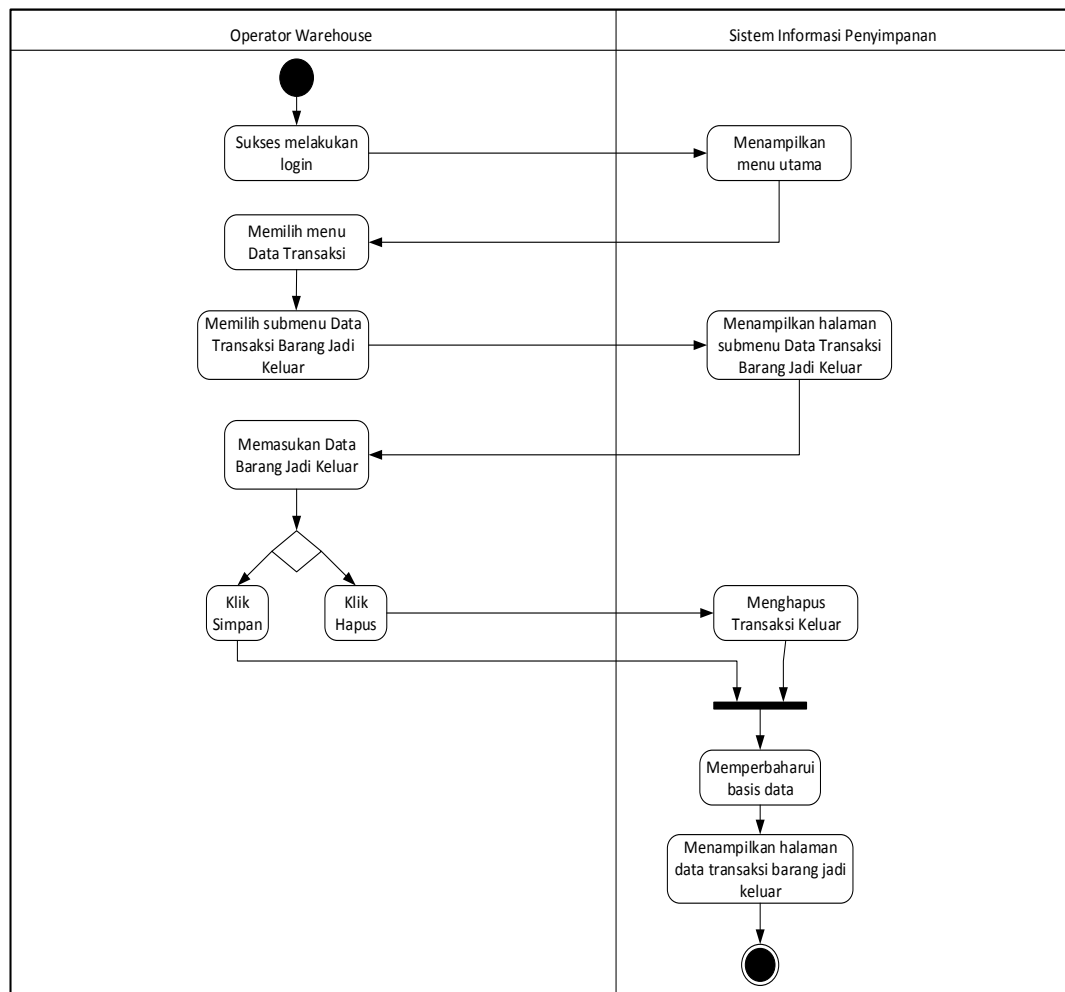
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mengisi data barang jadi masuk, dimana Operator Warehouse meng-*input* data barang jadi yang masuk kemudian disimpan ke database untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses mengisi data barang jadi masuk digambarkan pada Gambar V.9.



Gambar V.9 Activity Diagram Proses Mengisi Data Barang Jadi Masuk
Sumber: Hasil Analisis (2017)

8. Activity Diagram Mengisi Data Barang Jadi Keluar

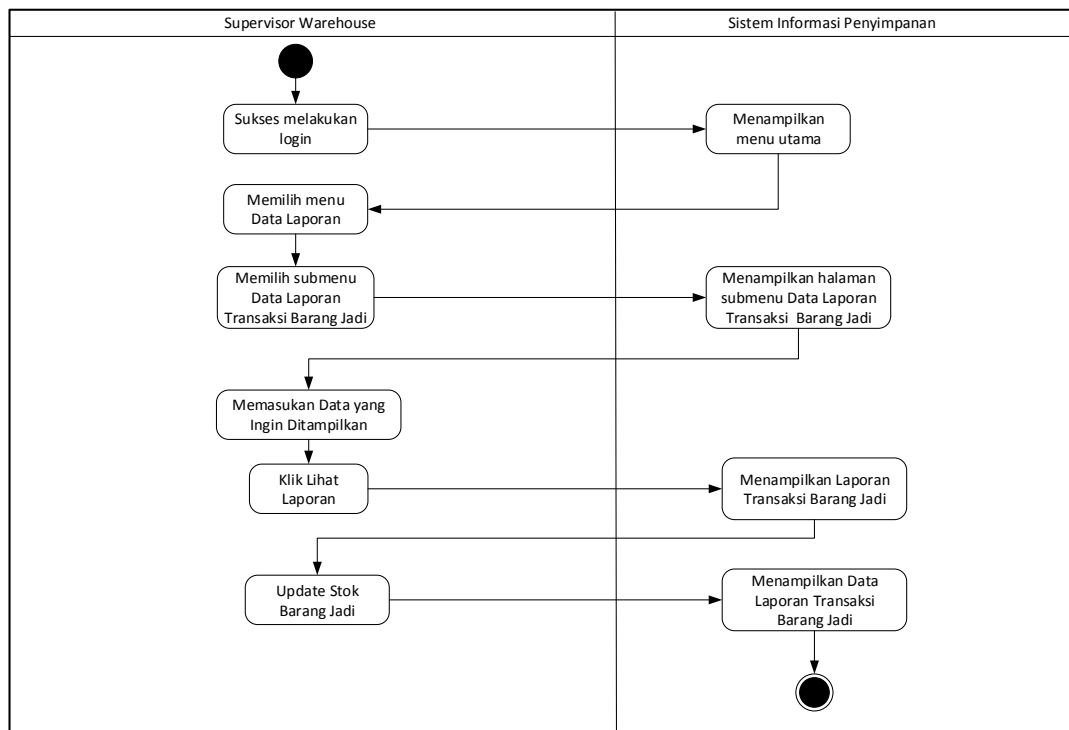
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mengisi data barang jadi keluar, dimana *Operator Warehouse* meng-input data barang jadi yang keluar kemudian disimpan ke database untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses mengisi data barang jadi masuk digambarkan pada Gambar V.10.



Gambar V.10 Activity Diagram Proses Mengisi Data Barang Jadi Keluar
Sumber: Hasil Analisis (2017)

9. Activity Diagram Meng-Update Data Stok Barang Jadi

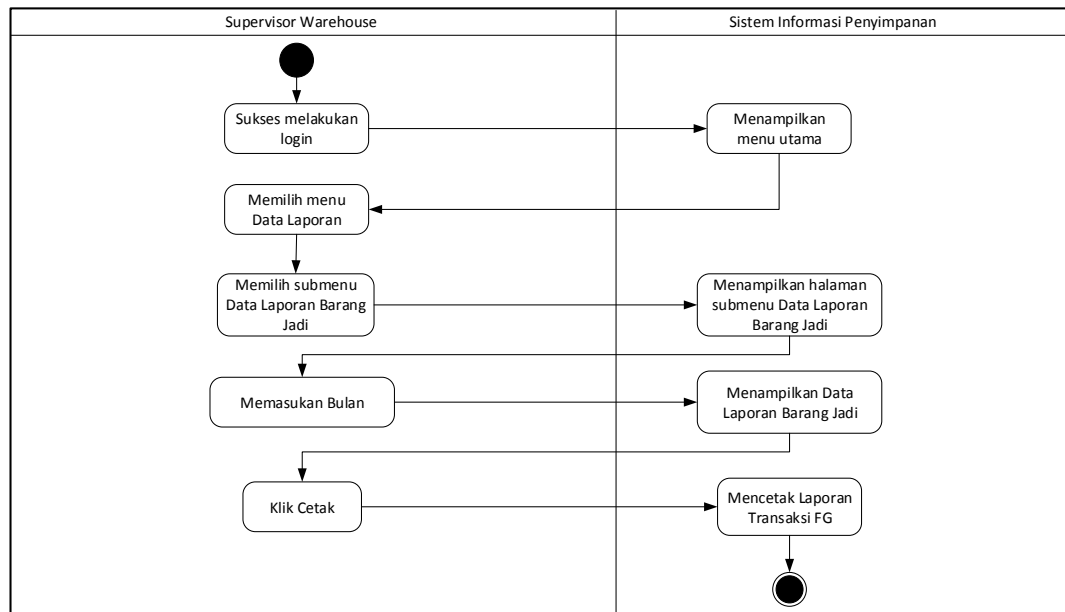
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan meng-update data stok barang jadi, dimana *Supervisor Warehouse* meng-update data stok barang jadi untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses meng-update data stok barang jadi digambarkan pada Gambar V.11.



Gambar V.11 Activity Diagram Proses Meng-Update Data Stok Barang Jadi
Sumber: Hasil Analisis (2017)

10. Activity Diagram Mencetak Laporan Rekapitulasi Data Stok Barang Jadi

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mencetak laporan rekapitulasi data stok barang jadi, dimana *Supervisor Warehouse* mencetak laporan rekapitulasi data stok barang jadi untuk mengetahui laporan bulanan dari stok barang jadi. Activity diagram proses mencetak laporan rekapitulasi data stok barang jadi digambarkan pada Gambar V.12.



Gambar V.12 *Activity Diagram* Proses Mencetak Laporan Rekapitulasi Data Stok Barang Jadi
Sumber: Hasil Analisis (2017)

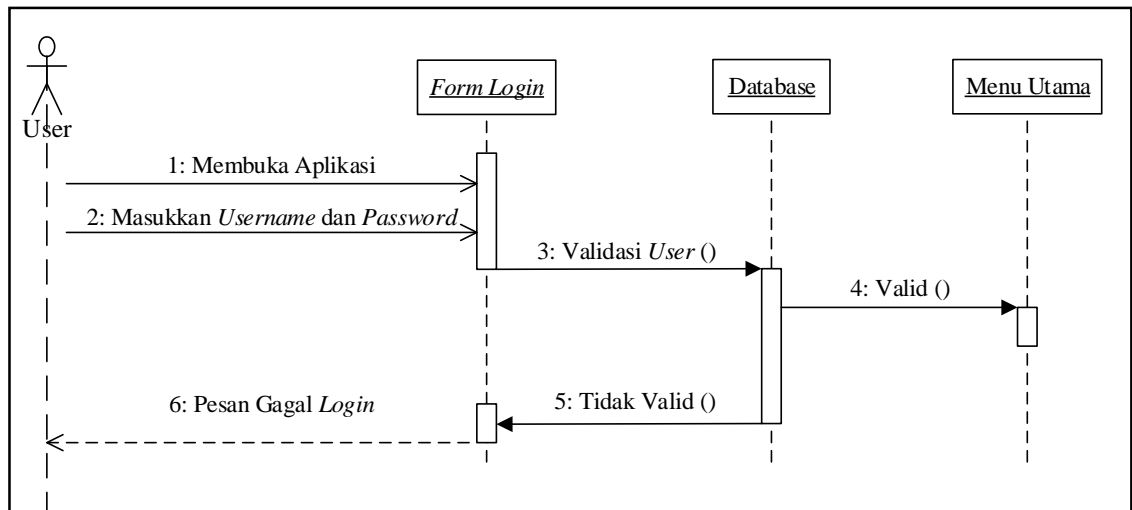
5.5.3 *Sequence Diagram*

Sequence diagram menggambarkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu, di mana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada diagram.

Hubungan yang ada pada gambar-gambar di bawah ini adalah proses yang dilakukan oleh sistem sesuai dengan objek pada *use case diagram*, berikut adalah *sequence diagram* pada sistem informasi penyimpanan barang jadi pada *warehouse finished goods* usulan:

1. *Sequence Diagram* pada Proses Login

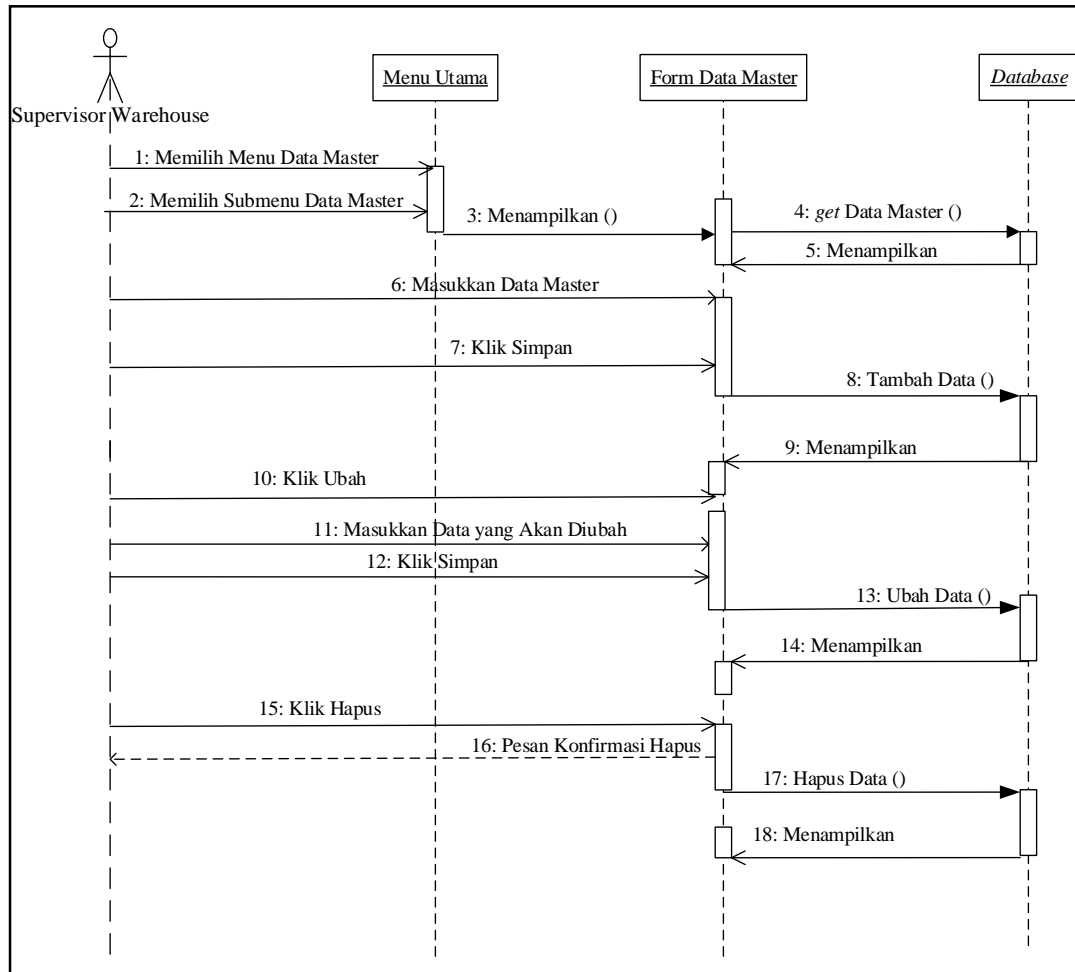
Sequence diagram login menggambarkan interaksi yang terjadi pada proses login. Proses ini dilakukan oleh seluruh aktor dalam *use case diagram*. *Sequence diagram* proses login digambarkan pada Gambar V.13.



Gambar V.13 *Sequence Diagram* Proses Login
Sumber: Hasil Analisis (2017)

2. *Sequence Diagram* Proses Mengelola Data Master

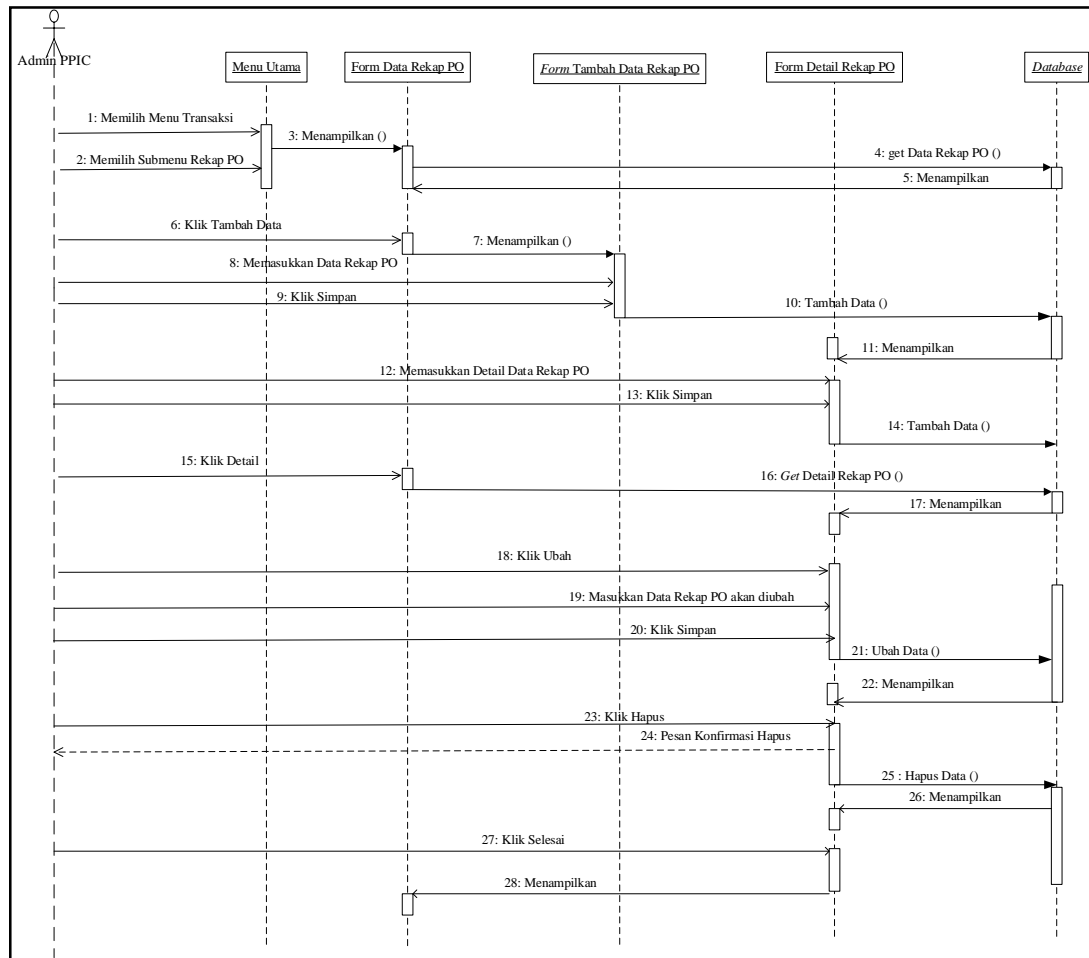
Sequence diagram mengelola data master menggambarkan interaksi yang terjadi dalam mengelola data master seperti menambah, menghapus dan mengubah data master. Adapun *sequence diagram* proses mengelola data master digambarkan pada Gambar V.14.



Gambar V.14 *Sequence Diagram* Mengelola Data Master
 Sumber: Hasil Analisis (2017)

3. *Sequence Diagram* Proses Memasukkan Data Rekap PO

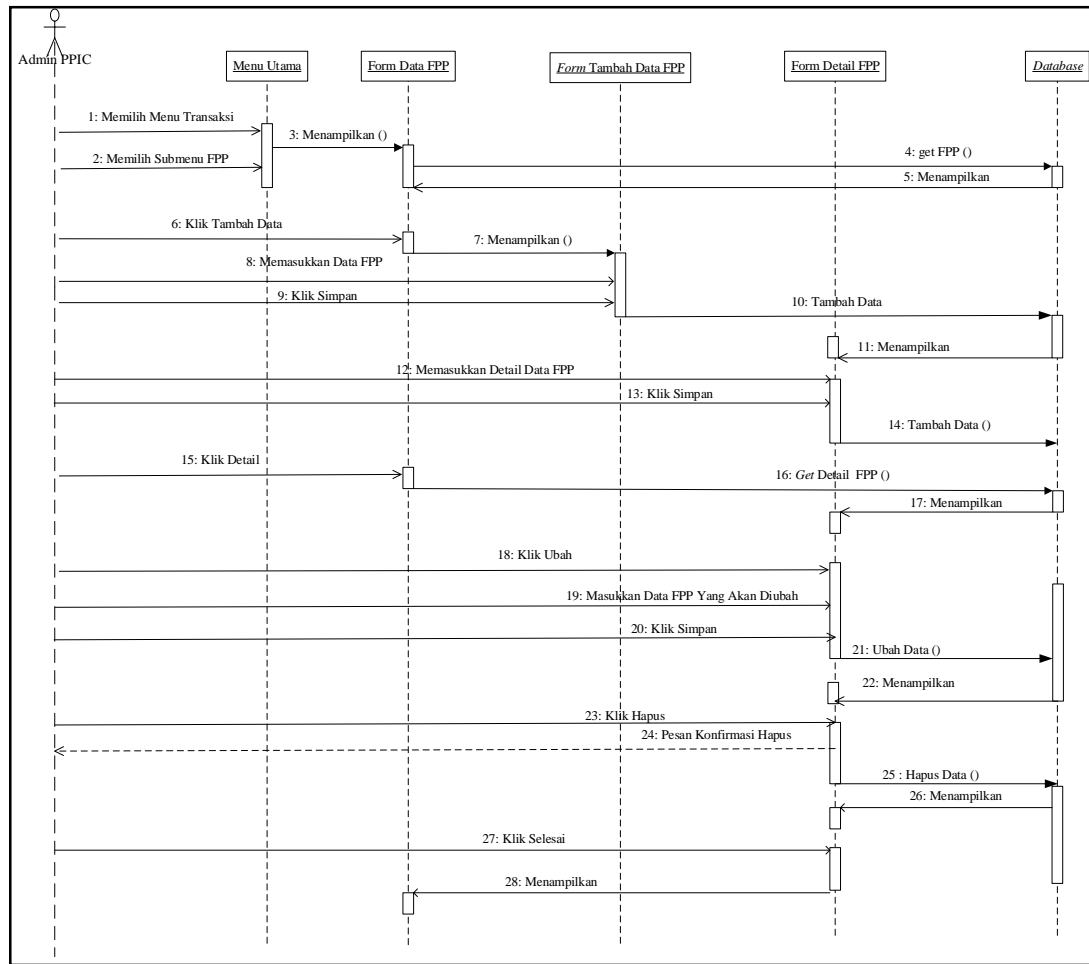
Sequence diagram memasukkan data rekap PO menjelaskan sebuah *sequence diagram* dalam proses memasukkan data rekap PO. Adapun *sequence diagram* proses memasukkan data rekap PO digambarkan pada Gambar V.15.



Gambar V.15 *Sequence Diagram* Memasukkan Data Rekap PO
 Sumber: Hasil Analisis (2017)

4. *Sequence Diagram* Proses Memasukkan Data Permintaan Produksi

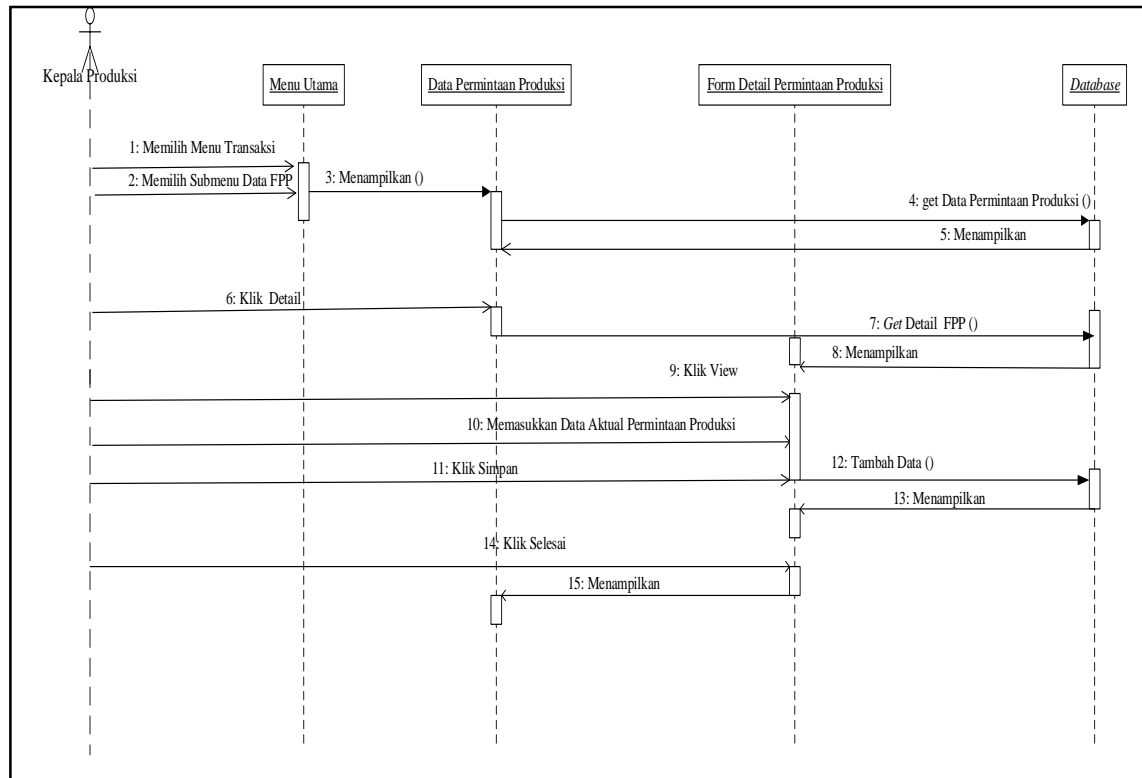
Sequence diagram memasukkan data permintaan produksi menjelaskan sebuah *sequence diagram* dalam proses memasukkan data permintaan produksi. Adapun *sequence diagram* proses memasukkan data permintaan produksi digambarkan pada Gambar V.16.



Gambar V.16 *Sequence Diagram* Memasukkan Data Permintaan Produksi
Sumber: Hasil Analisis (2017)

5. *Sequence Diagram* Proses Mengisi Data Aktual Permintaan Produksi

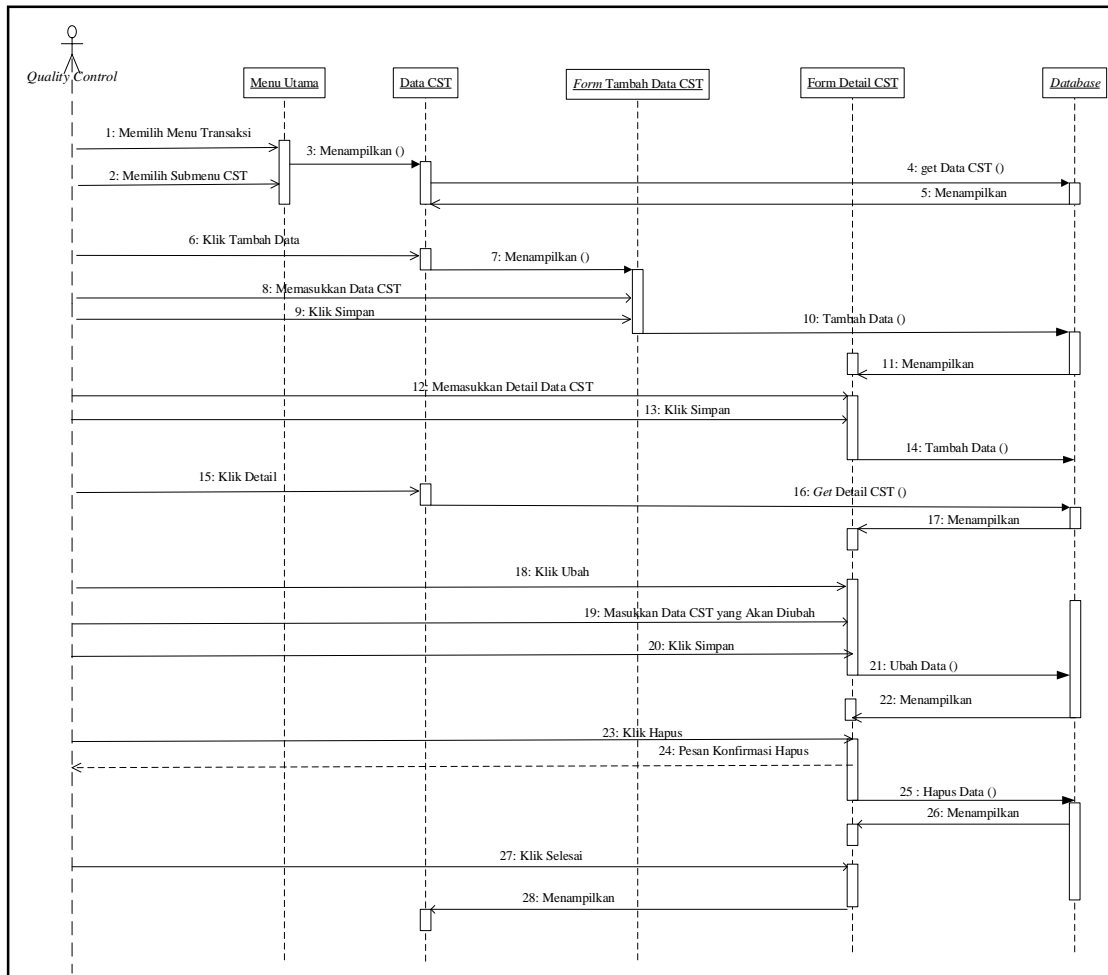
Sequence diagram mengisi data aktual permintaan produksi menjelaskan sebuah *sequence diagram* dalam proses mengisi data aktual permintaan produksi. Adapun *sequence diagram* proses mengisi data permintaan produksi digambarkan pada Gambar V.17.



Gambar V.17 *Sequence Diagram* Mengisi Data Aktual Permintaan Produksi
 Sumber: Hasil Analisis (2017)

6. *Sequence Diagram* Proses Mengisi Data *Check Sheet Temporary*

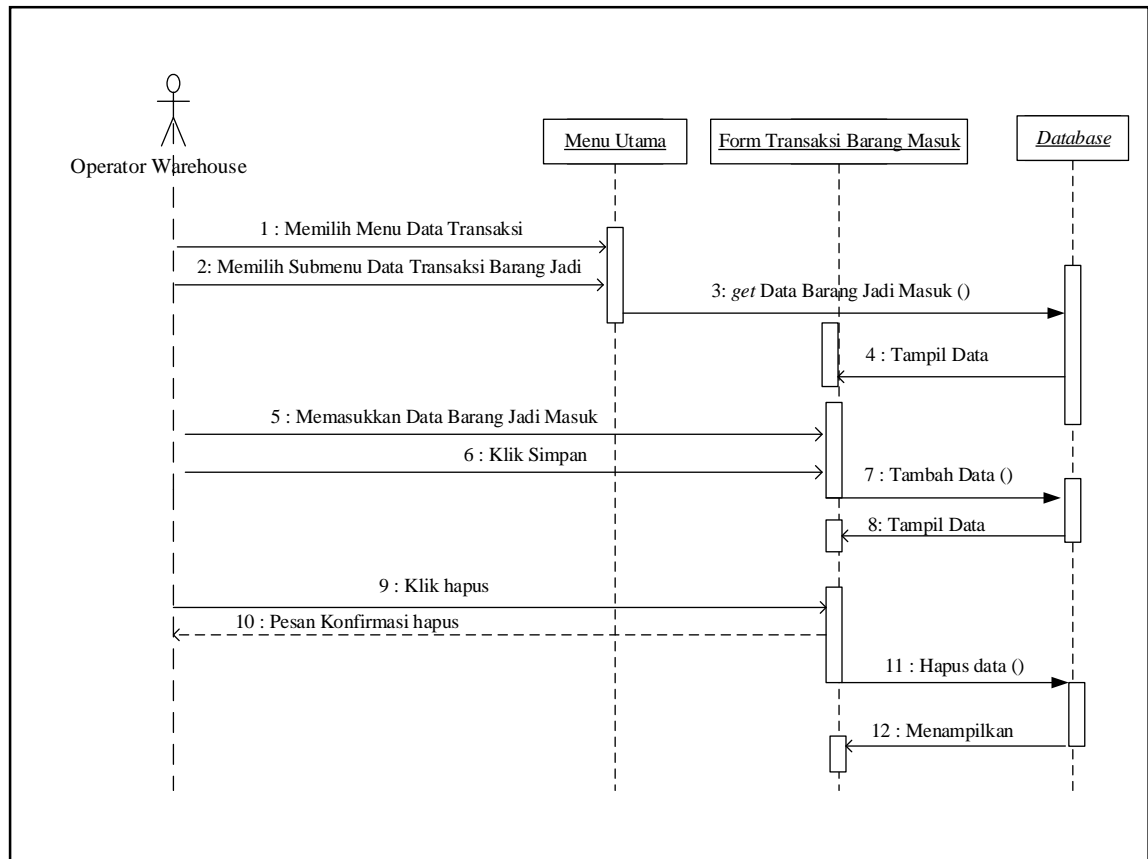
Sequence diagram mengisi data *check sheet temporary* menjelaskan sebuah *sequence diagram* dalam proses mengisi data *check sheet temporary*. Adapun *sequence diagram* proses mengisi data *check sheet temporary* digambarkan pada Gambar V.18.



Gambar V.18 *Sequence Diagram* Mengisi *Check Sheet Temporary*
 Sumber: Hasil Analisis (2017)

7. *Sequence Diagram* Proses Mengisi Data Barang Jadi Masuk

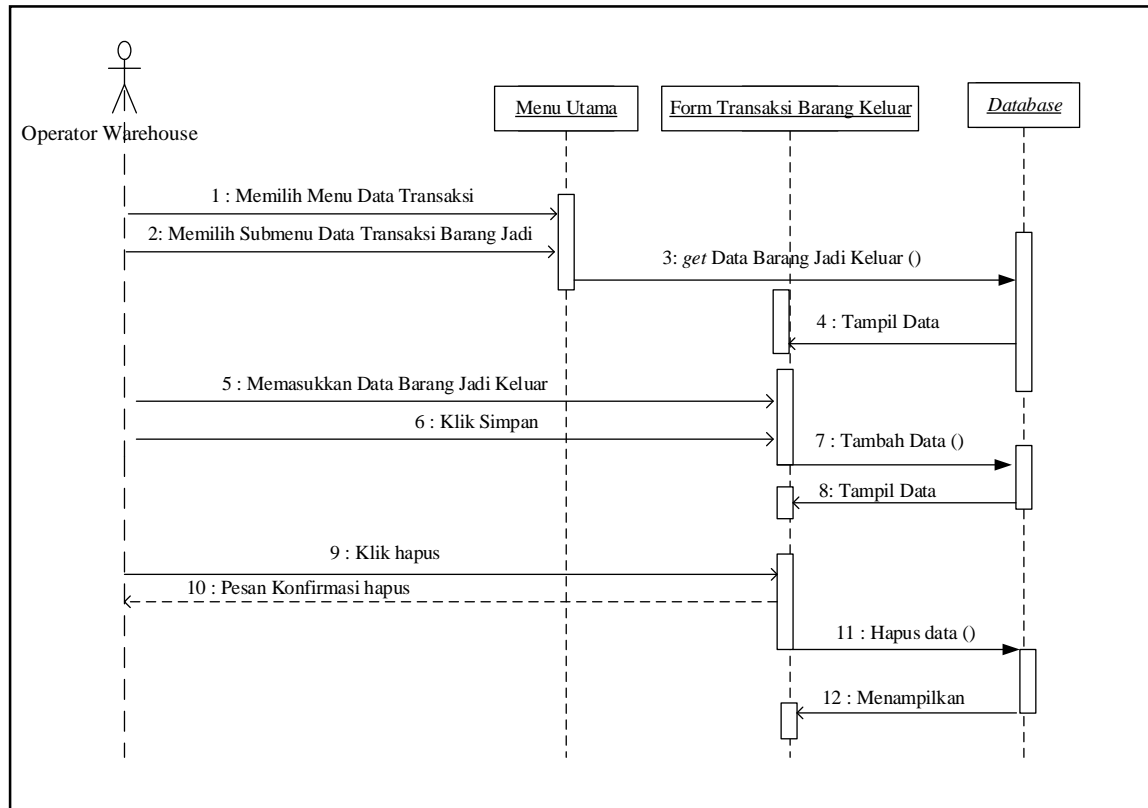
Sequence diagram mengisi data barang jadi masuk menjelaskan sebuah *sequence diagram* dalam proses mengisi data barang jadi yang masuk ke gudang. Adapun *sequence diagram* proses mengisi data barang jadi masuk digambarkan pada Gambar V.19.



Gambar V.19 *Sequence Diagram* Mengisi Data Barang Jadi Masuk
 Sumber: Hasil Analisis (2017)

8. *Sequence Diagram* Proses Mengisi Data Barang Jadi Keluar

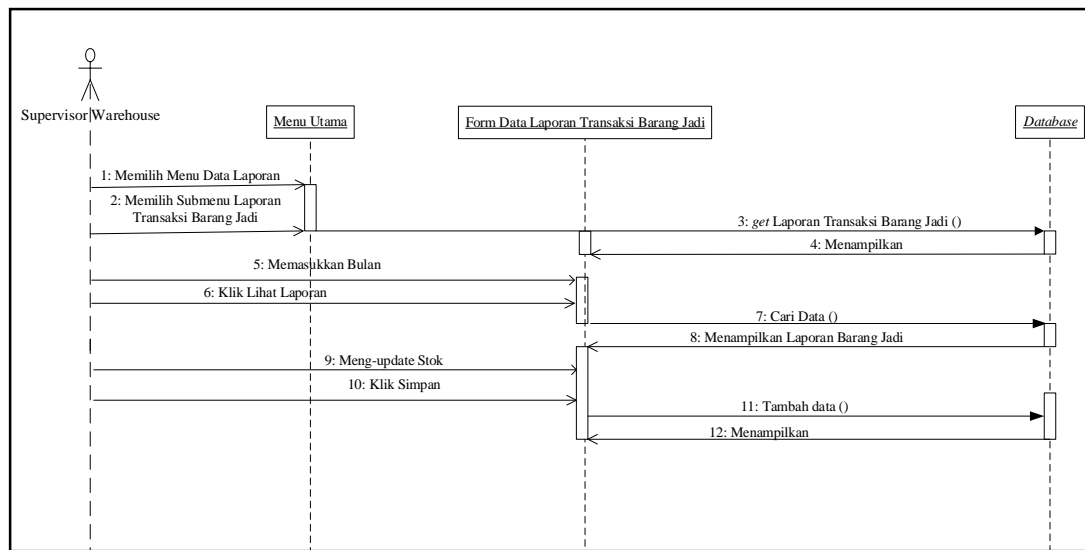
Sequence diagram mengisi data barang jadi keluar menjelaskan sebuah *sequence diagram* dalam proses mengisi data barang jadi yang keluar dari gudang. Adapun *sequence diagram* proses mengisi data barang jadi keluar digambarkan pada Gambar V.20.



Gambar V.20 *Sequence Diagram* Mengisi Data Barang Jadi Keluar
 Sumber: Hasil Analisis (2017)

9. *Sequence Diagram* Proses Meng-Update Data Stok Barang Jadi

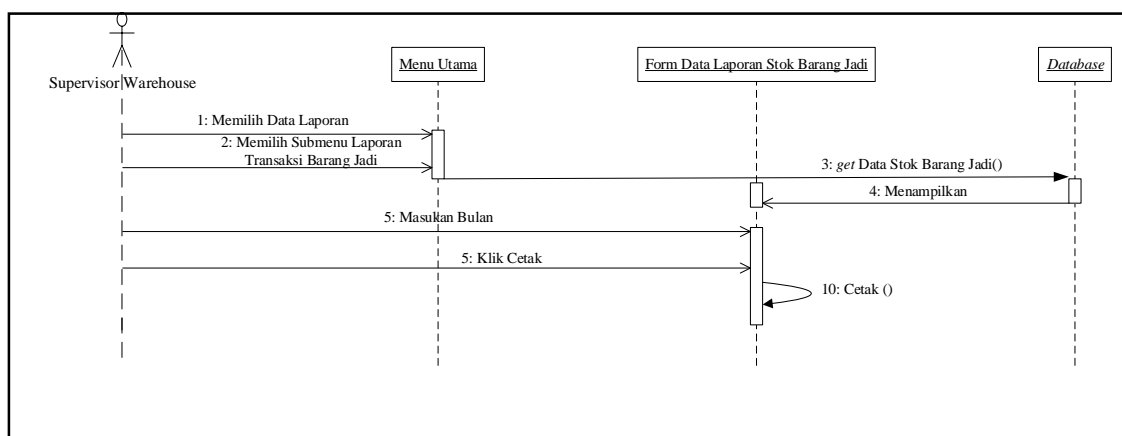
Sequence diagram meng-update data stok barang jadi menjelaskan sebuah *sequence diagram* dalam proses meng-update data stok barang jadi. Adapun *sequence diagram* proses meng-update data stok barang jadi digambarkan pada Gambar V.21.



Gambar V.21 *Sequence Diagram* Meng-update Data Stok Barang Jadi
 Sumber: Hasil Analisis (2017)

10. *Sequence Diagram* Proses Mencetak Laporan Data Stok Barang Jadi

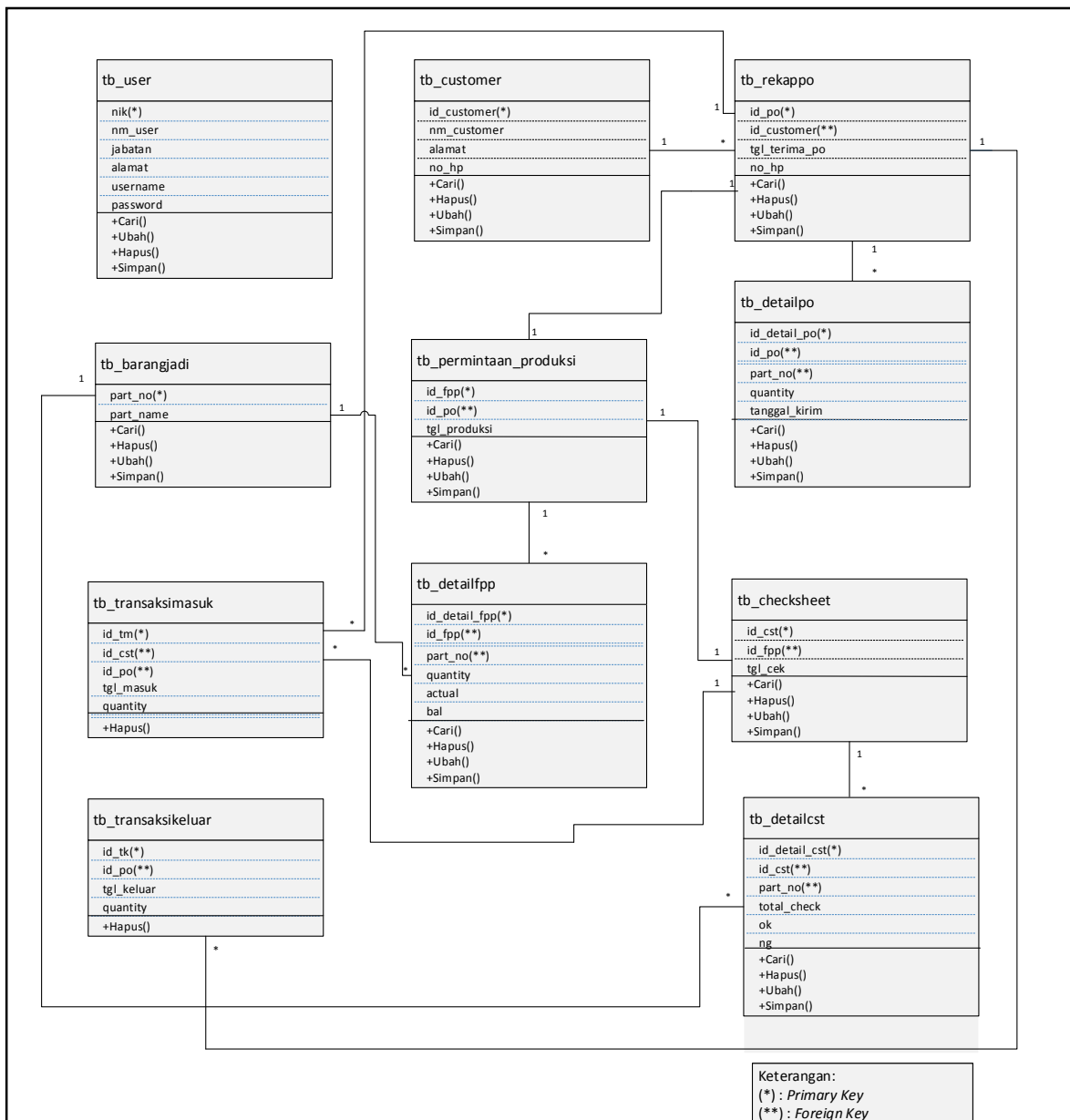
Sequence diagram mencetak laporan data stok barang jadi menjelaskan sebuah *sequence diagram* dalam proses mencetak laporan data stok barang jadi. Adapun *sequence diagram* proses mencetak laporan data stok barang jadi digambarkan pada Gambar V.22.



Gambar V.22 *Sequence Diagram* Mencetak Laporan Data Stok Barang Jadi
 Sumber: Hasil Analisis (2017)

5.5.4 Class Diagram

Class diagram digunakan untuk memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi penyimpanan barang jadi usulan dapat dilihat pada Gambar V.23 berikut ini:



Gambar V.23 *Class Diagram* Usulan
 Sumber: Hasil Analisis (2017)

5.5.5 Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Berikut adalah kamus data sistem informasi penyimpanan barang jadi *warehouse finished goods* usulan:

1. Spesifikasi Tabel *User*

Nama Tabel : *tb_user*
 Fungsi : Untuk menyimpan data pengguna
 Tipe : Data Master

Tabel V.14 Tabel *User*

Nama Field	Akronim	Tipe	Panjang	Keterangan
NIK	nik	<i>Char</i>	5	<i>Primary Key</i>
Nama User	nm_user	<i>Varchar</i>	30	
Jabatan	jabatan	<i>Varchar</i>	20	
Email	email	<i>Varchar</i>	30	
Username	username	<i>Varchar</i>	10	
Password	password	<i>Varchar</i>	8	

Sumber: Hasil Analisis (2017)

2. Spesifikasi Tabel *Customer*

Nama Tabel : *tb_customer*
 Fungsi : Untuk menyimpan data *customer*
 Tipe : Data Master

Tabel V.15 Tabel *Customer*

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Customer	id_customer	<i>Char</i>	5	<i>Primary Key</i>
Nama Customer	nm_customer	<i>Varchar</i>	30	
Alamat	alamat	<i>Varchar</i>	50	
No Hp	no_hp	<i>Varchar</i>	13	

Sumber: Hasil Analisis (2017)

3. Spesifikasi Tabel Barang Jadi

Nama Tabel : tb_barangjadi

Fungsi : Untuk menyimpan data barang jadi

Tipe : Data Master

Tabel V.16 Tabel Barang Jadi

Nama Field	Akronim	Tipe	Panjang	Keterangan
Part Nomor	part_no	<i>Char</i>	15	<i>Primary Key</i>
Part Name	part_name	<i>Varchar</i>	30	
Stock	stock	<i>Integer</i>	5	

Sumber: Hasil Analisis (2017)

4. Spesifikasi Tabel Rekap PO

Nama Tabel : tb_rekappo

Fungsi : Untuk menyimpan data rekap PO

Tipe : Data Transaksi

Tabel V.17 Tabel Rekap PO

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id PO	id_po	<i>Char</i>	10	<i>Primary Key</i>
Id Customer	id_customer	<i>Char</i>	5	<i>Foreign Key</i>
Tanggal terima PO	tgl_terima_po	<i>Date</i>		

Sumber: Hasil Analisis (2017)

5. Spesifikasi Tabel Rekap PO Detail

Nama Tabel : tb_detailpo

Fungsi : Untuk menyimpan data rekap PO detail

Tipe : Data Transaksi

Tabel V.18 Tabel Rekap PO Detail

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Detail PO	id_detail_po	<i>Char</i>	8	<i>Primary Key</i>
Id PO	id_po	<i>Varchar</i>	10	<i>Foreign Key</i>
Id Customer	id_customer	<i>Char</i>	5	<i>Foreign Key</i>
Part Nomor	part_no	<i>Varchar</i>	15	<i>Foreign Key</i>
Quantity	quantity	<i>Integer</i>	5	
Tanggal Kirim	Tanggal_kirim	<i>Date</i>		

Sumber: Hasil Analisis (2017)

6. Spesifikasi Tabel Permintaan Produksi

Nama Tabel : tb_permintaan_produksi

Fungsi : Untuk menyimpan data permintaan produksi

Tipe : Data Transaksi

Tabel V.19 Tabel Permintaan Produksi

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id FPP	id_fpp	<i>Char</i>	8	<i>Primary Key</i>
Id PO	id_po	<i>Varchar</i>	10	<i>Foreign Key</i>
Tanggal FPP	tgl_fpp	<i>Date</i>		

Sumber: Hasil Analisis (2017)

7. Spesifikasi Tabel Permintaan Produksi Detail

Nama Tabel : tb_detailfpp

Fungsi : Untuk menyimpan data permintaan produksi detail

Tipe : Data Transaksi

Tabel V.20 Tabel Permintaan Produksi Detail

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Detail FPP	id_detail_fpp	<i>Char</i>	10	<i>Primary Key</i>
Id FPP	id_fpp	<i>Char</i>	8	<i>Foreign Key</i>
Id Customer	id_customer	<i>Char</i>	5	<i>Foreign Key</i>
Part Nomor	part_no	<i>Char</i>	15	<i>Foreign Key</i>
Quantity	Quantity	<i>Integer</i>	5	
Actual	Actual	<i>Integer</i>	5	
Bal(+/-)	Bal	<i>Integer</i>	5	

Sumber: Hasil Analisis (2017)

8. Spesifikasi Tabel *Check Sheet Temporary*

Nama Tabel : tb_checksheetsheet

Fungsi : Untuk menyimpan data *check sheet temporary*

Tipe : Data Transaksi

Tabel V.21 Tabel *Check Sheet Temporary*

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id CST	id_cst	<i>Char</i>	8	<i>Primary Key</i>
Id FPP	id_fpp	<i>Char</i>	8	<i>Foreign Key</i>
Id PO	id_po	<i>Char</i>	10	<i>Foreign Key</i>
Tanggal Cek	tgl_cek	<i>Date</i>		

Sumber: Hasil Analisis (2017)

9. Spesifikasi Tabel *Check Sheet Temporary* Detail

Nama Tabel : tb_detailcst

Fungsi : Untuk menyimpan data detail *check sheet temporary*

Tipe : Data Transaksi

Tabel V.22 Tabel *Check Sheet Temporary* detail

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Detail CST	id_detail_cst	<i>Char</i>	10	<i>Primary Key</i>
Id CST	id_cst	<i>Char</i>	8	<i>Foreign Key</i>
Id Customer	id_customer	<i>Char</i>	5	<i>Foreign Key</i>
Part No	part_no	<i>Char</i>	15	<i>Foreign Key</i>
Total Check	total_check	<i>Integer</i>	5	
Part OK	Ok	<i>Integer</i>	5	
Part NG	Ng	<i>Integer</i>	5	
Status	status	<i>Varchar</i>	15	

Sumber: Hasil Analisis (2017)

10. Spesifikasi Tabel Transaksi Masuk

Nama Tabel : tb_transaksimasuk
 Fungsi : Untuk menyimpan data transaksi barang jadi masuk
 Tipe : Data Transaksi

Tabel V.23 Tabel Transaksi Masuk

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Transaksi Masuk	id_tm	<i>Char</i>	6	<i>Primary Key</i>
Id CST	id_cst	<i>Char</i>	8	<i>Foreign Key</i>
Id PO	id_po	<i>Char</i>	10	<i>Foreign Key</i>
Id Customer	id_customer	<i>Char</i>	5	<i>Foreign Key</i>
Part No	part_no	<i>Char</i>	15	<i>Foreign Key</i>
Tanggal Masuk	tgl_masuk	<i>Date</i>		
Quantity	Quantity	<i>Integer</i>	3	

Sumber: Hasil Analisis (2017)

11. Spesifikasi Tabel Transaksi Keluar

Nama Tabel : tb_transaksikeluar
 Fungsi : Untuk menyimpan data transaksi barang jadi keluar
 Tipe : Data Transaksi

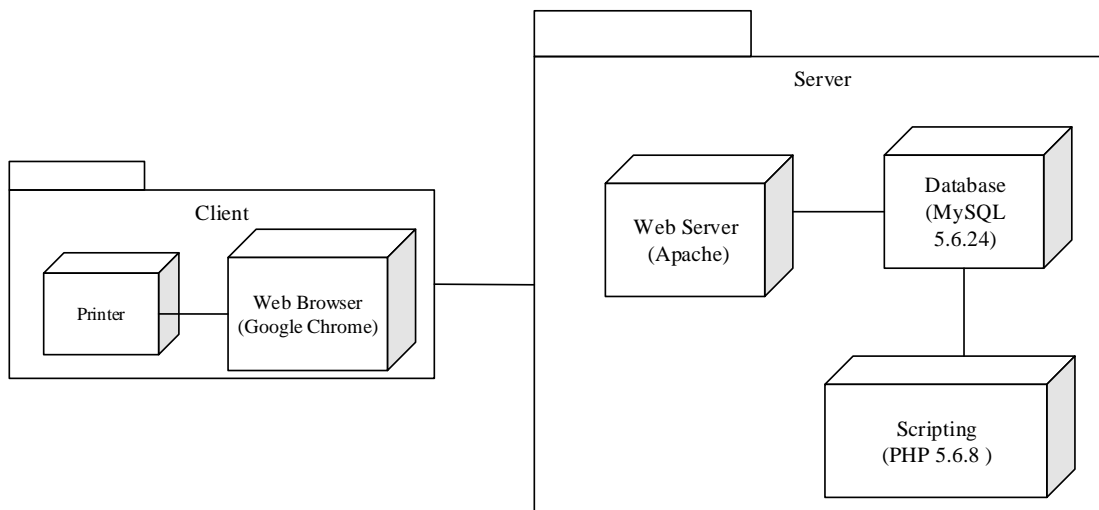
Tabel V.24 Tabel Transaksi Keluar

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Transaksi Keluar	id_tk	<i>Char</i>	6	<i>Primary Key</i>
Id PO	id_po	<i>Char</i>	10	<i>Foreign Key</i>
Id Customer	id_customer	<i>Char</i>	5	<i>Foreign Key</i>
Part No	part_no	<i>Char</i>	15	<i>Foreign Key</i>
Tanggal Keluar	tgl_keluar	<i>Date</i>		
Quantity	Quantity	<i>Integer</i>	3	

Sumber: Hasil Analisis (2017)

5.5.6 Deployment Diagram

Deployment diagram menggambarkan arsitektur fisik dari sistem, seperti *web server* dan semua perangkat lunak tambahan pendukung. *Deployment diagram* untuk sistem informasi penyimpanan barang jadi usulan dapat dilihat pada Gambar V.24 sebagai berikut:



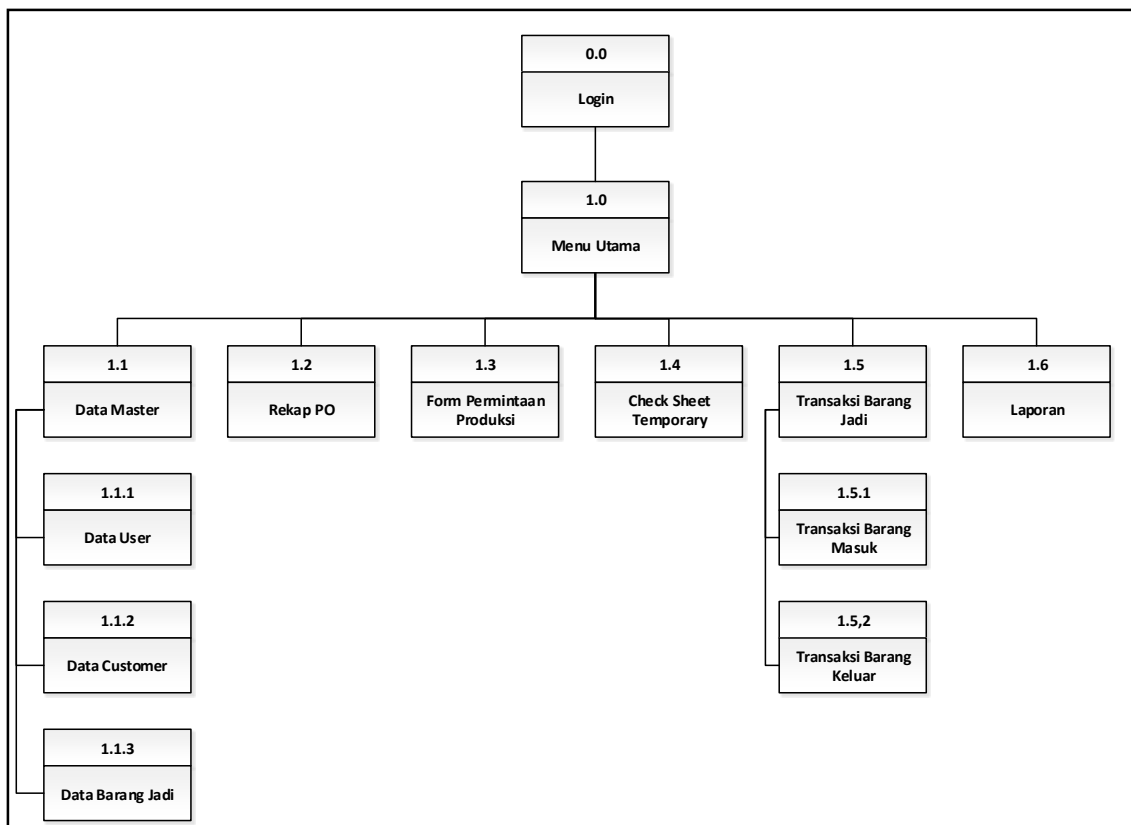
Gambar V.24 *Deployment Diagram* Sistem Informasi Penyimpanan Barang Jadi Usulan
Sumber: Hasil Analisis (2017)

Berikut adalah penjelasan Gambar V.24 *Deployment Diagram* sistem informasi penyimpanan barang jadi:

- *Client* adalah komputer *client* yang harus terinstal sebuah *web browser* contohnya *Google Chrome* untuk menjalankan aplikasi sistem informasi penyimpanan barang jadi dan terhubung dengan *server*. Kemudian untuk media pencetakan menggunakan *printer*.
- *Server* aplikasi sistem informasi penyimpanan barang jadi terdiri dari *web server* Apache, bahasa pemrograman PHP dan perangkat lunak manajemen basis data MySQL.

5.5.7 HIPO (*Hierarchy plus Input-Process-Output*)

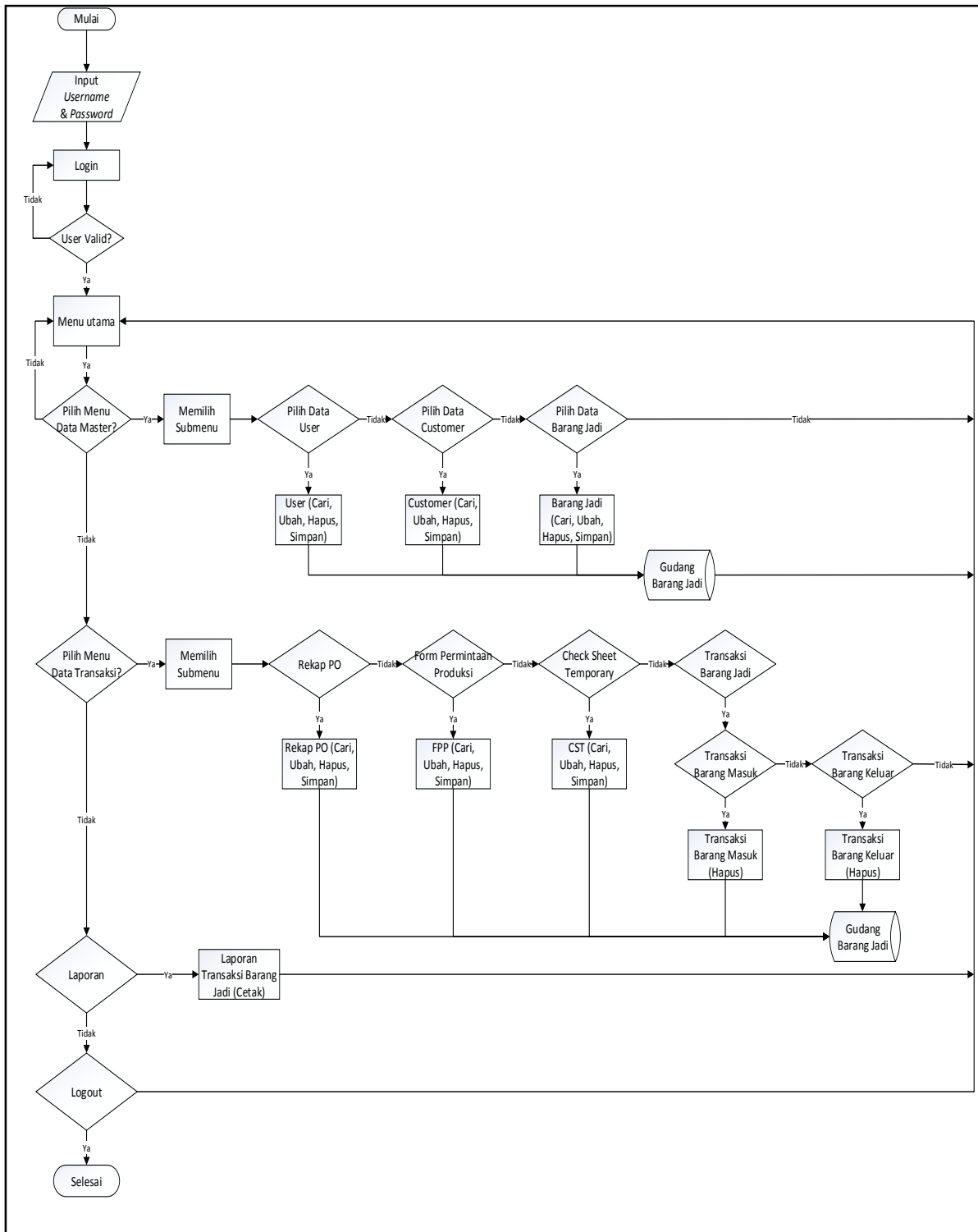
Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hirarkis. Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya. Perancangan HIPO sistem informasi penyimpanan barang jadi usulan digambarkan pada Gambar V.25 berikut ini.



Gambar V.25 HIPO Sistem Informasi Penyimpanan Barang Jadi Usulan
Sumber: Hasil Analisis (2017)

5.5.8 *Flowchart* Aplikasi Usulan

Flowchart yang digunakan untuk mendokumentasikan aplikasi sistem informasi penyimpanan barang jadi ini menggunakan bagan alir logika program (*program logic flowchart*). Bagan alir ini digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika, bukan untuk menggambarkan instruksi-instruksi program komputer secara terinci. *Program logic flowchart* sistem informasi penyimpanan barang jadi dapat dilihat pada Gambar V.26 .



Gambar V.26 *Flowchart* Sistem Informasi Penyimpanan Barang Jadi Usulan
Sumber: Hasil Analisis (2017)

5.5.9 Perancangan *Interface* Sistem Usulan

Interface aplikasi pada sistem informasi penyimpanan barang jadi berupa rancangan tampilan yang akan dibuat. Perancangan *interface* aplikasi usulan adalah sebagai berikut:

1. *Form Login*

Form login adalah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan nama pengguna dan sandi yang benar. Rancangan *form login* dapat dilihat pada Gambar V.27.

PT REKADAYA MULTI ADIPRIMA

Masukan Username dan Password

Username

Password

Login

Gambar V.27 Rancangan *Form Login*
Sumber: Hasil Analisis (2017)

2. *Form Tampilan Menu Utama*

Form menu utama adalah tampilan awal pada aplikasi yang berisi empat menu. Menu tersebut adalah *Home*, *Data Master*, *Transaksi*, dan *Laporan*. Rancangan menu utama dapat dilihat pada Gambar V.28 berikut:

Logo PT RMA PT REKADAYA MULTI ADIPRIMA • User Logout

- Home
- Data Master
- Data Transaksi
- Data Laporan

Selamat Datang di Aplikasi Sistem Penyimpanan Warehouse Finish Good

Footer

Gambar V.28 Tampilan Rancangan *Form* Menu Utama
Sumber: Hasil Analisis (2017)

3. *Form* Tampilan Menu Data *User*

Form menu data *user* adalah *form* yang digunakan untuk menampilkan input data *user* dan data *user* yang telah diinput. Rancangan *interface* dari *form* data *user* dapat dilihat pada Gambar V.29 berikut:

Logo PT RMA PT REKADAYA MULTI ADIPRIMA • User Logout

- Home
- Data Master
 - Data User
 - Data Customer
 - Data Barang Jadi
- Data Transaksi
- Data Laporan

Input Data User

NIK

Nama User

Jabatan

Email

Username

Password

Data User

Search

NIK	Nama User	Jabatan	Email	Username	Password	Aksi
						Ubah Hapus

Footer

Gambar V.29 Rancangan *Form* Menu Data *User*
Sumber: Hasil Analisis (2017)

4. *Form* Tampilan Menu Data Rekap PO

Form menu data rekap PO adalah *form* yang digunakan untuk menampilkan input data rekap PO dan data rekap PO yang telah diinput. Rancangan *interface* dari *form* data rekap PO dapat dilihat pada Gambar V.30 berikut:

The screenshot shows the web application interface for PT REKADAYA MULTI ADIPRIMA. The header includes the company logo and name, a user profile, and a logout button. The left sidebar contains a navigation menu with categories: Home, Data Master, Data Transaksi (highlighted), and Data Laporan. Under Data Transaksi, there are sub-items: - Rekap PO, - Form Permintaan Produksi, - Check Sheet Temporary, and - Data Transaksi Barang Jadi. The main content area features a 'Tambah(+)' button and an 'Input Data Rekap PO' form with fields for Id PO, Nama Customer, and Tanggal Terima PO, along with a 'Simpan' button. Below this is a 'Data Rekap PO' section with a search bar and a table. The table has columns for Id PO, Nama Customer, Tanggal Terima PO, and Aksi. The Aksi column contains links for 'Detail' and 'Hapus'. A footer is located at the bottom of the page.

Gambar V.30 Rancangan *Form* Menu Data Rekap PO
Sumber: Hasil Analisis (2017)

5. *Form* Tampilan Menu Data Rekap PO Detail

Form menu data rekap PO detail adalah *form* yang digunakan untuk menampilkan input data rekap PO detail dan data rekap PO detail yang telah diinput. Rancangan *interface* dari *form* data rekap PO detail dapat dilihat pada Gambar V.31 berikut:

The screenshot displays the web application interface for PT REKADAYA MULTI ADIPRIMA. The header includes the company logo and name, a user profile section with 'User' and 'Logout' options, and a navigation menu on the left with categories like Home, Data Master, Data Transaksi, and Data Laporan. The main content area is divided into two sections: 'Input Detail PO' and 'Data Rekap PO Detail'.

Input Detail PO Form:

- Id PO:
- Id Detail PO:
- Nama Customer:
- Part Name: (with a dropdown arrow)
- Part No:
- Quantity:
- Tanggal Kirim Barang:

Buttons:

Data Rekap PO Detail Table:

Search:

Id PO	Nm Customer	Part No	Part Name	Qty	Tgl Kirim Brg	Aksi
						Ubah Hapus

Footer

Gambar V.31 Rancangan *Form* Menu Data Rekap PO Detail
Sumber: Hasil Analisis (2017)

6. *Form* Tampilan Menu Data Permintaan Produksi

Form menu data permintaan produksi adalah *form* yang digunakan untuk menampilkan input data permintaan produksi dan data permintaan produksi yang telah diinput. Rancangan *interface* dari *form* data permintaan produksi dapat dilihat pada Gambar V.32 berikut:

The screenshot displays a web application interface for PT REKADAYA MULTI ADIPRIMA. The header includes the company logo, name, and a user profile with a 'Logout' button. A left sidebar contains a menu with categories: Home, Data Master, Data Transaksi (highlighted), and Data Laporan. Under 'Data Transaksi', there are sub-items: Rekap PO, Form Permintaan Produksi (highlighted), Check Sheet Temporary, and Data Transaksi Barang Jadi. The main content area features a 'Tambah(+)' button and a form titled 'Input Data Permintaan Produksi' with input fields for 'Id FPP', 'Id PO', and 'Tanggal Buat FPP', along with a 'Simpan' button. Below this is a 'Data Permintaan Produksi' section with a search bar and a table. The table has columns for 'Id FPP', 'Id PO', 'Tanggal Buat PO', and 'Aksi'. The 'Aksi' column contains links for 'Detail' and 'Hapus'. A 'Footer' label is positioned at the bottom of the main content area.

Gambar V.32 Rancangan *Form* Menu Data Permintaan Produksi
 Sumber: Hasil Analisis (2017)

7. *Form* Tampilan Menu Data Permintaan Produksi Detail

Form menu data permintaan produksi detail adalah *form* yang digunakan untuk menampilkan input data permintaan produksi detail dan data permintaan produksi detail yang telah diinput. Rancangan *interface* dari *form* data permintaan produksi detail dapat dilihat pada Gambar V.33 berikut:

The screenshot shows a web application interface for PT REKADAYA MULTI ADIPRIMA. The header includes the company logo and name, a user profile section with a 'Logout' button, and a navigation menu on the left. The main content area is divided into two sections: 'Input Detail Permintaan Produksi' and 'Data Permintaan Produksi Detail'.

Input Detail Permintaan Produksi

Id FPP

Id Detail FPP

Nama Customer

Id Detail PO

Part No

Part Name

Quantity

Data Permintaan Produksi Detail

Search

Id FPP	Nm Customer	Part No	Part Name	Qty	Aksi
					Ubah Hapus

Footer

Gambar V.33 Rancangan *Form* Menu Data Permintaan Produksi Detail
 Sumber: Hasil Analisis (2017)

8. *Form* Tampilan Menu Data *Check Sheet Temporary*

Form menu data *check sheet temporary* adalah *form* yang digunakan untuk menampilkan input data *check sheet temporary* dan data *check sheet temporary* yang telah diinput. Rancangan *interface* dari *form* data *check sheet temporary* dapat dilihat pada Gambar V.34 berikut:

The screenshot shows a web application interface for PT REKADAYA MULTI ADIPRIMA. The header includes the company logo and name, a user profile icon, and a 'Logout' button. The left sidebar contains a navigation menu with the following items: Home, Data Master, Data Transaksi (highlighted), - Rekap PO, - Form Permintaan Produksi, - Check Sheet Temporary, - Data Transaksi Barang Jadi, and Data Laporan. The main content area is titled 'Input Data Check Sheet Temporary' and contains a 'Tambah(+)' button and four input fields: Id Checksheet, Id FPP, Id PO, and Tanggal Pengecekan, with a 'Simpan' button below them. Below the input form is a section titled 'Data Check Sheet Temporary' which includes a search input field and a table with the following structure:

Id Checksheet	Id FPP	Tanggal Pengecekan	Aksi
			Detail Hapus

The footer of the page is labeled 'Footer'.

Gambar V.34 Rancangan *Form Menu Data Check Sheet Temporary*
 Sumber: Hasil Analisis (2017)

9. *Form Tampilan Menu Data Check Sheet Temporary Detail*

Form menu data check sheet temporary detail adalah *form* yang digunakan untuk menampilkan input data *check sheet temporary* detail dan data *check sheet temporary* detail yang telah diinput. Rancangan *interface* dari *form data check sheet temporary* detail dapat dilihat pada Gambar V.35 berikut:

Logo PT
RMA

PT REKADAYA MULTI ADIPRIMA

• User
Logout

- Home
- Data Master
- Data Transaksi
 - Rekap PO
 - Form Permintaan Produksi
 - Check Sheet Temporary
 - Data Transaksi Barang Jadi
- Data Laporan

Input Detail Check Sheet Temporary

Id CST

Id Detail CST

Id Detail FPP ▼

Nama Customer

Part No

Part Name

Total Check

Part OK

Data Check Sheet Temporary Detail

Search

Id CST	Nm Customer	Part No	Part Name	Total Check	OK	NG	Aksi
							Ubah Hapus

Footer

Gambar V.35 Rancangan *Form Menu Data Check Sheet Temporary Detail*
Sumber: Hasil Analisis (2017)

10. *Form Tampilan Menu Data Transaksi Masuk*

Form menu data transaksi masuk adalah *form* yang digunakan untuk menampilkan input data transaksi masuk dan data transaksi masuk yang telah *diinput*. Rancangan *interface* dari *form* data transaksi masuk dapat dilihat pada Gambar V.36 berikut:

Logo PT RMA

PT REKADAYA MULTI ADIPRIMA

• User
Logout

- Home
- Data Master
- Data Transaksi
- Rekap PO
- Form Permintaan Produksi
- Check Sheet Temporary
- Data Transaksi Barang Jadi
- Data Laporan

Barang Masuk
Barang Keluar

Id Transaksi Barang Masuk

Id Detail CST

Id PO

Tanggal Masuk

Nama Customer

Part Name

Quantity

Data Transaksi Masuk

Search

Id TM	Id Detail CST	Id PO	Tgl Masuk	Nm Customer	Part Name	Quantity	Aksi
							Hapus

Footer

Gambar V.36 Rancangan *Form* Menu Data Transaksi Masuk
Sumber: Hasil Analisis (2017)

11. *Form* Tampilan Menu Data Transaksi Keluar

Form menu data transaksi keluar adalah *form* yang digunakan untuk menampilkan input data transaksi keluar dan data transaksi keluar yang telah *diinput*. Rancangan *interface* dari *form* data transaksi keluar dapat dilihat pada Gambar V.37 berikut:

Logo PT RMA

PT REKADAYA MULTI ADIPRIMA

• User
 Logout

- Home
- Data Master
- Data Transaksi
- Rekap PO
- Form Permintaan Produksi
- Check Sheet Temporary
- Data Transaksi Barang Jadi
- Data Laporan

Barang Masuk
Barang Keluar

Id Transaksi Barang Keluar
 Id Detail PO
 Id PO
 Tanggal Keluar
 Nama Customer
 Part Name
 Quantity

Data Transaksi Keluar

Search

Id TK	Id Detail PO	Id PO	Tgl Keluar	Nm Customer	Part Name	Quantity	Aksi
							Hapus

Footer

Gambar V.37 Rancangan *Form* Menu Data Transaksi Keluar
 Sumber: Hasil Analisis (2017)

12. *Form* Tampilan Menu Laporan Data Transaksi Barang Jadi

Form menu laporan data transaksi barang jadi adalah *form* yang digunakan untuk menampilkan laporan data transaksi masuk dan keluar barang jadi. Rancangan *interface* dari *form* menu laporan data transaksi barang jadi dapat dilihat pada Gambar V.38 berikut:

The screenshot shows a web application interface for PT REKADAYA MULTI ADIPRIMA. The interface is divided into several sections:

- Header:** Logo PT RMA, PT REKADAYA MULTI ADIPRIMA, and a Logout button.
- Navigation Menu:** Home, Data Master, Data Transaksi, Data Laporan, and - Laporan Transaksi Barang Jadi.
- Main Content Area:**
 - Laporan Barang Jadi:** A section containing input fields for 'Tanggal Awal' (dd/mm/yyyy), 'Tanggal Akhir' (dd/mm/yyyy), and 'Nama Customer'. Below these are 'Lihat Laporan' and 'Cetak' buttons.
 - Table:** A table with 7 columns: Id PO, Part Name, Nama Cust, Byk Masuk, Byk Keluar, Stok Awal, and Stok Akhir. The table is currently empty.
- Footer:** A footer section at the bottom of the page.

Gambar V.38 Rancangan *Form* Laporan Data Transaksi Barang Jadi
Sumber: Hasil Analisis (2017)

5.5.10 Implementasi Sistem

Untuk dapat menggunakan aplikasi, tentunya diperlukan suatu spesifikasi perangkat keras (*hardware*) ataupun perangkat lunak (*software*) yang mendukung agar aplikasi dapat berjalan dengan baik. Adapun spesifikasinya sebagai berikut:

1. *Kebutuhan Hardware*
 - a. *Processor* : Minimal *Processor Pentium IV*.
 - b. *RAM* : Minimal *RAM 512 MB*.
 - c. *Harddisk* : Minimal *Harddisk 64 GB*.
 - d. *Media input* : *Mouse, Keyboard, Monitor*
 - e. *Media output* : *Printer*
2. *Kebutuhan Software*
 - a. *Sistem Operasi* : *Microsoft Windows 7*
 - b. *PDF Reader* : *Adobe Reader*
 - c. *Web Browser* : *Mozilla firefox, Google Chromes*

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil pengumpulan, pengolahan dan analisis data yang telah dilakukan dalam penelitian ini dapat disimpulkan bahwa:

1. Dengan adanya aplikasi sistem penyimpanan barang jadi ini diharapkan memudahkan bagian gudang barang jadi untuk dapat menginput data transaksi barang jadi masuk dan keluar ke dalam sistem, sehingga pembaharuan data lebih cepat dan tepat.
2. Dengan adanya sistem basis data untuk gudang barang jadi yang diharapkan dapat mempermudah proses penyimpanan data dan mencegah terjadinya kehilangan data karena setiap data yang dimasukkan kedalam sistem telah tersimpan dengan aman dengan menggunakan *database* MySQL .

6.2 Saran

Saran yang diberikan untuk pengembangan sistem informasi penyimpanan barang jadi ini selanjutnya adalah sebagai berikut:

1. Dalam penerapan sistem baru terhadap sistem lama sebaiknya dilakukan secara bertahap, untuk menghindari terjadinya kegagalan terhadap sistem yang baru dan diperlukan adanya sosialisasi serta pengenalan untuk penggunaan sistem ini kepada bagian yang menerapkannya.
2. Dikarenakan sistem baru yang dibuat ini memiliki keterbatasan dalam penggunaan, diharapkan *user* dapat menggunakan dengan baik dan diharapkan sistem dapat dikembangkan agar membantu proses bisnis yang lain.


DAFTAR PUSTAKA

- Amsyah, Zulkifli. 2003. *Manajemen Kearsipan*. Gramedia, Jakarta.
- Jogiyanto, 2005. *Analisis dan Desain Sistem Informasi*, Yogyakarta: Penerbit Andi.
- Kadir, Abdul, 2014. *Pengenalan Sistem Informasi*, Edisi Revisi, Yogyakarta: Andi Offset.
- Martono, Ricky. 2015. *Manajemen Logistik Terintegrasi.*, PPM. Jakarta
- McLeod, Raymond., S, George. 2011. *Sistem Informasi Manajemen*. Jakarta: PT. Indeks.
- Purnomo , Hari . 2004 . *Perencanaan dan Perancangan Fasilitas* , Edisi Pertama, Graha Ilmu . Yogyakarta
- Riadi, Muchlisin. 2016. *Pengertian, Tujuan dan Manfaat Gudang*. <http://www.kajianpustaka.com/2016/04/pengertian-tujuan-dan-manfaat-gudang.html> (Tanggal Akses; 9 Februari 2017)
- Rosa, A.S., Shalahuddin, M, 2014, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Siahaya, Williem. 2016. *Sukses Supply Chain Management: Akses Demand Chain Managenent*. In Media . Bogor
- Sitorus, Lamhot. 2015. *Algoritma dan Pemrograman*. Yogyakarta: Andi Offset.
- Sommerville, Ia,. 2003, *Software Engineering Rekayasa Perangkat Lunak*. Jakarta: Erlangga.

- Sutabri, Tata, 2012, *Analisis Sistem Informasi*, ANDI Yogyakarta, Yogyakarta.
- Sutaji, Deni, 2012, *Sistem Inventory Mini Market dengan PHP & JQuery*, Lokomedia, Yogyakarta.
- Trisna, Denny. 2012. *Pergudangan (Warehousing)*.
<http://www.rantaipasok.com/2012/12/pergudangan-warehousing.html>. (Tanggal akses; 9 Februari 2017).
- Warman, John . 2012 . *Manajemen Pergudangan* . Pustaka Sinar Harapan . Jakarta
- Welling, L., Thomson, L, 2003, *PHP and MySql Web Development, Second Edition*, Sams Publishing, Indianapolis.
- Yakub. 2012. *Pengantar Sistem Informasi*, Yogyakarta: Graha Ilmu.
- _____, *Barang Jadi* <http://id.wikipedia.org/wiki/Barangjadi>, 2013. (Tanggal Akses; 8 Februari 2017)

TAMPILAN PROGRAM

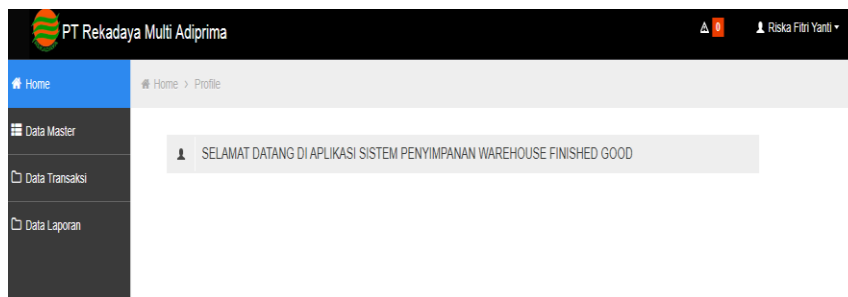
1. Login

PT Rekadaya Multi Adiprima 

Masukkan Username dan Password Anda











Login

2. Menu Utama



3. Data Master User

The screenshot displays the 'Data Master User' page. It features an 'Input Data User' form with fields for NIK, Nama User, Jabatan, Email, Username, and Password, along with 'Simpan' and 'Batal' buttons. Below the form is a table listing existing users.

NIK	Nama User	Jabatan	Email	Username	Password	Aksi
00001	Fahmi A Hueris	Admin PPIC	fahmi@gmail.com	fahmi	12345	 
00002	Sri Winarto	Kepala Produksi	sriwinarto@gmail.com	sriwinarto	12346	 
00003	Harsoyo	Operator Quality Control	harsoyo@gmail.com	harsoyo	12347	 
00004	August Tri Utomo	Operator Warehouse	august@gmail.com	august	12348	 
00005	Herry Boedyanto	Supervisor Warehouse	herry@gmail.com	herry	12349	 

4. Form Rekap PO

PT Rekadaya Multi Adiprima

Home > Profile

Data Rekap PO

Id PO	Nama Customer	Tanggal Terima PO	Aksi
PO-001	PT Bonecom Tricom	21-10-2017	[Green] [Red]

Showing 1 to 1 of 1 entries

Buat Rekap PO

Id PO: PO-002

Nama Customer: -

Tanggal Terima PO: 22-10-2017

Input Detail PO

Id PO: PO-002

Id Detail PO: D-PO-003

Nama Customer: PT Astra Daihatsu Motor

Part Name: Select an Option

Part No:

Quantity:

Tanggal Kirim Barang: dd/mm/yyyy

5. Form Permintaan Produksi

PT Rekadaya Multi Adiprima

Home > Profile

Data Permintaan Produksi

Id FPP	Id PO	Tanggal Buat FPP	Aksi
FPP-001	PO-001	2017-10-21	[Green] [Red]
FPP-002	PO-001	2017-10-23	[Green] [Red]

Showing 1 to 2 of 2 entries

Buat Form Permintaan Produksi

Id FPP: FPP-003

Id PO: -

Tanggal Buat FPP: dd/mm/yyyy

PT Rekadaya Multi Adiprima

Home > Profile

Input Detail FPP

Id FPP: FPP-003

Id Detail FPP: D-FPP-003

Nama Customer: PT Bonecom Tricom

Id Detail PO: Select an Option

Part No:

Part Name:

Quantity:

Simpan | Batal

6. *Form Checksheet Temporary*

PT Rekadaya Multi Adiprima

Home > Profile

Tambah(+)

Data Check Sheet Temporary

10 records per page

Search:

Id CST	Id FPP	Tanggal Pengecekan	Aksi
CST-001	FPP-001	2017-10-21	

Showing 1 to 1 of 1 entries

← Previous | 1 | Next →

Buat Form Check Sheet Temporary

Id CheckSheet: CST-002

ID FPP: Select an Option

ID PO:

Tanggal Pengecekan: dd/mm/yyyy

Tutup | Simpan

PT Rekadaya Multi Adiprima

Home > Profile

Check Sheet Temporary

ID CST: CST-002

Id Detail CST: DCST-003

ID Detail FPP: Select an Option

Nama Customer:

Part No:

Part Name:

Total Check:

Part OK:

Simpan | Batal

7. *Form Transaksi Barang Jadi Masuk*

The screenshot shows the 'Form Transaksi Barang Jadi Masuk' (Finished Goods Transaction Entry Form) in the PT Rekadaya Multi Adiprima system. The interface includes a sidebar menu with options like Home, Data Master, Data Transaksi, and Data Laporan. The main content area is titled 'Barang Masuk | Barang Keluar' and contains the following fields:

- ID Transaksi Barang Masuk:** TM-001
- ID Detail CST:** Select an Option (dropdown menu)
- ID PO:** (text input)
- Tanggal Masuk:** (date input)
- Nama Customer:** (text input)
- Part Name:** (text input)
- Quantity:** (text input)

At the bottom of the form, there are two buttons: 'Simpan' (Save) and 'Batal' (Cancel).

8. *Form Transaksi Barang Jadi Keluar*

The screenshot shows the 'Form Transaksi Barang Jadi Keluar' (Finished Goods Transaction Exit Form) in the PT Rekadaya Multi Adiprima system. The interface is similar to the previous form, with a sidebar menu and a main content area titled 'Barang Masuk | Barang Keluar'. The fields are:

- ID Transaksi Barang Keluar:** TK-001
- ID Detail PO:** Select an Option (dropdown menu)
- ID PO:** (text input)
- Tanggal Keluar:** (date input)
- Nama Customer:** (text input)
- Part Name:** (text input)
- Quantity:** (text input)

At the bottom of the form, there are two buttons: 'Simpan' (Save) and 'Batal' (Cancel).

9. *Form Laporan Barang Jadi*

The screenshot shows the 'Form Laporan Barang Jadi' (Finished Goods Report Form) in the PT Rekadaya Multi Adiprima system. The interface includes a sidebar menu with options like Home, Data Master, Data Transaksi, and Data Laporan. The main content area is titled 'Laporan Barang Jadi' and contains the following fields:

- Tanggal Awal:** dd/mm/yyyy (date input)
- Tanggal Akhir:** dd/mm/yyyy (date input)
- Nama Customer:** (dropdown menu)

At the bottom of the form, there are two buttons: 'Lihat Laporan' (View Report) and 'Cetak' (Print).

KODE PROGRAM

1. Login

```

<!DOCTYPE html>
<html lang="en">
<link rel="icon" type="image/png" href="image/logo_rekadaya.png">
  <head>
    <?php
    session_start();
    unset($_SESSION['name']);
    unset($_SESSION['jabatan']);

    error_reporting(0);
    ?>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <title>PT Rekadaya Multi Adiprima</title>

    <!-- CSS -->

    <link rel="stylesheet"
href="css_login/assets/bootstrap/css/bootstrap.min.css">
    <link rel="stylesheet" href="css_login/assets/font-awesome/css/font-
awesome.min.css">
    <link rel="stylesheet" href="css_login/assets/css/form-elements.css">
    <link rel="stylesheet" href="css_login/assets/css/style.css">

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements
and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via
file:// -->
    <!--[if lt IE 9]>
    <script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
    <script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></scrip
t>
    <![endif]-->

    <!-- Favicon and touch icons -->
    <link rel="apple-touch-icon-precomposed" sizes="144x144"
href="css_login/assets/ico/apple-touch-icon-144-precomposed.png">

```

```

    <link rel="apple-touch-icon-precomposed" sizes="114x114"
href="css_login/assets/ico/apple-touch-icon-114-precomposed.png">
    <link rel="apple-touch-icon-precomposed" sizes="72x720"
href="css_login/assets/ico/apple-touch-icon-72-precomposed.png">
    <link rel="apple-touch-icon-precomposed"
href="css_login/assets/ico/apple-touch-icon-57-precomposed.png">

</head>

<body>

    <!-- Top content -->

        <div class="row" align="center">
            <div class="col-sm-4 col-sm-offset-4 form-box">
                <div class="form-top">
                    <div class="form-top-left">
                        <h3><p>PT Rekadaya Multi Adiprima</p></h3>
                        <h4><p>Masukkan Username dan Password
Anda</p></h4>
                    </div>
                    <div class="form-top-right">
                        <i></i>
                    </div>
                </div>
                <div class="form-bottom">
                    <form role="form" method="post"
action="p_login.php">
                        <div class="form-group">
                            <label class="sr-only" for="form-
username"></label>
                            <input type="text" name="txt_username"
placeholder="Username" class="form-control">
                        </div>
                        <div class="form-group">
                            <input type="password" name="txt_password"
placeholder="Password" class="form-control">
                        </div>
                    <div align="middle">

                </div>
            <div align="middle">
                <?php

```

```

$pesan = $_POST["pesan"] ;
if ( $pesan != NULL) {
    echo "<font color='#ff3b3b'>" . $pesan . "</font>" ;
}
?>
</div>

```

```

        <button type="submit" class="btn">Login</button>
    </form>
</div>
</div>
</div>

```

```

<!-- Javascript -->
<script src="css_login/assets/js/jquery-1.11.1.min.js"></script>
<script src="css_login/assets/bootstrap/js/bootstrap.min.js"></script>
<script src="css_login/assets/js/jquery.backstretch.min.js"></script>
<script src="css_login/assets/js/scripts.js"></script>

```

```

<!--[if lt IE 10]>
    <script src="css_login/assets/js/placeholder.js"></script>
<![endif]-->

```

```

</body>

```

```

</html>

```

2. Home

```

<!DOCTYPE html>
<html lang="en">
<link rel="icon" type="image/png" href="image/logo_rekadaya.png">
<?php
include("koneksi.php");
session_start();
if ($_SESSION["jabatan"] == NULL) {
    header("location: login.php?pesan= Silahkan login terlebih dahulu");
}
?>
<head>

    <!-- start: Meta -->
    <meta charset="utf-8">
    <title>PT Rekadaya Multi Adiprima</title>

```

```

        <!-- start: CSS -->
        <link id="bootstrap-style" href="css/bootstrap.min.css"
rel="stylesheet">
        <link href="css/bootstrap-responsive.min.css" rel="stylesheet">
        <link id="base-style" href="css/style.css" rel="stylesheet">
        <link id="base-style-responsive" href="css/style-responsive.css"
rel="stylesheet">
        <link
href='http://fonts.googleapis.com/css?family=Open+Sans:300italic,400ital
ic,600italic,700italic,800italic,400,300,600,700,800&subset=latin,cyrillic-
ext,latin-ext' rel='stylesheet' type='text/css'>
        <!-- end: CSS -->

        <!-- The HTML5 shim, for IE6-8 support of HTML5 elements -->
        <!--[if lt IE 9]>
            <script
src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
            <link id="ie-style" href="css/ie.css" rel="stylesheet">
        <![endif]-->

        <!--[if IE 9]>
            <link id="ie9style" href="css/ie9.css" rel="stylesheet">
        <![endif]-->

        <!-- start: Favicon -->

        <!-- end: Favicon -->

</head>

<body>
    <!-- start: Header -->
    <div class="navbar">
        <div class="navbar-inner">
            <div class="container-fluid">
                <a class="btn btn-navbar" data-
toggle="collapse" data-target=".top-nav.nav-collapse,.sidebar-nav.nav-
collapse">
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>

```

```

        </a>
        <a class="brand"><span>PT Rekadaya Multi
Adiprima</span></a>

```

```

        <!-- start: Header Menu -->
        <div class="nav-no-collapse header-nav">
            <ul class="nav pull-right">
                <li class="dropdown hidden-
phone">
                    <a class="btn
dropdown-toggle" data-toggle="dropdown" href="#">
                        <i
class="halflings-icon white warning-sign"></i>

```

```
<?php
```

```

        $countid_detail_po="select count(id_detail_po) as banyak_notif
from tb_detailpo where tanggal_kirim=curdate() and status='belum
dikirim' "; $result= mysql_query($countid_detail_po) or
die("ERROR".mysql_error());
if(mysql_num_rows($result) != 0) {
    $row = mysql_fetch_array($result);
    $banyak_notif_id_detail_po = $row['banyak_notif'];
}
else{
    $banyak_notif_id_detail_po='0';
}

```

```

        $countid_detail_cst="select *, tb_detailcst.id_detail_cst,
tb_detailcst.id_detail_cst,(select count(id_cst) from tb_checksheet where
tgl_cek=curdate()) as banyak_notif FROM tb_checksheet, tb_detailcst
WHERE tb_checksheet.id_cst = tb_detailcst.id_cst and
tb_detailcst.status='belum disimpan' and
tb_checksheet.tgl_cek=curdate()"; $result=
mysql_query($countid_detail_cst) or die("ERROR".mysql_error());
if(mysql_num_rows($result) != 0) {
    $row = mysql_fetch_array($result);
    $banyak_notif_id_cst = $row['banyak_notif'];
}
else{
    $banyak_notif_id_cst = '0';
}

```

```

$banyak_notif= $banyak_notif_id_detail_po + $banyak_notif_id_cst;
    ?>

    <span
class="label label-important"><?php echo $banyak_notif; ?></span>
    </a>
    <ul class="dropdown-
menu notifications">
        <li
class="dropdown-menu-title">

            <span>Ada <?php echo $banyak_notif; ?> Notifikasi Baru</span>
            <a
href="#refresh"><i class="icon-repeat"></i></a>
            </li>
            <?php

                $selectpo=mysql_query("select * from tb_detailpo where
tanggal_kirim= curdate() and status='belum dikirim");

                while($su=mysql_fetch_array($selectpo)){

                    ?>

                    <li>
                    <a href="#">
                        <span
class="message">Barang <?php echo $su['id_detail_po']; ?> harus dikirim
hari ini</span>

                    <?php } ?>

                    </a>
                    </li>

                <?php

                $selectcst=mysql_query("select *, tb_detailcst.id_detail_cst,
tb_detailcst.id_detail_cst,(select count(id_cst) from tb_checksheets where

```

```

tgl_cek=curdate()) as banyak_notif FROM tb_checksheets, tb_detailcst
WHERE tb_checksheets.id_cst = tb_detailcst.id_cst and
tb_detailcst.status='belum disimpan' and
tb_checksheets.tgl_cek=curdate());

```

```

while($u=mysql_fetch_array($selectcst)){

```

```

    ?>

```

```

        <li>

```

```

        <a href="#">

```

```

            <span

```

```

            class="message">Barang <?php echo $u['id_detail_cst']; ?> Sudah Selesai
            Dicek</span>

```

```

        <?php } ?>

```

```

        </a>

```

```

    </li>

```

```

    </ul>

```

```

    </li>

```

```

    <!-- Start account setting -->

```

```

        <li class="dropdown">

```

```

            <a class="btn

```

```

            dropdown-toggle" data-toggle="dropdown" href="#">

```

```

            <i

```

```

            class="halflings-icon white user"></i> <?php echo

```

```

            "".$_SESSION["name"]."" ?>

```

```

            <span

```

```

            class="caret"></span>

```

```

        </a>

```

```

        <ul class="dropdown-

```

```

        menu">

```

```

            <li

```

```

            class="dropdown-menu-title">

```

```

            <span>Account Settings</span>

```

```

        </li>

```

```

        <li><a

```

```

        href="#"><i class="halflings-icon user"></i> Profile</a></li>

```

```

        <li><a

```

```

        href="login.php"><i class="halflings-icon off"></i> Logout</a></li>

```

```

        </ul>
        </li>
        <!-- end: User Dropdown -->
    </ul>
    <div class="row-fluid sortable">

        </div>
    </div>
    <!-- end: Header Menu -->

</div>
</div>
</div>
<!-- start: Header -->

    <div class="container-fluid-full">
    <div class="row-fluid">

        <!-- start: Main Menu -->
        <div id="sidebar-left" class="span2">
            <div class="nav-collapse sidebar-nav">
                <ul class="nav nav-tabs nav-stacked
main-menu">
                    <li><a href="home.php"><i
class="icon-home"></i><span class="hidden-tablet">
Home</span></a></li>

                    <?php

$stampil_menu=$_SESSION['jabatan'] == 'Admin PPIC';
                    if($stampil_menu){
                        echo
                        ,
                        <li>
                            <a class="dropdown"
href="#"><i class="icon-folder-close-alt"></i><span class="hidden-
tablet"> Data Transaksi</span></a>

                            <ul>
                                <li><a
class="submenu" href="rekappo.php"><i class="icon-file-alt"></i><span
class="hidden-tablet"> Rekap PO</span></a></li>

                                <li><a
class="submenu" href="datapermintaanproduksi.php"><i class="icon-file-
alt"></i><span class="hidden-tablet"> Form Permintaan
Produksi</span></a></li>

```



```

<li><a
class="submenu" href="transaksimasuk.php"><i class="icon-file-
alt"></i><span class="hidden-tablet"> Data Transaksi Barang
Jadi</span></a></li>

</ul>
</li>
<li>
<a class="dropmenu"
href="#"><i class="icon-folder-close-alt"></i><span class="hidden-
tablet"> Data Laporan</span></a>

<ul>
<li><a
class="submenu" href="formlaporanmasuk.php"><i class="icon-file-
alt"></i><span class="hidden-tablet"> Laporan Transaksi Barang Jadi
Masuk</span></a></li>
<li><a
class="submenu" href="formlaporankeluar.php"><i class="icon-file-
alt"></i><span class="hidden-tablet"> Laporan Transaksi Barang Jadi
Keluar</span></a></li>

</li>
</ul>

';}?'>

</div>
</div>
<!-- end: Main Menu -->

<noscript>
<div class="alert alert-block span10">
<h4 class="alert-
heading">Warning!</h4>
<p>You need to have <a
href="http://en.wikipedia.org/wiki/JavaScript"
target="_blank">JavaScript</a> enabled to use this site.</p>
</div>
</noscript>

<!-- start: Content -->
<div id="content" class="span10">

<ul class="breadcrumb">
<li>

```

```

        <i class="icon-home"></i>
        <a href="home.php">Home</a>
        <i class="icon-angle-right"></i>
    </li>
    <li><a href="#">Profile</a></li>
</ul>

<div class="row-fluid">

    <div class="clearfix"></div>

</div><!--/row-->

<div class="box span10" align="middle">
    <div class="box-header" data-
original-title>
        <h2><i class="halflings-icon
user"></i><span class="break"></span>SELAMAT DATANG DI
APLIKASI SISTEM PENYIMPANAN WAREHOUSE FINISHED
GOOD</h3>

        <div class="box-icon">
        </div>
        </h2>
        </div>
        </div>

</div><!--/.fluid-container-->

    <!-- end: Content -->
</div><!--/#content.span10-->
</div><!--/fluid-row-->

<div class="modal hide fade" id="myModal">
    <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">×</button>
        <h3>Settings</h3>
    </div>
    <div class="modal-body">
        <p>Here settings can be configured...</p>
    </div>
    <div class="modal-footer">

```

```

        <a href="#" class="btn" data-
dismiss="modal">Close</a>
        <a href="#" class="btn btn-primary">Save
changes</a>
    </div>
</div>

<div class="clearfix"></div>

<footer>

    <p>
        <span style="text-align:left;float:left">&copy; 2017
<a href="http://jiji262.github.io/Bootstrap_Metro_Dashboard/"
alt="Bootstrap_Metro_Dashboard">Riska Fitri Yanti</a></span>

    </p>

</footer>

<!-- start: JavaScript-->

    <script src="js/jquery-1.9.1.min.js"></script>
<script src="js/jquery-migrate-1.0.0.min.js"></script>

    <script src="js/jquery-ui-1.10.0.custom.min.js"></script>

    <script src="js/jquery.ui.touch-punch.js"></script>

    <script src="js/modernizr.js"></script>

    <script src="js/bootstrap.min.js"></script>

    <script src="js/jquery.cookie.js"></script>

    <script src="js/fullcalendar.min.js"></script>

    <script src="js/jquery.dataTables.min.js"></script>

    <script src="js/excanvas.js"></script>
<script src="js/jquery.flot.js"></script>
<script src="js/jquery.flot.pie.js"></script>
<script src="js/jquery.flot.stack.js"></script>
<script src="js/jquery.flot.resize.min.js"></script>

    <script src="js/jquery.chosen.min.js"></script>

```

```

<script src="js/jquery.uniform.min.js"></script>
<script src="js/jquery.cleditor.min.js"></script>
<script src="js/jquery.noty.js"></script>
<script src="js/jquery.elfinder.min.js"></script>
<script src="js/jquery.raty.min.js"></script>
<script src="js/jquery.iphone.toggle.js"></script>
<script src="js/jquery.uploadify-3.1.min.js"></script>
<script src="js/jquery.gritter.min.js"></script>
<script src="js/jquery.imagesloaded.js"></script>
<script src="js/jquery.masonry.min.js"></script>
<script src="js/jquery.knob.modified.js"></script>
<script src="js/jquery.sparkline.min.js"></script>
<script src="js/counter.js"></script>
<script src="js/retina.js"></script>
<script src="js/custom.js"></script>
<!-- end: JavaScript-->

</body>
</html>

```

3. *Data User*

```

<!DOCTYPE html>
<html lang="en">
<link rel="icon" type="image/png" href="image/logo_rekadaya.png">
<?php
include("../koneksi.php");
session_start();
if ($_SESSION["jabatan"] == NULL) {
    header("location: page/login.php?pesan=LOGIN TERLEBIH
DAHULU!");
}

```

```

?>
<head>

    <!-- start: Meta -->
    <meta charset="utf-8">
    <title>PT Rekadaya Multi Adiprima</title>

    <!-- start: CSS -->
    <link id="bootstrap-style" href="../css/bootstrap.min.css"
rel="stylesheet">
    <link href="../css/bootstrap-responsive.min.css" rel="stylesheet">
    <link id="base-style" href="../css/style.css" rel="stylesheet">
    <link id="base-style-responsive" href="../css/style-responsive.css"
rel="stylesheet">
    <link
href='http://fonts.googleapis.com/css?family=Open+Sans:300italic,400ital
ic,600italic,700italic,800italic,400,300,600,700,800&subset=latin,cyrillic-
ext,latin-ext' rel='stylesheet' type='text/css'>
    <!-- end: CSS -->

    <!-- The HTML5 shim, for IE6-8 support of HTML5 elements -->
    <!--[if lt IE 9]>
        <script
src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
        <link id="ie-style" href="css/ie.css" rel="stylesheet">
    <![endif]-->

    <!--[if IE 9]>
        <link id="ie9style" href="css/ie9.css" rel="stylesheet">
    <![endif]-->

    <!-- start: Favicon -->

    <!-- end: Favicon -->

</head>

<body>
    <!-- start: Header -->
    <div class="navbar">
        <div class="navbar-inner">

```

```

<div class="container-fluid">
  <a class="btn btn-navbar" data-
toggle="collapse" data-target=".top-nav.nav-collapse,.sidebar-nav.nav-
collapse">
    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
  </a>
  <a class="brand"><span>PT Rekadaya Multi
Adiprima</span></a>

<!-- start: Header Menu -->

<div class="nav-no-collapse header-nav">
  <ul class="nav pull-right">
    <li class="dropdown hidden-
phone">
      <a class="btn
dropdown-toggle" data-toggle="dropdown" href="#">
        <i
class="halflings-icon white warning-sign"></i>
      </a>
      <ul class="dropdown-
menu notifications">
        <li
class="dropdown-menu-title">
          <span>You have 11 notifications</span>
          <a
href="#refresh"><i class="icon-repeat"></i></a>
        </li>
        <li>
          <a href="#">
            <span class="icon blue"><i class="icon-user"></i></span>
            <span class="message">New user registration</span>
            <span class="time">1 min</span>
          </a>
        </li>
        <li>
          <a href="#">

```

```
<span class="icon green"><i class="icon-comment-
alt"></i></span>
```

```
<span class="message">New comment</span>
```

```
<span class="time">16 min</span>
```

```
</a>
```

```
</li>
```

```
<li>
```

```
<a href="#">
```

```
<span class="icon blue"><i class="icon-user"></i></span>
```

```
<span class="message">New user registration</span>
```

```
<span class="time">36 min</span>
```

```
</a>
```

```
</li>
```

```
<li>
```

```
<a href="#">
```

```
<span class="icon yellow"><i class="icon-shopping-
cart"></i></span>
```

```
<span class="message">2 items sold</span>
```

```
<span class="time">1 hour</span>
```

```
</a>
```

```
</li>
```

```
<li>
```

```
class="warning">
```

```
<a href="#">
```

```
<span class="icon red"><i class="icon-user"></i></span>
```

```
<span class="message">User deleted account</span>
```

```
<span class="time">2 hour</span>
```

```
</a>
```

```
</li>
```

```
<li>
```

```
class="warning">
```

```
<a href="#">
```

```

        <span class="icon red"><i class="icon-shopping-
cart"></i></span>

        <span class="message">New comment</span>

        <span class="time">6 hour</span>
            </a>
        </li>
            <li>
                <a href="#">

        <span class="icon green"><i class="icon-comment-
alt"></i></span>

        <span class="message">New comment</span>

        <span class="time">yesterday</span>
            </a>
        </li>
            <li>
                <a href="#">

        <span class="icon blue"><i class="icon-user"></i></span>

        <span class="message">New user registration</span>

        <span class="time">yesterday</span>
            </a>
        </li>
        <li class="dropdown-menu-sub-footer">
            <a>View all notifications</a>
        </li>
    </ul>
        </li>
        <!-- Start account setting -->
        <li class="dropdown">
            <a class="btn
dropdown-toggle" data-toggle="dropdown" href="#">
                <i
class="halflings-icon white user"></i> <?php echo
"".$_SESSION["name"]."" ?>
            </span>
            class="caret"></span>
        </a>

```

```

        <ul class="dropdown-
menu">
            <li
class="dropdown-menu-title">
                <span>Account Settings</span>
            </li>
            <li><a
href="#"><i class="halflings-icon user"></i> Profile</a></li>
            <li><a
href=" ../login.php"><i class="halflings-icon off"></i> Logout</a></li>
        </ul>
    </li>
    <!-- end: User Dropdown -->
</ul>
</div>
<!-- end: Header Menu -->
</div>
</div>
</div>
<!-- start: Header -->
<div class="container-fluid-full">
<div class="row-fluid">
    <!-- start: Main Menu -->
    <div id="sidebar-left" class="span2">
        <div class="nav-collapse sidebar-nav">
            <ul class="nav nav-tabs nav-stacked
main-menu">
                <li><a
href=" ../home.php"><i class="icon-home"></i><span class="hidden-
tablet"> Home</span></a></li>
                <?php
                $stampil_menu=$_SESSION['jabatan'] == 'Supervisor Warehouse';
                if($stampil_menu){
                    echo
                    ,
                <li>
                <a class="dropmenu"
href="#"><i class="icon-th-list"></i><span class="hidden-tablet"> Data
Master</span></a>
                </li>
            </ul>

```

```

        <li><a
class="submenu" href="datauser.php"><i class="icon-file-alt"></i><span
class="hidden-tablet"> Data User</span></a></li>
        <li><a
class="submenu" href="../customer/datacustomer.php"><i class="icon-
file-alt"></i><span class="hidden-tablet"> Data
Customer</span></a></li>
        <li><a
class="submenu" href="../barangjadi/databarangjadi.php"><i class="icon-
file-alt"></i><span class="hidden-tablet"> Data Barang
Jadi</span></a></li>
    </ul>
</li>
<li>
    <a class="dropmenu"
href="#"><i class="icon-folder-close-alt"></i><span class="hidden-
tablet"> Data Transaksi</span></a>
    <ul>
        <li><a
class="submenu" href="../transaksimasuk.php"><i class="icon-file-
alt"></i><span class="hidden-tablet"> Data Transaksi Barang
Jadi</span></a></li>
    </ul>
</li>
<li>
    <a class="dropmenu"
href="#"><i class="icon-folder-close-alt"></i><span class="hidden-
tablet"> Data Laporan</span></a>
    <ul>
        <li><a
class="submenu" href="../formlaporan.php"><i class="icon-file-
alt"></i><span class="hidden-tablet"> Laporan Transaksi Barang
Jadi</span></a></li>
    </ul>
</li>
    </ul>
';}?'>
</div>
</div>
<!-- end: Main Menu -->
<noscript>
    <div class="alert alert-block span10">

```



```

        <label class="control-label"
for="focusedInput">NIK</label>
        <div class="controls">
            <input class="input-xsmall focused" name="nik"
id="focusedInput" type="text" value="">
        </div>
    </div>
    <div class="control-group">
        <label class="control-label"
for="focusedInput">Nama User</label>
        <div class="controls">
            <input class="input-xsmall focused"
name="nm_user" id="focusedInput" type="text" value="">
        </div>
    </div>

    <div class="control-group">
        <label class="control-label"
for="focusedInput">Jabatan</label>
        <div class="controls">
            <input class="input-xsmall focused" name="jabatan"
id="focusedInput" type="text" value="">
        </div>
    </div>

    <div class="control-group">
        <label class="control-label"
for="focusedInput">Email</label>
        <div class="controls">
            <input class="input-xsmall focused" name="email"
id="focusedInput" type="text" value="">
        </div>
    </div>

    <div class="control-group">
        <label class="control-label"
for="focusedInput">Username</label>
        <div class="controls">
            <input class="input-xsmall focused"
name="username" id="focusedInput" type="text" value="">
        </div>
    </div>

    <div class="control-group">
        <label class="control-label"
for="focusedInput">Password</label>

```

```

        <div class="controls">
            <input class="input-xsmall focused"
name="password" id="focusedInput" type="text" value="">
        </div>
    </div>

    <table width="800px" border="0">
        <tr>
            <td width="200px">
                <button type="submit" class="btn btn-
primary">Simpan</button>
                <button type="reset" class="btn">Batal</button>
            </td>
        </tr>
    </table>

</div>
</fieldset>
</form>
</div>
<div class="control-group">

<div class="row-fluid sortable">
    <div class="box span12">
        <div class="box-header" data-
original-title>
            <h2><i class="halflings-icon
user"></i><span class="break"></span>Data User</h3>
            <div class="box-icon">
                <a href="#"
class="btn-setting"><i class="halflings-icon wrench"></i></a>
                <a href="#"
class="btn-minimize"><i class="halflings-icon chevron-up"></i></a>
                <a href="#"
class="btn-close"><i class="halflings-icon remove"></i></a>
            </div>
        </div>
        <div class="box-content">
            <table class="table table-
striped table-bordered bootstrap-datatable datatable">
                <thead>
                    <tr>
                        <th>NIK</th>

```

```

User</th>
<th>Jabatan</th>
<th>Email</th>
<th>Username</th>
<th>Password</th>
<th>Aksi</th>
</tr>
</thead>
<tbody>
<?php
$user=mysql_query("select *
from tb_user");
while($u=mysql_fetch_array($user)){
?>
<tr>
<td
class="center"><?php echo $u['nik']?></td>
<td
class="center"><?php echo $u['nm_user']?></td>
<td
class="center"><?php echo $u['jabatan']?></td>
<td
class="center"><?php echo $u['email']?></td>
<td
class="center"><?php echo $u['username']?></td>
<td
class="center"><?php echo $u['password']?></td>
<td
class="center">
<a
class="btn btn-info" href="edituser.php?nik=<?php echo $u['nik']?>">
<i class="halflings-icon white edit"></i>
</a>
<a
class="btn btn-danger" href="hapus.php?nik=<?php echo $u['nik']?>">

```

```

        <i class="halflings-icon white trash"></i>
    </a>
    <?php
} ?>
    </td>
</tr>
</tr>
</tbody>
</table>
</div>
</div><!--/span-->
</div><!--/row-->

</div><!--/.fluid-container-->

    <!-- end: Content -->
</div><!--/#content.span10-->
</div><!--/fluid-row-->

<div class="modal hide fade" id="myModal">
    <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">x</button>
        <h3>Settings</h3>
    </div>
    <div class="modal-body">
        <p>Here settings can be configured...</p>
    </div>
    <div class="modal-footer">
        <a href="#" class="btn" data-
dismiss="modal">Close</a>
        <a href="#" class="btn btn-primary">Save
changes</a>
    </div>
</div>

<div class="clearfix"></div>

<footer>

    <p>

```

```
<span style="text-align:left;float:left">&copy; 2017  
<a href="http://jiji262.github.io/Bootstrap_Metro_Dashboard/" alt="Riska  
Fitri Yanti">Bootstrap Metro Dashboard</a></span>
```

```
</footer>
```

```
<!-- start: JavaScript-->
```

```
<script src=" ../js/jquery-1.9.1.min.js"></script>
```

```
<script src=" ../js/jquery-migrate-1.0.0.min.js"></script>
```

```
<script src=" ../js/jquery-ui-1.10.0.custom.min.js"></script>
```

```
<script src=" ../js/jquery.ui.touch-punch.js"></script>
```

```
<script src=" ../js/modernizr.js"></script>
```

```
<script src=" ../js/bootstrap.min.js"></script>
```

```
<script src=" ../js/jquery.cookie.js"></script>
```

```
<script src=" ../js/fullcalendar.min.js"></script>
```

```
<script src=" ../js/jquery.dataTables.min.js"></script>
```

```
<script src=" ../js/excanvas.js"></script>
```

```
<script src=" ../js/jquery.flot.js"></script>
```

```
<script src=" ../js/jquery.flot.pie.js"></script>
```

```
<script src=" ../js/jquery.flot.stack.js"></script>
```

```
<script src=" ../js/jquery.flot.resize.min.js"></script>
```

```
<script src=" ../js/jquery.chosen.min.js"></script>
```

```
<script src=" ../js/jquery.uniform.min.js"></script>
```

```
<script src=" ../js/jquery.cleditor.min.js"></script>
```

```
<script src=" ../js/jquery.noty.js"></script>
```

```
<script src=" ../js/jquery.elfinder.min.js"></script>
```

```
<script src=" ../js/jquery.raty.min.js"></script>
```

```
<script src=" ../js/jquery.iphone.toggle.js"></script>
```

```
<script src="../js/jquery.uploadify-3.1.min.js"></script>
<script src="../js/jquery.gritter.min.js"></script>
<script src="../js/jquery.imagesloaded.js"></script>
<script src="../js/jquery.masonry.min.js"></script>
<script src="../js/jquery.knob.modified.js"></script>
<script src="../js/jquery.sparkline.min.js"></script>
<script src="../js/counter.js"></script>
<script src="../js/retina.js"></script>
<script src="../js/custom.js"></script>
<!-- end: JavaScript-->
</body>
</html>
```