

**RANCANG BANGUN SISTEM INFORMASI
PERGUDANGAN DI BAGIAN *TRIMMING* BERBASIS WEB
DENGAN MENGGUNAKAN CODE IGNITER 3.1.5 DAN
MYSQL 5.5.27 PADA PT KRAMA YUDHA RATU MOTOR**

TUGAS AKHIR

Untuk Memenuhi Sebagian Syarat Penyelesaian Jenjang Diploma Empat (D-IV)
Program Studi Sistem Informasi Industri Otomotif Pada Politeknik STMI Jakarta

OLEH

YENI LESTARI 1413009



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2017**

ABSTRAK

PT. Krama Yudha Ratu Motor merupakan sebuah perusahaan perseroan terbatas yang bergerak dalam bidang perakitan kendaraan bermotor jenis niaga. Dalam menjalankan proses pergudangan, ada beberapa bagian yang mengelola manajemen pergudangan, salah satunya yaitu bagian *part control trimming*. Dalam menjalankan proses pergudangan di bagian *trimming* terutama di kegiatan pengolahan data pergudangan masih menggunakan *Microsoft Excel* sehingga proses dapat terjadinya *human error* seperti salah dalam penginputan. Dalam proses pengolahan data penerimaan admin gudang harus terlebih dahulu membuat *form* penerimaan kemudian input sesuai dengan dokumen. Dalam proses pengolahan data pengeluaran, admin gudang *input* berdasarkan permintaan dari bagian *production and planning control (PPC)*, Hal ini juga dapat menyebabkan *human error* dan stok tidak akurat. Untuk mengatasi masalah tersebut dibuatlah suatu aplikasi rancang bangun sistem informasi pergudangan. Dengan menggunakan metode pengembangan *prototype evolutioner*. Aplikasi untuk sistem informasi tersebut dirancang dengan menggunakan PHP dengan menggunakan *Framework Code Igniter 3.1.5* dan database MySQL 5.5.27. Metode yang digunakan yaitu *Evolutionary Prototype* dalam pengembangan sistem pergudangan menggunakan beberapa alat bantu seperti Diagram *Use Case*, Diagram *Activity*, Diagram *Sequence*, Diagram *Class*, Kamus Data, HIPO dan *flowchart*. Aplikasi ini dapat memudahkan dalam proses penerimaan, pengeluaran, laporan dan stok

Kata kunci: Sistem informasi, Pergudangan, *Prototyping*, UML, *Use Case*, *Code Igniter 3.1.5*, dan MySQL.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PENGESAHAN	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN TUGAS AKHIR	
LEMBAR PERNYATAAN KEASLIAN	
Abstrak	i
Kata Pengantar	ii
Daftar Isi	iii
Daftar Tabel	vi
Daftar Gambar.....	viii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan.....	2
1.3 Tujuan Penelitian.....	2
1.4 Batasan Masalah.....	3
1.5 Manfaat Tugas Akhir.....	3
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Rancang Bangun	6
2.2 Pengertian Sistem.....	6
2.2.1 Karakteristik Sistem	7
2.2.2 Klasifikasi Sistem	9
2.3 Konsep Dasar Informasi	10
2.3.1 Siklus Informasi.....	11
2.3.2 Kualitas Informasi	12
2.4 Konsep Dasar Sistem Informasi	13
2.4.1 Komponen Sistem Informasi	13
2.5 Metodologi Pengembangan Sistem	15
2.6 Konsep Dasar Gudang	17
2.6.1 Tipe dan Jenis Gudang.....	17
2.6.2 Manfaat Gudang	18
2.6.3 Fungsi Manajemen Pergudangan.....	19
2.6.4 Operasi Pergudangan	20
2.6.5 Mekanisme Dasar Sistem Gudang	21

	2.6.6	Metode dalam Penyimpanan Gudang.....	23
	2.6.7	Sistem Informasi Pergudangan	24
	2.7	Proses <i>Trimming</i>	25
	2.8	<i>Flowmap</i> dan <i>Flowchart</i>	25
	2.8.1	Jenis-Jenis <i>Flowchart</i>	26
	2.8.2	Simbol <i>Flowchart</i>	27
	2.9	<i>Unified Modeling Language (UML)</i>	29
	2.10	Sistem Manajemen Basis Data Relational (RDBMS).....	37
	2.11	Kamus Data.....	37
	2.12	HIPO (<i>Hierarchy plus Input-Process-Output</i>)	38
	2.13	PHP	40
	2.14	<i>Code Igniter</i>	41
	2.15	MySQL.....	42
	2.16	XAMPP	44
BAB III		METODOLOGI PENELITIAN	46
	3.1	Jenis dan Sumber Data.....	46
	3.2	Metode Pengumpulan Data	47
	3.3	Metode Pengembangan Sistem	48
	3.4	Kerangka Penelitian.....	48
BAB IV		PENGUMPULAN DAN PENGOLAHAN DATA	52
	4.1	Sejarah Perusahaan.....	52
	4.2	Profil Perusahaan.....	53
	4.3	Visi dan Misi Perusahaan.....	54
	4.4	Budaya Kerja Perusahaan	55
	4.5	Kebijakan Sistem Manajemen Keselamatan dan Kesehatan Kerja.....	56
	4.6	Struktur Organisasi Perusahaan.....	56
	4.7	Gudang Penyimpanan <i>Part</i> dan Bagian <i>Part Control</i>	61
	4.8	Dokumen pada Sistem Pergudangan yang Berjalan	62
	4.9	Prosedur Pergudangan	68
	4.10	Penggambaran Sistem Pergudangan dengan Berjalan dengan <i>Unified Modeling Language (UML)</i>	71
BAB V		ANALISIS DAN PEMBAHASAN	75
	5.1	Analisis Sistem.....	75
	5.2	Analisis Kebutuhan Rinci Sistem	75
	5.3	Perancangan <i>Flowmap</i> Sistem Usulan	77
	5.4	Analisis Sistem Informasi Usulan.....	79
	5.4.1	<i>Use Case Diagram</i>	79

5.4.2	<i>Activity Diagram</i>	89
5.4.3	<i>Sequence Diagram</i>	98
5.4.4	<i>Class Diagram</i>	108
5.4.5	Kamus Data	108
5.4.6	<i>Relational Diagram</i>	114
5.4.7	<i>Deployment diagram</i>	114
5.4.8	HIPO (<i>Hierarchy plus Input-Process-Output</i>).....	115
5.5.9	<i>Flowchart Aplikasi Usulan</i>	116
5.5.10	<i>Interface</i>	118
5.5.11	Implementasi Sistem	130
BAB VI	Kesimpulan dan Saran	132
6.1	Kesimpulan	132
5.2	Saran.....	132

DAFTAR PUSTAKA 133

LAMPIRAN

DAFTAR GAMBAR

		Halaman
Gambar II.1	Karakteristik Sistem	9
Gambar II.2	Siklus Informasi	11
Gambar II.3	Kriteria Kualitas Informasi	12
Gambar II.4	Blok Sistem Informasi yang Saling Berinteraksi	13
Gambar II.5	Langkah Pengembangan <i>Prototype Evolutionary</i>	16
Gambar II.6	<i>Visual Table of Contents</i>	39
Gambar II.7	<i>Overview Diagram</i>	39
Gambar II.8	Logo <i>Code Igniter</i>	41
Gambar III.1	Kerangka Pemecahan Masalah	49
Gambar IV.1	Struktur Organisasi Perusahaan	57
Gambar IV.2	<i>Delivery Note</i>	63
Gambar IV.3	Surat Jalan	63
Gambar IV.4	<i>Input Purchase Order</i>	64
Gambar IV.5	<i>Master Production Schedule (MPS)</i>	64
Gambar IV.6	<i>Daily Production Schedule (DPS)</i>	65
Gambar IV.7	<i>Form Trimming on Day</i>	66
Gambar IV.8	<i>Balance Produksi</i>	67
Gambar IV.9	Stok	68
Gambar IV.10	<i>Flowmap</i> Sistem Informasi Pergudangan yang Berjalan	70
Gambar IV.11	<i>Use Case Diagram</i> Sistem Informasi Pergudangan	74
Gambar V.1	<i>Flowmap</i> Sistem Informasi Pergudangan	78
Gambar V.2	<i>Use Case Sistem Informasi Pergudangan Usulan</i>	80
Gambar V.3	<i>Activity Diagram</i> Proses Login	89
Gambar V.4	<i>Activity Diagram</i> Proses Mengelola Data <i>Master</i>	90
Gambar V.5	<i>Activity Diagram</i> Proses Mengisi Data Penerimaan <i>Part</i>	91
Gambar V.6	<i>Activity Diagram</i> Proses Melihat Data Stok <i>Part</i>	92
Gambar V.7	<i>Activity Diagram</i> Proses Mengisi Data Permintaan <i>Part</i>	93

Gambar V.8	<i>Activity Diagram</i> Proses Melihat Data Pengeluaran <i>Part</i>	94
Gambar V.9	<i>Activity Diagram</i> Proses Mencetak Data Pengeluaran <i>Part</i> ..	95
Gambar V.10	<i>Activity Diagram</i> Proses Mengisi Data Penerimaan <i>Part</i>	96
Gambar V.11	<i>Activity Diagram</i> Proses Mencetak Laporan Pengeluaran <i>Part</i>	97
Gambar V.12	<i>Sequence Diagram</i> Proses Login	98
Gambar V.13	<i>Sequence Diagram</i> Mengelola Data Master <i>User</i>	99
Gambar V.14	<i>Sequence Diagram</i> Mengelola Data Master <i>Vendor</i>	100
Gambar V.15	<i>Sequence Diagram</i> Mengelola Data Master <i>Part</i>	101
Gambar V.16	<i>Sequence Diagram</i> Mengelola Data Master Jenis	102
Gambar V.17	<i>Sequence Diagram</i> Proses Penerimaan	103
Gambar V.18	<i>Sequence Diagram</i> Proses Pengeluaran	104
Gambar V.19	<i>Sequence Diagram</i> Proses Permintaan.....	105
Gambar V.20	<i>Sequence Diagram</i> Proses Stok	106
Gambar V.21	<i>Sequence Diagram</i> Proses Stok Detail.....	107
Gambar V.22	<i>Class Diagram</i> Usulan	108
Gambar V.23	<i>Relational Diagram</i> Sistem Informasi Pergudangan.....	114
Gambar V.24	<i>Deployment Diagram</i> Sistem Informasi Pergudangan.....	115
Gambar V.25	HIPO Sistem Informasi Pergudangan	116
Gambar V.26	<i>Flowchart</i> Sistem Informasi Pergudangan	117
Gambar V.27	Rancangan <i>Form Login</i>	118
Gambar V.28	Rancangan <i>Form Menu Utama</i>	119
Gambar V.29	Rancangan Menu Data Master <i>User</i>	120
Gambar V.30	Rancangan Menu Data Master <i>Vendor</i>	121
Gambar V.31	Rancangan Menu Data Master <i>Part</i>	122
Gambar V.32	Rancangan Menu Data Master Jenis	123
Gambar V.33	Rancangan Menu Transaksi Penerimaan	124
Gambar V.34	Rancangan Menu Transaksi Pengeluaran.....	125
Gambar V.35	Rancangan Menu Transaksi Permintaan	126
Gambar V.36	Rancangan Menu Data Stok Detail.....	127
Gambar V.37	Rancangan Menu Data Stok Per Unit	128

Gambar V.38	Rancangan Menu Data Laporan Penerimaan	129
Gambar V.39	Rancangan Menu Data Laporan Pengeluaran.....	130

ABSTRAK DAFTAR TABEL

		Halaman
Tabel II.1	Simbol <i>Flowchart</i>	27
Tabel II.2	Notasi-Notasi <i>Use Case Diagram</i>	30
Tabel II.3	Notasi-Notasi <i>Activity Diagram</i>	31
Tabel II.4	Notasi-Notasi <i>Sequence Diagram</i>	32
Tabel II.5	Notasi-Notasi <i>Class Diagram</i>	33
Tabel II.6	Notasi-Notasi <i>Object Diagram</i>	35
Tabel II.7	Notasi-Notasi <i>Component Diagram</i>	36
Tabel II.8	Notasi-Notasi <i>Deployment Diagram</i>	36
Tabel II.9	Contoh Kamus Data Untuk Tabel Pemasok.....	37
Tabel II.10	Beberapa Tipe Data Pada MySQL	43
Tabel IV.1	Definisi Aktor <i>Use Case</i> Sistem Informasi Pergudangan	71
Tabel IV.2	Definisi <i>Use Case Diagram</i> Sistem Informasi Pengudangan.....	71
Tabel V.1	Kebutuhan Rinci Sistem	75
Tabel V.2	Definisi Aktor <i>Use Case</i>	81
Tabel V.3	Definisi <i>Use Case</i> Mengelola Data <i>Master</i>	81
Tabel V.4	Definisi <i>Use Case Login</i>	82
Tabel V.5	Definisi <i>Use Case</i> Mengisi Data Penerimaan <i>Part</i>	83
Tabel V.6	Definisi <i>Use Case</i> Melihat Data Stok <i>Part</i>	83
Tabel V.7	Definisi <i>Use Case</i> Mengisi Data Permintaan <i>Part</i>	84
Tabel V.8	Definisi <i>Use Case</i> Melihat Data Permintaan <i>Part</i>	84
Tabel V.9	Definisi <i>Use Case</i> Mencetak Data Permintaan <i>Part</i>	85
Tabel V.10	Definisi <i>Use Case</i> Mengisi Data Pengeluaran <i>Part</i>	85
Tabel V.11	Definisi <i>Use Case</i> Mencetak Data Pengeluaran <i>Part</i>	86
Tabel V.12	Definisi <i>Use Case</i> Melihat Data Laporan Penerimaan <i>Part</i>	87
Tabel V.13	Definisi <i>Use Case</i> Mencetak Laporan Penerimaan <i>Part</i>	87
Tabel V.14	Definisi <i>Use Case</i> Melihat Data Laporan Pengeluaran <i>Part</i>	88
Tabel V.15	Definisi <i>Use Case</i> Mencetak Laporan Pengeluaran <i>Part</i>	88
Tabel V.16	Tabel <i>User</i>	109

Tabel V.17	Tabel <i>Vendor</i>	109
Tabel V.18	Tabel <i>Part</i>	110
Tabel V.19	Tabel Jenis	110
Tabel V.20	Tabel Stok	110
Tabel V.21	Tabel Stok Detail.....	111
Tabel V.22	Tabel Penerimaan.....	111
Tabel V.23	Tabel Pengeluaran	112
Tabel V.24	Tabel Permintaan.....	112
Tabel V.25	Tabel Laporan Penerimaan	113
Tabel V.26	Tabel Laporan Pengeluaran	113

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi dan informasi menyebabkan persaingan antar perusahaan semakin ketat. Hal ini mendorong suatu organisasi atau perusahaan untuk lebih kreatif, inovatif dan efisien dalam menghasilkan produk yang berkualitas tinggi. Untuk menciptakan produk yang berkualitas tinggi, perusahaan harus memperhatikan proses produksi yang sedang berjalan mulai pengambilan bahan baku dari gudang penyimpanan hingga menjadi produk yang mempunyai nilai jual dan sampai ke tangan konsumen.

Bagi suatu perusahaan, gudang merupakan kegiatan yang penting. Oleh karena itu, manajemen pergudangan harus dikelola dengan baik. Apabila manajemen pergudangan tidak dikelola dengan baik maka akan mengakibatkan proses produksi tidak berjalan dengan lancar atau mengalami keterlambatan pengiriman material. Untuk menunjang kelancaran proses pergudangan diperlukan adanya sistem informasi mulai dari penerimaan bahan baku, pendistribusian bahan baku ke *line* produksi dan dapat menyajikan informasi secara cepat dan akurat.

PT. Krama Yudha Ratu Motor mempunyai 2 gudang penyimpanan *part* atau bahan baku yaitu gudang *welding* dan gudang *trimming*. Gudang *welding* hanya menyimpan *part* atau bahan baku untuk keperluan produksi di *line welding* sedangkan gudang *trimming* hanya menyimpan *part* atau bahan baku untuk keperluan produksi di *line trimming*. Dalam mengelola manajemen pergudangan, PT. Krama Yudha Ratu Motor mempunyai bagian khusus untuk mengelola gudang yaitu bagian *part control*. Bagian *part control* adalah bagian yang bertanggung jawab dalam hal penyimpanan dan pengelolaan *part* atau bahan baku sehingga dapat terjamin ketersediaannya pada saat dibutuhkan untuk kepentingan proses produksi.

Pada saat ini sistem informasi pengolahan data pergudangan di bagian *trimming* PT. Krama Yudha Ratu Motor masih menggunakan *Microsoft Excel* dan sistem pengolahan data pergudangan yang saat ini digunakan belum terintegrasi dengan *database*. Hal ini dapat memungkinkan terjadinya *human error* seperti: data yang di-*input double*, tidak teliti saat penginputan, dapat memungkinkan terjadinya kehilangan data yang menyebabkan stok menjadi tidak akurat. Dengan sistem seperti ini bagian admin gudang sering kesulitan untuk mencari data *part*.

Untuk mengatasi masalah tersebut, diperlukan aplikasi untuk membantu perusahaan dalam mengatasi permasalahan tersebut. Adapun judul tugas akhir ini adalah “RANCANG BANGUN SISTEM INFORMASI PERGUDANGAN DI BAGIAN *TRIMMING* BERBASIS WEB DENGAN MENGGUNAKAN *CODE IGNITER 3.1.5* DAN *MYSQL 5.5.27* PADA PT. KRAMA YUDHA RATU MOTOR” .

1.2 Pokok Permasalahan

Berdasarkan uraian latar belakang di atas maka dapat di simpulkan pokok permasalahan yang ada adalah:

1. Pengolahan data penerimaan *part* dan pengeluaran *part* masih menggunakan *Microsoft Excel*, hal ini bisa memungkinkan terjadinya *human error* seperti data yang di *input double* dan tidak teliti saat penginputan yang mengakibatkan data tidak akurat.
2. Pengolahan data penerimaan *part* dan pengeluaran *part* masih menggunakan *Microsoft Excel* sehingga data mudah hilang dan data *part* sulit untuk dicari.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir yang dilakukan di PT. Krama Yudha Ratu Motor adalah sebagai berikut:

1. Merancang dan membangun sistem informasi yang dapat meminimalisir kesalahan (*human error*) dalam penginputan penerimaan *part* dan pengeluaran *part*.

2. Membuat sistem informasi yang memiliki penyimpanan data atau *database* agar data tidak hilang dan memudahkan mencari data *part* .

1.4 Batasan Masalah

Agar dalam penulisan Tugas Akhir ini lebih fokus dan terarah, maka perlu diadakan batasan-batasan yang berkaitan dengan permasalahan ini adalah sebagai berikut:

1. Tempat penelitian dilakukan pada gudang bagian *trimming* PT Krama Yudha Ratu Motor.
2. Analisis dan perancangan yang dilakukan hanya sebatas menangani masalah pengolahan data penerimaan *part* dari *vendor* dan pengolahan data pengeluaran *part* yang dibutuhkan untuk keperluan produksi mobil CJM. Tidak membahas tentang pengiriman *part* ke *line kitting*, *sub assembling* dan produksi.
3. *Part* yang diamati merupakan *part* lokal yang diterima sesuai dengan dokumen-dokumen. Tidak membahas tentang pemesanan *part* dan *kanban*.
4. Informasi yang dihasilkan berupa penerimaan *part*, stok, permintaan produksi, pengeluaran *part* dan laporan penerimaan dan pengeluaran.

1.5 Manfaat Tugas Akhir

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

1. Bagi penulis
Rancang Bangun Sistem Informasi Pergudangan memberikan pembelajaran untuk menerapkan ilmu yang diperoleh selama perkuliahan serta menambah wawasan dan pengetahuan penulis, khususnya dalam analisis dan perancangan sistem informasi.
2. Bagi perusahaan
Rancang Bangun Sistem Informasi Pergudangan diharapkan dapat mempermudah dalam mendapatkan informasi tentang *part* yang datang hari

ini dalam proses penerimaan *part*, pengeluaran *part*, stok *part* yang ada di gudang dan laporan penerimaan *part* dan pengeluaran *part*.

3. Bagi pembaca

Hasil penelitian ini diharapkan dapat memberikan pengetahuan dan referensi dalam hal rancang bangun sistem informasi pergudangan.

1.6 Sistematika Penulisan

Agar lebih mempermudah perumusan dan pemecahan masalah yang akan dibahas pada tugas akhir ini, maka diuraikan tahapan-tahapan dalam penyusunan laporan 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 membahas berbagai teori-teori mengenai definisi rancang bangun, pengertian sistem, karakteristik sistem, klasifikasi sistem, konsep dasar informasi, siklus informasi, kualitas informasi, konsep dasar sistem informasi, komponen sistem informasi, metodologi pengembangan sistem, konsep dasar gudang, tipe dan jenis gudang, manfaat gudang, fungsi manajemen gudang, operasi-operasi pergudangan, mekanisme dasar sistem gudang, metode dalam penyimpanan gudang, sistem informasi pergudangan, *flowmap* dan *flowchart*, jenis-jenis *flowchart*, simbol-simbol *flowchart*, *Unified Modeling Language* (UML), kamus data, *Hierarchy plus Input-Process-Output* (HIPO), PHP, MySQL dan XAMPP yang dikutip dari buku-buku referensi yang dijadikan sebagai acuan dalam penulisan Tugas Akhir sesuai dengan tema yang dibahas.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metode yang digunakan dalam pengumpulan data selama kegiatan penelitian. Selain itu dijelaskan pula kerangka pemecahan masalah yang menguraikan tahap-tahap untuk mengatasi permasalahan yang telah diidentifikasi dan menggunakan metode *prototype evolutionary* untuk melakukan pengembangan sistem.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini menguraikan data yang telah dikumpulkan dan diolah untuk mendeskripsikan hal-hal yang berkaitan dengan tema dari Tugas Akhir.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini membahas tentang pengembangan dan perancangan sistem informasi pergudangan yang sesuai dengan metode yang digunakan.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari hasil pengembangan sistem dan saran-saran yang sekiranya perlu disampaikan sebagai masukan bagi perusahaan.

BAB II

LANDASAN TEORI

2.1 Rancang Bangun

Rancang bangun adalah tahap dari setelah analisis dari siklus pengembangan sistem yang merupakan pendefinisian dari kebutuhan-kebutuhan fungsional, serta menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi, termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat keras dan perangkat lunak dari suatu sistem. (Jogiyanto,2005)

2.2 Pengertian Sistem

Kata sistem sendiri berasal dari bahasa Latin “*systema*” dan bahasa Yunani “*sustema*” adalah suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Istilah sistem merupakan istilah dari bahasa Yunani, *system* yang artinya adalah himpunan bagian atau unsur yang saling berhubungan secara teratur untuk mencapai tujuan bersama. Untuk pengertian sistem yang diintisarikan dari beberapa sumber referensi, mendefinisikan bahwa sistem :

- a. Menurut McLeod (2004) “Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. Dengan kata lain sistem juga merupakan sekelompok elemen–elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan”.
- b. Menurut Lucas (1987) “Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsure, komponen atau variable-variable yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu”.
- c. Menurut Jogiyanto (2005) “Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu”.

Suatu sistem dapat terdiri dari sistem-sistem bagian (*subsystem*). Sebagai contoh, sistem komputer terdiri dari subsistem perangkat keras (*hardware*) dan perangkat lunak (*software*). Masing-masing subsistem dapat terdiri dari subsistem- subsistem yang lebih kecil atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar. Subsistem-subsistem saling berinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran system tersebut dapat tercapai. Interaksi dari subsistem-subsistem sedemikian rupa sehingga dicapai suatu

kesatuan yang terpadu atau terintegrasi. (Jogiyanto, 2005).

2.2.1 Karakteristik Sistem

Menurut Jogiyanto (2005), karakteristik sistem adalah sistem yang mempunyai komponen (*components*), batas sistem (*boundary*), lingkungan sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*) dan sasaran (*objectives*).

a. Komponen Sistem (*components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu mempengaruhi proses sistem secara keseluruhan.

b. Batasan Sistem (*boundary*)

Batasan sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

c. Lingkungan Luar Sistem (*environment*)

Lingkungan luar (*environment*) dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan berupa energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

d. Penghubung (*interface*)

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

e. Masukan (*input*)

Masukan (*input*) sistem adalah energi yang masuk kedalam sistem. Masukan tersebut dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk mendapatkan sebuah keluaran. Sebagai contoh data adalah *signal input* untuk diolah menjadi informasi.

f. Keluaran (*output*)

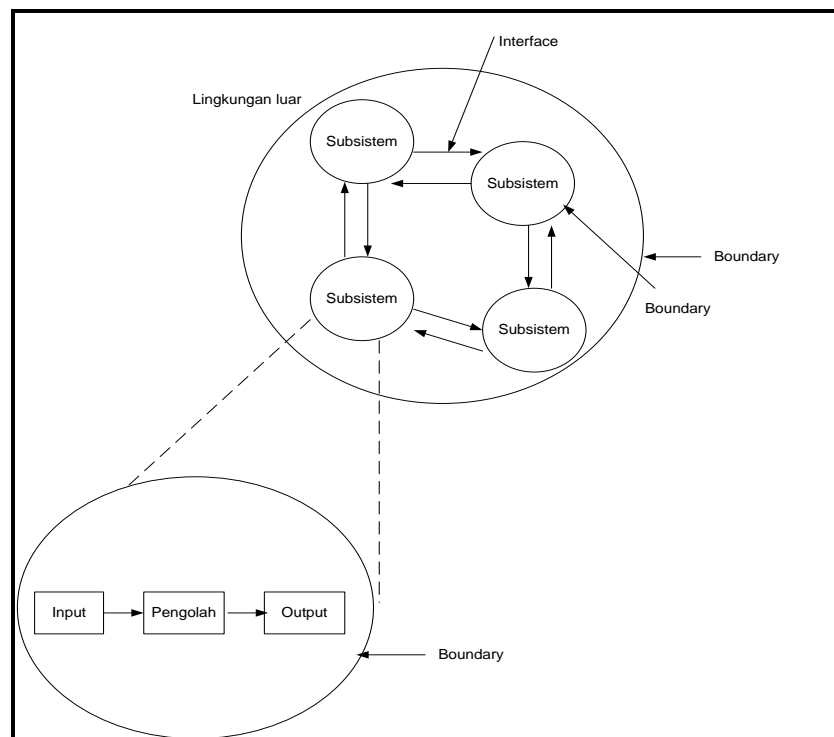
Keluaran (*output*) sistem adalah hasil dari energi yang diolah serta diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

g. Pengolah (*process*)

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

h. Sasaran (*objectives*)

Sebuah sistem sudah tentu mempunyai sasaran ataupun tujuan. Dengan adanya sasaran sistem, maka kita dapat menentukan masukan yang dibutuhkan sistem dan keluaran apa yang akan dihasilkan sistem tersebut dapat dikatakan berhasil apabila mencapai atau mengenai sasaran atau pun tujuan.



Gambar II.1 Karakteristik Sistem

(Sumber: Jogiyanto, 2005)

2.2.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya adalah sebagai berikut:

- a. Sistem Abstrak (*abstract system*) dan Sistem Fisik (*physical system*).

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem *teologia* yaitu sistem yang berupa pemikiran-pemikiran hubungan antara manusia dan tuhan. Sistem fisik merupakan sistem yang ada secara fisik, misalnya sistem komputer, sistem akuntansi dan sistem produksi.

- b. Sistem Alamiah (*natural system*) dan Sistem Buatan Manusia (*human made system*).

Sistem alamiah adalah sistem yang terjadi melalui proses alam dan tidak dibuat manusia, misalnya sistem perputaran bumi. Sistem buatan manusia adalah sistem yang dirancang oleh manusia dan melibatkan interaksi antara manusia dengan mesin, misalnya sistem informasi akuntansi.

- c. Sistem Tertentu (*deterministic system*) dan Sistem Tidak Tentu (*probabilistic system*).

Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer adalah contoh dari sistem tertentu yang tingkah lakunya dapat dipastikan melalui program yang dijelankannya. Sistem tidak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

- d. Sistem Tertutup (*closed system*) dan Sistem Terbuka (*open system*).

Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak diluarnya. Pada kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanyalah *relatively closed system*. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya, sehingga harus memiliki sistem pengendalian yang baik. (Jogiyanto, 2005).

2.3 Konsep Dasar Informasi

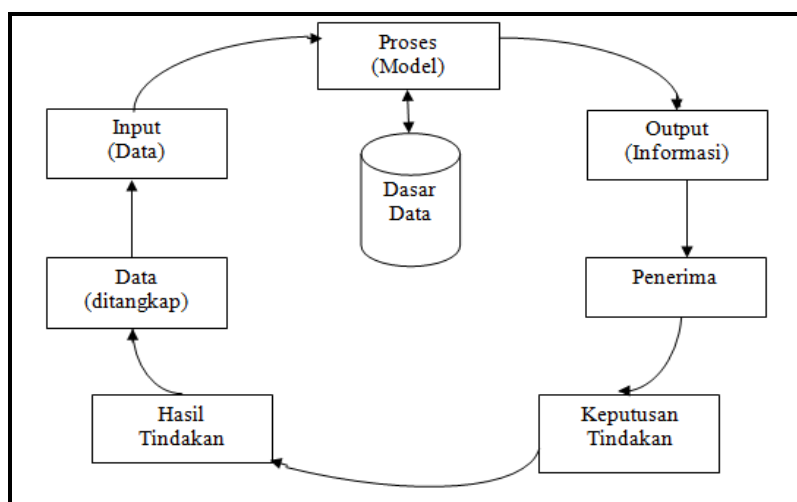
Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*event*) yang nyata digunakan untuk pengambilan keputusan.

Sumber dari informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal atau data-idem. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu (Jogiyanto, 1999).

2.3.1 Siklus Informasi

Di dalam kegiatan suatu perusahaan, misalnya dari hasil transaksi penjualan oleh sejumlah *salesman*, dihasilkan sejumlah faktor-faktor yang merupakan data dari penjualan pada suatu periode tertentu. Faktor-faktor penjualan tersebut masih belum dapat bercerita banyak kepada manajemen. Untuk pengambilan keputusan, maka faktor-faktor tersebut perlu diolah lebih lanjut untuk menjadi suatu informasi (Jogiyanto, 2005).

Data agar menjadi lebih berarti dan berguna dalam bentuk informasi maka perlu diolah menjadi suatu model tertentu. Data yang telah diolah tersebut kemudian diterima oleh penerima, lalu penerima membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, dan diproses kembali lewat suatu model dan seterusnya sehingga membentuk suatu siklus. Siklus ini disebut dengan siklus informasi (*information cycle*) atau disebut pula siklus pengolahan data (*processing cycles*) (Jogiyanto, 2005).



Gambar II.2 Siklus Informasi

(Sumber: Jogiyanto, 2005)

2.3.2 Kualitas Informasi

Menurut Jogiyanto (2005) kualitas informasi (*quality of information*) sangat dipengaruhi atau ditentukan oleh tiga hal, yaitu:

a. Relevan (*Relevancy*)

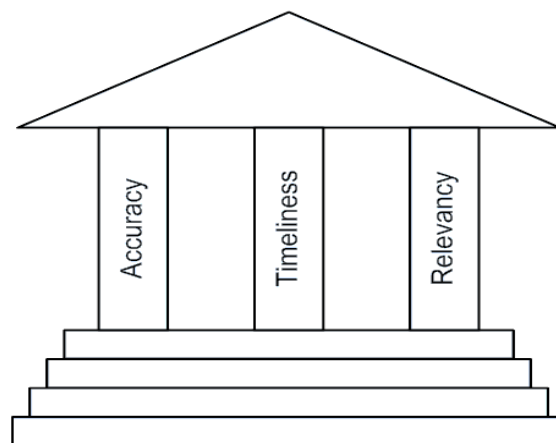
Relevan berarti informasi harus memberikan manfaat bagi pemakainya. Relevansi informasi berbeda-beda untuk tiap-tiap orang satu dengan yang lainnya.

b. Akurat (*Accuracy*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak bisa menyesatkan dan harus jelas mencerminkan maksudnya. Ketidakakuratan dapat terjadi karena sumber informasi (data) mengalami gangguan atau kesengajaan sehingga merusak atau merubah data-data asli tersebut. Komponen akurat: lengkap (*completeness*), benar (*correctness*), dan aman (*security*).

c. Tepat Waktu (*timeliness*)

Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat. Informasi yang terlambat tidak mempunyai nilai yang baik, sehingga kalau digunakan sebagai dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan dalam keputusan dan tindakan. Kondisi demikian menyebabkan mahalny suatu nilai informasi, sehingga kecepatan untuk mendapatkan, mengolah, dan mengirimkannya memerlukan teknologi-teknologi terbaru.



Gambar II.3

Kriteria Kualitas

Informasi

(Sumber: Jogiyanto, 2005)

2.4 Konsep Dasar Sistem Informasi

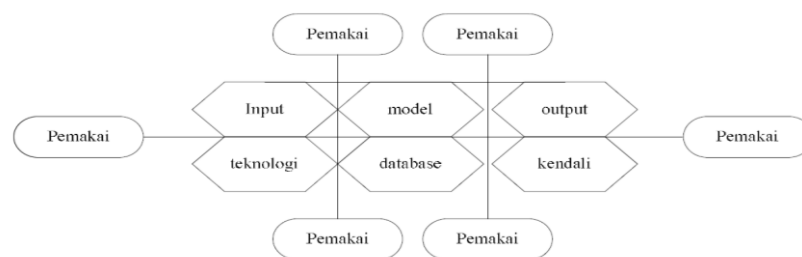
Informasi merupakan hal yang sangat penting bagi manajemen di dalam pengambilan keputusan. Informasi dapat diperoleh dari sistem informasi (*information system*) atau disebut juga dengan *processing terms* atau *information processing system* atau *information generating systems*.

Jogiyanto (2005), sistem informasi didefinisikan sebagai berikut: “Suatu sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Sebagai tambahan terhadap pendukung pengambilan keputusan, koordinasi, dan kendali, sistem informasi dapat juga membantu para manajer dan karyawan untuk meneliti permasalahan, memvisualisasikan pokok-pokok yang kompleks dan menciptakan produk-produk baru.

2.4.1 Komponen Sistem Informasi

Sistem informasi dapat terdiri dari beberapa komponen yang disebut blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*modelblock*), blok keluaran (*output block*), dan blok teknologi (*technology block*), blok dasar data (*database block*), dan blok kendali (*control block*). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarannya.



Gambar II.4 Blok Sistem Informasi yang Saling Berinteraksi

(Sumber: Jogiyanto, 2005)

Keterangan gambar blok sistem informasi yang berinteraksi :

1. Blok Masukan

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, dapat berupa dokumen-dokumen dasar.

2. Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematika yang akan memanipulasi data input dan data yang tersimpan di dasar data dengan cara tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran

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

Teknologi merupakan “kotak alat” (*tool-box*) dari pekerjaan sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem keseluruhan. Teknologi terdiri dari 2 bagian utama, yaitu perangkat lunak (*software*) dan perangkat keras (*hardware*).

5. Blok Basis Data

Basis data merupakan kumpulan data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut DBMS (*Database Management Systems*).

6. Blok Kendali

Agar sistem informasi dapat berjalan sesuai dengan yang diinginkan, maka perlu adanya pengendalian-pengendalian di dalamnya. Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperatur, debu, air, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidakefisienan dan sabotase. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

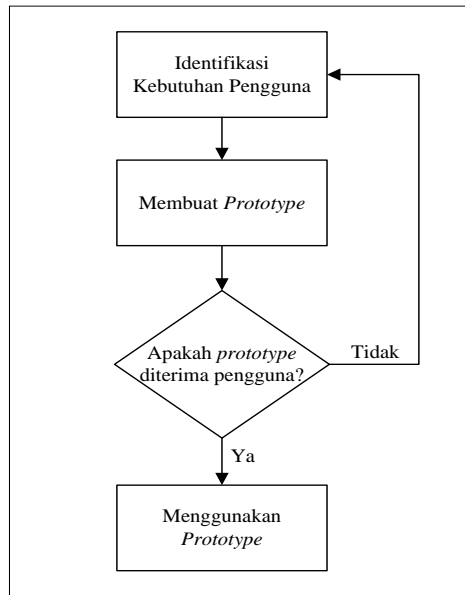
2.5 Metodologi Pengembangan Sistem

Metodologi pengembangan sistem adalah pendekatan formal untuk menerapkan *systems development life cycle* (SDLC). Ada berbagai metodologi pengembangan sistem, dan setiap metodologi memiliki keunikan tersendiri. SDLC memiliki beberapa metode dalam penerapan tahapan prosesnya, salah satunya adalah metode *prototyping*.

Sebuah metodologi berbasis *prototyping* melakukan tahap identifikasi, desain, dan pelaksanaan bersamaan, dan ketiga tahap tersebut dilakukan berulang kali dalam suatu siklus sampai sistem selesai. Prototipe pertama biasanya bagian pertama sistem yang akan digunakan pengguna. Hal ini ditunjukkan kepada pengguna dan sponsor proyek yang akan memberikan komentar, yang digunakan untuk menganalisis, mendesain, dan mengimplementasikan kembali prototipe kedua yang menyediakan fitur yang lebih sedikit. Proses ini berlanjut dalam suatu siklus sampai para analis, pengguna, dan sponsor setuju bahwa prototipe menyediakan fungsionalitas yang cukup untuk diinstal dan digunakan dalam organisasi. Setelah prototipe (sekarang disebut sistem) terinstal, perbaikan terus dilakukan sampai diterima sebagai sistem baru. (Mc.Leod, 2008). Ada tiga jenis metode *prototype* yang dikembangkan, yaitu:

1. *Prototype evolutionary*, yaitu *prototype* yang terus menerus disempurnakan sampai memiliki seluruh fungsionalitas yang dibutuhkan pengguna dari sistem yang baru. (Mc.Leod, 2008)

Beberapa langkah dalam metode pengembangan *prototype evolutionary* digambarkan seperti berikut ini:



Gambar II.5 Langkah Pengembangan *Prototype Evolutionary*
(Sumber: Mc.Leod, 2008)

2. *Prototype* persyaratan, yaitu *prototype* yang dikembangkan sebagai salah satu cara untuk mendefinisikan persyaratan-persyaratan fungsional dari sistem baru ketika pengguna tidak mampu mengungkapkan dengan jelas apa yang mereka inginkan. (McLeod, 2008)
3. *Throwaway Prototyping*
Metodologi *throwaway prototyping* dilakukan pada titik yang berbeda dalam SDLC. Metodologi *throwaway prototyping* ini memiliki tahap analisis yang relatif menyeluruh, yang digunakan untuk mengumpulkan informasi dan mengembangkan ide-ide untuk konsep sistem. Namun, banyak fitur yang disarankan oleh pengguna mungkin tidak dipahami dengan baik, dan hal ini menjadi tantangan teknis untuk dipecahkan. Setiap masalah ini diperiksa oleh analis, perancang, dan pembuat desain *prototype*. (Mc.Leod, 2008)

Dengan metode *prototyping* ini pengembang dan pengguna dapat saling berinteraksi selama proses pembuatan sistem. Sering terjadi, seorang pengguna hanya mendefinisikan secara umum apa yang dikehendakinya tanpa menyebutkan secara detail *output* apa saja yang dibutuhkan, pemrosesan dan data-data apa saja yang dibutuhkan. (Mc. Leod, 2008)

2.6 Konsep Dasar Gudang

Gudang merupakan area yang berfungsi menyimpan barang untuk produksi atau hasil produksi dalam jumlah dan rentan waktu yang kemudian didistribusikan ke lokasi yang dituju berdasarkan permintaan. Area gudang dalam praktek nyata dapat terbagi menjadi 3 macam bagian seperti area gudang bahan baku, area gudang hasil produksi dan area gudang bahan penunjang produksi (Warman, 2004).

2.6.1 Tipe dan Jenis Gudang

Gudang adalah jenis yang paling umum dari tempat penyimpanan meskipun memang ada bentuk-bentuk lain misalnya, tangki penyimpanan. Beberapa gudang memiliki bangunan besar dan luas sehingga memungkinkan kegiatan pembongkaran barang dari truk pemasok dan kegiatan memuat barang ke pelanggan secara bersamaan (Warman, 2004). Berikut ini adalah tipe-tipe dari gudang:

a. Gudang Pribadi/ Swasta

Jenis gudang ini dimiliki dan dioperasikan oleh pemasok dan *reseller* untuk digunakan dalam kegiatan distribusi mereka sendiri. Sebagai contoh jaringan ritel besar menyediakan gudang untuk toko mereka atau grosir mengoperasikan sebuah gudang dimana ia menerima dan mendistribusikan produk.

b. Gudang Publik/ Umum

Gudang umum pada dasarnya adalah ruang yang dapat disewakan untuk mengatasi kebutuhan distribusi dalam jangka pendek. Pengecer yang memiliki gudang sendiri terkadang mereka mencari ruang penyimpanan tambahan jika kapasitas gudang mereka tidak mencukupi atau jika mereka melakukan pembelian produk dalam jumlah besar dengan alasan tertentu. Sebagai contoh, pengecer bisa memesan tambahan barang untuk memaksimalkan penjualan di toko atau ketika ada harga promosi dari pemasok jika membeli dalam jumlah besar.

c. Gudang Otomatis

Dengan kemajuan teknologi komputer dan robotika banyak gudang sekarang memiliki kemampuan otomatis. Tahapan *otomatisasi* sampai pada pemakaian *conveyor belt* untuk mengangkut barang sehingga memerlukan orang yang lebih sedikit untuk menangani kegiatan penyimpanan.

d. *Climate Controlled Warehouse*

Gudang penyimpanan yang menangani berbagai jenis produk dengan penanganan khusus kondisi seperti *freezer* untuk menyimpan produk beku dan kelembaban lingkungan.

e. *Distribution Centre*

Ada beberapa gudang yang hanya menyimpan produk dalam jangka pendek, jadi pada produk yang diterima dari berbagai pemasok dan segera dikirimkan ke banyak pelanggan.

Misalkan, seperti dengan penanganan untuk *perishable food* besar produk yang masuk di pagi hari dan didistribusikan pada akhir ini.

2.6.2 Manfaat Gudang

Secara garis besar semua jenis gudang memiliki manfaat yang sama. Manfaat pergudangan adalah untuk (Warman, 2004):

1. Terjaganya kualitas dan kuantitas logistik dan peralatan.
2. Tertatanya logistik dan peralatan.
3. Peningkatan pelayanan pendistribusian.
4. Tersedianya data dan informasi yang lebih akurat, aktual dan akuntabel.
5. Kemudian akses dalam pengendalian dan pengawasan.
6. Tertib administrasi.

Untuk mewujudkan efektifitas dan efisiensi dalam pengelolaan persediaan dalam penanggulangan bencana perlu pedoman pergudangan. Aktifitas gudang yang dijalankan dengan baik akan mempengaruhi (Warman, 2004):

1. Sistem *just in time* (JIT).
2. Kecepatan dalam merespon perubahan.
3. Respon yang efisien terhadap permintaan konsumen.
4. Menjaga kelancaran distribusi.
5. Meningkatkan kepuasan konsumen.
6. Meningkatkan keamanan dan keselamatan kerja operator.
7. Melindungi barang dari lingkungan.

2.6.3 Fungsi Manajemen Pergudangan

Adapun fungsi dasar manajemen pergudangan guna mengatur subsistem organisasi pergudangan, yaitu (Warman, 2004):

1. *Receiving*: menerima barang.
2. *Inspection and quality control*.
3. *Repackaging*: pengepakan ulang.
4. *Put away*: menempatkan barang yang baru datang di *storagenya*.
5. *Storage*: menjaga barang hingga diperlukan.
6. *Order picking*: mengambil barang dari gudang jika ada permintaan.
7. *Postponement*: mengepak barang di dalam box untuk memudahkan *material handlingnya*.
8. *Packing and shipping*:

- a. Cek kelengkapan pesanan.
 - b. Siapkan *container* untuk *shipping*.
 - c. Siapkan dokumen yang diperlukan.
 - d. Timbang.
 - e. Muat kedalam truk.
9. *Cross docking*: barang yang diterima langsung dikirimkan pada saat yang sama.
10. *Replenishing*: pemesanan kembali jika stok sudah habis.

2.6.4 Operasi-operasi Pergudangan

Pergudangan terdapat tiga fungsi yaitu perpindahan (*movement*), penyimpanan (*storage*) dan transfer informasi (*information transfer*) (Yunarto, 2005).

1. Perpindahan (*Movement*)

Fungsi *movement* ini merupakan fungsi utama, salah satu kegiatannya adalah memperbaiki perputaran *inventory* dan mempercepat proses pemesanan dari produksi hingga pengiriman utama. Menurut Yunarto (2005), fungsi *movement* dibagi menjadi aktivitas-aktivitas yang meliputi:

a. Penerimaan

Merupakan aktifitas penerimaan barang dimana di dalamnya terdapat aktivitas-aktivitas seperti pembongkaran muatan, perhitungan kuantitas yang diterima dan inspeksi kualitas dan kerusakan, juga aktivitas-aktivitas lain yang berkaitan dengan penerimaan barang di gudang.

b. *Putaway*

Merupakan proses pemindahan barang dari dok penerimaan ke gudang penyimpanan.

c. *Customer Order Picking*

Merupakan aktivitas pemindahan barang dari gudang penyimpanan atau dari lokasi *picking* untuk kemudian disiapkan untuk proses pengiriman.

d. *Packing*

Proses *packing* merupakan proses pengepakan barang yang akan dikirim ke konsumen.

e. *Cross Docking*

Proses ini merupakan proses pemindahan barang dari area *receiving* langsung ke lokasi *shipping* tanpa melalui aktivitas penyimpanan di gudang.

f. *Shipping*

Aktivitas ini merupakan aktivitas pengiriman dan meliputi proses pembuatan dokumen barang yang akan dikirim.

2. Penyimpanan (*Storage*)

Storage merupakan aktivitas penyimpanan barang baik yang merupakan barang bahan baku ataupun barang hasil produksi. Gudang *finished good* dan *sparepart* dapat menjadi satu atau dapat dipisahkan.

3. Transfer Informasi (*Information Transfer*)

Aktivitas ini adalah aktivitas transfer informasi seperti informasi stok barang yang ada di gudang atau informasi-informasi lain yang berguna, informasi ini dapat merupakan informasi untuk pihak di luar gudang atau pihak gudang itu sendiri.

2.6.5 Mekanisme Dasar Sistem Gudang

Mekanisme dasar sistem persediaan dalam gudang meliputi proses sebagai berikut (Yunarto dan Santika, 2005):

1. Penerimaan

Penerimaan merupakan proses penyerahan dan penerimaan persediaan di gudang. Dalam proses penyerahan dan penerimaan ini dilakukan:

- a. Pendataan jumlah dan mutu barang persediaan harus sesuai dengan ketentuan yang berlaku/ layak untuk diberikan kepada konsumen.
- b. Pencatatan administratif sebagai dokumen yang dapat dipertanggungjawabkan oleh petugas yang bersangkutan.

2. Penyimpanan

Penyimpanan merupakan proses kegiatan penyimpanan persediaan di gudang dengan cara menempatkan barang persediaan yang diterima:

- a. Penempatan sesuai dengan denah.
- b. Aman dari pencurian.
- c. Aman dari gangguan fisik.
- d. Aman dari pencemaran secara kimiawi dan biologi yang dapat merusak kualitas dan kuantitas.
- e. Aman dari kebakaran.
- f. Penataan sesuai dengan standar pergudangan.

3. Pemeliharaan

Pemeliharaan merupakan kegiatan perawatan barang persediaan agar kondisi tetap terjamin dan siap pakai untuk dipergunakan dalam penanggulangan bencana secara efektif, efisien dan akuntabel melalui prinsip:

- a. 5R = Ringkas, Rapih, Resik (bersih), Rawat, Rajin.
- b. Logistik dan peralatan disusun di atas *pallet* secara rapih dan teratur sesuai dengan ketentuan.

4. Pendistribusian

Pendistribusian merupakan proses kegiatan pengeluaran dan penyaluran persediaan dari gudang untuk diserahkan kepada yang *user* atau pengguna, melalui suatu proses serah terima yang dapat dipertanggungjawabkan. Hal ini dilakukan berdasarkan permintaan sesuai dengan kebutuhan produksi.

5. Pengendalian

Pengendalian merupakan proses kegiatan pengawasan atas pergerakan masuk keluarnya persediaan dan penempatan dapat diketahui dengan menggunakan formulir dalam lampiran.

6. Penghapusan

Penghapusan merupakan rangkaian kegiatan pemusnahan barang persediaan dalam rangka pembebasan barang milik/ kekayaan negara dari tanggung jawab berdasarkan peraturan perundang-undangan yang berlaku. Tujuan penghapusan adalah sebagai berikut:

- a. Penghapusan merupakan bentuk pertanggungjawaban administrasi petugas terhadap persediaan yang dikelola, yang sudah ditetapkan untuk dihapuskan/ dimusnahkan sesuai dengan ketentuan yang berlaku.
- b. Menghindari pembiayaan (biaya penyimpanan, pemeliharaan, penjagaan dan lain-lain) atau barang yang sudah tidak layak untuk dipelihara.
- c. Menjaga keselamatan agar terhindar dari pencemaran lingkungan.

Kegiatan penghapusan adalah sebagai berikut:

- a. Membuat daftar barang persediaan yang akan dihapuskan beserta alasan-alasannya.
- b. Pisahan barang persediaan yang kadaluarsa/ rusak pada tempat tertentu sampai pelaksanaan pemusnahan.
- c. Melaporkan kepada atasan mengenai barang persediaan yang akan dihapuskan.
- d. Membentuk panitia pencelaan dan penghapusan barang persediaan melalui surat keputusan dari pejabat yang berwenang.
- e. Membuat berita acara hasil pencelaan dan penghapusan barang persediaan yang akan dihapuskan.
- f. Melaporkan hasil pencelaan dan penghapusan kepada pejabat yang berwenang.

- g. Melaksanakan penghapusan dan pemusnahan setelah ada keputusan dari pejabat yang berwenang.

2.6.6 Metode dalam Penyimpanan Gudang

Pengolahan barang di gudang memiliki metode tertentu guna meminimalisir biaya penyimpanan tapi juga seimbang dengan biaya pemesanan. Berikut merupakan metode yang digunakan (Yunarto dan Santika, 2005):

- a. Metode FIFO (*First In First Out*), yaitu sistem pengeluaran barang yang terlebih dahulu masuk ke gudang, didahulukan keluar. Biasanya sistem seperti ini banyak dipakai di perusahaan manufaktur makanan karena untuk menghindari bahan baku busuk di dalam gudang. Namun tidak menutup kemungkinan digunakan juga untuk perusahaan manufaktur selain makanan, untuk menjaga kualitas bahan baku tidak rusak di dalam gudang. Singkatnya, sistem ini seperti sistem antrian (*queue*).
- b. Metode LIFO (*Last In First Out*), yaitu sistem pengeluaran barang dari gudang dimana barang yang terakhir masuk adalah yang pertama dikeluarkan. Biasanya sistem seperti ini tidak banyak digunakan perusahaan manufaktur, namun banyak digunakan di perusahaan dagang yang mengutamakan kesegaran barang jual. Singkatnya sistem ini seperti sistem tumpukan (*stack*).
- c. Metode FEFO (*First Expired Date First Out*), yaitu barang persediaan yang pertama kadaluarsa harus yang pertama keluar untuk didistribusikan. Dalam penyusunan persediaan yang masa kadaluarsanya lebih awal atau yang diterima lebih awal biasanya juga diproduksi lebih awal dan umurnya relatif lebih tahan masa kadaluarsanya mungkin lebih awal.

2.6.7 Sistem Informasi Pergudangan

Dalam banyak perusahaan, pemecahan terhadap masalah-masalah sangat tergantung pada ingatan dari para spesialis gudang yang ada. Informasi yang bersangkutan ada dalam laci-laci arsip, tetapi proses pencarian sangat terbatas. Pencarian arsip adalah pencarian yang terpisah-pisah. Jika suatu inventarisasi tentang keahlian tersedia dalam suatu bank memori komputer, semua karyawan yang memenuhi syarat paling tidak akan menjadi perhatian manajer personalia (Warman, 2004).

Sistem informasi pergudangan adalah keseluruhan proses dalam penghimpun, mencatat, menyimpan, menganalisis, dan melengkapi atau menghimpun kembali data yang berkenaan dengan gudang untuk melayani pihak yang berwenang menggunakannya dalam upaya mewujudkan eksistensi organisasi yang kompetitif (Warman, 2004).

Dari fungsi sistem informasi pergudangan terdapat di dalam pengertiannya seperti tersebut diatas, dapat ditelusuri dan diidentifikasi unsur-unsurnya. Unsur-unsur sistem informasi gudang adalah sebagai berikut :

- a. Pengelola informasi, yang menjalankan fungsi penghimpun, mencatat, menyimpan, menganalisis dan melengkapi atau menghimpun kembali informasi.
- b. Sumber informasi, semua proses yang terkait dengan keluar dan masuknya barang.
- c. Pemakaian informasi (para manajer sesuai divisi atau departemen dan wewenangnya masing-masing)
- d. Saluran (*channel*), seluruh saluran kegiatan pengadaan barang secara efektif dan efisien.
- e. Informasi yang berkenaan dengan kecukupan, ketepatan dan keakuratannya.

Pada dasarnya sistem informasi pergudangan untuk kepentingan manajemen gudang yang efektif dapat menghimpun segala sesuatu informasi mengenai pergudangan dalam hubungan pekerjaannya, tidak saja kondisinya yang sekarang tetapi juga sebelumnya. Untuk itu beberapa diantaranya akan dikemukakan dengan catatan bahwa informasi atau data itu masih selalu dapat dikembangkan sesuai dengan keperluan organisasi atau perusahaan. (Warman, 2004).

2.7 Proses *Trimming*

Proses *trimming* (proses produksi) berfungsi untuk memasang dan merakit komponen komponen kendaraan (produk) menjadi produk CBU (*Completely Built Up*). Pada proses *trimming* ini terdapat 3 step yang akan dilalui untuk sampai tahap *Quality Control* (QC) yaitu (PT. Krama Yudha Ratu Motor, 2017) :

- *Trimming On*

Dimana pada proses ini kabin baru turun dari dari proses welding dan painting atau sering disebut orang-orang yang ada di pabrik ini kap dan bak mobil menyatu.

- *Hi Line*

Dimana pada proses ini adalah pemasangan *engine*, transmisi dan roda-roda pada mobil itu dipasang dan tingkat keamanan pun lebih ditingkatkan pada fase ini seperti penambahan APD ataupun *special tools* untuk menunjang pemasangan pada *part* yang akan dipasang.

- *Trimming Off*

Dimana proses ini adalah pemasangan yang berhubungan dengan *part-part electrical* seperti pemasangan aki mobil dan lampu-lampu pada mobil.

2.8 *Flowmap dan Flowchart*

Flowmap adalah diagram yang menunjukkan aliran data berupa formulir-formulir ataupun keterangan berupa dokumentasi yang mengalir atau beredar dalam suatu sistem (Jogiyanto, 2005). *Flowchart* adalah bagan yang menunjukkan alir di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Notasi yang digunakan dalam suatu *flowmap* merupakan penggabungan notasi *flowchart* program.

Pada waktu akan menggambar suatu bagan alir, analisis sistem atau pemrogram dapat mengikuti pedoman-pedoman sebagai berikut:

- a. Bagan alir sebaiknya di gambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
- b. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.
- c. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhirnya.
- d. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan, misal: "Persiapkan" dokumen, "Hitung" gaji.
- e. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
- f. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
- g. Gunakan simbol-simbol bagan alir yang standar

2.8.1 Jenis-Jenis *Flowchart*





Terdapat beberapa jenis bagan alir (*flowchart*), yaitu sebagai berikut (Jogiyanto, 2005):

1. Bagan alir sistem (*system flowchart*).
Bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem yang menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem.
2. Bagan alir dokumen (*document flowchart*).
Flowchart yang menunjukkan arus dari laporan dan formulir.
3. Bagan alir skematik (*schematic flowchart*).
Bagan yang menggambarkan prosedur di dalam sistem. *Flowchart* ini menggunakan gambar-gambar komputer dan peralatan lainnya dalam pembuatan *flowchart* dengan tujuan untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol *flowchart*.
4. Bagan alir program (*program flowchart*).
Bagan yang menjelaskan secara rinci langkah-langkah dari proses program dan dibuat atas dasar *flowchart* sistem
5. Bagan alir proses (*process flowchart*).
Bagan alir yang banyak digunakan di teknik industri dan berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur


2.8.2 Simbol *Flowchart*


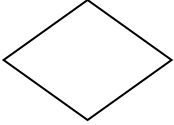
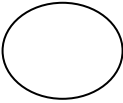
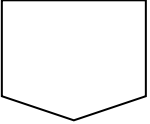
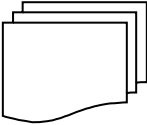
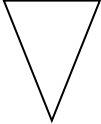
Suatu *flowchart* memberi gambaran dua dimensi berupa simbol-simbol grafis. Masing-masing simbol memiliki fungsi dan arti tersendiri. Berikut ini merupakan simbol *flowchart* menurut Jogiyanto (2005) :

Tabel II.1 Simbol *Flowchart*

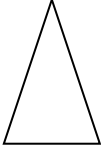
Simbol	Nama	Fungsi
	Terminator	Awal dan akhir dari suatu proses.
	Garis Akhir (<i>Front Line</i>)	Arus dari suatu proses
	<i>Preparation</i>	Proses inisialisasi awal
	Proses	Proses pengolahan data

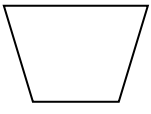
Tabel II.1 Simbol *Flowchart* (Lanjutan)

Simbol	Nama	Fungsi
	<i>Input/Output Data</i>	Mewakili data masukan atau keluaran

	<i>Predefined Process (Sub Proses)</i>	Permulaan sub proses
	<i>Decision</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	<i>On Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman
	<i>Off Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda
	Dokumen Rangkap	Menggambarkan dokumen asli dan tembusannya
	Arsip Sementara	Menunjukkan tempat penyimpanan dokumen

Tabel II.1 Simbol *Flowchart* (Lanjutan)

Simbol	Nama	Fungsi
	Arsip Permanen	Menunjukkan tempat penyimpanan dokumen secara permanen yang tidak akan diproses lagi

	Proses Manual	Untuk menggambarkan kegiatan manual seperti : menerima order, mengisi formulir, membandingkan dll
---	---------------	---

Sumber: (Jogiyanto, 2005)

2.9 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. UML merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri piranti lunak dan pengembangan sistem. (Rama dan Jones, 2008). Didalam UML terdapat beberapa diagram yakni (Munawar, 2005):


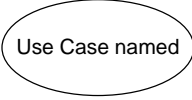

1. Use Case Diagram

Use case 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).

Use case menampilkan *actor*, *use case* dan hubungan antara mereka. use case diagram dapat sangat membantu bila kita sedang menyusun kebutuhan sistem, mengkomunikasikan rancangan dengan klien dan merancang tes khusus untuk semua desain yang ada pada sistem. Berikut dibawah ini merupakan notasi-notasi *use case* (Munawar, 2005) :

Tabel II.2 Notasi- Notasi *Use Case Diagram*

Elemen	Nama Elemen	Fungsi Elemen
	<i>Actor</i>	<i>Actor</i> adalah pengguna sistem. <i>Actor</i> tidak terbatas hanya manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan <i>input</i> atau



 Mahasiswa		memberikan <i>output</i> , maka aplikasi tersebut juga bisa dianggap sebagai <i>actor</i> .
	<i>Use case</i>	<i>Use Case</i> digambarkan sebagai lingkaran elips dengan nama <i>use case</i> dituliskan didalam elips tersebut.
	<i>Association</i>	<i>Association</i> digunakan untuk Menghubungkan actor dengan <i>use case</i> . Asosiasi digambarkan dengan sebuah garis yang menghubungkan antara <i>Actor</i> dengan <i>Use Case</i> .


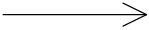
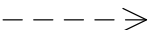



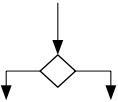
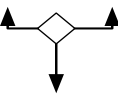
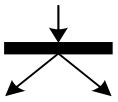
Sumber: (Munawar, 2005)

2. Activity Diagram

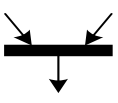
Menurut Munawar (2005) *Activity diagram* menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, mulai dari titik awal, melalui kondisi (*decision*) sampai pada titik akhir. Diagram ini mampu menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Diagram ini tidak menggambarkan perilaku atau proses internal sebuah sistem maupun interaksi antar subsistem, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas secara umum. Diagram ini dilambangkan dengan segi empat dengan sudut membulat (lebih bulat dari *statechart*) untuk menggambarkan aktivitas, dan memiliki kondisi yang digunakan untuk mengambil keputusan atau percabangan. Berikut ini adalah elemen-elemen yang dapat digunakan pada saat pembuatan *Activity Diagram*:

Tabel II.3 Notasi-notasi *Activity Diagram*

Elemen	Nama Elemen	Fungsi Elemen
	<i>Action</i>	Untuk menggambarkan perilaku yang sederhana dan bersifat <i>non decomposable</i> .
	<i>Activity</i>	Untuk mewakili serangkaian aksi (<i>action</i>).

	<i>Object Node</i>	Untuk mewakili objek yang terhubung dengan serangkaian <i>object flow</i> .
	<i>Control Flow</i>	Untuk mewakili serangkaian pelaksanaan
	<i>Object Flow</i>	Untuk menunjukkan aliran sebuah objek dari sebuah aktivitas (aksi), atau kesebuah aktivitas (aksi).
	<i>Initial Node</i>	Untuk menandakan awal dari serangkaian aksi atau aktivitas.
	<i>Final-Activity Node</i>	Untuk menghentikan seluruh <i>control flows</i> atau <i>object flows</i> pada sebuah aktivitas (aksi).
	<i>Final-Flow Node</i>	Untuk menghentikan <i>control flow</i> atau <i>object flow</i> tertentu.
	<i>Decision Node</i>	Untuk mewakili suatu kondisi pengujian, yang bertujuan untuk memastikan bahwa <i>control flow</i> atau <i>object flow</i> hanya menuju ke satu arah.
	<i>Merge Node</i>	Untuk menyatukan kembali <i>decision path</i> yang dibuat dengan menggunakan <i>decision node</i> .
	<i>Fork Node</i>	Untuk memisahkan perilaku menjadi serangkaian aktivitas yang berjalan secara paralel atau bersamaan.

Tabel II.3 Notasi-notasi *Activity Diagram* (Lanjutan)

Elemen	Nama Elemen	Fungsi Elemen
	<i>Join Node</i>	Untuk menyatukan kembali serangkaian aktivitas yang berjalan secara paralel atau bersamaan.

Swimlane 1	Swimlane 2	Swimlane	Untuk membagi sebuah <i>activity diagram</i> menjadi beberapa kolom, dengan tujuan menempatkan aktivitas (aksi) tertentu pada individu (objek) yang bertanggung jawab atas terlaksananya aktivitas (aksi) tersebut.
------------	------------	----------	---

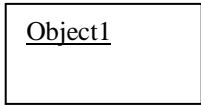
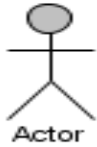
Sumber: (Munawar, 2005)

3. *Sequence Diagram*

Menurut Munawar (2005) *Sequence diagram* menggambarkan interaksi antar objek di dalam maupun di luar sistem berupa *message* yang digambarkan terhadap waktu. Terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Bisa digunakan untuk menggambarkan skenario atau rangkaian langkah langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan *output*. Diagram ini menunjukkan sejumlah contoh objek dan pesan yang diletakkan diantara objek-objek ini di dalam *use case*.



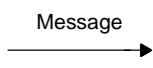
Berikut ini adalah notasi-notasi yang digunakan pada *Sequence Diagram* :

Tabel II.4 Notasi-notasi *Sequence Diagram*

Simbol	Nama	Keterangan
	<i>Object</i>	<i>Object</i> merupakan instance dari sebuah <i>class</i> dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah <i>class</i> (kotak) dengan nama obyek didalamnya yang diawali dengan sebuah titik koma.
	<i>Actor</i> .	<i>Actor</i> juga dapat berkomunikasi dengan <i>object</i> , maka <i>actor</i> juga dapat diurutkan sebagai kolom. Simbol <i>Actor</i> sama dengan simbol pada <i>Actor Use Case Diagram</i>

Tabel II.4 Notasi-notasi *Sequence Diagram* (Lanjutan)

Simbol	Nama	Keterangan
--------	------	------------

	<i>Lifeline</i>	<i>Lifeline</i> mengindikasikan keberadaan sebuah object dalam basis waktu. Notasi untuk <i>Lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah obyek.
	<i>Activation</i>	<i>Activation</i> dinotasikan sebagai sebuah kotak segiempat yang digambar pada sebuah <i>lifeline</i> . <i>Activation</i> mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.
	<i>Message</i>	<i>Message</i> digambarkan dengan anak panah horizontal antara <i>Activation</i> . <i>Message</i> mengindikasikan komunikasi antara <i>object-object</i> .

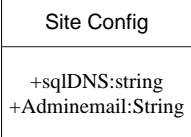
Sumber: (Munawar, 2005)

4. Class Diagram

Menurut Munawar (2005) *Class Diagram* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi obyek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).





Berikut adalah notasi-notasi yang ada pada *class diagram*:

Tabel II.5 Notasi-notasi *Class Diagram*

Simbol	Nama	Keterangan
	<i>Class</i>	<i>Class</i> adalah blok-blok pembangun pada pemrograman berorientasi obyek. Sebuah <i>class</i> digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari <i>class</i> . Bagian tengah mendefinisikan <i>property</i> /atribut <i>class</i> . Bagian akhir mendefinisikan <i>method-method</i> dari sebuah <i>class</i> .

Tabel II.5 Notasi-notasi *Class Diagram* (Lanjutan)

Simbol	Nama	Keterangan
--------	------	------------

<u>1..n Owned by 1</u>	<i>Assosiation</i>	Sebuah asosiasi merupakan sebuah <i>relationship</i> paling umum antara 2 <i>class</i> , dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 <i>class</i> . Garis ini bisa melambangkan tipe-tipe <i>relationship</i> dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah <i>relationship</i> (Contoh: <i>One-to-one</i> , <i>one-to-many</i> , <i>many-to-many</i>).
	<i>Composition</i>	Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>Composition</i> terhadap <i>class</i> tempat dia bergantung tersebut. Sebuah <i>relationship composition</i> digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi/solid.
	<i>Dependency</i>	Kadangkala sebuah <i>class</i> menggunakan <i>class</i> yang lain. Hal ini disebut <i>dependency</i> . Umumnya penggunaan <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain. Sebuah <i>dependency</i> dilambangkan sebagai sebuah panah bertitik titik.
	<i>Aggregation</i>	<i>Aggregation</i> mengindikasikan keseluruhan bagian <i>relationship</i> dan biasanya disebut sebagai relasi “mempunyai sebuah” atau “bagian dari”. Sebuah <i>aggregation</i> digambarkan sebagai sebuah garis dengan sebuah jajaran genjang yang tidak berisi/tidak solid.
	<i>Generalization</i>	Sebuah relasi <i>generalization</i> sepadan dengan sebuah relasi <i>inheritance</i> pada konsep berorientasi <i>obyek</i> . Sebuah <i>generalization</i> dilambangkan dengan sebuah panah dengan kepala panah yang tidak solid yang mengarah ke kelas “ <i>parent</i> ”-nya/induknya.

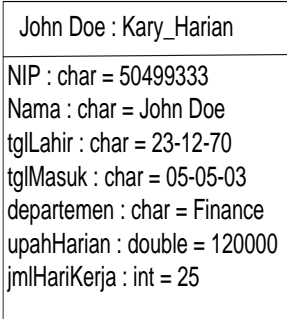

Sumber: (Munawar 2005)

5. *Object Diagram*

Menurut Munawar (2005) *Object diagram* adalah diagram yang memberikan gambaran model *instance-instance* dari sebuah *class*. Diagram ini digunakan untuk menggambarkan sebuah sistem pada sebuah sudut pandang waktu tertentu. Dengan menggunakan diagram ini anda dapat memeriksa keabsahan kelas-kelas diagram berikut aturan-aturan multiplisitasnya dengan “*real data*”

dan mengujinya dengan scenario-skenario tertentu. Notasi diagramnya dapat dilihat pada table II.6.

Tabel II.6 Notasi-notasi *Object Diagram*

Simbol	Nama	Keterangan
	<i>Obyek-obyek</i>	<p><i>Obyek-obyek</i> diidentifikasi dengan cara meletakkan nama <i>instance</i>-nya kemudian diikuti oleh tanda titik dua didepan nama <i>class</i>-nya.</p> <p>Nilai properti atau atribut dituliskan berpasangan seperti "nama_atribut=nilai".</p> <p>Sedangkan notasi sebuah obyek digambarkan segi empat yang terbagi atas 2 bagian.</p>
	<i>Association</i>	<p><i>Object diagram</i> juga dapat mengandung asosiasi. Biasanya constraint, detail relationship, multiplisitas yang ada di class diagram tidak disertakan dalam <i>object diagram</i> sebagai upaya memfokuskan perhatian hanya terhadap obyek dan property/atributnya. Asosiasi antar 2 obyek biasanya dinotasikan dengan sebuah garis yang menghubungkan kedua obyek.</p>

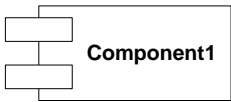

Sumber: (Munawar 2005)

6. *Component Diagram*

Menurut Munawar (2005) *Component diagram* mengandung komponen, *interface* dan *relationship*. Komponen diagram ini digunakan pada saat anda ingin memecah sistem menjadi komponen-komponen dan ingin menampilkan hubungan-hubungan mereka dengan antarmuka atau pemecahan komponen menjadi struktur yang lebih rendah. Berikut adalah notasi dari *component diagram*:

Tabel II.7 Notasi-notasi *Component Diagram*

Simbol	Nama	Keterangan
--------	------	------------

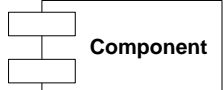
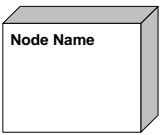

	<i>Component</i>	Sebuah komponen melambangkan sebuah entitas software dalam sebuah sistem. Sebuah komponen dinotasikan sebagai sebuah kotak segiempat dengan dua kotak kecil tambahan yang menempel disebelah kirinya.
	<i>Dependency</i>	Sebuah <i>dependency</i> digunakan untuk menotasikan relasi antara dua komponen. Notasinya adalah tanda panah putus-putus yang diarahkan kepada komponen tempat sebuah komponen itu bergantung.

Sumber: (Munawar 2005)

7. Deployment Diagram

Menurut Munawar (2005) *Deployment Diagram* ialah terdiri dari node-node di mana setiap node diwakili oleh sebuah kubus dan garis yang menghubungkan antara kubus tersebut menunjukkan hubungan antara kedua node tersebut. Berikut ini adalah elemen-elemen yang digunakan pada *Deployment Diagram*:

Tabel II.8 Notasi-notasi *Deployment Diagram*

Simbol	Nama	Keterangan
	<i>Component</i>	Pada <i>deployment</i> diagram, komponen komponen yang ada diletakkan didalam <i>node</i> untuk memastikan keberadaan posisi mereka.
	<i>Node</i>	<i>Node</i> menggambarkan bagian-bagian <i>hardware</i> dalam sebuah sistem. Notasi untuk <i>node</i> digambarkan sebagai sebuah kubus 3 dimensi.
	<i>Association</i>	Sebuah <i>association</i> digambarkan sebagai sebuah garis yang menghubungkan dua <i>node</i> yang mengindikasikan jalur komunikasi antara elemen-elemen <i>hardware</i> .

Sumber: (Munawar, 2005)

2.10 Sistem Manajemen Basis Data *Relational* (RDBMS)

Sebuah sistem manajemen basis data relational atau dalam bahasa inggrisnya dikenal sebagai *relational database management system* (RDBMS) adalah sebuah program komputer (secara lebih tipikal adalah seperangkat komputer) yang dirancang untuk mengatur/memanajemen sebuah basis data sebagai sekumpulan data yang disimpan secara terstruktur, dan melakukan operasi-operasi data atas permintaan penggunaanya (id.wikipedia.org).

Suatu *database* terdiri dari banyak tabel dan tabel ini terdiri dari banyak *field* yang merupakan kolomnya, isi tiap baris dari tabel inilah merupakan data. Sedangkan untuk membuat data saling terhubung dan terintegrasi, maka satu tabel dengan tabel yang lainnya harus memiliki hubungan antara satu dengan yang lainnya. Setiap tabel mempunyai sebuah *primary key*, lalu *primary key* ini dihubungkan dengan tabel kedua dan menjadi *foreign key* di tabel kedua ini.

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. Berikut adalah contoh penulisan kamus data:

Spesifikasi tabel pemasok

Nama tabel : Pemasok

Tipe : File master

Tabel II.9 Contoh Kamus Data Untuk Tabel Pemasok

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

Sumber: (Jogiyanto, 2005)

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

HIPO (*Hierarchy plus Input-Process-Output*) merupakan metodologi yang dikembangkan dan didukung oleh IBM. HIPO sebenarnya adalah alat dokumentasi program. Akan tetapi sekarang HIPO juga banyak digunakan sebagai alat desain dan teknik dokumentasi dalam

siklus pengembangan sistem. HIPO berbasis pada fungsi, yaitu tiap-tiap modul di dalam sistem digambarkan oleh fungsi utamanya. (Jogiyanto, 2005)

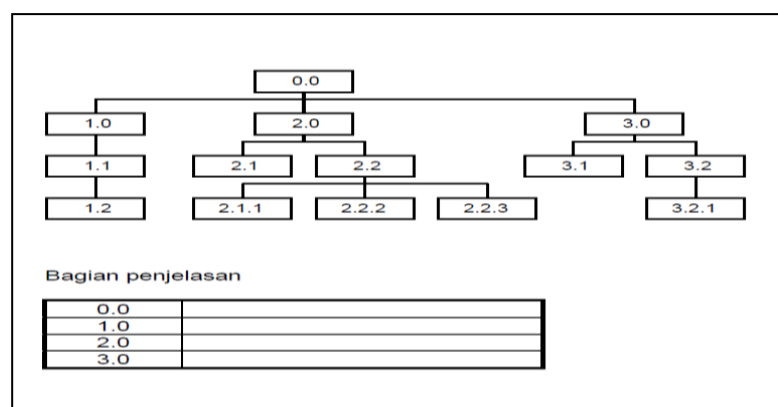
HIPO dapat digunakan sebagai alat pengembangan sistem dan teknik dokumentasi program dan penggunaan HIPO ini mempunyai sasaran utama sebagai berikut:

1. Untuk menyediakan suatu struktur guna memahami fungsi-fungsi dari sistem.
2. Untuk lebih menekankan fungsi-fungsi yang harus diselesaikan oleh program, bukan menunjukkan pernyataan-pernyataan program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan dari *input* yang harus digunakan dan *output* yang harus dihasilkan oleh masing-masing fungsi pada tiap-tiap tingkatan.
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. Daftar Isi Visual/ *Visual Tabel of Contents* (VTOC)

Visual tabel of contents menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur. Pada diagram ini nama dan nomor dari program HIPO diidentifikasi. Struktur paket diagram dan hubungan fungsi juga diidentifikasi dalam bentuk hirarki. Keterangan masing-masing fungsi diberikan pada bagian penjelasan yang diikutsertakan dalam diagram ini. *Visual tabel of contents* ini dapat digambarkan sebagai berikut (Jogiyanto, 2005):

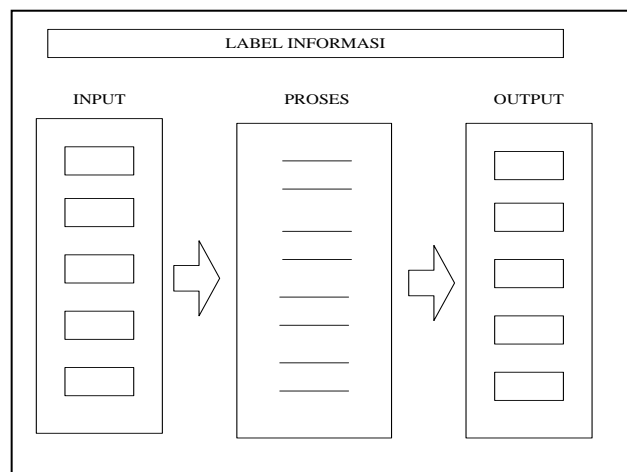


Gambar II.6 *Visual Tabel of Contents*
(Sumber: Jogiyanto, 2005)

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. *Overview diagram* ini dapat digambarkan sebagai berikut:



Gambar II.7 *Overview Diagram*
(Sumber: Jogiyanto, 2005)

3. *Detail Diagram*

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

.1 2.13 *PHP*

Menurut Anhar (2010) PHP singkatan dari PHP: *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada *server (server side HTML embedded scripting)*. PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru atau *up to date*. Semua *script* PHP dieksekusi pada *server* dimana *script* tersebut dijalankan.

PHP merupakan sebuah bahasa pemrograman *web* yang memiliki sintak atau aturan dalam menuliskan *script* atau kode-kodenya. Salah satu contoh penulisan kode PHP dapat dilihat sebagai berikut ini:

```
<?php
    echo ("Tes Pakai PHP");
?>
```

Kode-kode PHP memiliki tata aturan, yaitu diawali dengan tanda `<?php` dan diakhiri dengan tanda `?>`. Tiap akhir baris harus selalu diber tanda titik koma (;). PHP bersifat *case sensitive*, artinya penulisan huruf besar dan kecil pada kode PHP sangat berpengaruh.(Anhar, 2010)

Menurut Anhar (2010) beberapa kelebihan PHP dari bahasa pemrograman lain adalah sebagai berikut:

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai *apache*, *IIS*, *Lighttpd*, hingga *Xitami* dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin seperti *Linux*, *Unix*, *Macintosh*, dan *Windows* serta dapat dijalankan secara *runtime* melalui *console* dan dapat menjalankan perintah-perintah sistem.

2.14 Codeigniter Framework

Codeigniter merupakan *framework* PHP yang diklaim memiliki eksekusi tercepat dibandingkan dengan *framework* lainnya. *Codeigniter* bersifat *open source* dan menggunakan model basis MVC (*Model*, *View*, *Controller*), yang merupakan model konsep modern *framework* yang digunakan saat ini (Saputra, 2011).

Framework adalah suatu kerangka kerja dalam aplikasi web yang didalamnya memiliki suatu potongan-potongan program yang disusun (modul), sehingga *programmer* tidak perlu membuat kode dari nol, karena *framework* telah menyediakannya (Saputra, 2011).



Gambar II.8 Logo

Framework

Codeigniter

Sumber: Saputra (2011)

Kelebihan dari *Codeigniter* ialah :

1. PHP adalah bahasa multiplatform yang artinya dapat berjalan di berbagai mesin dan sistem operasi (*Linux, Unix, Macintosh, Windows*) dan dapat dijalankan secara *runtime* melalui console serta juga dapat menjalankan perintah perintah system lainnya.
2. PHP bersifat *Open Source* yang berarti dapat digunakan oleh siapa saja secara gratis.
3. Web server yang mendukung PHP dapat ditemukan di mana-mana dari mulai apache,IIS ,Lighttpd ,nginx, hingga Xitami dengan konfigurasi yang relatif mudah dan tidak berbelit-belit, bahkan yang membuat dalam bentuk paket atau package (PHP, MySQL dan Web Server).
4. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis, komunitas dan *developer* yang siap membantu dalam pengembangan.
5. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.

Sedangkan kekurangan yang dimiliki oleh *Codeigniter* ialah

1. PHP Tidak mengenal Package.
2. Jika tidak di-*encoding*, maka kode PHP dapat dibaca semua orang dan untuk meng-*encoding*-nya dibutuhkan *tool* dari zend yang mahal biayanya.
3. PHP memiliki kelemahan keamanan. jadi programmer harus jeli dan berhati hati dalam melakukan pemograman dan Konfigurasi PHP.

.2 2.15 MySQL

MySQL adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS (*Database Management System*), sifat dari DBMS ini adalah *open source*. MySQL sebenarnya produk yang berjalan pada *platform* Linux, dengan adanya perkembangan dan banyaknya pengguna serta lisensi dari *database* ini adalah *open source*, maka para pengembang kemudian merilis versi Windows. (Nugroho, 2007)

SQL adalah suatu bahasa permintaan yang terstruktur dan telah distandarkan untuk semua program pengakses *database* seperti Oracle, PosgreSQL, SQL Server, dan lain-lain. (Kadir, 2008)

Pada MySQL masing-masing tipe data memiliki kegunaan dan keterangan untuk digunakan dalam merancang tabel. Berikut ini beberapa tipe data yang terdapat pada MySQL

Tabel II.10 Beberapa Tipe Data Pada MySQL

Tipe Data	Ukuran	Keterangan
-----------	--------	------------

CHAR	M	Menampung maksimal M karakter (kombinasi huruf, angka, dan simbol-simbol). Jumlah memori yang dibutuhkan selalu M byte. M terbesar adalah 255.
VARCHAR	M	Karakter yang disimpan maksimal M karakter. Jumlah memori yang dibutuhkan tergantung jumlah karakter. M bisa mencapai 65535.
DATE	8 byte	Menyatakan tanggal.
TIME	8 byte	Menyatakan waktu (jam:menit:detik).
TINYINT	1 byte	Bilangan antara -128 sampai dengan +127.
SMALLINT	2 byte	Bilangan antara -32768 sampai dengan +32768.
INT	4 byte	Bilangan antara -2147683648 sampai dengan +2147683647.
FLOAT	4 byte	Bilangan pecahan.
DOUBLE	8 byte	Bilangan pecahan dengan presisi tinggi.
BOOL	1 byte	Untuk menampung nilai <i>true</i> (benar) dan <i>false</i> (salah). Identik dengan TINYINT.
ENUM	-	Menyatakan suatu tipe yang nilainya tertentu (disebutkan dalam pendefinisian).
TEXT	-	Menyimpan teks yang ukurannya sangat panjang.

Tabel II.10 Beberapa Tipe Data Pada MySQL (Lanjutan)

Tipe Data	Ukuran	Keterangan
-----------	--------	------------

BLOB	-	Untuk menyimpan data biner (misalnya gambar atau suara).
------	---	--

(Sumber: Kadir, 2008)

2.15 XAMPP

XAMPP adalah sebuah software yang berfungsi untuk menjalankan website berbasis PHP dan menggunakan pengolahan data MySQL di komputer lokal. XAMPP merupakan paket PHP yang berbasis open source yang dikembangkan oleh sebuah komunitas open source. XAMPP berperan sebagai web server pada komputer (Nugroho, 2008). XAMPP merupakan singkatan dari masing-masing hurufnya. Berikut ini merupakan penjelasan tentang singkatan dari XAMPP (Wikipedia, 2017) :

- X
Program ini dapat dijalankan dibanyak sistem operasi, seperti [Windows](#), [Linux](#), [Mac OS](#), dan [Solaris](#).
- A
Apache, merupakan aplikasi [web server](#). Tugas utama *apache* adalah menghasilkan halaman *web* yang benar kepada *user* berdasarkan [kode](#) PHP yang dituliskan oleh pembuat halaman *web*. jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu [database](#) diakses terlebih dahulu (misalnya dalam [MySQL](#)) untuk mendukung halaman web yang dihasilkan.
- M
MySQL, merupakan aplikasi *database server*. Perkembangannya disebut SQL yang merupakan kepanjangan dari *Structured Query Language*. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah *database*. MySQL dapat digunakan untuk membuat dan mengelola *database* beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah, dan menghapus data yang berada dalam *database*.
- P
PHP, bahasa pemrograman *web*. Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat *web* yang bersifat *server-side scripting*.

PHP memungkinkan kita untuk membuat halaman web yang bersifat dinamis. [Sistem manajemen basis data](#) yang sering digunakan bersama PHP adalah MySQL. PHP juga mendukung sistem manajemen *database* [Oracle](#), [Microsoft Access](#), [Interbase](#), [d-base](#), [PostgreSQL](#), dan sebagainya

- P

Perl, bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin *unix*. *Perl* dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya *Perl* 1. Pada versi-versi selanjutnya, *perl* tersedia pula untuk berbagai sistem operasi varian *unix* (*SunOS*, *Linux*, *BSD*, *HP-UX*), juga tersedia untuk sistem operasi seperti *DOS*, *Windows*, *PowerPC*, *BeOS*, *VMS*, *EBCDIC*, dan *PocketPC*.

Bagian yang terpenting dari XAMPP adalah sebagai berikut (Nugroho, 2008):

1. *htdocs* adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.
2. *PhpMyAdmin* merupakan bagian untuk mengelola basis data MySQL yang ada di komputer. Untuk membukanya, buka *browser* lalu ketikkan alamat <http://localhost/phpMyAdmin>, maka akan muncul halaman *phpMyAdmin*.
3. Kontrol Panel yang berfungsi untuk mengelola layanan (service) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian berasal dari kata “Metode” yang artinya cara yang tepat untuk melakukan sesuatu dan “Logos” yang artinya ilmu atau pengetahuan. Metodologi yaitu cara melakukan sesuatu dengan menggunakan pikiran secara saksama untuk mencapai suatu tujuan. Sedangkan “Penelitian” adalah suatu kegiatan untuk mencari, mencatat, merumuskan dan menganalisis sampai menyusun laporannya. Berdasarkan pengertian di atas maka dapat disimpulkan bahwa Metodologi penelitian yaitu ilmu yang mempelajari cara-cara melalui tahapan-tahapan yang disusun secara ilmiah untuk mencari, menyusun serta menganalisis dan menyimpulkan data-data.

3.2 Jenis dan Sumber Data

Data yang dibutuhkan dalam penelitian ini dikelompokkan menjadi dua jenis, yaitu sumber data primer dan data sekunder.

1. Data primer

Data yang diperoleh melalui pengamatan langsung terhadap sistem yang sedang berjalan dan wawancara dengan karyawan sebagai sumber informasinya. Dalam penelitian ini, data tersebut berupa data yang berkaitan dengan proses pergudangan pada PT. Krama Yudha Ratu Motor.

2. Data sekunder

Data sekunder adalah data yang diperoleh dari data yang tersedia dan telah terlebih dahulu dikumpulkan dan dilaporkan oleh orang lain, buku-buku dan kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian. Data tersebut adalah data umum perusahaan, profil perusahaan dan struktur organisasi perusahaan.

3.3 Metode Pengumpulan Data

Metode yang digunakan dalam pengumpulan data adalah sebagai berikut :

1. Studi lapangan

Studi lapangan adalah usaha melakukan pengumpulan data secara langsung pada objek yang dilakukan dengan cara sebagai berikut :

a. Observasi

Tahap ini merupakan tahap untuk mengumpulkan data dengan cara mengamati secara langsung keadaan maupun kegiatan dalam perusahaan yang kemudian dilakukan pencatatan informasi terhadap objek yang dianggap perlu. Observasi ini dilakukan di gudang bagian *trimming*. Pada tahap ini juga dilakukan pengamatan langsung terhadap pengguna sistem dan pengamatan langsung terhadap sistem yang berjalan.

b. Wawancara

Tahap ini penulis melakukan wawancara dengan pengguna sistem untuk memperoleh keterangan mengenai proses pergudangan khususnya pada penerimaan *part* dan pengeluaran *part* dalam menghasilkan informasi data part yang tersedia di gudang serta laporan-laporannya.

2. Studi Pustaka

Studi pustaka dilakukan dengan membaca buku dan literatur dalam lingkup perkuliahan maupun di luar lingkup perkuliahan yang berhubungan dengan judul dan permasalahan sehingga dapat menunjang dalam penulisan tugas akhir ini. Studi kepustakaan yang dilakukan adalah dengan menggunakan buku yang dimiliki, buku yang dipinjam dari perpustakaan dan mencari data yang diperlukan melalui internet.

3.4 Metode Pengembangan Sistem

Pengembangan sistem yaitu menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah berjalan. Metodologi pengembangan sistem yang digunakan yaitu model *prototype* jenis *evolutionary*. Model *prototype* cocok digunakan untuk menjabarkan kebutuhan *user* secara lebih detail. Tahapan-tahapan dalam model *prototype evolutionary* adalah sebagai berikut:

1. Mengidentifikasi kebutuhan pengguna.

Pengembang melakukan diskusi dengan bagian gudang *trimming* tentang kebutuhan sistem yang diinginkan.

2. Membuat *prototype*.

Pengembang membuat *prototype* dari sistem yang telah dijelaskan oleh bagian gudang *trimming*.

3. Menyesuaikan dan evaluasi *prototype* dengan keinginan pengguna.

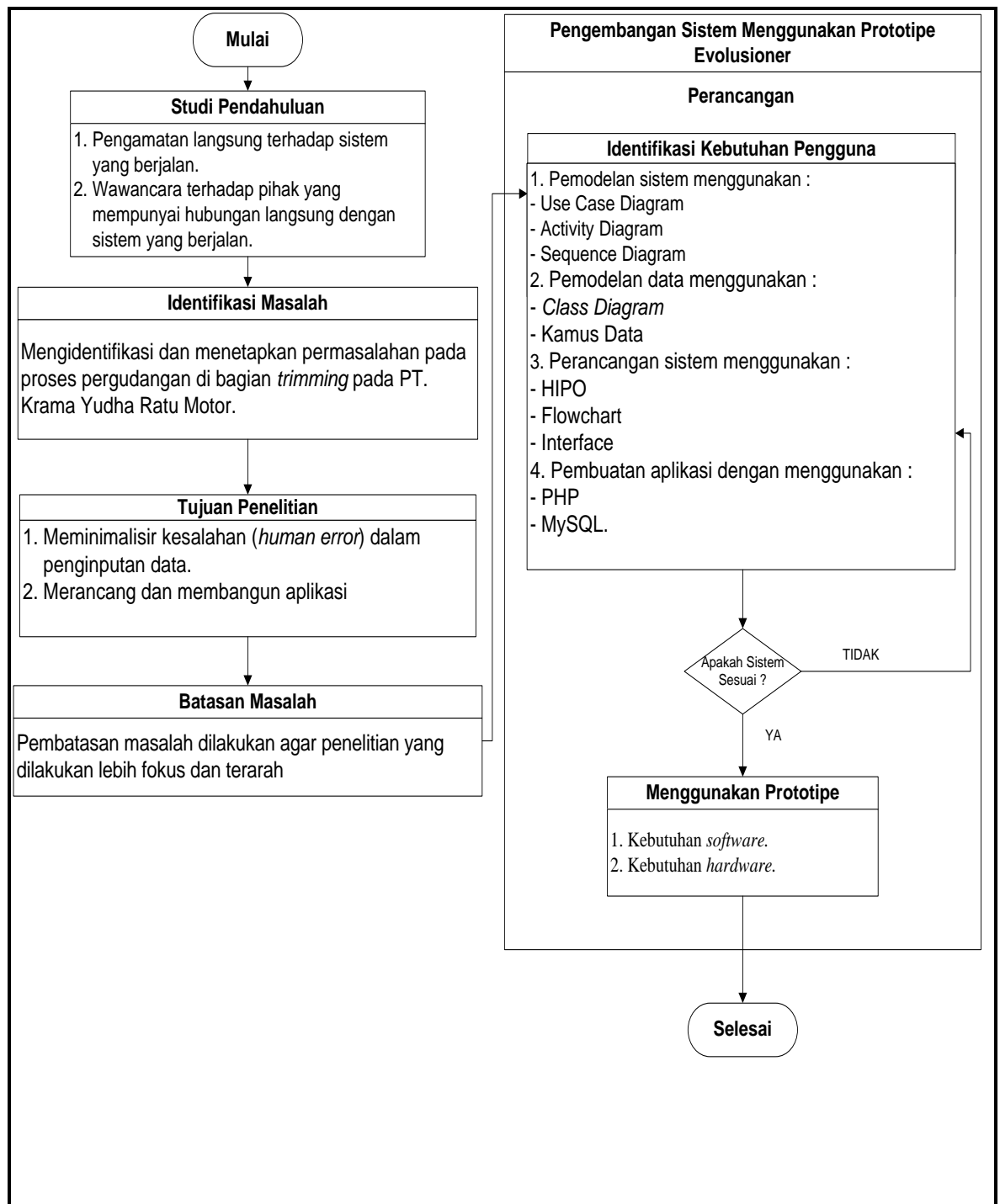
Pengembang menanyakan kepada bagian gudang *trimming* tentang *prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.

4. Menggunakan *prototype*.

Sistem mulai dikembangkan dengan *prototype* yang sudah dibuat.

3.5 Kerangka Penelitian

Dari permasalahan yang telah diidentifikasi melalui penelitian, maka dibuat sebuah kerangka yang menjelaskan tahap-tahap dalam memecahkan permasalahan yang terdapat pada sistem informasi pergudangan pada PT. Krama Yudha Ratu Motor. Kerangka penelitian dibuat dalam bentuk *flowchart* seperti pada Gambar III.1. Kerangka tersebut menggambarkan tahap-tahap kegiatan yang dilakukan mulai dari awal hingga akhir :



Gambar III.1 Kerangka Pemecahan Masalah
(Sumber: Hasil Analisis Data, 2017)

1. Studi Pendahuluan

Kegiatan yang dilakukan adalah studi lapangan dan studi pustaka. Studi lapangan yang dilakukan yaitu pengamatan dan wawancara. Sedangkan studi

pustaka yang dilakukan yaitu membaca buku dan literatur dalam lingkup perkuliahan maupun di luar lingkup perkuliahan yang berhubungan dengan judul dan permasalahan.

2. Identifikasi Masalah

Mengidentifikasi dan merumuskan masalah yang terjadi selama penelitian pada sistem pergudangan di bagian *trimming* yang ada pada PT. Krama Yudha Ratu Motor.

3. Tujuan Penelitian

Berdasarkan pokok permasalahan yang ada maka tujuan penelitian dari tugas akhir ini adalah memberikan kemudahan bagi *user* dengan cara merancang dan membangun aplikasi pergudangan untuk mempermudah user dalam mendapatkan informasi berkaitan dengan proses pergudangan.

4. Batasan Masalah

Batasan masalah pada penelitian ini yaitu hanya hanya sebatas menangani masalah pengolahan data penerimaan part dari vendor dan pengolahan data pengeluaran part yang dibutuhkan untuk keperluan produksi. Tidak membahas tentang pengiriman part ke line kitting, sub assembling dan produksi.

5. Pengembangan Sistem Menggunakan *Prototype Evolutionary*

Tahap selanjutnya adalah melakukan pengembangan sistem menggunakan metode *prototype evolutionary*. Adapun langkah-langkah pengembangan sistem menggunakan metode *prototype evolutionary* yaitu:

a. Mengidentifikasi kebutuhan pengguna.

Tahap ini dilakukan untuk mengidentifikasi apa saja yang dibutuhkan oleh pengguna pada sistem informasi pergudangan.

b. Perancangan.

Tahap ini dilakukan untuk merancang sebuah sistem berdasarkan hasil dari identifikasi kebutuhan pengguna. Pada tahap perancangan terdapat tiga langkah yang dilakukan yaitu:

- Pemodelan Sistem Menggunakan UML.
- Pemodelan Data Menggunakan *Class Diagram*.
- Perancangan Sistem Dengan Menggunakan HIPO, *Flowchart* dan perancangan *interface*.
- Pembuatan Aplikasi Dengan Menggunakan PHP dan MySQL.

6. Menyesuaikan dan evaluasi *prototype* dengan keinginan pengguna

Tahap selanjutnya yaitu dimana pengembang menanyakan kepada bagian admin gudang tentang *prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.

7. Menggunakan *Prototype*

Tahap ini merupakan hasil dari keputusan yang diambil berdasarkan tahap evaluasi yang berarti bahwa sistem yang telah dirancang sudah sesuai dengan kebutuhan pengguna dan akan digunakan.

8. Kesimpulan dan Saran

Tahap ini merupakan akhir dari rangkaian tahap-tahap dalam kerangka pemecahan masalah, yaitu membuat kesimpulan serta mengemukakan saran yang sekiranya perlu disampaikan sebagai masukan bagi perusahaan.

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sejarah Perusahaan

PT. Krama Yudha Ratu Motor merupakan sebuah perusahaan perseroan terbatas yang bergerak dalam bidang perakitan kendaraan bermotor jenis niaga yang berdiri pada tanggal 1 Juni 1973 dan mulai merekrut karyawan pada tahun 1974 selanjutnya baru memproduksi secara masal pada tahun 1975.

Pendirian PT. Krama Yudha Ratu Motor tersebut berdasarkan akte notaris Abdul latief No. 10 tanggal 1 Juni tahun 1973. Pendirian tersebut diperkuat dengan :

1. Perijinan dari BKPM dalam bidang usaha No. 92/A/BPKPM/73/PMDN tanggal 4 Juli 1973.
2. Perijinan dari departemen perindustrian dalam bidang teknis No. 27/IIA/D/IV/74 tanggal 21 Maret 1974, pada saat itu perusahaan ini masuk dalam kelompok *assembling*, mesin dan perbengkelan yang kini menjadi kelompok otomotif (beroda 4 atau lebih).
3. Perijinan dari Departemen Kehakiman dalam bidang hukum No. 16.A.S.105/18/74 tanggal 15 April 1974.
4. Perijinan dari pengadilan negeri Jakarta dalam bidang hukum kewilayahan No. 1374 tanggal 18 April 1974.

PT. Krama Yudha Ratu Motor merupakan bagian dari Krama Yudha Mitshubishi Group dan perusahaan lainnya yang tergabung diantaranya adalah PT. Colt Engingeer Manufacturer, PT. Mitsubishi Krama Yudha Motor and Manufacturer, PT. Krama Yudha Tiga Berlian, PT. Krama Yudha Kesuma Motor dan PT. Krama Yudha Ratu Motor.

Tujuan awal berdirinya Group Mitsubishi di Indonesia pada tahun 1970 adalah untuk mengurangi impor kendaraan jenis niaga dari Eropa sehingga mengurangi beban ekonomi negara pada saat itu dan untuk memenuhi kebutuhan.

pasar didalam negeri sendiri serta untuk membuka lapangan kerja bagi para pengangguran pada saat itu.

Hingga saat ini PT. Krama Yudha Ratu Motor merakit kendaraan jenis niaga dan penumpang. Jenis kendaraan yang dirakit saat ini yaitu :

1. Fuso yang terdiri dari tipe FM-H, FM-517, FN-517, FN-527.
2. Colt Diesel Maru-T dengan tipe 304 Bus Chassis, tipe 304, tipe 334, tipe 347, tipe 349, tipe 349 HD.
3. L-300 dengan tipe DPU dan DMB.
4. T120 atau CJM terdiri dari model CJM 1.3 & CJM 1.5.
5. *Outlander Sport*.

4.2 Profile Perusahaan

Nama Perusahaan : PT. Krama Yudha Ratu Motor

Logo Perusahaan :



Alamat : Jl. Raya Bekasi Km. 21-22 Rawa Terate
Cakung, Pulogadung-Jakarta 13920

Telp : 021-4602905

Fax : 021-4602904

Tanggal Berdiri : 1 Juni 1973

Status Modal : Penanaman Modal Asing (PMA)

Bangunan Pabrik : 20.360 m²

Luas Tanah : 143.035 m²

Bangunan Pendukung : 6.600 m²

(Gudang, Kantin, Masjid)

Produksi Komersial : Tahun 1975

Jenis Usaha : Perakitan (*Assembling*) kendaraan bermotor merek Mitsubishi

Produksi Utama	: Colt L-300 (SL) Colt Diesel (TD) Colt T120 SS (CJM) Fuso (FM/FN) <i>Outlander Sport (ZC)</i>
Jam Kerja	
Senin-Kamis	: 07.10 - 11.35 WIB
Istirahat	: 11.35 - 12.25 WIB
Snack	: 10.00 - 10.10 WIB 14.00 - 14.10 WIB
Jumat	: 07.10 - 11.40 WIB
Istirahat	: 11.40 - 13.00 WIB
Snack	: 10.00 - 10.10 WIB 15.00 - 15.10 WIB
Tenaga Kerja	
Produksi	: 993 orang
Umum	: 220 orang
Total	: 1213 orang

4.3 Visi dan Misi Perusahaan

PT. Krama Yudha Ratu Motor adalah perusahaan manufaktur yang bergerak dalam bidang perakitan kendaraan roda empat jenis niaga. Perusahaan ini mempunyai visi dan misi sebagai berikut :

Visi :

- Untuk pembangunan dan kesejahteraan masyarakat Indonesia.
- Menjadi perusahaan yang mempunyai reputasi tinggi.
- Menjadi perakit mobil Mitsubishi yang terkemuka diluar Jepang.
- Dalam kuruh waktu berikutnya sudah merakit mobil dengan merek Krama Yudha.

Misi :

- Menjadi perusahaan yang paling menguntungkan dalam kalangan industri.

- b. Menghasilkan keuntungan yang cukup memuaskan bagi shareholders dan kesejahteraan yang baik bagi seluruh karyawan.
- c. Memperlakukan seluruh karyawan dengan cara yang adil dan terhormat.
- d. Menjadi perusahaan yang terpercaya untuk merakit mobil dengan merek Mitshubishi.
- e. Menjadi perusahaan yang kuat dan bertumbuh, serta siap untuk menghadapi persaingan regional maupun global.

4.4 Budaya Kerja Perusahaan

Dalam menjalankan proses produksinya PT. Krama Yudha Ratu Motor menerapkan 5S yang bertujuan untuk meningkatkan efisiensi dan kualitas lingkungan kerja. Berikut adalah uraian budaya kerja 5S yang di terapkan pada PT. Krama Yudha Ratu Motor yaitu :

- 1. *Seiri* : Memisahkan (memisahkan antara barang yang di perlukan dengan barang yang tidak diperlukan).
- 2. *Seiton* : Menyimpan (Menyimpan barang yang di perlukan sesuai dengan urutan yang mudah di gunakan dalam keadaan bersih).
- 3. *Seisou* : Membersihkan (Membersihkan barang yang hendak di gunakan dalam keadaan bersih).
- 4. *Shiketsu* : Menjaga (Menjaga dengan konsisten kondisi *Seiri*, *Seiton*, dan *Seisou*).
- 5. *Shitsuke* : Melakukan 5S (*Seiri*, *Seiton*, dan *Seisou* kepada seluruh karyawan sehingga menjadi suatu kebiasaan).

4.5 Kebijakan Sistem Manajemen Keselamatan dan Kesehatan Kerja

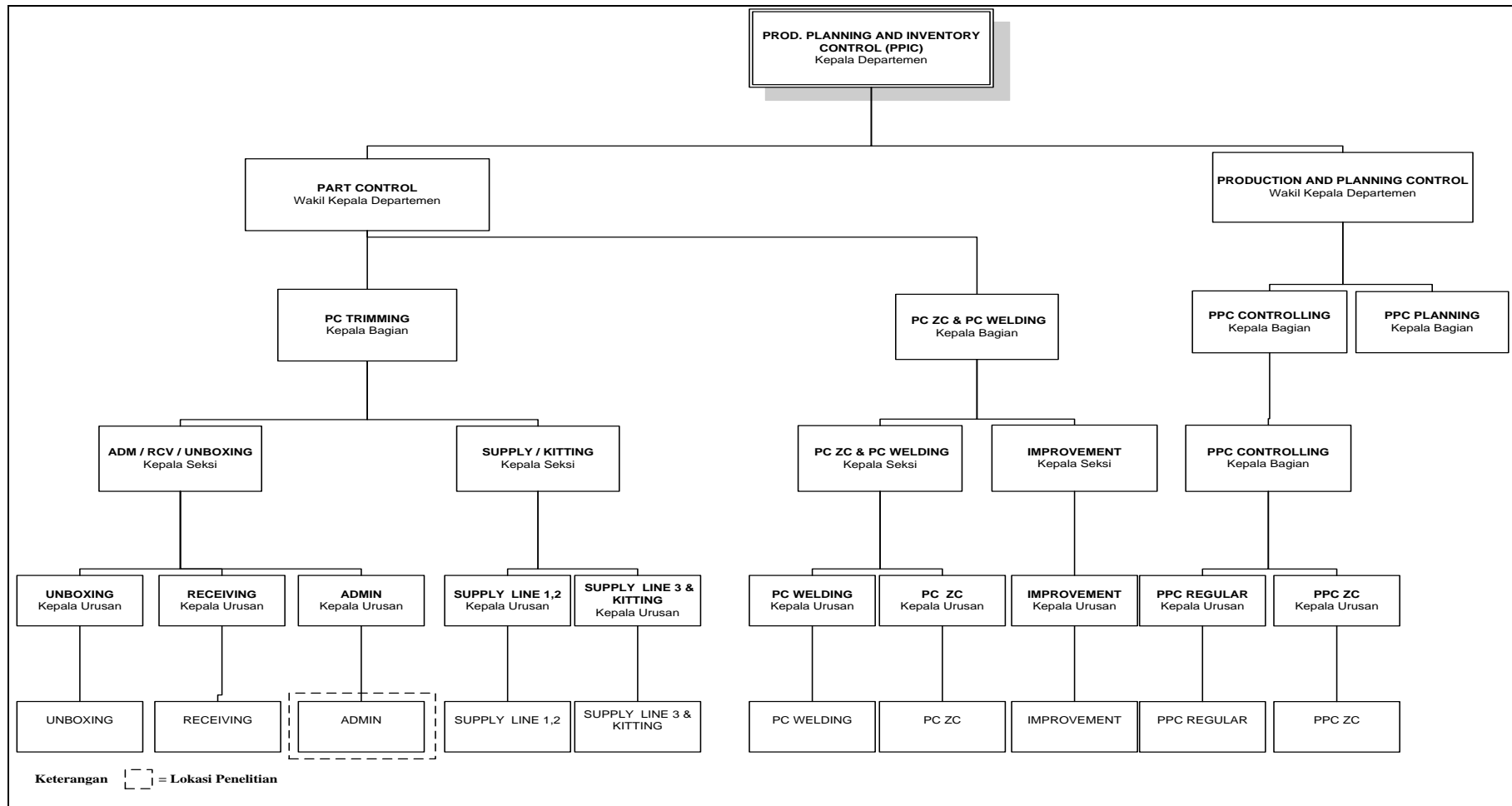
Dalam upaya mengelola bisnis, resiko bisnis dan manajemen, seluruh karyawan PT. Krama Yudha Ratu Motor, senantiasa berpedoman kepada ISO 9001, ISO 14001 dan SMK3 dengan strategi :

1. Mengelola perseroan yang aman dan berkembang dengan menerapkan prinsip : PQCDS (*Produktivitas, Quality, Cost, Delivery, Safety*) dengan melakukan perbaikan produktivitas, penghematan biaya, melakukan perbaikan kualitas, mengirim tepat waktu dan mengusahakan nihil kecelakaan, mencegah gangguan kesehatan dan penyakit akibat kerja serta mencegah kebakaran dalam pabrik.
2. Senantiasa melakukan pengelolaan lingkungan dengan mencegah pencemaran lingkungan dan penghematan sumber daya alam dan atau energi.
3. Menyediakan induksi dan pembelajaran kepada karyawan dan mitra kerja.
4. Mematuhi peraturan dan persyaratan lain yang terkait untuk mutu, lingkungan dan K3.
5. Senantiasa melakukan peningkatan atau perbaikan berkesinambungan.
6. Memastikan setiap karyawan dan atau mitra kerja PT. Krama Yudha Ratu Motor memahami, bertanggung jawab dan mempertanggung jawabkan untuk peduli pada mutu, lingkungan, keselamatan dan kesehatan kerja dalam aktifitasnya.

4.6 Struktur Organisasi Perusahaan

Setiap perusahaan dalam menjalankan roda usahanya memerlukan suatu pengorganisasian. Struktur organisasi dapat diartikan sebagai susunan dan hubungan antar bagian dan posisi dalam suatu perusahaan. Suatu struktur organisasi menggambarkan pembagian kerja, pelimpahan wewenang, kesatuan perintah dan tanggung jawab yang jelas.

Struktur organisasi yang tersusun dengan baik akan memudahkan koordinasi, integrasi, serta meningkatkan efektivitas dan efisiensi suatu perusahaan didalam mencapai tujuannya. Berikut adalah struktur organisasi PT. Krama Yudha Ratu Motor departemen *production planning and inventory control* (PPIC).



Gambar IV.1 Struktur Organisasi Departemen *Production Planning and Inventory Control* (PPIC)

(Sumber : PT. Krama Yudha Ratu Motor, 2017)

Berikut adalah uraian tugas dari masing-masing jabatan yang tercantum pada struktur organisasi di atas :

1. Kepala Departemen *Production Planning and Inventory Control* (PPIC).
 - a. Membuat rencana kerja dan melaksanakan pengawasan serta pengendalian pada bagian *Production Planning Control* (PPC), *Part Control* dan *Information Technology* (IT).
 - b. Memberdayakan dan memonitor kegiatan proses produksi dengan menggunakan tenaga kerja yang cukup sesuai dengan kondisi alat dan mesin agar mencapai produktivitas sesuai standar.
 - c. Mengevaluasi dan mengembangkan sumber daya manusia secara terpadu untuk mencapai sasaran yang telah ditentukan.
 - d. Menjalankan, memelihara, meningkatkan sistem manajemen perusahaan.
 - e. Melakukan identifikasi, mengelola dan memonitoring lingkungan ditempat kerja masing-masing.
2. Wakil Kepala Departemen *Production Planning and Inventory Control* (PPIC).
 - a. Membantu atau mewakili kepala departemen membuat rencana dan melaksanakan pengawasan serta pengendalian pada bagian PPC, *Part Control* dan *Information Technology* (IT).
 - b. Membantu atau mewakili kepala departemen memberdayakan dan memonitor kegiatan proses produksi dengan menggunakan tenaga kerja yang sesuai kondisi alat dan mesin agar mencapai produktivitas sesuai standar.
 - c. Membantu atau mewakili kepala departemen mengevaluasi dan mengembangkan sumber daya manusia secara terpadu untuk mencapai sasaran yang telah ditetapkan.
 - d. Melaksanakan tugas kepala bagian, bila kepala bagian mendapat tugas keluar perusahaan atau berhalangan.

- e. Menjalankan, memelihara, meningkatkan sistem manajemen perusahaan.
 - f. Melakukan identifikasi, mengelola dan memonitoring lingkungan ditempat kerja masing-masing.
3. Kepala Bagian *Production Planning Control* (PPC)
- a. Menerima *order* produksi dari PT. Krama Yudha Tiga Berlian.
 - b. Membuat *daily production schedule* terhadap 4 titik hitung yaitu : *welding, trimming on, trimming off* dan *delivery* dan mendistribusikan ke bagian terkait.
 - c. Membuat rencana produksi *sparepart*.
 - d. Mengatur laju proses produksi dan sebagai penghubung kebagian terkait jika ada hambatan dan perubahan-perubahan.
 - e. Membuat laporan bulanan dan tahunan seluruh kegiatan produksi.
 - f. Membuat laporan hasil produksi *spareparts* ke departemen keuangan.
 - g. Memberdayakan dan memonitor kegiatan proses produksi dan menggunakan tenaga kerja yang cukup sesuai kondisi alat dan mesin agar mencapai produktivitas sesuai standar.
 - h. Memelihara lingkungan, alat-alat, mesin-mesin dengan melaksanakan program K3 (kesehatan dan keselamatan kerja), 5R (ringkas, rapih, resik, rawat dan rajin) dan GKM (gugus kendali mutu).
4. Kepala Bagian *Part Control*
- a. Membuat *order* dan memonitor penerimaan bahan baku untuk kebutuhan produksi.
 - b. Mengawasi dan mengendalikan penerimaan, penyimpanan, pemeliharaan dan pendistribusian *parts* ke *line* produksi
 - c. Berkoordinasi dengan bagian terkait dalam pengaturan dan perubahan jadwal produksi apabila terjadi keterlambatan pengadaan *parts* dan bahan baku.
 - d. Memberikan *feedback* kepada *vendor* maupun *distributor* atas terjadinya keterlambatan penerimaan *parts*.
 - e. Mengendalikan *Line Damage* dan membuat order pengadaan *parts*

penggantinya.

f. Bertanggung jawab atas penyimpanan dan penggunaan kunci gudang.

5. Kepala Seksi *Production Planning Control* (PPC)

a. Membuat schedule jam-jaman terhadap titik hitung antara lain: *welding, trimming on, trimming off* dan *delivery* dan mendistribusikan.

b. Mencatat hambatan-hambatan proses produksi dan melaporkan kepada atasan.

c. Membuat *schedule spareparts*.

d. Membuat laporan tahunan seluruh kegiatan hasil produksi.

e. Membuat laporan hasil produksi *spareparts* bulanan ke departemen keuangan.

f. Memelihara lingkungan kerja, alat-alat dan mesin-mesin dengan cara melaksanakan program K3 (kesehatan dan keselamatan kerja), 5R (ringkas, rapih, resik, rawat dan rajin) dan GKM (gugus kendali mutu).

g. Mengevaluasi, mengarahkan dan membina karyawan sehingga menghasilkan karyawan yang bertanggung jawab, disiplin dan berprestasi guna tercapainya sasaran mutu.

6. Kepala Seksi *Part Control*

a. Bertanggung jawab atas tersedianya *part* lokal sesuai dengan kebutuhan produksi.

b. Memantau dan mengendalikan *line damaged, part claim, part* rawan hilang, serta kedatangan *part* lokal.

c. Membuat laporan harian dan bulanan meliputi antara lain : *line damaged part, claim part*, dll.

d. Memimpin *meeting* koordinasi harian, memantau kehadiran karyawan, membuka dan menutup pintu gudang.

e. Mengawasi dan mengkoordinir bawahan agar penerimaan *parts*, penyimpanan di gudang dan distribusi dapat terkendali dengan baik

f. Melakukan pembinaan dan penilaian atas prestasi kerja bawahan meliputi mutu hasil kerja, disiplin, absensi, produktivitas dan hubungan kerja.

- g. Melakukan pembagian tugas karyawan sedemikian rupa sehingga tercapai pembagian kerja yang seimbang, efektif dan efisien.
7. Admin
- a. Membuat *part balance*.
 - b. *Input* hasil dan rencana produksi harian ke *part balance*.
 - c. *Check part balance* (Laporkan jika ada penyimpangan *part* terhadap produksi).
 - d. Isi hasil produksi ke lembar produksi dari *Production Planning Control* (PPC).
 - e. *File* surat jalan.
 - f. Membuat daily time schedule (DTS).
 - g. Membuat Kanban PT. GKD dan PT. Mitshubishi Krama Yudha Motors and Manufacturing (MKM).
 - h. *Input part* ke *order list vendor*.
 - i. Membuat laporan harian.

4.7 Gudang Penyimpanan Part dan Bagian Part Control

Gudang penyimpanan berfungsi sebagai tempat penyimpanan *part* atau bahan baku untuk diproduksi maupun hasil produksi yang siap untuk didistribusikan ke pelanggan. Gudang sangat berperan penting oleh karena itu manajemen pergudangan harus dilakukan dengan baik agar proses produksi berjalan dengan lancar. Aktifitas-aktifitas pergudangan meliputi penerimaan barang, meletakkan barang sesuai lokasi yang ditentukan, pendistribusian dan penyimpanan. Dalam proses penyimpanan, bahan baku atau *part* dijaga dan dilindungi dari kerusakan agar kualitas suatu produk terjaga hingga sampai ke pelanggan. Setiap perusahaan yang memiliki gudang terdapat satu atau beberapa bagian yang bertugas untuk mengatur atau mengelola gudang tersebut. Namun bagian ini tetap berhubungan dengan bagian lain khususnya bagian produksi.

Bagian *part control* adalah bagian yang bertanggung jawab dalam hal penyimpanan dan pengelolaan *part* atau bahan baku sehingga dapat terjamin ketersediaannya pada saat dibutuhkan untuk kepentingan proses produksi. Pada

PT. Krama Yudha Ratu Motor memiliki satu bagian yang khusus mengatur dan mengelola gudang penyimpanan yaitu bagian *part control*. Bagian *part control* terdiri dari dua tim yaitu bagian *part control welding* dan bagian *part control trimming*. Adapun tugas dan wewenang bagian *part control welding* dan bagian *part control trimming* pada PT. Krama Yudha Ratu Motor, yaitu:

a. Bagian *part control welding*

bertanggung jawab pada gudang *welding* dalam hal penyimpanan dan pengelolaan *part* atau bahan baku sehingga dapat terjamin ketersediaannya pada saat dibutuhkan untuk kepentingan proses produksi pada bagian produksi di *line welding*.

b. Bagian *part control trimming*

bertanggung jawab pada gudang *trimming* dalam hal penyimpanan dan pengelolaan *part* atau bahan baku sehingga dapat terjamin ketersediaannya pada saat dibutuhkan untuk kepentingan proses produksi pada bagian produksi di *line trimming*.

PT. Krama Yudha Ratu Motor memiliki dua gudang penyimpanan *part* yaitu gudang *welding* dan gudang *trimming*. Gudang *welding* yaitu gudang yang berfungsi untuk menyimpan *part* untuk keperluan produksi pada *line welding* sedangkan gudang *trimming* yaitu gudang yang berfungsi untuk menyimpan *part* untuk keperluan produksi pada *line trimming*.

4.8 Dokumen pada Sistem Pergudangan yang Berjalan

Sistem informasi pergudangan yang sedang berjalan di PT. Krama Yudha Ratu Motor masih menggunakan *Microsoft Excel*. Dokumen yang terlibat dalam sistem informasi pergudangan adalah :

1. *Delivery Note*

Delivery Note digunakan dalam proses pemesanan *part* dan pengiriman ke *vendor*. PT. Krama Yudha Tiga Berlian mengirimkan *delivery note* kepada *vendor*, *vendor* akan mencetak *delivery note*, kemudian *delivery note* ini akan dibawa *vendor* ketika pengiriman akan dilakukan. *Delivery note* ini terdiri dari beberapa bagian yaitu :

- *Plant slip* : Sebagai bukti *part* telah dikirim, diterima baik oleh gudang dan disimpan oleh PT. Krama Yudha Ratu Motor.
- *KTB slip* : Sebagai bukti *part* telah dikirim sesuai dengan pesanan dan disimpan oleh PT. Krama Yudha Tiga Berlian.
- *Vendor slip* : Sebagai bukti penagihan pembayaran atas pesanan tersebut.

DELIVERY NOTE					DELIVERY NOTE					DELIVERY NOTE											
Vendor Slip					KTB Slip					Plant Slip											
Date : 26.01.2017					Date : 26.01.2017					Date : 26.01.2017											
Page : 1 (1)					Page : 1 (1)					Page : 1 (1)											
Vendor Name : KAYABA INDONESIA					D/N No. 1530501596					D/N No. 1530501596											
Vendor Code : 100085					Destination : KRM					Vendor Name : KAYABA INDONESIA											
Contract No : 72100085					Delivery Date : 01.02.2017					Vendor Code : 100085											
Destination : KRM					Shift /Hour : /07.30-11.00, 12.30-15.00 WIB					Contract No : 72100085											
Delivery Date : 01.02.2017					Vendor : KAYABA INDONESIA																
Hour : 07.30-11.00, 12.30-15.00 WIB																					
<table border="1"> <tr> <td>KTB</td> <td>KRM</td> <td>Vendor</td> </tr> </table>					KTB	KRM	Vendor	<table border="1"> <tr> <td>KRM</td> <td>Vendor</td> </tr> </table>					KRM	Vendor	<table border="1"> <tr> <td>KRM</td> <td>Vendor</td> </tr> </table>					KRM	Vendor
KTB	KRM	Vendor																			
KRM	Vendor																				
KRM	Vendor																				
Item	Material	Description	U/M	Qty	Item	Material	Description	U/M	Qty	Item	Material	Description	U/M	Qty							
10	41608-77511	FRONT STRUT SET SUSPEN	PC	48	10	41608-77511	FRONT STRUT SET SUSPEN	PC	48	10	41608-77511	FRONT STRUT SET SUSPEN	PC	48							
20	41608-77511	FRONT STRUT SET SUSPEN	PC	48	20	41608-77511	FRONT STRUT SET SUSPEN	PC	48	20	41608-77511	FRONT STRUT SET SUSPEN	PC	48							
30	41700-77500	SIABSORBER ASSY REAR	PC	96	30	41700-77500	SIABSORBER ASSY REAR	PC	96	30	41700-77500	SIABSORBER ASSY REAR	PC	96							

Gambar IV.2 Delivery Note

(Sumber : PT. Krama Yudha Ratu Motor, 2017)

2. Surat Jalan

Surat jalan merupakan surat pengantar dalam melakukan pengiriman barang yang dapat digunakan untuk informasi kepada PT. Krama Yudha Tiga Berlian (*customer*) bahwa barang pesanan yang diminta atau dipesan sudah dikirim ke alamat tujuan dan surat jalan ini berguna bagi *vendor* untuk proses penagihan pembayaran.

Gambar IV.3 Surat Jalan

(Sumber : PT. Krama Yudha Ratu Motor, 2017)

Trimming on day berisi form pengeluaran *part*. admin gudang akan meng-*input form* pengeluaran sehari sebelum tanggal yang tertera di *daily production schedule* (DPS). Admin akan meng-*input* data per tipe mobil sesuai dengan permintaan atau *daily production schedule* (DPS) produksi yang dikirim oleh bagian *production planning control* (PPC).

MICCode	MINName	Serial no.	1	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	Total	E/Serial	BCL	
DB	COLT L3000 CHASSIS	10560																															60	10620	60		
DUC	COLT L3000 CHASSIS	720							6	9	5	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	120	840	120	
DH	FUSO DIESEL TRUCK FM527M3	744																																744	744		
PN	FUSO DIESEL TRUCK FM527M2	5020																															24	5844	24		
DPF	COLT L3000 PICK UP FLAT	104177	102	102	96	81		97	91	86	90	57								21	72	59												1589	165766	1589	
DP	COLT L3000 PICK UP STD	46360										21	59		77	82		61															300	47160	300		
CU	SLI DPS	123																																	123		123
CUF	SLI OFF	1095									10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	110	1915	110
CA	SLI DB	125																																	125		125
EFB0	COLT DIESEL LT TRUCK FE447B	239																																	239		239
EF00	COLT DIESEL LT TRUCK FE447	5461																																	5461		5461
EF10	COLT DIESEL LT TRUCK FE447	1020																																	1020		1020
ET00	COLT DIESEL LT TRUCK FE304	20520																																	20520		20520
ET10	COLT DIESEL LT TRUCK FE304	27000																																	27000		27000
ET11	COLT DIESEL LT TRUCK FE304	1680																																	1680		1680
EU00	COLT DIESEL LT TRUCK FE334	17640																																	17640		17640
EU10	COLT DIESEL LT TRUCK FE334	20066																																	20066		20066
EY00	COLT DIESEL LT TRUCK FE347	5280																																	5280		5280
EY10	COLT DIESEL LT TRUCK FE347	4020																																	4020		4020
EW00	COLT DIESEL LT TRUCK FE349	39720																																	39720		39720
EW10	COLT DIESEL LT TRUCK FE349	61440																																	61440		61440

Gambar IV.7 Form *Trimming On Day*

(Sumber : PT. Krama Yudha Ratu Motor, 2017)

2. Balance Produksi

Setelah admin gudang selesai meng-*input form* pengeluaran, *balance* produksi akan otomatis dan memotong stock sesuai dengan tipe mobil. *balance* produksi merupakan selisih antara permintaan produksi dengan part yang ada di gudang per satu unit mobil. Sebagai contoh : permintaan produksi 10 unit mobil tipe CJM, ketika di-*input* di *form trimming on day* kemudian *balance* produksi secara otomatis menunjukkan angka -1 artinya part yang ada digudang hanya tersedia 9 unit mobil tipe CJM. Berikut ini merupakan gambar *form part balance*:

		BALANCE PRODUKSI																											
TYPE	BLC	1	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
JPS	0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
JPF0	0	0	0	0	0	0	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4
JB00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3 way	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DUc	0	0	0	0	0	0	6	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DU(std)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
DUF	0	0	0	-2	0	0	-21	-39	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31	-31
CU(std)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CUF(Flat)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CA(mb)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FE 71PBSSRDK2 (TQ)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FE 71PBSSRDK1 (TZ)	0	0	0	0	-1	-1	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
FE 71PBSSRDK1 (TH)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Gambar IV.8 Balance Produksi

(Sumber : PT. Krama Yudha Ratu Motor, 2017)

3. Stok

Sheet stok yaitu menampilkan data stok type mobil menjadi beberapa komponen-komponen *part* yang lain. Berikut ini merupakan gambar stok sebagai berikut :

NO	TYPE	Vend	Item	Name	STOCK	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	KM002094	HOLDER JACK ASSY	318	205	212	172	221	221	247	254	241	196	217	205	205	227	179	179	58
2	TQ,TF	TRUAYA UNON	KM002943	BRKT.BOX A/C/LNR UPPER	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0
3	TQ,TF	TRUAYA UNON	KM002985	AIR CLEANER BOX ASSY	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0
4	TQ,TF	KARYA BAHANA UNIG	KM002367	REINF.BRACE LH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0
5	TQ,TF	KARYA BAHANA UNIG	KM002369	REINF.BRACE RH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0
6	TQ,TF	KARYA BAHANA UNIG	KM002416	WASHER PLATE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	48	0	0
7	TU,TT,TL	KARYA BAHANA UNIG	KM003123	SPACER A	0	240	88	240	170	170	0	0	120	60	240	176	176	82	0	0	0
8	TU,TT,TL	KARYA BAHANA UNIG	KM003124	SPACER B	0	240	88	240	170	170	0	0	120	60	240	176	176	82	0	0	0
9	TW,TV,TV,TT,TL,TK	KARYA BAHANA UNIG	MC112902	BRKT.CROSS SHAFT	260	207	214	124	145	145	219	138	120	180	120	148	148	41	0	0	60
10	TS,TR	KARYA BAHANA UNIG	MC112914	BRKT.CROSS SHAFT	0	0	24	14	24	24	4	25	29	36	23	30	30	54	30	30	6
11	TG,TK,TZ,TF,TQ,TY,TH	KARYA BAHANA UNIG	MC122520	SUPT.ASSY.UPPER,RRR ENG.MTG	0	0	12	36	42	42	50	69	106	42	40	41	41	122	115	115	28
12	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	ME235408	PROTECTOR.HOSE	266	213	220	190	193	193	239	249	196	168	202	207	207	193	109	109	70
13	TS,TR	KARYA BAHANA UNIG	ME408754	ROD.RAD SUPPORT	0	0	24	14	24	24	4	25	29	36	23	30	30	54	30	30	6
14	TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	ME412311	BRKT.SNORKEL LWR	266	213	220	190	193	193	239	249	196	168	202	207	207	193	133	133	34
15	TDSM,GT,TT,TL,TH,TK	KARYA BAHANA UNIG	ME412653	SUPPORT.RADIATOR UPPER	266	213	220	190	193	193	239	249	196	168	202	207	207	193	109	109	70
16	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	ME413649	ROD.RAD SUPPORT	272	219	226	196	199	199	245	255	202	174	148	213	213	199	115	115	28
17	TQ,TZ,TY,TK,TG,TH	KARYA BAHANA UNIG	ME511264	BRKT.CROSS SHAFT	0	0	12	36	42	42	50	69	106	42	40	41	41	122	115	115	28
18	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	ME515118	BRKT.SHIPT CABLE	306	253	200	170	173	173	219	229	176	208	182	187	187	173	149	149	50
19	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	ME515119	BRKT.CABLE	272	219	226	196	199	199	245	255	202	174	148	213	213	199	115	115	28
20	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	MK411036	BRKT.CLUTCH CYLINDER	314	261	208	168	217	217	243	250	237	252	213	201	201	223	175	175	52
21	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	MK412230	BRACKET.P.CABLE A	296	243	190	160	223	223	209	219	226	190	172	177	177	163	139	139	40
22	TW,TV,TV,TS,TT,TL,TK	KARYA BAHANA UNIG	MK412730	SUPT.ASSY.UPPER RR ENG.MTG	318	205	212	172	179	179	173	209	111	214	141	212	212	141	76	76	52
23	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	MK513682	BRKT.DUCT	828	669	510	480	549	549	627	597	558	474	456	531	531	489	357	357	120
24	TQ,TZ,TY,TK,TW,TV,TK	KARYA BAHANA UNIG	MK515694	BRKT.VALVE EXHBRK	266	213	220	190	193	193	239	249	196	168	202	207	207	193	109	109	70

Gambar IV.9 Stok

(Sumber : PT. Krama Yudha Ratu Motor, 2017)

4.9 Prosedur Pergudangan

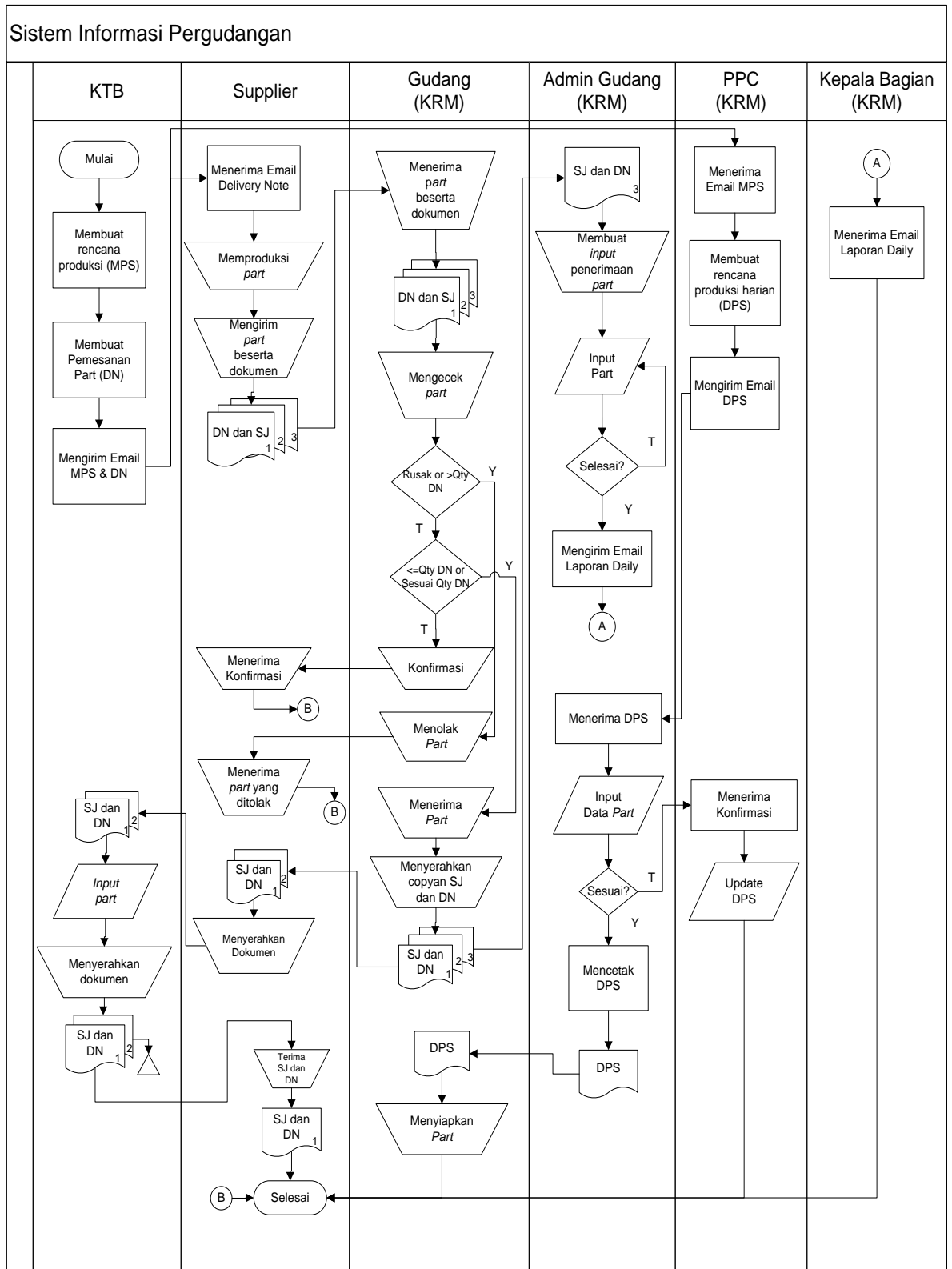
Berikut ini merupakan prosedur sistem informasi pergudangan pada bagian *trimming* di PT. Krama Yudha Ratu Motor :

1. PT. Krama Yudha Tiga Belian (KTB) membuat rencana produksi (MPS). Setelah membuat pemesanan *part* (DN) serta mengirim *delivery note* ke *vendor* dan mengirim MPS ke bagian PPC PT. Krama Yudha Ratu Motor (KRM).
2. Vendor menerima pemesanan kemudian memproduksi pemesanan lalu mengirim *part* ke gudang PT. Krama Yudha Ratu Motor beserta dokumen-dokumen (surat jalan dan *delivery note*).
3. Gudang menerima dokumen-dokumen lalu memeriksa dokumen beserta *part* yang dikirim apakah sesuai dengan dokumen seperti nama *part*, jumlah *part* dan kondisi *part*.
 - Jika fisik (*part*) yang dikirim mengalami kekurangan jumlah atau kuantiti dengan dokumen maka dokumen akan ditahan oleh gudang sampai kekurangannya dikirim oleh *vendor* sedangkan jika *part* (fisik) yang

dikirim mengalami kelebihan maka pihak gudang akan menolak atau meminuskan dokumen sesuai dengan jumlah fisik.

- Apabila fisik (*part*) mengalami kerusakan seperti patah maka gudang akan menolak serta meminuskan dokumen agar tidak terjadi selisih dikemudian hari.
4. Jika *part* sudah diterima atau proses *receiving* selesai maka gudang akan membawa *part* masuk kedalam gudang kemudian bagian gudang akan menandatangani dokumen-dokumen (surat jalan dan *delivery note*) dan menyerahkan *copy*-an surat jalan dan *delivery note* ke vendor dan bagian admin gudang.
 5. *Vendor* akan menerima dokumen lalu dokumen diserahkan ke PT. Krama Yudha Tiga Berlian (KTB) untuk proses peng-*input*-an.
 6. KTB akan memproses peng-*input*-an penerimaan *part* kemudian menyerahkan dokumen ke vendor. Vendor akan menerima dokumen.
 7. Bagian admin gudang akan menerima dokumen dari bagian gudang. Kemudian admin gudang input penerimaan *part* sesuai dengan dokumen. Jika sudah selesai maka akan mengirim *email* laporan penerimaan *part* ke kepala bagian. Kepala bagian menerima *email* laporan.
 8. Bagian *Production Planning Control* (PPC KRM) menerima email *master production schedule* (MPS) dari PT. Krama Yudha Tiga Berlian (KTB). Dari *master production schedule* (MPS) kemudian bagian *Production Planning Control* (PPC KRM) akan membuat *daily production schedule* (DPS) dan mengirim *email daily production schedule* (DPS) ke bagian Admin gudang.
 9. Admin Gudang menerima email *daily production schedule* (DPS) kemudian meng-*input* *daily production schedule* (DPS) dan mencetak *daily production schedule* (DPS) kemudian diserahkan ke bagian gudang.
 10. Bagian gudang menerima *daily production schedule* (DPS) kemudian bagian gudang akan menyiapkan barang sesuai dengan permintaan.

Berikut ini merupakan flowmap sistem informasi pergudangan yang berjalan dibagian *trimming* pada PT. Krama Yudha Ratu Motor :



Gambar IV.10 Flowmap Sistem Informasi Pergudangan yang Berjalan

(Sumber : Hasil Analisis, 2017)

4.10 Penggambaran Sistem Pegudangan dengan *Unified Modeling Language (UML)*

Usecase diagram sistem informasi pergudangan yang sedang berjalan pada bagian *trimming* di PT. Krama Yudha Ratu Motor adalah sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case* sistem informasi pergudangan pada PT. Krama Yudha Ratu Motor dapat dilihat pada tabel di bawah ini :

Tabel IV.1 Definisi Aktor *Use Case Diagram* Sistem Informasi Pengudangan

No.	Aktor	Deskripsi
1	KTB	Bagian dari PT. Krama Yudha Tiga Berlian yang membuat rencana produksi dan permintaan part.
2	<i>Vendor</i>	Salah satu pemasok atau penyedia bahan baku untuk proses produksi.
3	Gudang (KRM)	Bagian yang akan menjalankan proses penerimaan part dan pengeluaran part.
4	Admin Gudang (KRM)	Bagian yang menginput data penerimaan dan pengeluaran serta menyajikan informasi berdasarkan jumlah part yang datang.
5	Kepala Bagian (KRM)	Kepala bagian part control yang akan menerima email laporan daily penerimaan.
6	<i>Production Planning Control</i> (KRM)	Bagian yang membuat daily production schedule (DPS).

(Sumber: Hasil Analisis, 2017)

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* sistem informasi pergudangan pada PT. Krama Yudha Ratu Motor dapat dilihat pada table di bawah ini:

Tabel IV.2 Definisi *Use Case Diagram* Sistem Informasi Pengudangan

No.	<i>Use Case</i>	Deskripsi
1	Membuat rencana produksi (MPS)	Proses pembuatan rencana produksi (MPS) yang dilakukan oleh PT. Krama Yudha Tiga Berlian

Tabel IV.2 Definisi *Use Case Diagram* Sistem Informasi Pengudangan (Lanjutan)

No.	Use Case	Deskripsi
2	Membuat pemesanan <i>part</i> (DN)	Proses pembuatan pemesanan <i>part</i> (DN) yang dilakukan oleh PT. Krama Yudha Tiga Berlian
3	Email MPS dan DN	Proses pengiriman email yang dilakukan oleh PT. Krama Yudha Tiga Berlian untuk PPC KRM dan <i>Vendor</i>
4	Menerima Email DN	Proses penerimaan email yang dilakukan oleh <i>Vendor</i>
5	Memproduksi <i>part</i>	<i>Vendor</i> melakukan proses produksi untuk memenuhi pesanan
6	Mengirim <i>part</i> beserta dokumen	<i>Vendor</i> melakukan pengiriman <i>part</i> ke gudang beserta surat jalan dan delivery note.
7	Menerima <i>part</i> beserta dokumen	Gudang menerima dokumen dari <i>Vendor</i> , <i>checker</i> melakukan proses penerimaan <i>part</i> .
8	Mengecek <i>part</i>	<i>Checker</i> melakukan pengecekan <i>part</i> sesuai dengan dokumen.
9	Menolak <i>part</i>	Proses penolakan <i>part</i> apabila <i>part</i> tidak sesuai dengan dokumen.
10	Konfirmasi	Proses konfirmasi ke <i>vendor</i> apabila <i>part</i> mengalami masalah pengiriman.
11	Menerima <i>part</i>	Proses penerimaan <i>part</i> apabila <i>part</i> sesuai dengan dokumen.
12	Menyerahkan SJ dan DN	<i>Checker</i> menandatangani dokumen kemudian dokumen diserahkan ke <i>Vendor</i> dan admin gudang.
13	Membuat <i>input</i> penerimaan <i>part</i>	Proses pembuatan <i>input</i> penerimaan yang dilakukan oleh admin gudang.
14	<i>Input</i> penerimaan <i>part</i>	Proses penginputan penerimaan yang dilakukan oleh admin gudang.
15	Mengirim <i>Email</i> laporan penerimaan <i>daily</i>	Proses pengiriman <i>email</i> yang dilakukan oleh admin gudang untuk kepala bagian <i>part</i> control.

Tabel IV.2 Definisi *Use Case Diagram* Sistem Informasi Pengudangan (Lanjutan)

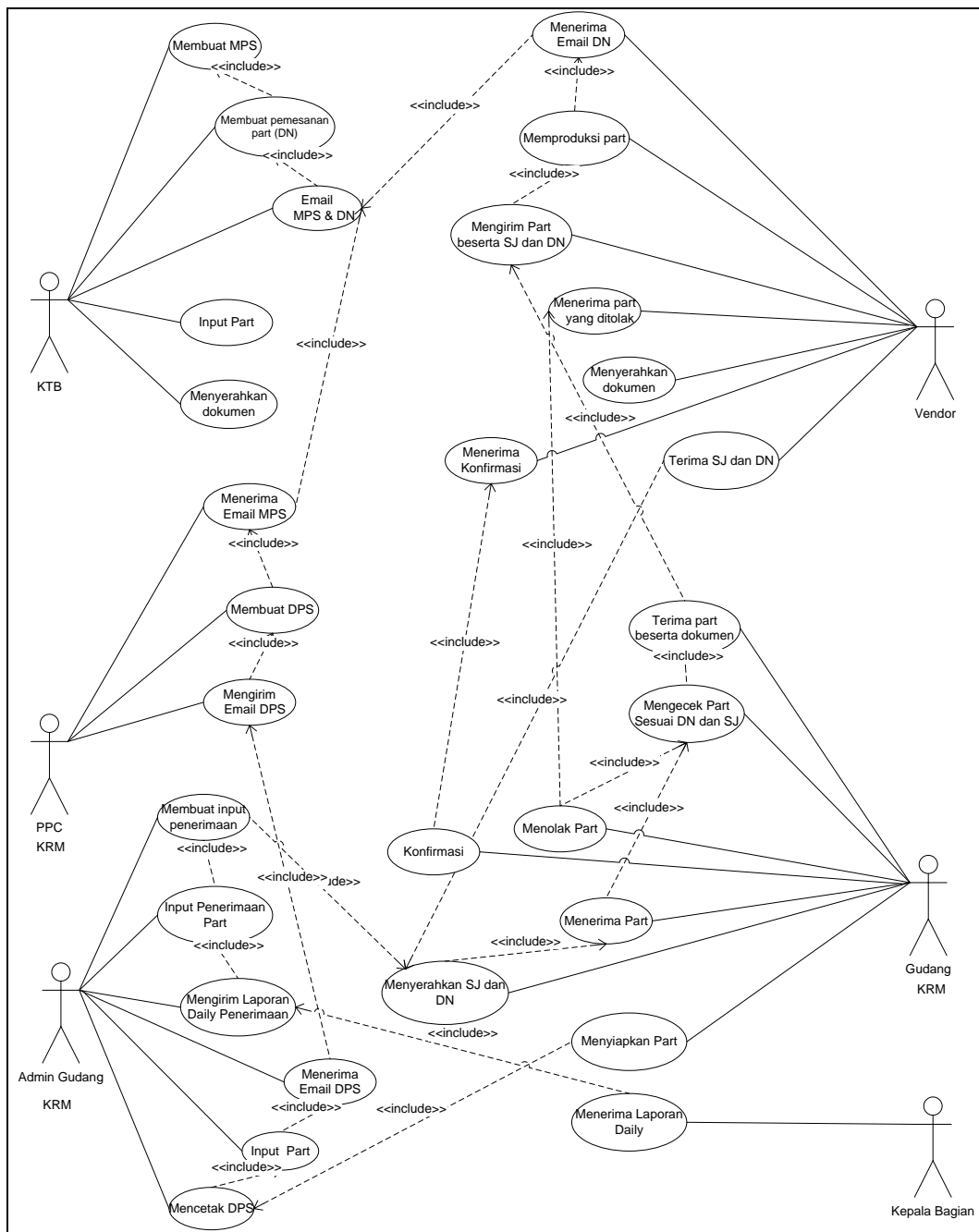
No.	Use Case	Deskripsi
-----	----------	-----------

16	Menerima laporan penerimaan <i>daily</i>	Proses penerimaan <i>email</i> yang dilakukan oleh kepala bagian <i>part control</i> .
17	Menerima Konfirmasi	<i>Vendor</i> menerima konfirmasi dari gudang mengenai pengiriman yang tidak sesuai.
18	Menerima <i>part</i> yang ditolak	<i>Vendor</i> menerima <i>part</i> yang tidak sesuai dokumen.
19	Menyerahkan Dokumen	<i>Vendor</i> menyerahkan dokumen PT. Krama Yudha Tiga Berlian.
20	<i>Input part</i>	Proses penginputan yang dilakukan oleh PT. Krama Yudha Tiga Berlian.
21	Menyerahkan dokumen	Bagian gudang memisahkan <i>copy</i> -an dokumen kemudian diserahkan ke <i>Vendor</i> .
22	Menerima SJ dan DN	<i>Vendor</i> menerima dokumen dari PT. Krama Yudha Tiga Berlian.
23	Menerima <i>Email</i> MPS	Proses penerimaan <i>email</i> yang dilakukan oleh PPC KRM.
24	Membuat rencana produksi harian (DPS)	Proses pembuatan rencana produksi harian (DPS) oleh PPC KRM.
25	Mengirim <i>Email</i> DPS	Proses pengiriman <i>email</i> DPS ke admin gudang.
26	Menerima DPS	Proses penerimaan <i>email</i> DPS dari PPC KRM.
27	<i>Input data part</i>	proses penginputan data pengeluaran <i>part</i> yang dilakukan oleh admin gudang sesuai dengan DPS
28	Mencetak DPS	Proses pencetakan DPS kemudian DPS diserahkan kepada gudang.
29	Menyiapkan <i>part</i>	Proses penyiapan <i>part</i> sesuai dengan dokumen DPS.

(Sumber: Hasil Analisis, 2017)

3. Use case diagram

Berikut ini adalah gambar *use case* diagram sistem informasi pergudangan yang sedang berjalan pada PT. Krama Yudha Ratu Motor :



Gambar IV.11 Use Case Diagram Sistem Informasi Pengudangan

(Sumber: Hasil Analisis, 2017)

BAB V ANALISIS DAN PEMBAHASAN

5.1 Analisis Sistem

Berdasarkan hasil pengumpulan dan pengolahan data yang dilakukan pada sistem informasi pergudangan yang sedang berjalan di gudang bagian *trimming* pada PT Krama Yudha Ratu Motor, dapat disimpulkan bahwa proses pengolahan data penerimaan dan pengeluaran *part* masih menggunakan *Microsoft Excel* dan belum mempunyai *database*, sehingga data mudah hilang dan dapat terjadi penggandaan data.

5.2 Analisis Kebutuhan Rinci Sistem

Analisis kebutuhan rinci sistem untuk sistem informasi pergudangan di bagian *trimming* pada PT. Krama Yudha Ratu Motor dijelaskan pada Tabel V.2

Tabel V.1 Kebutuhan Rinci Sistem

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

Tabel V.1 Kebutuhan Rinci Sistem (Lanjutan)

No.	Kebutuhan Rinci Sistem	Uraian
-----	------------------------	--------

3.	Pengelolaan data <i>master vendor</i>	Proses yang dapat dilakukan oleh Kepala Bagian pada pengelolaan data <i>master vendor</i> yaitu: a. Kepala Bagian dapat menambah <i>vendor</i> baru, mencari dan memilih <i>vendor</i> tertentu, kemudian mengubah atau menghapus data <i>vendor</i> tersebut.
3.	Pengelolaan data <i>master jenis</i>	Proses yang dapat dilakukan oleh Kepala Bagian pada pengelolaan data <i>master jenis</i> yaitu: b. Kepala Bagian dapat menambah jenis baru pada data master jenis. c. Kepala Bagian dapat mencari dan memilih data jenis tertentu, kemudian mengubah atau menghapus data jenis tersebut.
4	Membuat transaksi penerimaan <i>part</i>	Proses yang dapat dilakukan oleh Admin Gudang pada membuat transaksi penerimaan <i>part</i> yaitu: a. Admin Gudang dapat meng- <i>input</i> data dan menyimpan data penerimaan <i>part</i> . b. Admin Gudang dapat mencari data dan memilih data <i>part</i> tertentu, kemudian mengubah atau menghapus data.
5	Membuat transaksi pengeluaran <i>part</i>	Proses yang dapat dilakukan oleh Admin Gudang pada membuat transaksi pengeluaran <i>part</i> yaitu: a. Admin Gudang dapat meng- <i>input</i> data pengeluaran <i>part</i> , menyimpan dan mencetak data pengeluaran <i>part</i> b. Admin Gudang dapat mencari data dan memilih data <i>part</i> tertentu, kemudian mengubah atau menghapus data.

Tabel V.1 Kebutuhan Rinci Sistem (Lanjutan)

No.	Kebutuhan Rinci Sistem	Uraian
6	Membuat transaksi permintaan <i>part</i> .	Proses yang dapat dilakukan oleh PPC pada membuat transaksi permintaan <i>part</i> yaitu: a. PPC dapat meng- <i>input</i> data, menyimpan permintaan <i>part</i> , dapat mencari data dan memilih data tertentu,

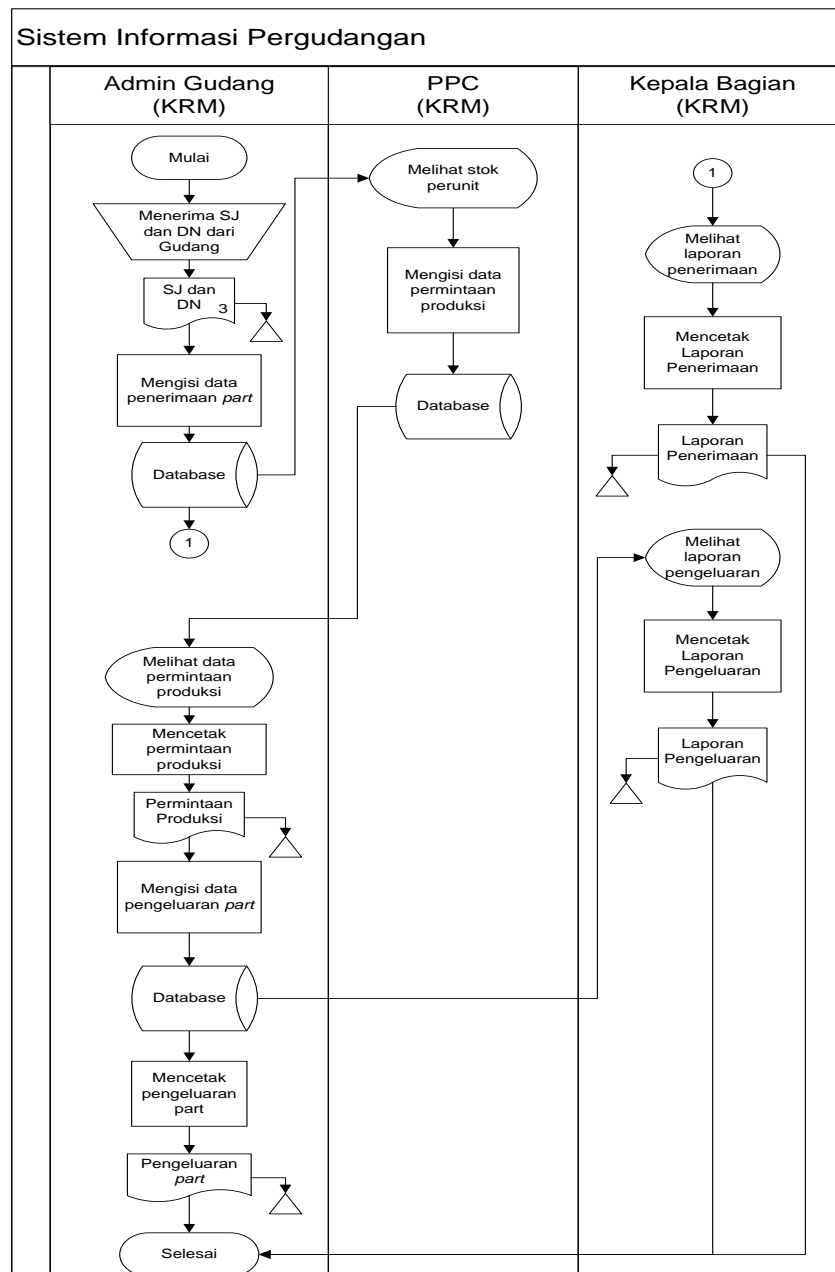
		kemudian mengubah atau menghapus data.
7	Menampilkan dan mencetak stok	Admin Gudang, Kepala Bagian dan PPC dapat menampilkan dan mencetak stok.
8	Menampilkan dan mencetak laporan penerimaan	Kepala Bagian dapat menampilkan dan mencetak laporan penerimaan.
9	Menampilkan dan mencetak laporan pengeluaran	Kepala Bagian dapat menampilkan dan mencetak laporan pengeluaran.

Sumber: Pengolahan Data (2017)

5.3 Perancangan *Flowmap* Sistem Usulan

Urutan prosedur dan *Flowmap* sistem informasi pergudangan yang diusulkan sebagai berikut:

1. Prosedur sistem informasi pergudangan yang diusulkan sebagai berikut:
 - a. Admin Gudang menerima dokumen-dokumen (surat jalan dan *delivery note*) penerimaan dari Gudang.
 - b. Admin Gudang meng-*input* data penerimaan ke dalam *database*.
 - c. PPC melihat stok *part* untuk membuat permintaan produksi.
 - d. PPC meng-*input* data permintaan *part* ke dalam *database*.
 - e. Admin Gudang melihat permintaan *part* meng-*input* jumlah permintaan *part* ke data pengeluaran ke dalam *database* dan mencetak pengeluaran *part* dan di serahkan ke gudang untuk proses pengeluaran *part*.
 - f. Kepala bagian dapat melihat laporan penerimaan *part* dan laporan pengeluaran *part* dengan cara mengisi tanggal awal dan tanggal akhir yg ingin di lihat kemudian mencetak laporan penerimaan dan mencetak laporan pengeluaran.
2. *Flowmap* sistem informasi pergudangan yang diusulkan sebagai berikut:



Gambar V.1 Flowmap Sistem Informasi Pergudangan

Sumber: Hasil Analisis (2017)

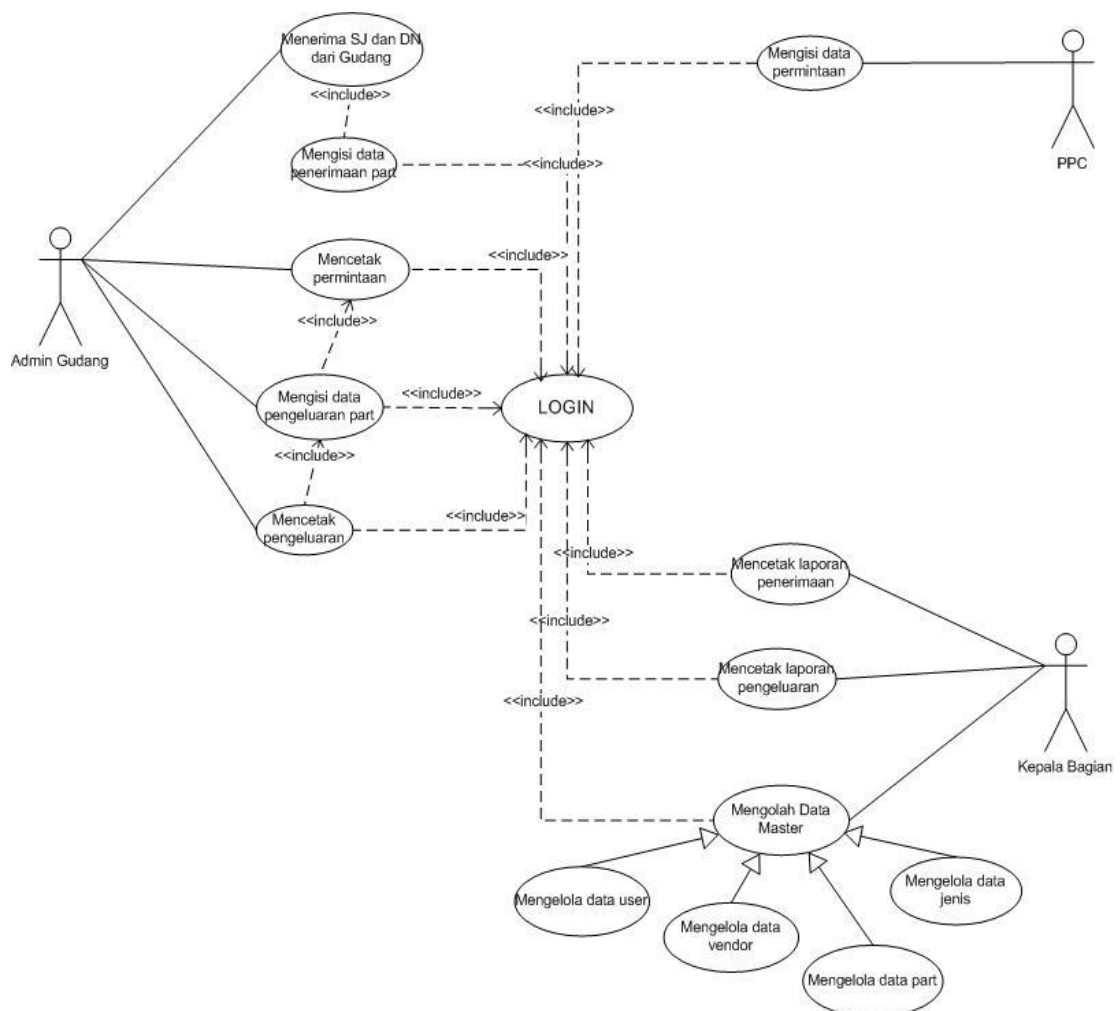
5.4 Analisis Sistem Informasi Usulan

Analisis proses sistem informasi pergudangan di bagian *trimming* 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

5.4.1 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 pergudangan usulan dapat dilihat pada Gambar V.2 berikut ini:



Gambar V.2 Use Case Sistem Informasi Pergudangan Usulan

Sumber: Hasil Analisis (2017)

Penjelasan *use case diagram* sistem informasi pergudangan bagian *trimming* usulan dijelaskan sebagai berikut ini:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi pergudangan usulan dijelaskan pada Tabel V.2

Tabel V.2 Definisi Aktor *Use Case*

No.	Aktor	Deskripsi
1	Admin Gudang	Admin Gudang merupakan aktor yang akan menginput penerimaan <i>part</i> dan pengeluaran <i>part</i> .
2	Kepala Bagian	Kepala Bagian merupakan aktor yang akan melihat laporan penerimaan serta mendapat akses mencetak laporan penerimaan.
3	<i>Production Planning Control</i> (PPC)	<i>Production Planning Control</i> (PPC) merupakan aktor yang akan melihat data stok yang ada di gudang.

Sumber: Hasil Analisis (2017)

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* sistem informasi pergudangan usulan dijelaskan pada tabel berikut:

Tabel V.3 di bawah berisi deskripsi *use case* mengelola data *master*

Tabel V.3 Definisi *Use Case* Mengelola Data *Master*

<i>Use Case Name</i>	Mengelola data <i>master</i>
<i>Primary Actor</i>	Kepala Bagian
<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>, Kepala Bagian membuat akun <i>user</i> baru serta mencari, mengubah, dan menghapus data <i>user</i> yang telah terdaftar sebelumnya. 2. Mengelola data <i>master vendor</i>, Kepala Bagian dapat menambah, mencari, mengubah dan menghapus data <i>master vendor</i>. 3. Mengelola data <i>master part</i>, Kepala Bagian dapat menambah, mencari, mengubah dan menghapus data

	<p><i>master part.</i></p> <p>4. Mengelola data <i>master</i> jenis, Kepala Bagian dapat menambah, mencari, mengubah dan menghapus data <i>master part.</i></p>
<i>Relationship</i>	<p><i>Generalization:</i> Mengelola data <i>master user</i>, data <i>master vendor</i> dan mengelola data <i>master part.</i></p> <p><i>Include:</i> <i>Login.</i></p>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Bagian masuk ke tampilan menu utama. 2. Kepala Bagian memilih menu data <i>master user</i>, data <i>master vendor</i> dan data <i>master part.</i> 3. Kepala Bagian melakukan proses menambah data baru atau memodifikasi data <i>master</i> sebelumnya.
<i>Precondition</i>	Kepala Bagian telah melakukan <i>login</i> terlebih dahulu.

Sumber: Hasil Analisis (2017)

Tabel V.4 di bawah ini berisi deskripsi untuk *use case login*.

Tabel V.4 Definisi *Use Case Login*

<i>Use Case Name</i>	<i>Login</i>
<i>Primary Actor</i>	Admin Gudang, Kepala Bagian dan PPC.
<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 Gudang, Kepala Bagian dan PPC
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> mengisi <i>username</i> dan <i>password</i> di menu <i>login</i>. 2. Sistem memvalidasi data yang di-<i>input</i>. Apabila data tidak sesuai maka <i>user</i> tersebut tidak dapat menggunakan sistem. 3. Apabila data sesuai maka <i>user</i> dapat menggunakan sistem sesuai dengan hak aksesnya.

Sumber: Hasil Analisis (2017)

Tabel V.5 di bawah ini berisi deskripsi untuk *use case* mengisi data penerimaan *part*

Tabel V.5 Definisi *Use Case* Mengisi Data Penerimaan *Part*

<i>Use Case Name</i>	Mengisi data penerimaan <i>part</i>
<i>Primary Actor</i>	Admin Gudang
<i>Brief Description</i>	Proses mengisi data penerimaan <i>part</i> dilakukan apabila Admin Gudang telah menerima dokumen-dokumen (surat jalan dan <i>delivery note</i>) dari gudang.
<i>Relationship</i>	<i>Association:</i> Admin Gudang <i>Include:</i> Login
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin Gudang masuk ke menu utama 2. Admin Gudang memilih menu data transaksi, pilih sub menu penerimaan. 3. Admin Gudang meng-<i>input</i> data <i>penerimaan part</i> 4. Data penerimaan <i>part</i> yang tersimpan di <i>database</i>.
<i>Precondition</i>	Admin Gudang telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.6 di bawah ini berisi deskripsi untuk *use case* melihat data stok *part*.

Tabel V.6 Definisi *Use Case* Melihat Data Stok *Part*

<i>Use Case Name</i>	Melihat data stok <i>part</i> .
<i>Primary Actor</i>	Admin Gudang, PPC dan Kepala Bagian
<i>Brief Description</i>	Proses menampilkan data stok <i>part</i> .
<i>Relationship</i>	<i>Association:</i> Admin Gudang, PPC dan Kepala Bagian <i>Include:</i> Login
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin Gudang, PPC dan Kepala Bagian masuk ke menu utama 2. Admin Gudang, PPC dan Kepala Bagian memilih menu stok 3. Sistem menampilkan data stok <i>part</i>
<i>Precondition</i>	Admin Gudang, PPC dan Kepala Bagian telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.7 di bawah ini berisi deskripsi untuk *use case* mengisi data permintaan *part*

Tabel V.7 Definisi *Use Case* Mengisi Data Permintaan *Part*

<i>Use Case Name</i>	Mengisi data permintaan <i>part</i> .
<i>Primary Actor</i>	PPC
<i>Brief Description</i>	Proses mengisi data permintaan <i>part</i> dilakukan apabila PPC telah membuat DPS (<i>daily production schedule</i>).
<i>Relationship</i>	<i>Association</i> : PPC <i>Include</i> : Login
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. PPC masuk ke menu utama 2. PPC memilih menu data transaksi, pilih sub menu permintaan. 3. PPC meng-<i>input</i> data permintaan <i>part</i>. 4. Data permintaan yang tersimpan di <i>database</i>.
<i>Precondition</i>	PPC telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.8 di bawah ini berisi deskripsi untuk *use case* melihat data permintaan *Part*

Tabel V.8 Definisi *Use Case* Melihat Data Permintaan *Part*

<i>Use Case Name</i>	Melihat data permintaan <i>part</i> .
<i>Primary Actor</i>	Admin Gudang
<i>Brief Description</i>	Proses melihat data permintaan <i>part</i>
<i>Relationship</i>	<i>Association</i> : Admin Gudang <i>Include</i> : Login, Mengisi data permintaan <i>part</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin Gudang masuk ke menu utama 2. Admin Gudang memilih menu data transaksi, pilih sub menu permintaan 3. Sistem menampilkan data permintaan <i>part</i>
<i>Precondition</i>	Admin Gudang telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.9 dibawah ini berisi deskripsi untuk *use case* mencetak data permintaan *part*

Tabel V.9 Definisi *Use Case* Mencetak Data Permintaan *Part*

<i>Use Case Name</i>	Mencetak data permintaan <i>part</i> .
<i>Primary Actor</i>	Admin Gudang
<i>Brief Description</i>	Proses mencetak data permintaan <i>part</i>

<i>Relationship</i>	<i>Association:</i> Admin Gudang <i>Include:</i> Login, Melihat data permintaan <i>part</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin Gudang masuk ke menu utama 2. Admin Gudang memilih menu data transaksi, pilih sub menu permintaan 3. Sistem menampilkan data permintaan <i>part</i> 4. Admin Gudang mencetak laporan permintaan <i>part</i>
<i>Precondition</i>	Admin Gudang telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.10 dibawah ini berisi deskripsi untuk *use case* mengisi data pengeluaran *part*

Tabel V.10 Definisi *Use Case* Mengisi Data Pengeluaran *Part*

<i>Use Case Name</i>	Mengisi data pengeluaran <i>part</i> .
<i>Primary Actor</i>	Admin Gudang
<i>Brief Description</i>	Proses mengisi data pengeluaran <i>part</i> dilakukan apabila Admin Gudang sudah mencetak permintaan <i>part</i> .
<i>Relationship</i>	<i>Association:</i> Admin Gudang <i>Include:</i> Login, Mencetak Permintaan <i>Part</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin Gudang masuk ke menu utama 2. Admin Gudang memilih menu transaksi, pilih sub menu Pengeluaran 3. Admin Gudang meng-<i>input</i> data pengeluaran <i>part</i> 4. Data pengeluaran <i>part</i> yang tersimpan di <i>database</i>.
<i>Precondition</i>	Admin Gudang telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.11 dibawah ini berisi deskripsi untuk *use case* mencetak data pengeluaran *part*

Tabel V.11 Definisi *Use Case* Mencetak Data Pengeluaran *Part*

<i>Use Case Name</i>	Mencetak data pengeluaran <i>part</i> .
<i>Primary Actor</i>	Admin Gudang
<i>Brief Description</i>	Proses mencetak data pengeluaran <i>part</i> dilakukan apabila Admin Gudang sudah mengisi data pengeluaran <i>part</i> .
<i>Relationship</i>	<i>Association:</i> Admin Gudang <i>Include:</i> Login, Mengisi Data Pengeluaran <i>Part</i>

<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Admin Gudang masuk ke menu utama 2. Admin Gudang memilih menu transaksi, pilih sub menu pengeluaran 3. Sistem menampilkan data pengeluaran <i>part</i> 4. Admin Gudang mencetak data pengeluaran <i>part</i>
<i>Precondition</i>	Admin Gudang telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.12 di bawah ini berisi deskripsi untuk *use case* melihat data laporan penerimaan *part*

Tabel V.12 Definisi *Use Case* Melihat Data Laporan Penerimaan *Part*

<i>Use Case Name</i>	Melihat data laporan penerimaan
<i>Primary Actor</i>	Kepala Bagian
<i>Brief Description</i>	Proses menampilkan data laporan penerimaan <i>part</i>
<i>Relationship</i>	<i>Association:</i> Kepala Bagian <i>Include:</i> <i>Login</i> , Mengisi data penerimaan <i>part</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Bagian masuk ke menu utama 2. Kepala Bagian memilih menu laporan, pilih sub penerimaan 3. Kepala Bagian Meng-<i>input</i> tanggal awal dan tanggal akhir 4. Sistem menampilkan laporan penerimaan
<i>Precondition</i>	Kepala Bagian telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.13 dibawah ini berisi deskripsi untuk *use case* mencetak laporan penerimaan *part*

Tabel V.13 Definisi *Use Case* Mencetak Laporan Penerimaan *Part*

<i>Use Case Name</i>	Mencetak data laporan penerimaan
<i>Primary Actor</i>	Kepala Bagian
<i>Brief Description</i>	Proses mencetak laporan penerimaan <i>part</i>
<i>Relationship</i>	<i>Association:</i> Kepala Bagian <i>Include:</i> <i>Login</i> , Melihat laporan penerimaan
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Bagian masuk ke menu utama 2. Kepala Bagian memilih menu laporan, pilih sub penerimaan 3. Sistem menampilkan laporan penerimaan 4. Kepala Bagian mencetak laporan penerimaan
<i>Precondition</i>	Kepala Bagian telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.14 dibawah ini berisi deskripsi untuk *use case* melihat data laporan pengeluaran *part*

Tabel V.14 Definisi *Use Case* Melihat Data Laporan Pengeluaran *Part*

<i>Use Case Name</i>	Melihat data laporan pengeluaran
<i>Primary Actor</i>	Kepala Bagian
<i>Brief Description</i>	Proses menampilkan data pengeluaran <i>part</i>
<i>Relationship</i>	<i>Association:</i> Kepala Bagian <i>Include:</i> Login, Mengisi pengeluaran <i>part</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Bagian masuk ke menu utama 2. Kepala Bagian memilih menu laporan, pilih sub pengeluaran 3. Kepala Bagian Meng-<i>input</i> tanggal awal dan tanggal akhir 4. Sistem menampilkan laporan pengeluaran
<i>Precondition</i>	Kepala Bagian telah melakukan <i>login</i> terlebih dahulu

Sumber: Hasil Analisis (2017)

Tabel V.15 dibawah ini berisi deskripsi untuk *use case* mencetak laporan pengeluaran *part*

Tabel V.15 Definisi *Use Case* Mencetak Laporan Pengeluaran *Part*

<i>Use Case Name</i>	Mencetak data laporan pengeluaran
<i>Primary Actor</i>	Kepala Bagian
<i>Brief Description</i>	Proses mencetak laporan pengeluaran <i>part</i> .
<i>Relationship</i>	<i>Association:</i> Kepala Bagian <i>Include:</i> Login, Melihat Pengeluaran <i>Part</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Bagian masuk ke menu utama 2. Kepala Bagian memilih menu laporan, pilih sub pengeluaran 3. Sistem menampilkan laporan pengeluaran 4. Kepala Bagian mencetak laporan pengeluaran
<i>Precondition</i>	Kepala Bagian telah melakukan <i>login</i> terlebih dahulu

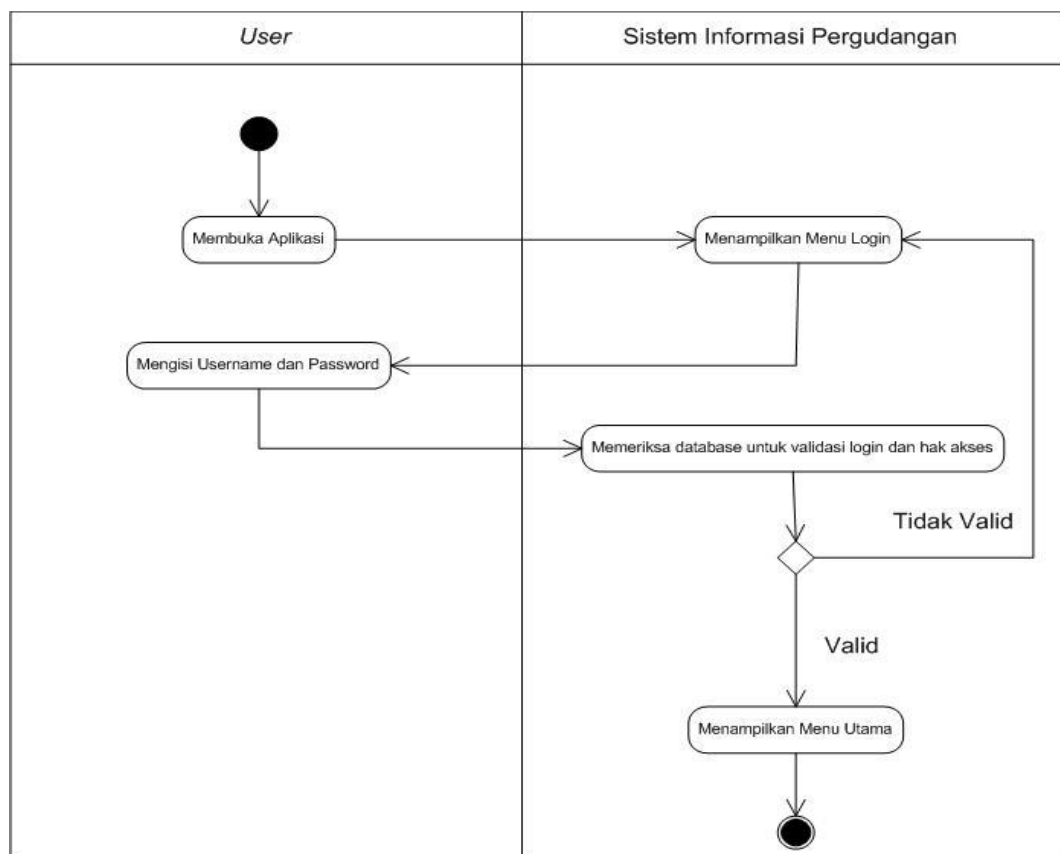
Sumber: Hasil Analisis (2017)

5.4.2 Activity Diagram

Activity Diagram digunakan untuk menggambarkan aliran kerja tiap *use case* pada sistem informasi pergudangan. *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.



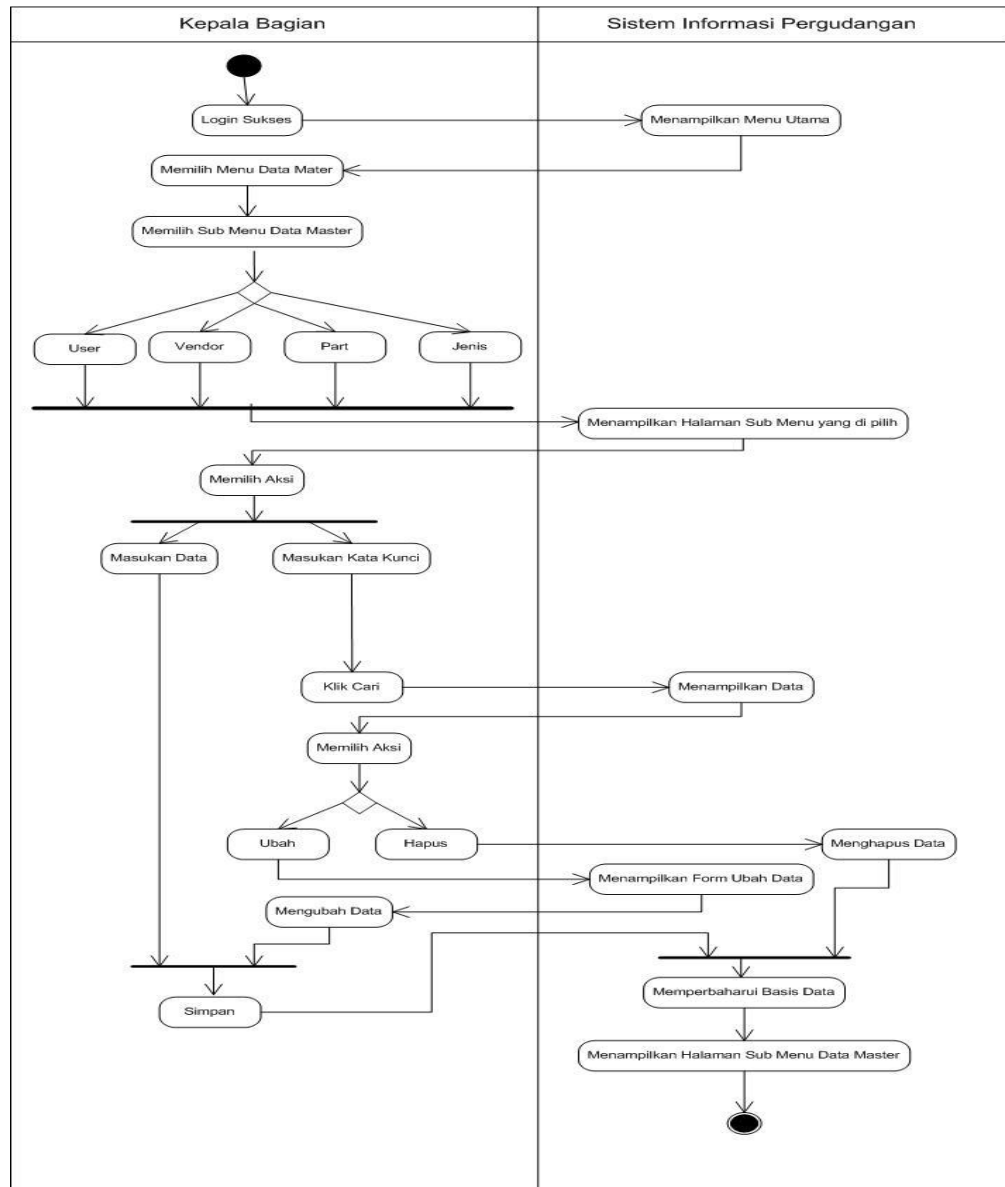
Gambar V.3 *Activity Diagram* Proses Login

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 Kepala Bagian dapat

menambah, mencari, mengubah dan menghapus data *user*. *Activity diagram* proses mengelola data *master user* digambarkan pada Gambar V.4



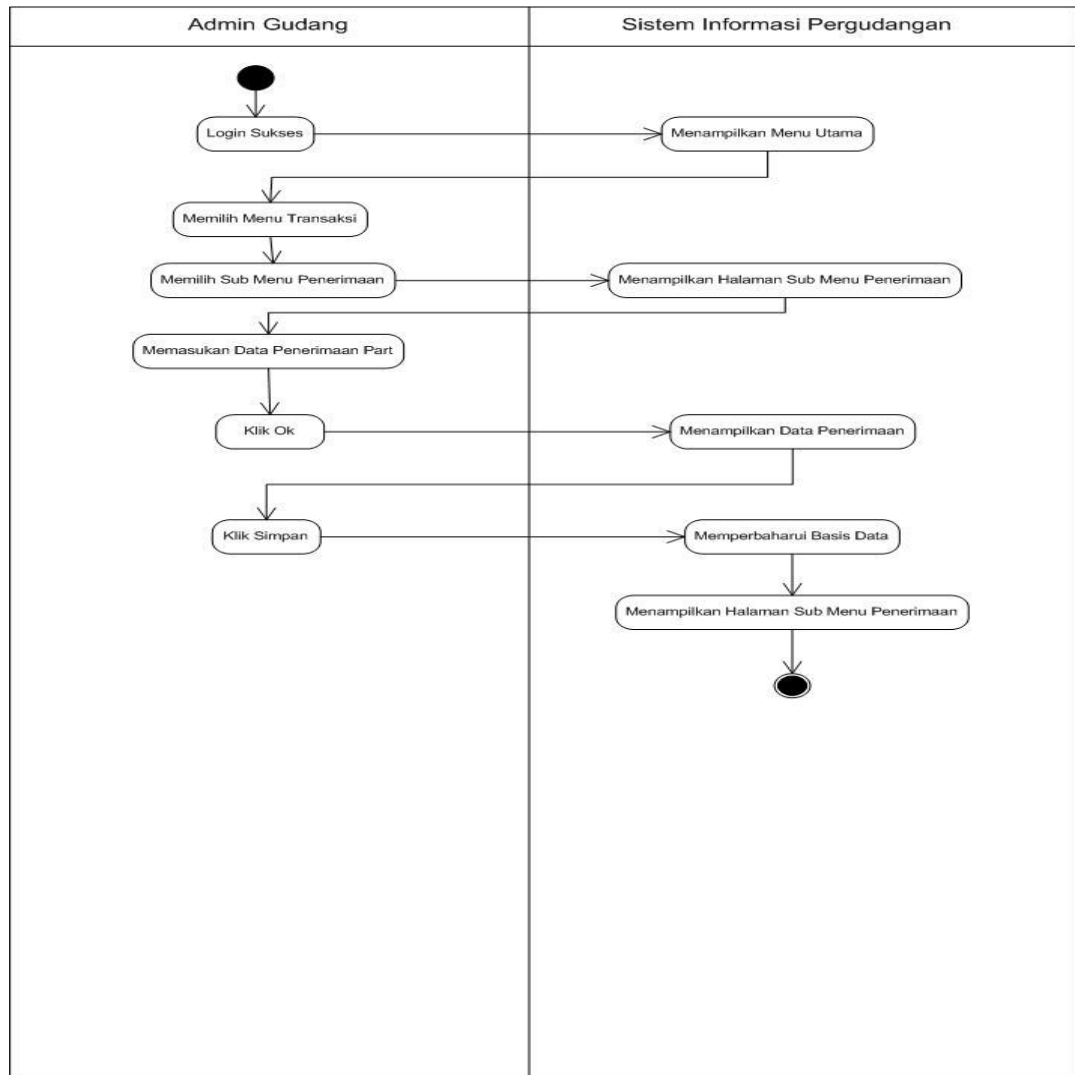
Gambar V.4 *Activity Diagram* Proses Mengelola Data *Master*

Sumber: Hasil Analisis (2017)

3. *Activity Diagram* Proses Mengisi Data Penerimaan *Part*

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat mengisi data penerimaan *part*, dimana Admin Gudang menerima dokumen-dokumen (surat jalan dan *delivery note*) dari Gudang kemudian meng-*input* data penerimaan *part* yang masuk ke database untuk memudahkan proses

pengolahan data berikutnya. *Activity diagram* proses mengisi data penerimaan *part* digambarkan pada Gambar V.5.

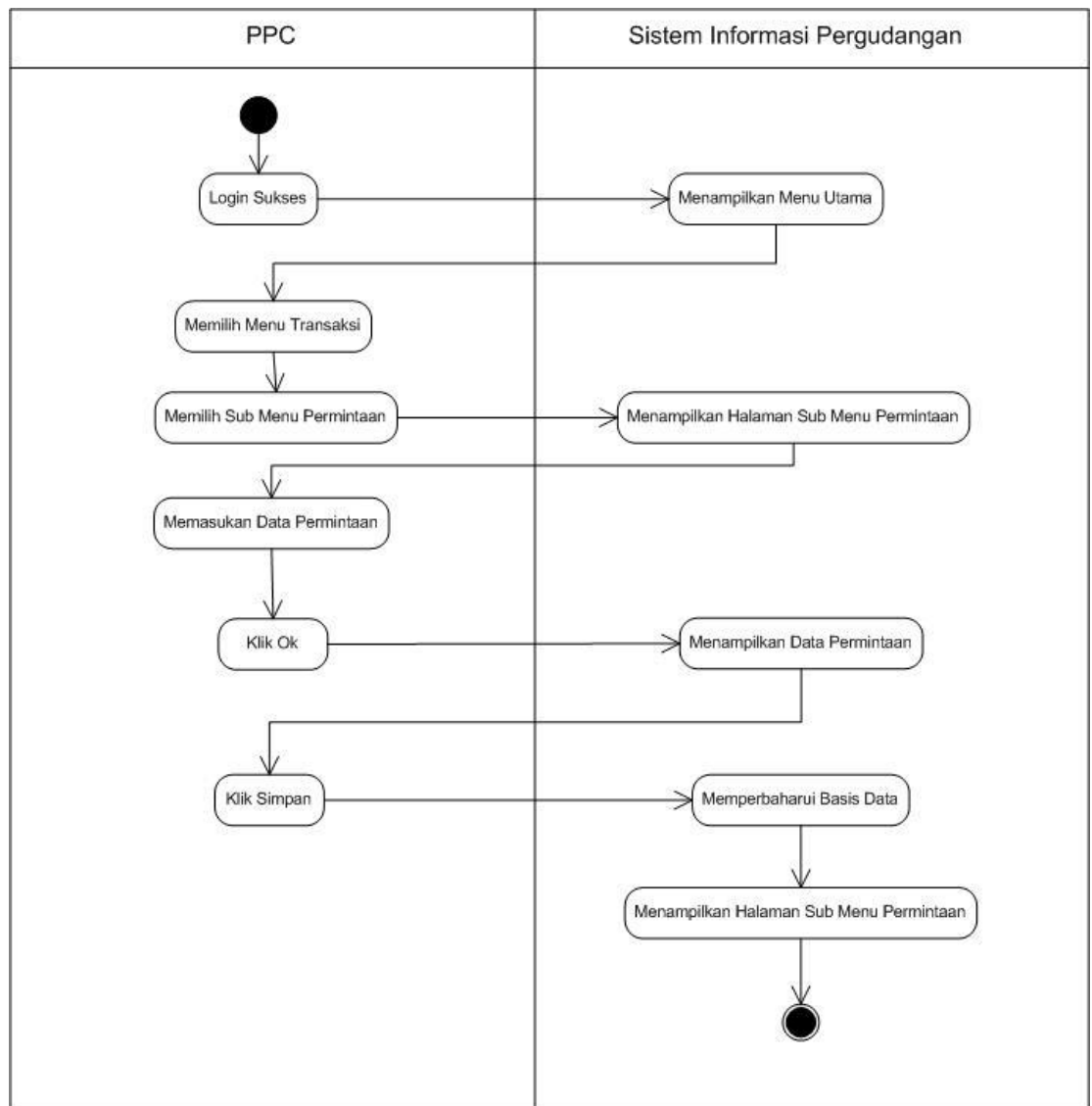


Gambar V.5 *Activity Diagram* Proses Mengisi Data Penerimaan *Part*

Sumber: Hasil Analisis (2017)

4. *Activity Diagram* Mengisi Data Permintaan *Part*

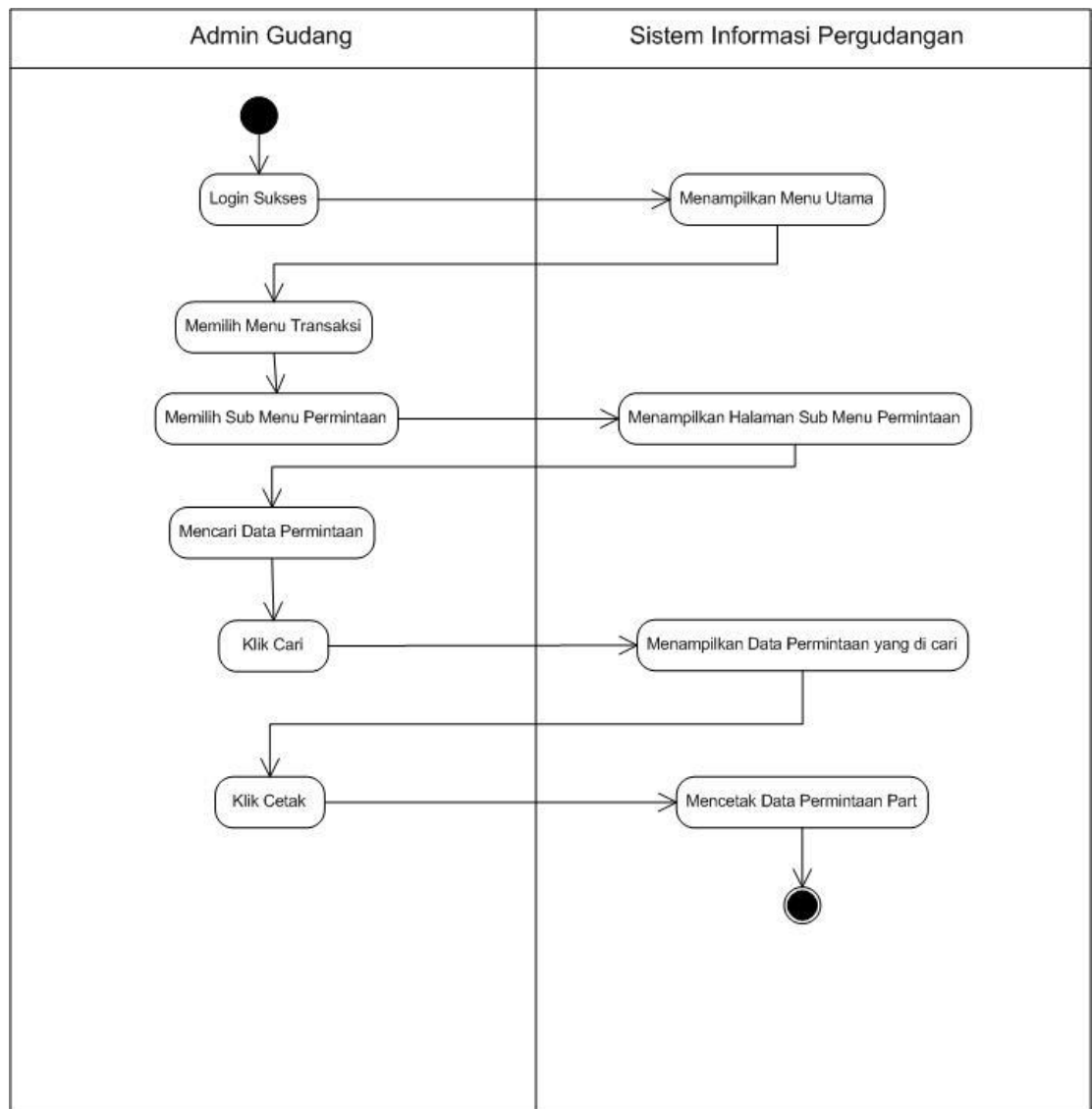
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mengisi data permintaan *part*, dimana PPC meng-*input* data permintaan *part* kemudian disimpan ke *database* untuk memudahkan proses pengolahan data berikutnya. *Activity diagram* proses mengisi data permintaan *part* digambarkan pada Gambar V.6.



Gambar V.6 *Activity Diagram* Proses Mengisi Data Permintaan *Part*
 Sumber: Hasil Analisis (2017)

5. *Activity Diagram* Mencetak Data Permintaan *Part*

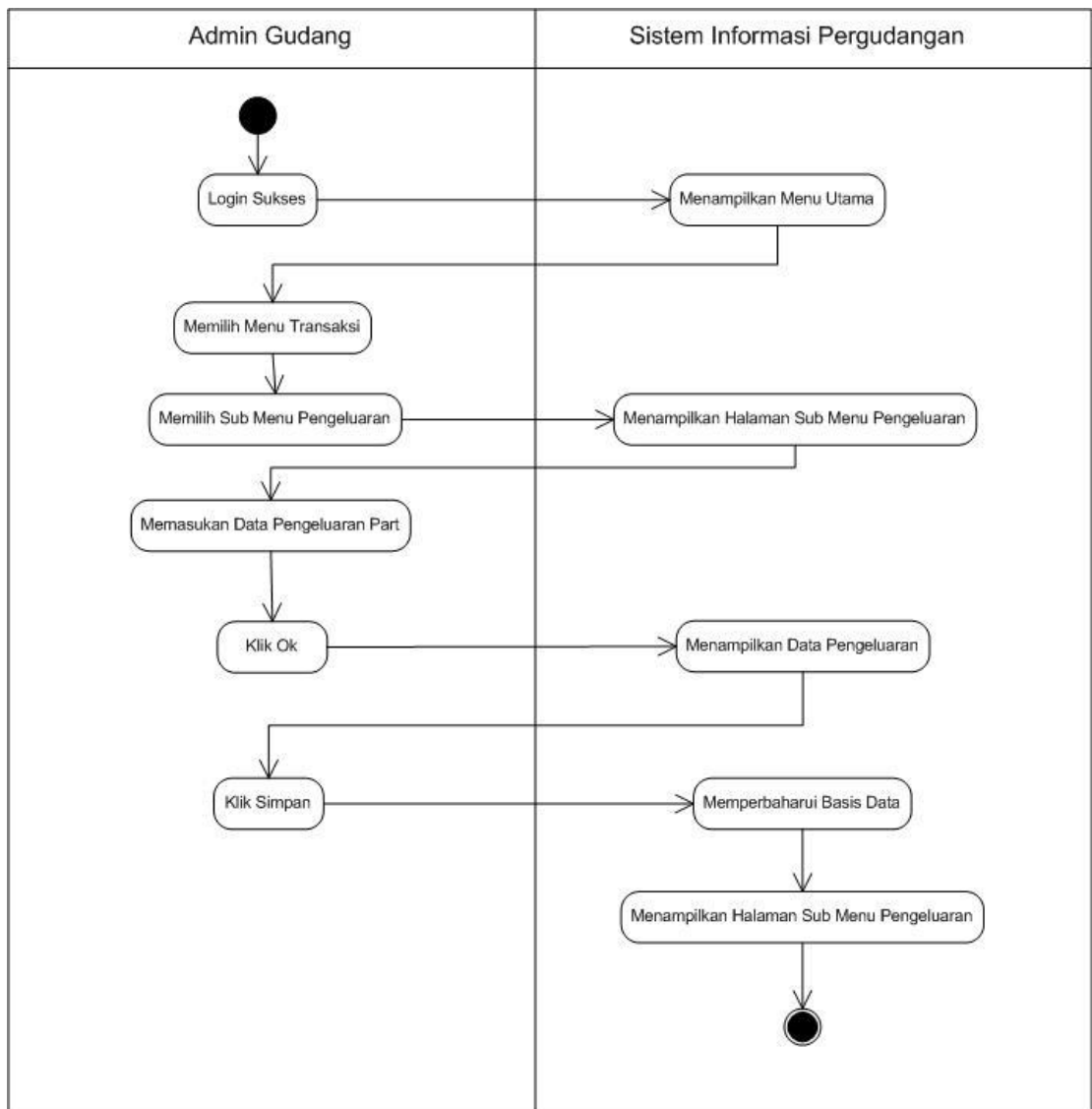
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mencetak permintaan *part*, dimana Admin Gudang mencetak data permintaan *part*. *Activity diagram* proses mencetak data permintaan *part* digambarkan pada Gambar V.7.



Gambar V.7 Activity Diagram Proses Mencetak Data Permintaan Part
 Sumber: Hasil Analisis (2017)

6. Activity Diagram Mengisi Data Pengeluaran Part

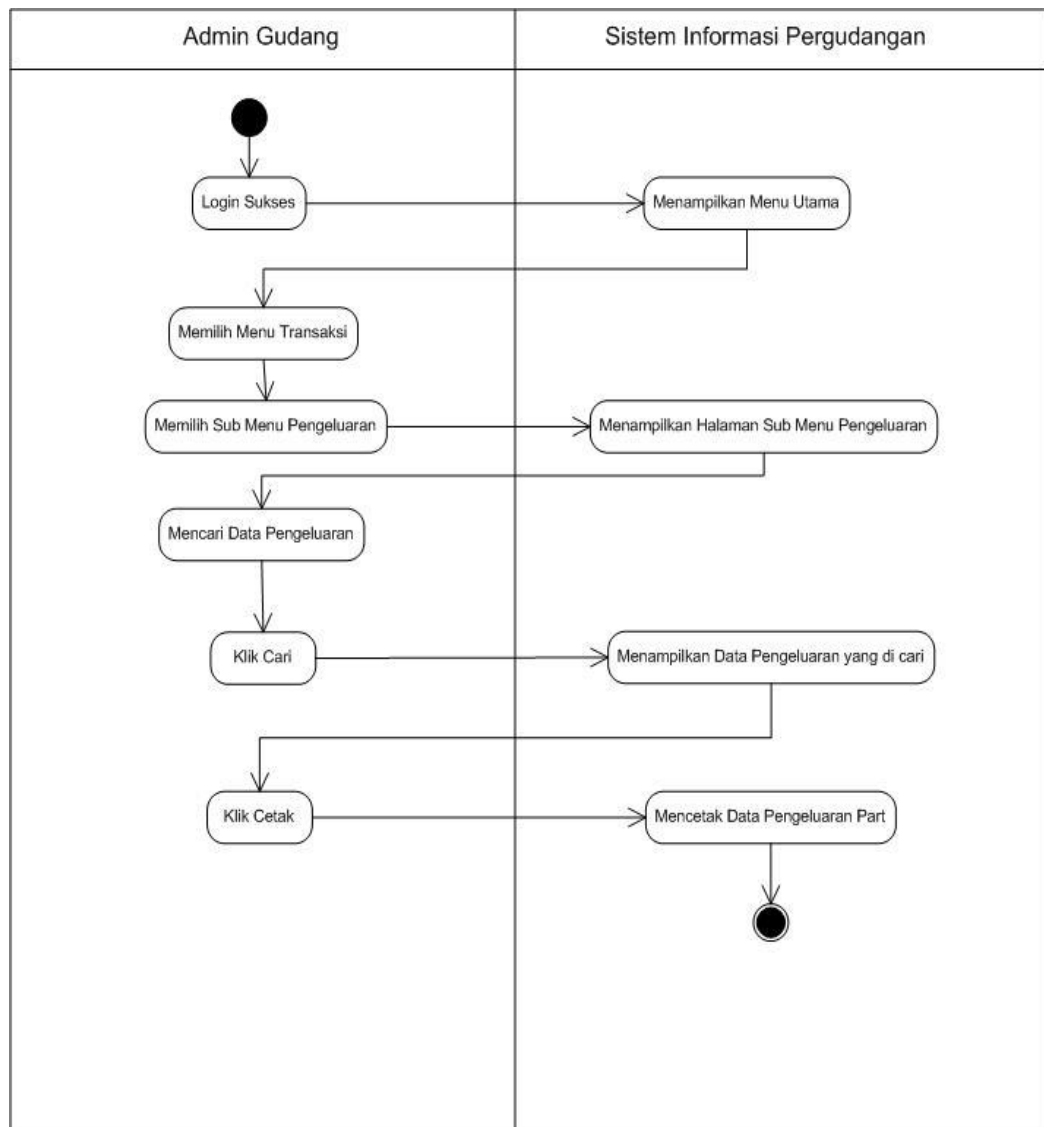
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mengisi data pengeluaran *part*, dimana Admin Gudang meng-*input* data pengeluaran *part* kemudian disimpan ke *database* untuk memudahkan proses pengolahan data berikutnya. Activity diagram proses mengisi data pengeluaran *part* digambarkan pada Gambar V.8.



Gambar V.8 Activity Diagram Proses Mengisi Data Pengeluaran Part
Sumber: Hasil Analisis (2017)

7. ActivityDiagram Mencetak Data Pengeluaran Part

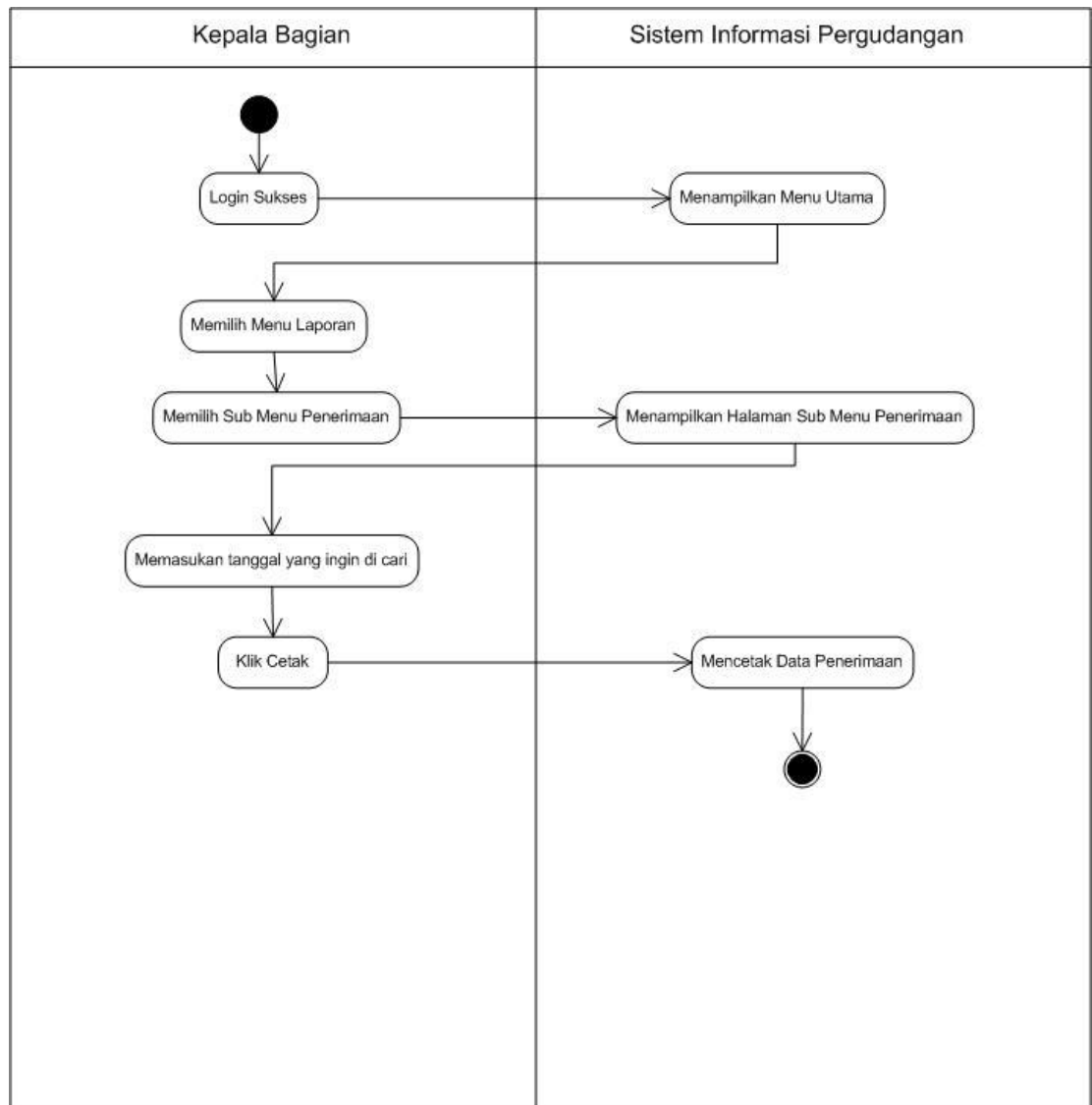
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mencetak data pengeluaran *part*, dimana Admin Gudang mencetak data pengeluaran *part*. *Activity diagram* proses mencetak data pengeluaran *part* digambarkan pada Gambar V.9.



Gambar V.9 *Activity Diagram* Proses Mencetak Data Pengeluaran *Part*
 Sumber: Hasil Analisis (2017)

8. *Activity Diagram* Mencetak Laporan Penerimaan *Part*

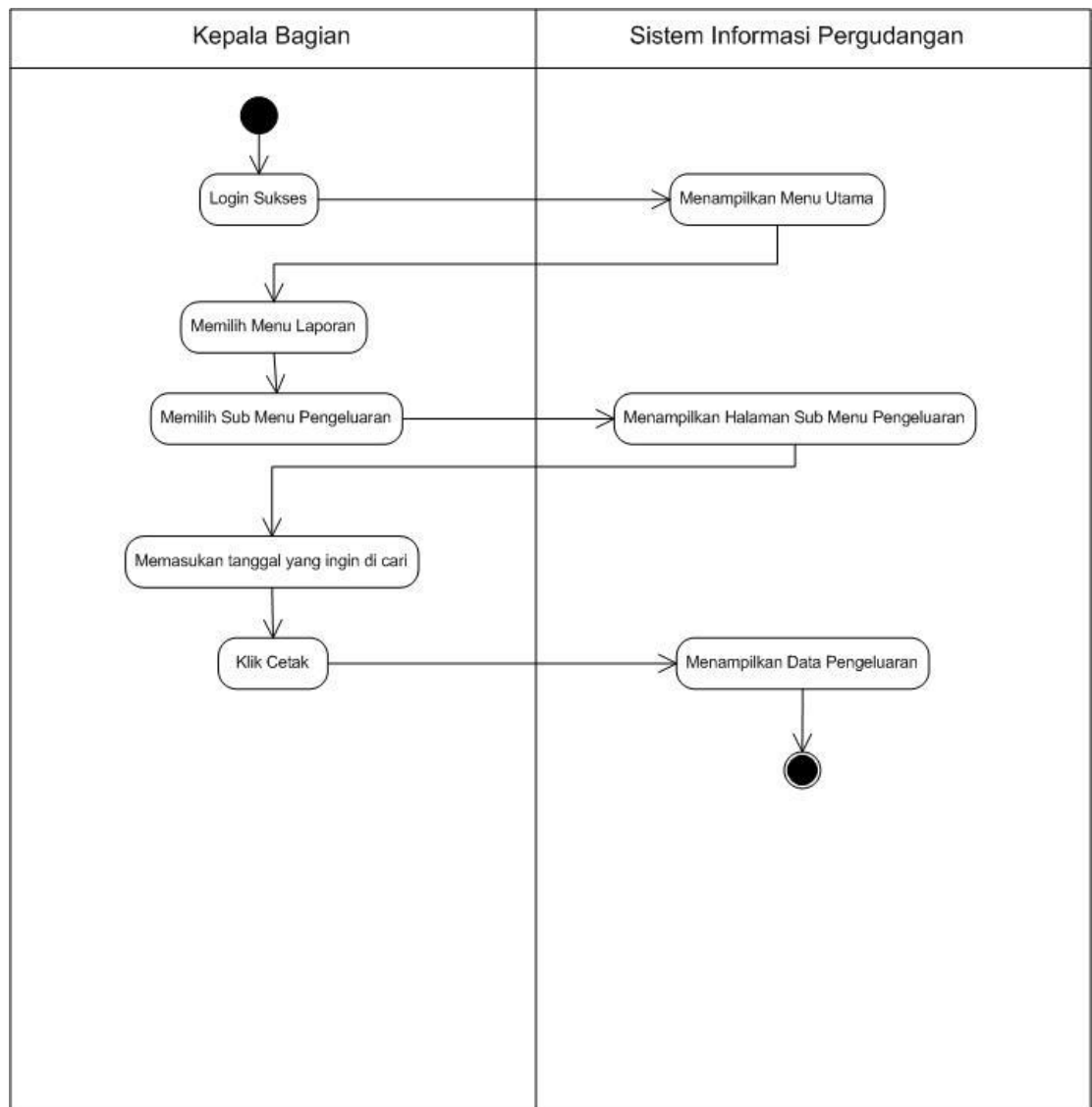
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mencetak laporan penerimaan, dimana Kepala Bagian mencetak laporan penerimaan. *Activity diagram* proses mencetak laporan penerimaan digambarkan pada Gambar V.10.



Gambar V.10 *Activity Diagram* Proses Mencetak Laporan Penerimaan *Part*
Sumber: Hasil Analisis (2017)

9. *Activity Diagram* Mencetak Laporan Pengeluaran *Part*

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mencetak laporan pengeluaran *part*, dimana Kepala Bagian mencetak laporan pengeluaran *part*. *Activity diagram* proses mencetak laporan pengeluaran *part* digambarkan pada Gambar V.11.



Gambar V.11 *Activity Diagram* Proses Mencetak Laporan Pengeluaran *Part*
Sumber: Hasil Analisis (2017)

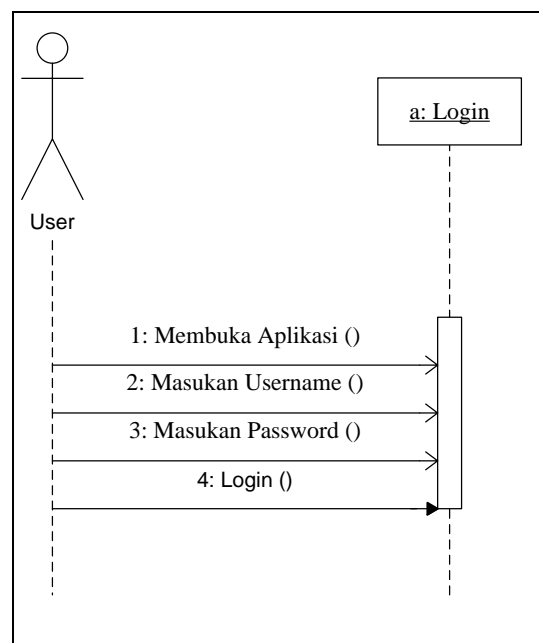
5.4.3 *Sequence Diagram*

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. Berikut adalah *sequence diagram* pada sistem informasi pergudangan usulan:

1. *Sequence Diagram* pada Proses *Login*

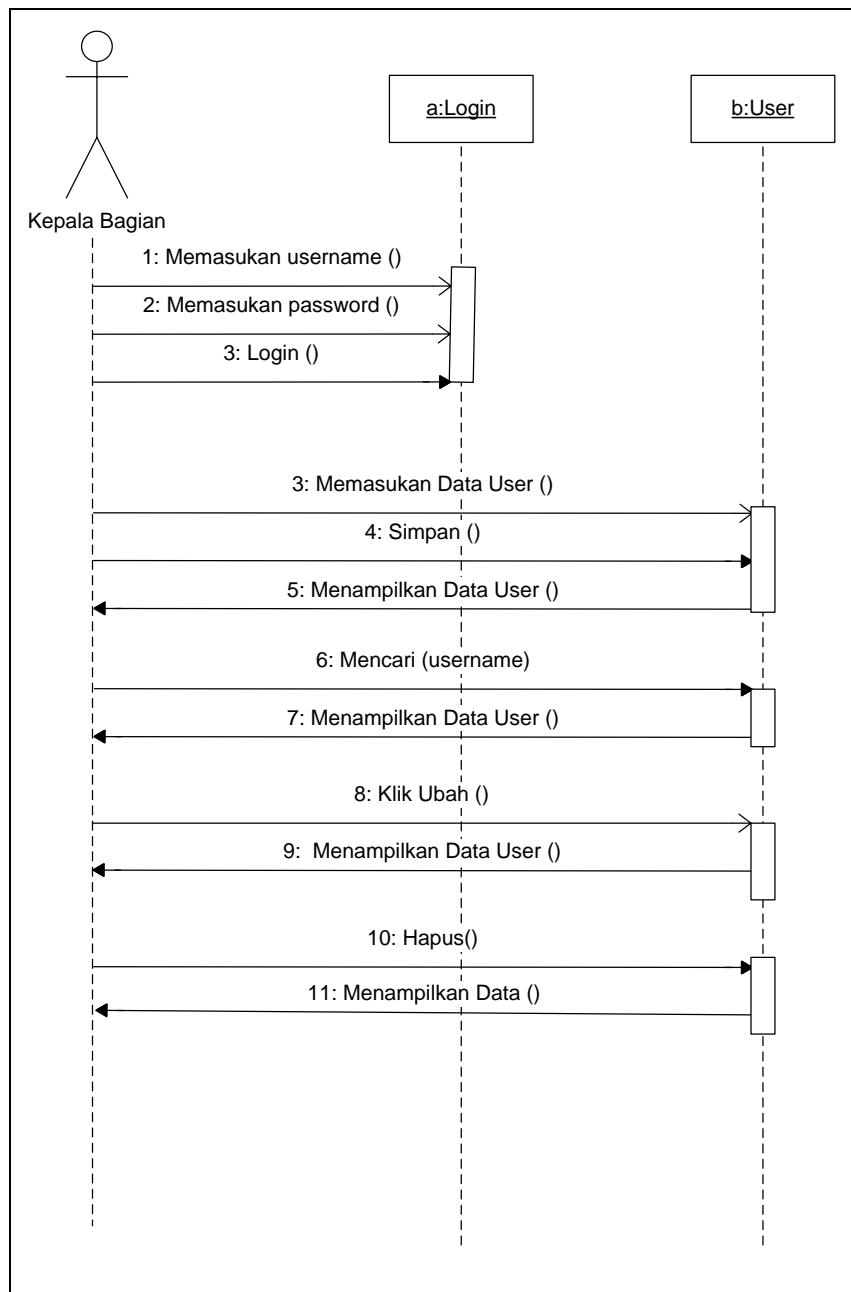
Sequence diagram login menggambarkan interaksi yang terjadi pada proses *login*. Proses ini dilakukan oleh seluruh aktor dalam *use case diagram*. *Sequence diagram* proses *login* dapat dilihat pada Gambar V.12 sebagai berikut :



Gambar V.12 *Sequence Diagram* Proses *Login*
Sumber: Hasil Analisis (2017)

2. *Sequence Diagram* Proses Mengelola Data Master *User*

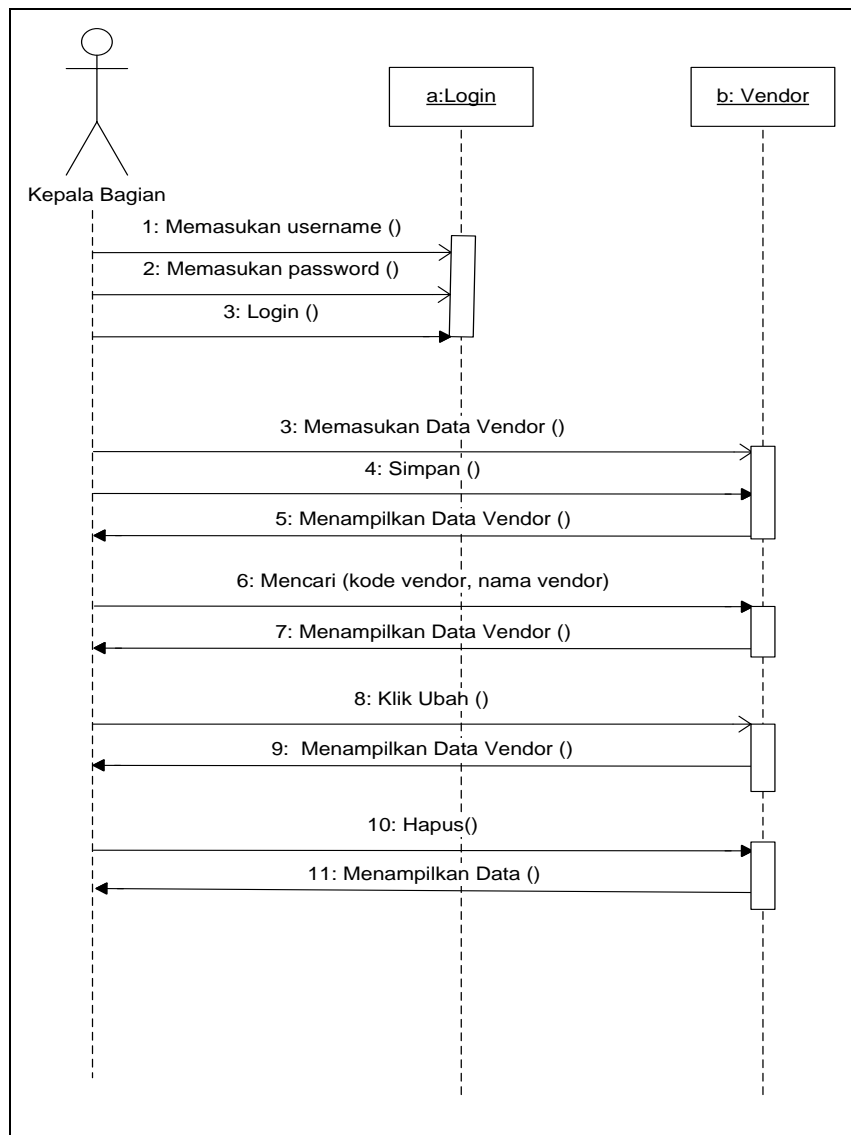
Sequence diagram mengelola data master *user* menggambarkan interaksi yang terjadi dalam mengelola data master seperti menambah, menghapus dan mengubah data master. Adapun *sequence diagram* proses mengelola data master *user* dapat dilihat pada Gambar V.13 sebagai berikut :



Gambar V.13 *Sequence Diagram* Mengelola Data Master User
 Sumber: Hasil Analisis (2017)

3. *Sequence Diagram* Proses Mengelola Data Master Vendor

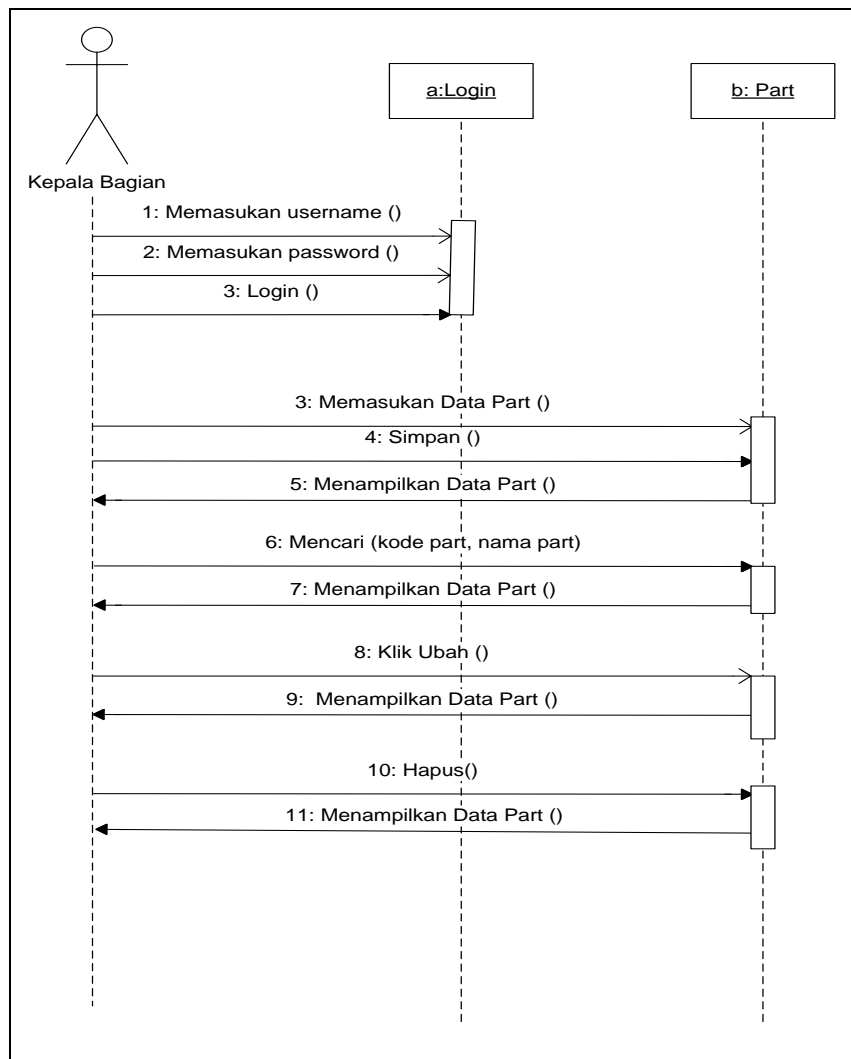
Sequence diagram mengelola data master vendor menggambarkan interaksi yang terjadi dalam mengelola data master seperti menambah, menghapus dan mengubah data master. Adapun *sequence diagram* proses mengelola data master vendor dapat dilihat pada Gambar V.14 sebagai berikut :



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

4. *Sequence Diagram* Proses Mengelola Data Master Part

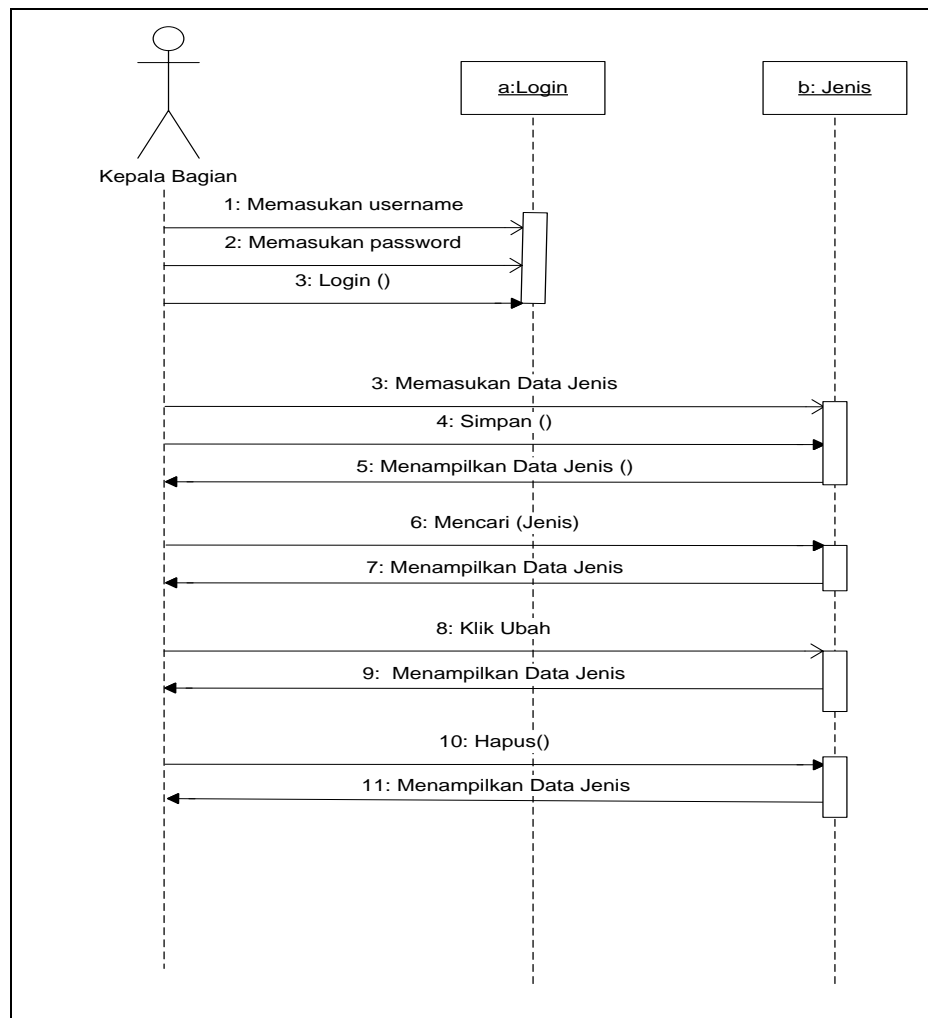
Sequence diagram mengelola data master *part* menggambarkan interaksi yang terjadi dalam mengelola data master seperti menambah, menghapus dan mengubah data master. Adapun *sequence diagram* proses mengelola data master *part* dapat dilihat pada Gambar V.15 sebagai berikut :



Gambar V.15 *Sequence Diagram* Mengelola Data Master Part
 Sumber: Hasil Analisis (2017)

5. *Sequence Diagram* Proses Mengelola Data Master Jenis

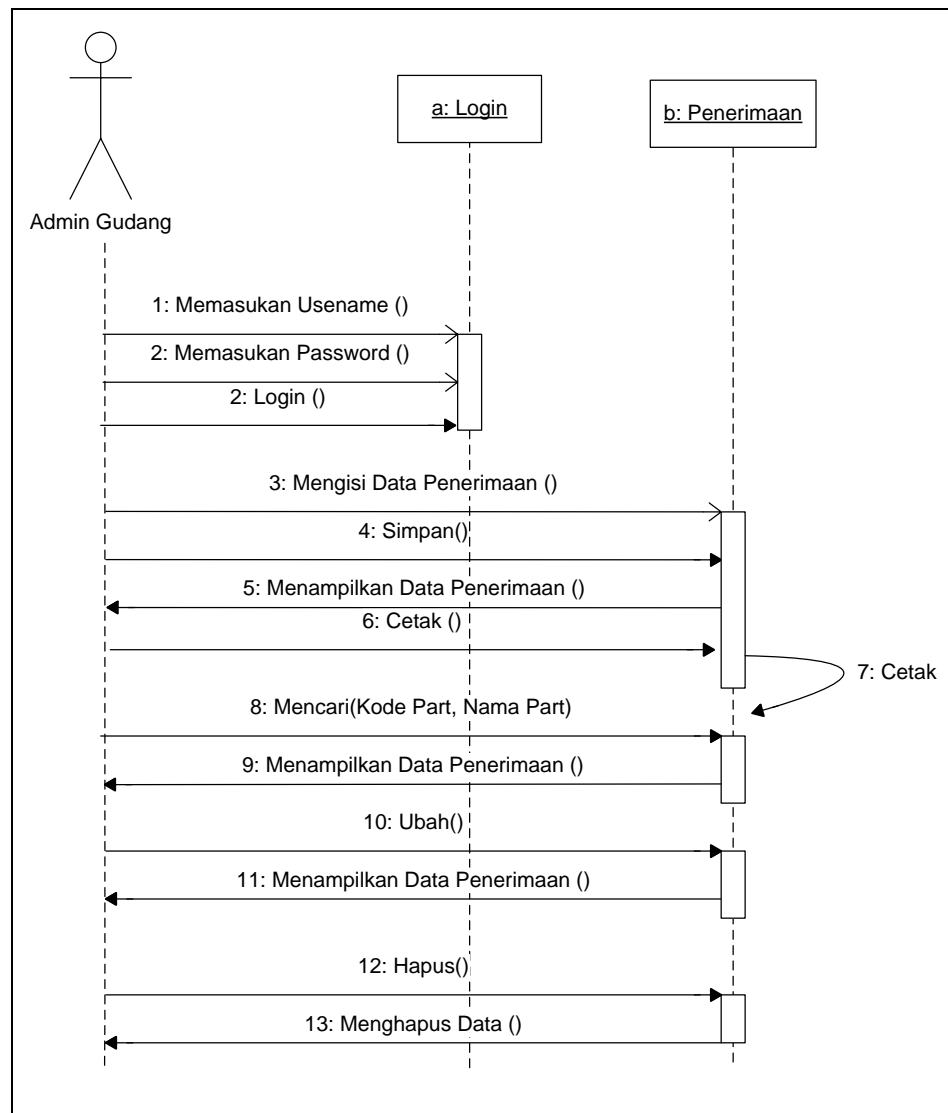
Sequence diagram mengelola data master jenis menggambarkan interaksi yang terjadi dalam mengelola data master seperti menambah, menghapus dan mengubah data master. Adapun *sequence diagram* proses mengelola data master jenis dapat dilihat pada Gambar V.16 sebagai berikut :



Gambar V.16 *Sequence Diagram* Mengelola Data Master Jenis
 Sumber: Hasil Analisis (2017)

6. *Sequence Diagram* Proses Penerimaan

Sequence diagram penerimaan menjelaskan sebuah *sequence diagram* dalam proses penerimaan. Adapun *sequence diagram* proses penerimaan dapat dilihat pada Gambar V.17 sebagai berikut :

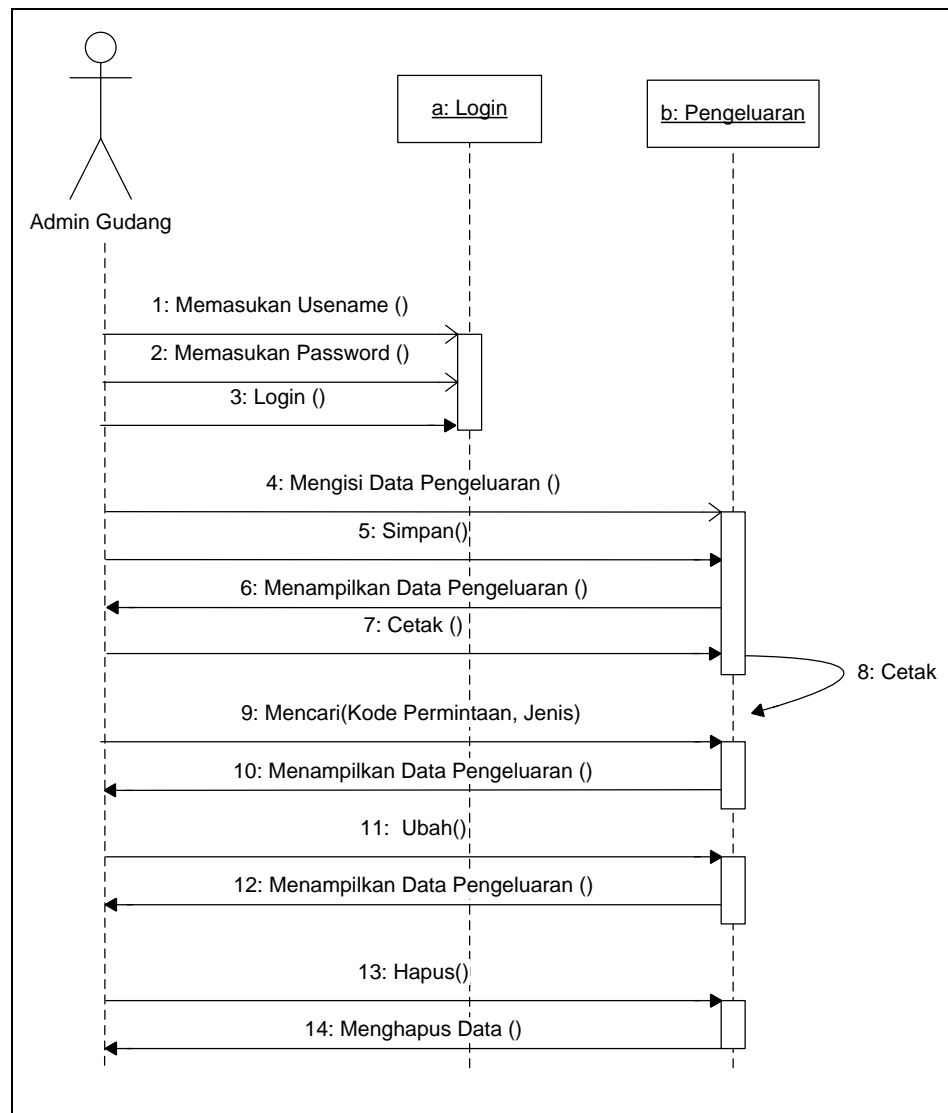


Gambar V.17 *Sequence Diagram* Proses Penerimaan

Sumber: Hasil Analisis (2017)

7. *Sequence Diagram* Proses Pengeluaran

Sequence diagram pengeluaran menjelaskan sebuah *sequence diagram* dalam proses pengeluaran. Adapun *sequence diagram* proses pengeluaran dapat dilihat pada Gambar V.18 sebagai berikut :

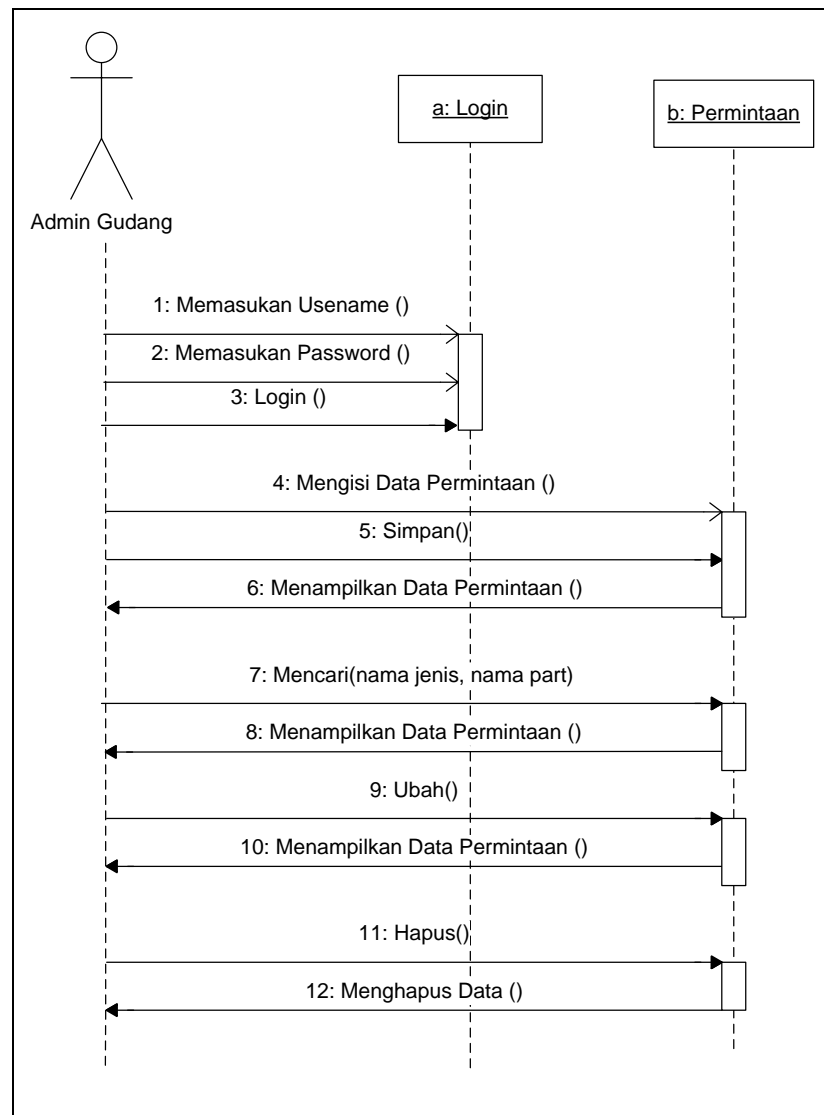


Gambar V.18 *Sequence Diagram* Proses Pengeluaran

Sumber: Hasil Analisis (2017)

8. *Sequence Diagram* Proses Permintaan

Sequence diagram permintaan menjelaskan sebuah *sequence diagram* dalam proses permintaan. Adapun *sequence diagram* proses permintaan dapat dilihat pada Gambar V.19 sebagai berikut :

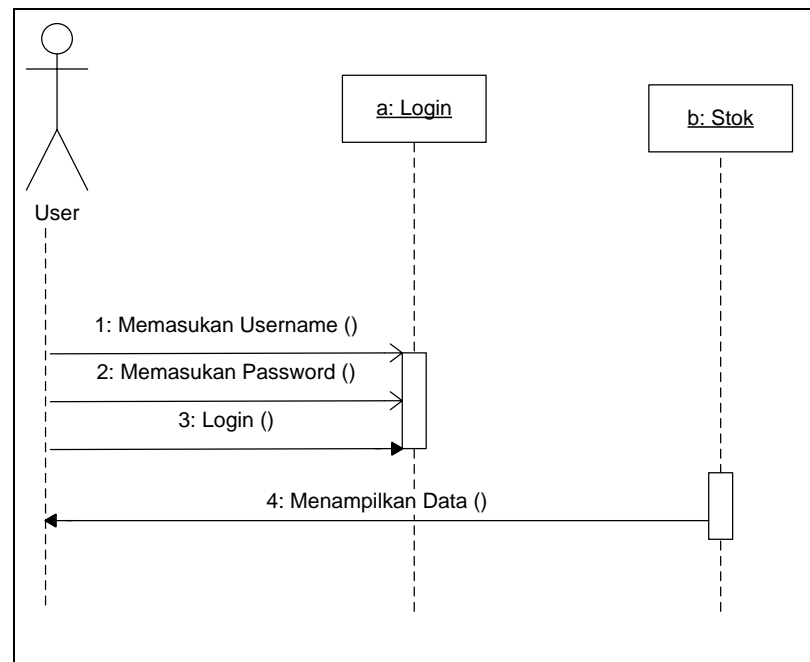


Gambar V.19 *Sequence Diagram* Proses Permintaan

Sumber: Hasil Analisis (2017)

9. *Sequence Diagram* Proses Stok

Sequence diagram stok menjelaskan sebuah *sequence diagram* dalam proses stok. Adapun *sequence diagram* proses data stok *part* dapat dilihat pada Gambar V.20 sebagai berikut .:

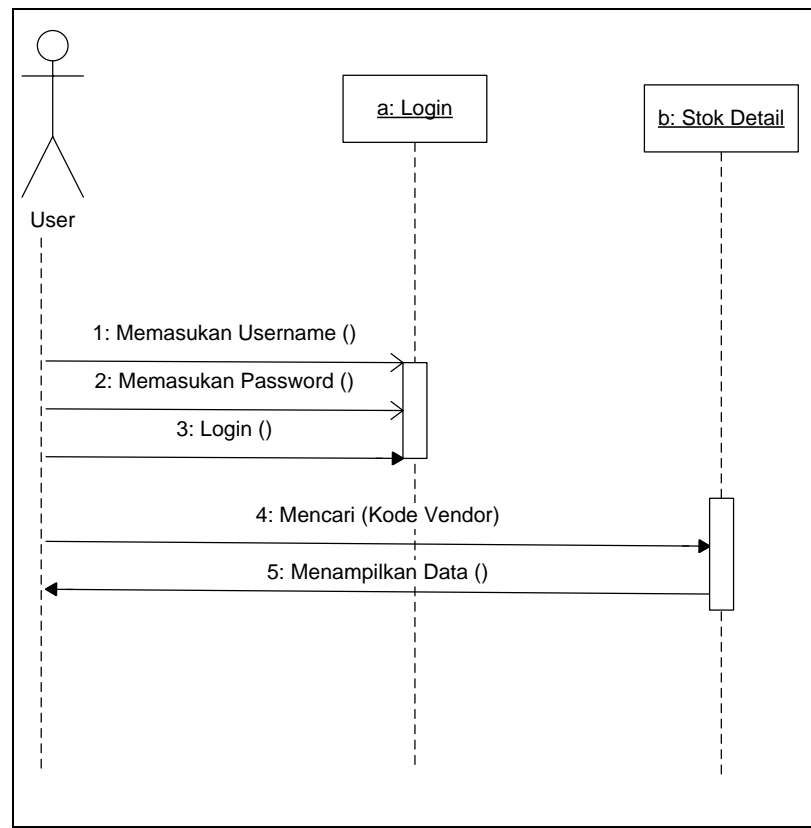


Gambar V.20 *Sequence Diagram* Proses Stok

Sumber: Hasil Analisis (2017)

10. *Sequence Diagram* Proses Stok Detail

Sequence diagram stok menjelaskan sebuah *sequence diagram* dalam proses stok. Adapun *sequence diagram* proses data stok detail *part* dapat dilihat pada Gambar V.21 sebagai berikut .:

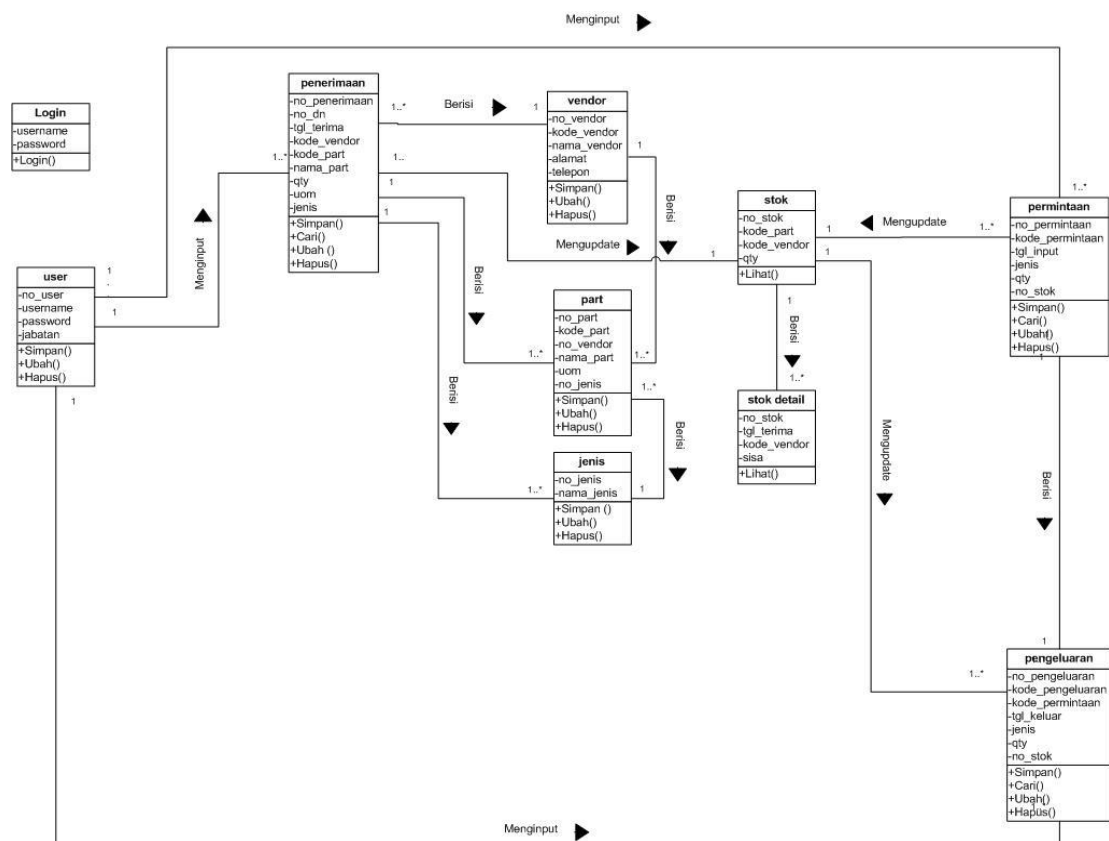


Gambar V.21 *Sequence Diagram* Proses Stok Detail

Sumber: Hasil Analisis (2017)

5.4.4 *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 pergudangan usulan dapat dilihat pada Gambar V.22 berikut ini:



Gambar V.22 *Class Diagram* Usulan

Sumber: Hasil Analisis (2017)

5.4.5 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 *datastore*. Berikut adalah kamus data sistem informasi pergudangan usulan:

1. Spesifikasi Tabel *User*

Nama Tabel : tbl_user

Fungsi : Untuk menyimpan data pengguna (*user*)

Tipe : Data Master

Tabel V.16 Tabel *User*

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id User	id_user	<i>Integer</i>	3	<i>Primary Key</i>
Username	Username	<i>Varchar</i>	15	
Password	Password	<i>Varchar</i>	100	
Jabatan	Jabatan	<i>Varchar</i>	20	

Sumber: Hasil Analisis (2017)

2. Spesifikasi Tabel *Vendor*

Nama Tabel : tbl_vendor

Fungsi : Untuk menyimpan data *vendor*

Tipe : Data Master

Tabel V.17 Tabel *Vendor*

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Vendor	id_vendor	<i>Integer</i>	5	<i>Primary Key</i>
Kode Vendor	kode_vendor	<i>Varchar</i>	10	
Nama Vendor	nama_vendor	<i>Varchar</i>	30	
Alamat	Alamat	<i>Varchar</i>	50	
Telepon	Telepon	<i>Varchar</i>	15	

Sumber: Hasil Analisis (2017)

3. Spesifikasi Tabel *Part*

Nama Tabel : tbl_part

Fungsi : Untuk menyimpan data *part*

Tipe : Data Master

Tabel V.18 Tabel *Part*

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Part	id_part	<i>Integer</i>	5	<i>Primary Key</i>
Kode Part	kode_part	<i>Varchar</i>	10	
Id Vendor	id_vendor	<i>Integer</i>	5	<i>Foreign Key</i>
Id Jenis	id_jenis	<i>Integer</i>	5	<i>Foreign Key</i>
Nama Part	nama_part	<i>Varchar</i>	30	
UOM	Uom	<i>Varchar</i>	15	

Sumber: Hasil Analisis (2017)

4. Spesifikasi Tabel Jenis

Nama Tabel : tbl_jenis

Fungsi : Untuk menyimpan data jenis per unit

Tipe : Data Jenis

Tabel V.19 Tabel Jenis

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Jenis	id_jenis	<i>Integer</i>	3	<i>Primary Key</i>
Nama Jenis	nama_jenis	<i>Varchar</i>	10	

Sumber: Hasil Analisis (2017)

5. Spesifikasi Tabel Stok

Nama Tabel : tbl_stok

Fungsi : Untuk menyimpan data stok

Tipe : Data Stok

Tabel V.20 Tabel Stok

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Stok	id_stok	<i>Integer</i>	5	<i>Primary Key</i>
Id Part	id_part	<i>Integer</i>	5	<i>Foreign Key</i>
Id Vendor	id_vendor	<i>Varchar</i>	5	<i>Foreign Key</i>
Qty	Qty	<i>Varchar</i>	15	

Sumber: Hasil Analisis (2017)

6. Spesifikasi Tabel Stok Detail

Nama Tabel : tbl_stok_detail

Fungsi : Untuk menyimpan data stok detail

Tipe : Data Stok

Tabel V.21 Tabel Stok Detail

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Stok	id_stok	<i>Integer</i>	5	<i>Primary Key</i>
Tanggal Terima	Tanggal	<i>Date</i>		
Id Part	id_part	<i>Varchar</i>	10	<i>Foreign Key</i>
Qty	qty	<i>Integer</i>	11	

Sisa	Sisa	Varchar	15	
------	------	---------	----	--

Sumber: Hasil Analisis (2017)

7. Spesifikasi Tabel Penerimaan

Nama Tabel : tbl_penerimaan

Fungsi : Untuk menyimpan data penerimaan

Tipe : Data Transaksi

Tabel V.22 Tabel Penerimaan

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Penerimaan	id_penerimaan	Integer	5	Primary Key
Nomor DN	no_dn	Varchar	15	
Tanggal Terima	tgl_terima	Date		
Id Vendor	id_vendor	Varchar	10	Foreign Key
Id Part	id_part	Varchar	15	Foreign Key
Qty Terima	qty_terima	Integer	5	

Sumber: Hasil Analisis (2017)

8. Spesifikasi Tabel Pengeluaran

Nama Tabel : tbl_pengeluaran

Fungsi : Untuk menyimpan data pengeluaran

Tipe : Data Transaksi

Tabel V.23 Tabel Pengeluaran

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Pengeluaran	id_pengeluaran	Integer	5	Primary Key
Kode Pengeluaran	kode_pengeluaran	Varchar	10	
Kode Permintaan	kode_permintaan	Varchar	10	
Tanggal Keluar	tgl_keluar	Date		
Id Part	id_part	Integer	5	Foreign Key
Qty	Qty_keluar	Integer	5	

Sumber: Hasil Analisis (2017)

9. Spesifikasi Tabel Permintaan

Nama Tabel : tbl_permintaan

Fungsi : Untuk menyimpan data permintaan

Tipe : Data Transaksi

Tabel V.24 Tabel Permintaan

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Permintaan	id_permintaan	<i>Integer</i>	5	<i>Primary Key</i>
Kode Permintaan	Kode_permintaan	<i>Varchar</i>	10	
Tanggal Input	tgl_input	<i>Date</i>		
Id Part	id_part	<i>Integer</i>	5	<i>Foreign Key</i>
Qty	qty_permintaan	<i>Integer</i>	5	

Sumber: Hasil Analisis (2017)

10. Spesifikasi Tabel Laporan Penerimaan

Nama Tabel : tbl_laporan_penerimaan

Fungsi : Untuk menyimpan data laporan penerimaan

Tipe : Data Laporan

Tabel V.25 Tabel Laporan Penerimaan

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Laporan Penerimaan	id_laporan_penerimaan	<i>Integer</i>	5	<i>Primary Key</i>
No Delivery Note	no_dn	<i>Varchar</i>	15	
Tanggal Terima	tgl_terima	<i>Date</i>		
Id Part	id_part	<i>Integer</i>	5	<i>Foreign Key</i>
Nama Part	nm_part	<i>Varchar</i>	30	

Sumber: Hasil Analisis (2017)

11. Spesifikasi Tabel Laporan Pengeluaran

Nama Tabel : tbl_laporan_pengeluaran

Fungsi : Untuk menyimpan data pengeluaran *part*

Tipe : Data Laporan

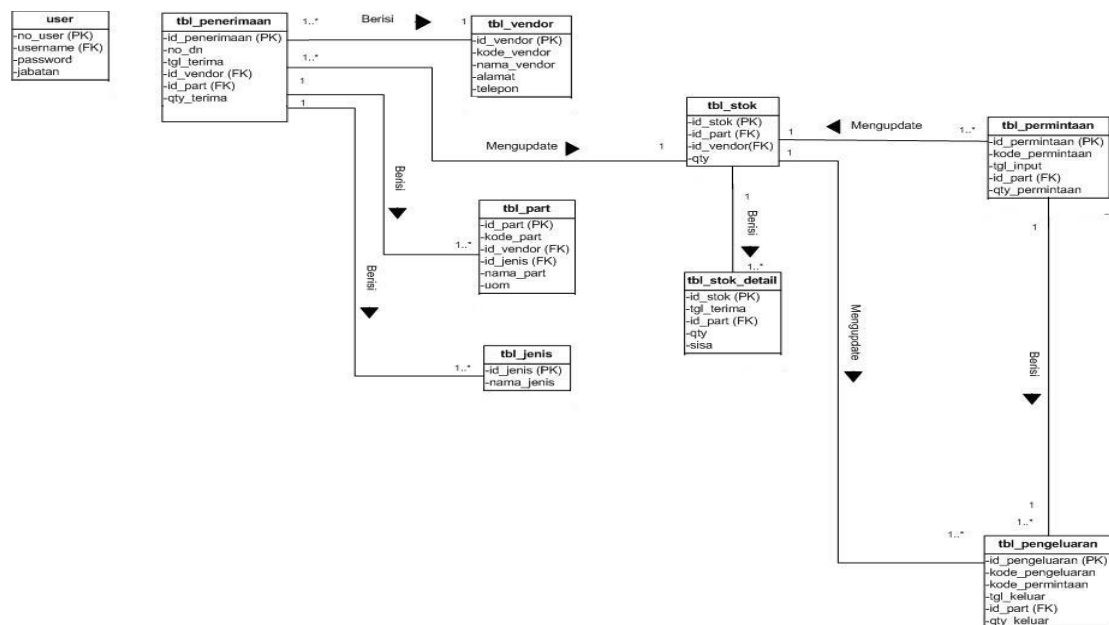
Tabel V.26 Tabel Laporan Pengeluaran

Nama Field	Akronim	Tipe	Panjang	Keterangan
Id Laporan Pengeluaran	id_laporan_pengeluaran	Integer	5	Primary Key
Id Pengeluaran	id_pengeluaran	Integer	5	Foreign Key
Id Permintaan	id_permintaan	Integer	5	Foreign Key
Tanggal Keluar	tgl_keluar	Date		
Jenis	Jenis	Varchar	15	
Qty	qty_keluar	Varchar	15	

Sumber: Hasil Analisis (2017)

5.4.6 Relational Diagram

Relational Diagram merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data. *Relational Diagram* untuk sistem informasi pengiriman barang usulan dapat dilihat pada Gambar V.23 sebagai berikut:

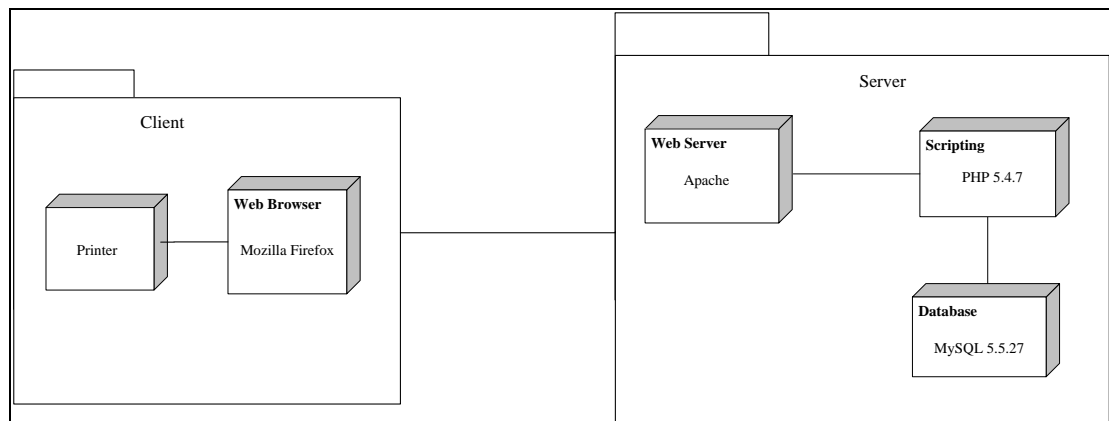


Gambar V.23 *Relational Diagram* Sistem Informasi Pergudangan

Sumber: Hasil Analisis (2017)

5.4.7 Deployment Diagram

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



Gambar V.24 *Deployment Diagram* Sistem Informasi Pergudangan

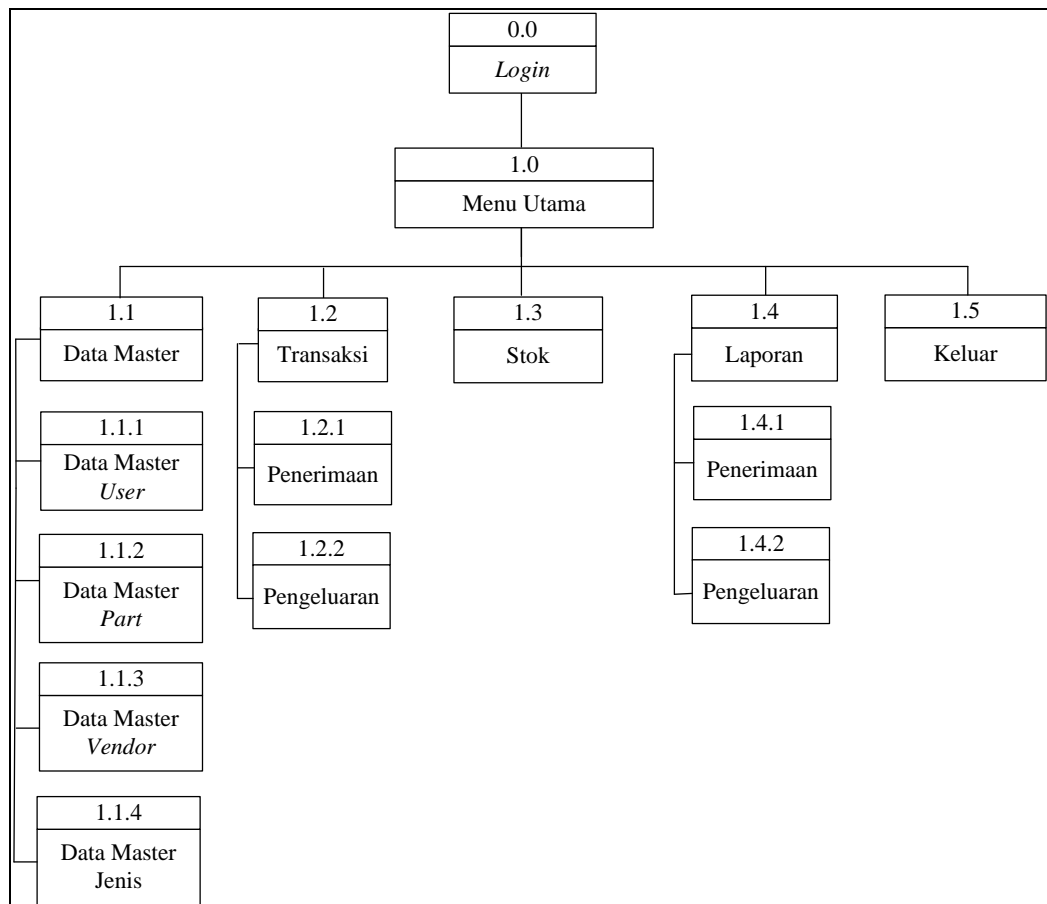
Sumber: Hasil Analisis (2017)

Berikut adalah penjelasan Gambar V.24 *Deployment Diagram* sistem informasi pergudangan :

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

5.4.8 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 pergudangan usulan digambarkan pada Gambar V.25 berikut ini.

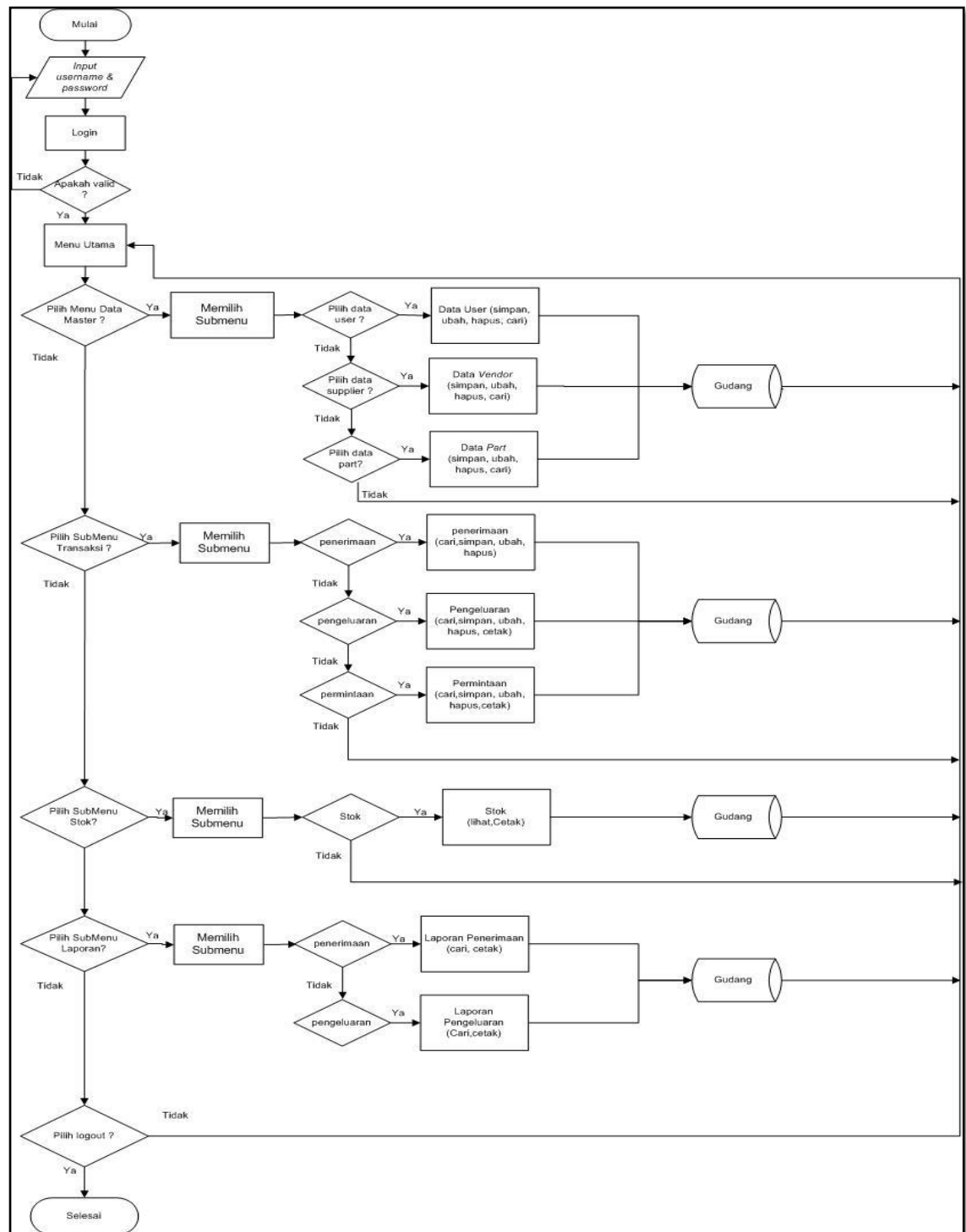


Gambar V.25 HIPO Sistem Informasi Pergudangan

Sumber: Hasil Analisis (2017)

5.4.9 Flowchart Aplikasi Usulan

Flowchart yang digunakan untuk mendokumentasikan aplikasi sistem informasi pergudangan 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 pergudangan dapat dilihat pada Gambar V.26



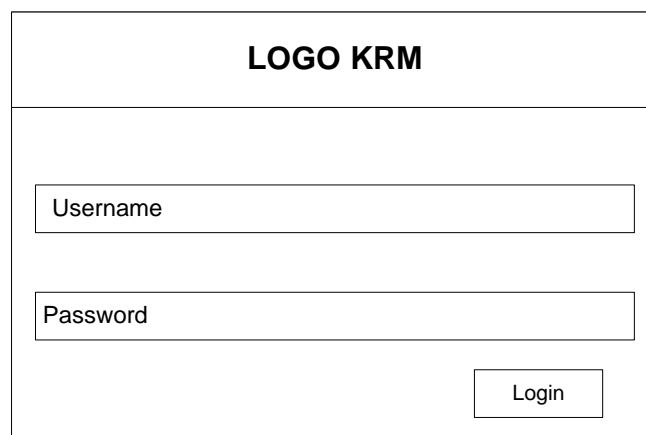
Gambar V.26 Flowchart Sistem Informasi Pergudangan
Sumber: Hasil Analisis (2017)

5.4.10 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.



The image shows a login form titled "LOGO KRM". It contains two input fields: "Username" and "Password". Below these fields is a "Login" button.

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

2. *Form 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 dibawah ini :

Logo KRM	SISTEM INFORMASI PERGUDANGAN PT. Krama Yudha Ratu Motor	Logout
<ul style="list-style-type: none">• Home• Data Master• Transaksi• Stok• Laporan	<div style="border: 1px solid black; height: 300px; width: 100%;"></div>	
	Footer	

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

3. *Form* Menu Data Master *User*

Form menu data master *user* adalah *form* yang digunakan untuk menampilkan data *user* yang telah di *input* dan *form* data *user*. Rancangan *interface* dari data master *user* dapat dilihat pada gambar dibawah ini:

Logo		SISTEM INFORMASI PERGUDANGAN PT. Krama Yudha Ratu Motor		Logout																																				
<ul style="list-style-type: none"> • Home • Data Master • Data User • Data Vendor • Data Jenis • Data Part 		<p>Input User Sistem Informasi Pergudangan</p> <p>Username <input type="text"/></p> <p>Password <input type="text"/></p> <p>Jabatan <input type="text"/> ▼</p> <p><input type="button" value="Save Data"/></p>																																						
<ul style="list-style-type: none"> • Transaksi • Stok • Laporan 		<p>Data User Sistem Informasi Pergudangan</p> <p style="text-align: right;"><input type="button" value="Search"/> <input type="text"/></p> <table border="1"> <thead> <tr> <th>No</th> <th>Username</th> <th>Password</th> <th>Jabatan</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td>Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td>Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td>Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td>Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td>Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td>Ubah Hapus</td> </tr> </tbody> </table>				No	Username	Password	Jabatan	Action					Ubah Hapus					Ubah Hapus					Ubah Hapus					Ubah Hapus					Ubah Hapus					Ubah Hapus
No	Username	Password	Jabatan	Action																																				
				Ubah Hapus																																				
				Ubah Hapus																																				
				Ubah Hapus																																				
				Ubah Hapus																																				
				Ubah Hapus																																				
				Ubah Hapus																																				
Footer																																								

Gambar 5.29 Rancangan Menu Data Master *User*
 Sumber: Hasil Analisis (2017)

4. *Form* Menu Data Master *Vendor*

Form menu data master *vendor* adalah *form* yang digunakan untuk menampilkan data *user* yang telah di *input* dan *form* data *vendor*. Rancangan *interface* dari data master *vendor* dapat dilihat pada gambar dibawah ini :

Logo
KRM

SISTEM INFORMASI PERGUDANGAN
 PT. Krama Yudha Ratu Motor

Logout

- Home
- Data Master
- Data User
- Data Vendor
- Data Jenis
- Data Part
- Transaksi
- Stok
- Laporan

Input Vendor Sistem Informasi Pergudangan

Kode Vendor

Nama Vendor

Telepon

Alamat

Data Vendor Sistem Informasi Pergudangan

Search

No	Kode Vendor	Alamat	Telepon	Action
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus

Footer

Gambar V.30 Rancangan Menu Data Master *Vendor*

Sumber: Hasil Analisis (2017)

5. *Form* Menu Data Master *Part*

Form menu data master *part* jadi adalah *form* yang digunakan untuk menampilkan *input* data *part* dan data data *part* di-*input*. Rancangan *interface* dari data master *part* dapat dilihat pada gambar dibawah ini :

Logo KRM	SISTEM INFORMASI PERGUDANGAN PT. Krama Yudha Ratu Motor	Logout																																										
<ul style="list-style-type: none"> • Home • Data Master <ul style="list-style-type: none"> • Data User • Data Vendor • Data Jenis • Data Part • Transaksi • Stok • Laporan 	<div style="border: 1px solid black; padding: 5px;"> <p>Input Data Barang</p> <p>Vendor <input type="text"/></p> <p>Jenis <input type="text"/></p> <p>Kode Part <input type="text"/></p> <p>Kode Part <input type="text"/></p> <p>UOM <input style="width: 100px;" type="text"/> ▼</p> <p style="text-align: center;"><input type="button" value="Save Data"/></p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p>Data Part Sistem Informasi Pergudangan</p> <p style="text-align: right;">Search : <input style="width: 80px;" type="text"/></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No</th> <th>Kode Part</th> <th>Nama Part</th> <th>UOM</th> <th>Jenis</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">Ubah Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">Ubah Hapus</td> </tr> </tbody> </table> </div>		No	Kode Part	Nama Part	UOM	Jenis	Action						Ubah Hapus						Ubah Hapus						Ubah Hapus						Ubah Hapus						Ubah Hapus						Ubah Hapus
No	Kode Part	Nama Part	UOM	Jenis	Action																																							
					Ubah Hapus																																							
					Ubah Hapus																																							
					Ubah Hapus																																							
					Ubah Hapus																																							
					Ubah Hapus																																							
					Ubah Hapus																																							
Footer																																												

Gambar V.31 Rancangan Menu Data Master Part
Sumber: Hasil Analisis (2017)

6. *Form* Menu Data Master Jenis

Form menu data jenis jadi adalah *form* yang digunakan untuk menampilkan *input* data jenis dan data jenis yang di-*input*. Rancangan *interface* dari *form* data master jenis dapat dilihat pada gambar dibawah ini :

Logo

SISTEM INFORMASI PERGUDANGAN
PT. Krama Yudha Ratu Motor

Logout

- Home
- Data Master
- Data User
- Data Vendor
- Data Jenis
- Data Part
- Transaksi
- Stok
- Laporan

Input Jenis Sistem Informasi Pergudangan

Nama Jenis

Data Jenis Sistem Informasi Pergudangan

Search :

No	Nama Jenis	Action
		Ubah Hapus
		Ubah Hapus
		Ubah Hapus
		Ubah Hapus
		Ubah Hapus
		Ubah Hapus

Footer

Gambar V.32 Rancangan Menu Data Master Jenis

Sumber: Hasil Analisis (2017)

7. *Form* Menu Transaksi Penerimaan

Form menu transaksi penerimaan adalah *form* yang digunakan untuk menampilkan *input* data penerimaan dan data penerimaan yang telah di-*input*. Rancangan *interface* dari menu transaksi penerimaan dapat dilihat pada gambar dibawah ini :

Logo
KRM

SISTEM INFORMASI PERGUDANGAN
PT. Krama Yudha Ratu Motor

Logout

- Home
- Data Master
- Transaksi
- Penerimaan
- Pengeluaran
- Permintaan
- Stok
- Laporan

Input Transaksi Penerimaan Sistem Informasi Pergudangan

No.DN

Tanggal Terima

Vendor

Nama Part

Qty Penerimaan

Data Penerimaan

Search

No	No. DN	Tanggal Terima	Nama Vendor	Nama Part	UOM	Jenis	Qty	Action
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus

Footer

Gambar V.33 Rancangan Menu Transaksi Penerimaan

Sumber: Hasil Analisis (2017)

8. *Form* Menu Transaksi Pengeluaran

Form menu data transaksi pengeluaran adalah *form* yang digunakan untuk menampilkan input data pengeluaran dan data pengeluaran yang telah di-*input*. Rancangan *interface* dari menu transaksi data pengeluaran dapat dilihat pada gambar dibawah ini :

Logo
KRM

SISTEM INFORMASI PERGUDANGAN
 PT. Krama Yudha Ratu Motor

Logout

- Home
- Data Master
- Transaksi
 - Penerimaan
 - Pengeluaran
 - Permintaan
- Stok
- Laporan

Input Transaksi Pengeluaran Sistem Informasi Pergudangan

Kode Pengeluaran

Nama Permintaan

Tanggal Keluar

Qty

Data Pengeluaran Search

No	Tanggal Input Permintaan	Kode Permintaan	Jenis	Qty Permintaan	Tanggal Keluar	Kode Keluar	Qty Keluar	Aksi
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus
								Ubah Hapus

Footer

Gambar V.34 Rancangan Menu Transaksi Pengeluaran
Sumber: Hasil Analisis (2017)

9. *Form* Menu Transaksi Permintaan

Form menu transaksi permintaan adalah *form* yang digunakan untuk menampilkan *input* data permintaan produksi dan data permintaan produksi yang telah di-*input*. Rancangan *interface* dari transaksi data permintaan dapat dilihat pada gambar dibawah ini :

Logo
KRM

SISTEM INFORMASI PERGUDANGAN
PT. Krama Yudha Ratu Motor

Logout

- Home
- Data Master
- Transaksi
- Penerimaan
- Pengeluaran
- Permintaan
- Stok
- Laporan

Input Transaksi Permintaan Sistem Informasi Pergudangan

Kode Permintaan

Tanggal Input

Jenis

Qty

Data Permintaan Search:

No	Kode Permintaan	Tanggal Input	Nama Jenis	Nama Part	Qty	Action
						Ubah Hapus
						Ubah Hapus
						Ubah Hapus
						Ubah Hapus
						Ubah Hapus
						Ubah Hapus

Footer

Gambar V.35 Rancangan Menu Transaksi Permintaan
Sumber: Hasil Analisis (2017)

10. *Form* Menu Data Stok

Form menu data stok adalah *form* yang digunakan untuk menampilkan data stok detail dan stok jenis per unit. Rancangan *interface* dari data stok dapat dilihat pada gambar dibawah ini :

Logo KRM	SISTEM INFORMASI PERGUDANGAN PT. Krama Yudha Ratu Motor	Logout																																																								
<ul style="list-style-type: none"> • Home • Data Master • Transaksi • Stok • Laporan 	<p style="text-align: center;"><u>Stok Part</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No</th> <th>Part</th> <th>Vendor</th> <th>Jenis</th> <th>Qty Terima</th> <th>Qty Keluar</th> <th>Jumlah Sekarang</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table> <p style="text-align: center;">Cetak Laporan Stok</p>		No	Part	Vendor	Jenis	Qty Terima	Qty Keluar	Jumlah Sekarang																																																	
No	Part	Vendor	Jenis	Qty Terima	Qty Keluar	Jumlah Sekarang																																																				
Footer																																																										

Gambar V.36 Rancangan Menu Data Stok Detail
Sumber: Hasil Analisis (2017)

Logo KRM	SISTEM INFORMASI PERGUDANGAN PT. Krama Yudha Ratu Motor	Logout																								
<ul style="list-style-type: none"> • Home • Data Master • Transaksi • Stok • Laporan 	<p style="text-align: center;"><u>Stok per Unit</u></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No</th> <th>Jenis</th> <th>Qty per unit</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td></tr> </tbody> </table>		No	Jenis	Qty per unit																					
No	Jenis	Qty per unit																								
Footer																										

Gambar V.37 Rancangan Menu Data Stok Per Unit

Sumber: Hasil Analisis (2017)

11. *Form* Menu Data Laporan Penerimaan

Form menu data laporan penerimaan adalah *form* yang digunakan untuk menampilkan input data laporan penerimaan dan data laporan penerimaan yang telah di-*input*. Rancangan *interface* dari *form* data laporan penerimaan dapat dilihat pada gambar dibawah ini :

The screenshot shows a web application interface for 'SISTEM INFORMASI PERGUDANGAN PT. Krama Yudha Ratu Motor'. At the top left is the 'Logo KRM' and at the top right is a 'Logout' button. A navigation menu on the left lists: Home, Data Master, Transaksi, Stok, Laporan (highlighted), Penerimaan (highlighted), and Pengeluaran. The main content area is titled 'Laporan Penerimaan Part' and contains a search bar, a table with the following columns: No, No. DN, Tanggal Terima, Vendor, Part, Qty, UOM, and Jenis. Below the table is a 'Cetak' button. The footer contains the text 'Footer'.

Gambar V.38 Rancangan Menu Data Laporan Penerimaan
Sumber: Hasil Analisis (2017)

12. *Form* Menu Data Laporan Pengeluaran

Form menu data pengeluaran adalah *form* yang digunakan untuk menampilkan input data pengeluaran dan data pengeluaran yang telah di-*input*. Rancangan *interface* dari *form* data pengeluaran dapat dilihat pada gambar dibawah ini :

Logo KRM	SISTEM INFORMASI PERGUDANGAN PT. Krama Yudha Ratu Motor	Logout																																																																		
<ul style="list-style-type: none"> • Home • Data Master • Transaksi • Stok • Laporan <ul style="list-style-type: none"> • Penerimaan • Pengeluaran 	<p>Laporan Penerimaan Part</p> <p>Tanggal Awal <input type="text"/></p> <p>Tanggal Akhir <input type="text"/></p> <p style="text-align: center;">Tampilkan <input type="button" value="Tampilkan"/></p>	<p>Laporan Pengeluaran Part</p> <p style="text-align: right;">Search <input type="text"/></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>No</th> <th>Kode Pengeluaran</th> <th>Kode Permintaan</th> <th>Tanggal Keluar</th> <th>Jenis</th> <th>Qty</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table> <p style="text-align: right;"><input type="button" value="Cetak"/></p>	No	Kode Pengeluaran	Kode Permintaan	Tanggal Keluar	Jenis	Qty																																																												
No	Kode Pengeluaran	Kode Permintaan	Tanggal Keluar	Jenis	Qty																																																															
Footer																																																																				

Gambar V.39 Rancangan *Form* Menu Data Laporan Pengeluaran
Sumber: Hasil Analisis (2017)

5.4.11 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 maka dapat disimpulkan bahwa:

1. Dengan penggunaan aplikasi sistem informasi pergudangan ini sebaiknya mampu mengurangi *human error* (data yang di *input double* dan tidak teliti saat penginputan) dalam proses pengisian data penerimaan *part* dan pengeluaran *part*.
2. Sistem informasi pergudangan yang telah terintegrasi dengan *database* diharapkan lebih cepat selesai jika dibandingkan dengan penginputan manual dengan *Microsoft Excel*.

6.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem informasi pergudangan adalah sebagai berikut :

1. Untuk penerapan sistem baru, sebaiknya dilakukan *training* kepada bagian-bagian yang terkait yang akan menggunakan sistem baru ini.
2. Sebaiknya dilakukan pemeliharaan terhadap sistem pergudangan secara berkala, sehingga sistem tersebut bisa berjalan dengan lancar.
3. Diharapkan bagi pembaca tugas akhir ini dapat membuat sistem informasi pergudangan menjadi sistem yang lebih luas dengan menambah beberapa fungsi tambahan, sehingga dapat meningkatkan efisiensi dalam menggunakan sistem.

DAFTAR PUSTAKA

- Anhar. 2010. *Panduan Menguasai PHP dan MySQL Secara Otodidak*. Jakarta: Mediakita.
- Bunafit, Nugroho. 2008. *Aplikasi Pemrograman Web Dinamis Menggunakan PHP dan MySQL*. Yogyakarta: Gava Media
- Handoko, 1984, *Dasar-Dasar Manajemen Produksi dan Operasi*, BFE, Yogyakarta.
- Jogiyanto, HM 1999, *Analisis & Desain Sistem Informasi : Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*, cetakan ketiga, Andi, Yogyakarta.
- Jogiyanto, HM. 2005. *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Edisi Ketiga Andi Offset, Yogyakarta.
- Kumorotomo, Wahyudi dan Margono, Subando Agus. *Sistem Informasi Manajemen*. Gadjah Mada University Press, Yogyakarta.
- Laudon, Kenneth C dan Jane P. Laudon. 2007. *Sistem Informasi Manajemen*. Edisi ke-10. Terjemahan Chriswan Sungkono dan Machmudin Eka P. Jakarta: Salemba Empat
- McLeod. 2004. *Sistem Informasi Manajemen*. PT. Indeks, Jakarta.
- McLeod, 2008. *Sistem Informasi Manajemen*. Edisi 10. Jakarta: Salemba Empat
- Munawar, 2005. *Pemodelan Visual menggunakan UML*. Yogyakarta: Penerbit : Graha Ilmu.
- Nancy, R.Tague. 2004. *The Toolbox Kualitas* Edisi Kedua. Milwaukee, Wisconsin : ASQ Quality Press.
- Warman, John. 2004. *Manajemen Pergudangan*. Pustaka Sinar Harapan, Jakarta.
- Yunarto dan Santika, 2005. *Business Concepts Implementation Series in Inventory Management*. PT. Elex Media Komputindo, Jakarta.

____, Pengguna Sistem, <http://fairuzelsaid.wordpress.com/2011/10/02/mana-jemen-proyek-teknologi-informasi-stakeholder-sistem>, (Tanggal Akses; 23 Maret 2017).

____, Proses *Trimming*, Manual Operasi, PT. Krama Yudha Ratu Motor
____, Sistem Manajemen Basis Data *Relational*, <https://id.wikipedia.org/wiki/Sistem>

manajemen basis data relasional (Tanggal Akses; 23 November 2017)

____, Simbol Flowchart, <http://susi22.staff.gunadarma.ac.id/Downloads/files/27505/SIMBOL.doc> (Tanggal Akses; 23 Maret 2017).

____, XAMPP, <https://id.wikipedia.org/wiki/XAMPP> (Tanggal Akses; 23 Agustus 2017).