

**RANCANG BANGUN SISTEM INFORMASI
PERMINTAAN DAN PENGELUARAN PAKET *PART*
BERBASIS *WEB* MENGGUNAKAN PHP 4.2.11 DAN
MYSQL 5.6.21 PADA PT KENCANA GEMILANG**

TUGAS AKHIR

Untuk Memenuhi Salah Satu Syarat Penyelesaian Program Sarjana Terapan
Program Studi Sistem Informasi Industri Otomotif Pada Politeknik STMI Jakarta

**OLEH
TRIA NOVIANA
1311046**



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN RI
JAKARTA
2018**

LEMBAR PERSETUJUAN PEMBIMBING
POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN RI

TANDA PERSETUJUAN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PERMINTAAN DAN PENGELUARAN PAKET
PART BERBASIS WEB MENGGUNAKAN PHP
4.2.11 DAN MYSQL 5.6.21 PADA PT KENCANA
GEMILANG

Disusun Oleh

Nama : TriaNoviana
Nim : 1311046
Program Studi : Sistem Informasi Industri Otomotif

Telah diperiksa dan disetujui untuk diajukan dalam sidang Tugas Akhir Politeknik
STMI Jakarta

Jakarta, 8 Agustus 2018

Pembimbing

Fifi Lailasari Hadianastuti, S.Kom., M.Kes.

NIP. 197310162005022001

LEMBAR PERSETUJUAN ASISTEN PEMBIMBING

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN RI

TANDA PERSETUJUAN ASISTEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PERMINTAAN DAN PENGELUARAN PAKET
PART BERBASIS WEB MENGGUNAKAN PHP
4.2.11 DAN MYSQL 5.6.21 PADA PT KENCANA
GEMILANG

Disusun Oleh

Nama : Tria Noviana
Nim : 1311046
Program Studi : Sistem Informasi Industri Otomotif

Telah diperiksa dan disetujui untuk diajukan dalam sidang Tugas Akhir Politeknik
STMI Jakarta.

Jakarta, 8 Agustus 2018

Asisten Pembimbing

Ahmad Juniar, S.Kom., MT
NIP. 197906052006041002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir : **RANCANG BANGUN SISTEM INFORMASI PERMINTAAN DAN PENGELUARAN PAKET PART BERBASIS WEB MENGGUNAKAN PHP 4.2.11 DAN MYSQL 5.6.21 PADA PT KENCANA GEMILANG**

Disusun Oleh :
Nama : Tria Noviana
NIM : 1311046
Program Studi : Sistem Informasi Industri Otomotif

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia pada hari Rabu tanggal 19 September 2018.

Jakarta, 19 September 2018

Dosen Pembimbing

Ketua Penguji

Fifi L. Hadianastuti, S.Kom., M.Kes
NIP. 197310162005022001

Drs. Jacob Saragih, MM
NIP : 195404281986031002

Dosen Penguji

Dosen Penguji

Ahlan Ismono, S.Kom., MMSI
NIP. 197901072006041002

Dewi Auditva Marizka, ST, MT
NIP. 197503182001122001



LEMBAR BIMBINGAN PENYUSUNAN TUGAS

AKHIR

Nama : TriaNoviana

NIM : 1311046

Judul TA : Rancang Bangun Sistem Informasi Permintaan dan
Pengeluaran
Paket Part Berbasis Web Menggunakan PHP 4.2.11 dan
MySql 5.6.21
Pada PT KENCANA GEMILANG.

Pembimbing : Fifi Lailasari Hadianastuti, S.Kom., M.Kes

Asisten Pembimbing : Ahmad Juniar, S.Kom., MT

Tanggal	Keterangan	Paraf
26/01/2017	Bimbingan Bab I, II, III	
20/02/2017	Revisi Bab I, II	
23/02/2017	Revisi Bab I, II, III	
27/02/2017	Revisi Bab I, III	
28/03/2018	Revisi Bab I, II, III, Bimbingan Bab IV	
18/07/2018	Revisi Bab I, II, III	
24/07/2018	Revisi Bab V	
27/07/2018	Revisi Bab V	
01/08/2018	Semua Bab	
07/08/2018	Keseluruhan	

Mengetahui,
Ketua Program Studi

Pembimbing

Sistem Informasi Industri Otomotif

Drs. Jacob Saragih, MM
S.Kom., M.Kes
NIP: 195404281986031002
197310162005022001

Fifi Lailasari Hadianastuti,
NIP.



LEMBAR BIMBINGAN PENYUSUNAN TUGAS

AKHIR

Nama : Tria Noviana

NIM : 1311046

Judul TA : Rancang Bangun Sistem Informasi Permintaan dan
Pengeluaran
Paket Part Berbasis Web Menggunakan PHP 4.2.11 dan
MySql 5.6.21
Pada PT KENCANA GEMILANG.

Asisten Pembimbing : Ahmad Juniar, S.Kom., MT

Tanggal	Keterangan	Paraf
23/02/2017	Bab I, II, III	
27/02/2017	Revisi Bab I, III	
28/03/2018	Revisi Bab I, II, III, Bimbingan Bab IV	
18/07/2018	Revisi Bab I, II, III	
24/07/2018	Revisi Bab V	
27/07/2018	Revisi Bab V	
01/08/2018	Semua Bab	
07/08/2018	Keseluruhan	

Mengetahui,
Ketua Program Studi
Pembimbing
Sistem Informasi Industri Otomotif

Asisten

Drs. Jacob Saragih, MM
S.Kom., MT
NIP: 195404281986031002
197906052006041002

Ahmad Juniar,
NIP.

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Tria Noviana

Nim : 1311046

Program Studi : Sistem Informasi Industri Otomotif

Dengan ini menyatakan bahwa karya Tugas Akhir yang saya buat dengan Judul:

“RANCANG BANGUN SISTEM INFORMASI PERMINTAAN DAN PENGELUARAN PAKET *PART* BERBASIS WEB MENGGUNAKAN PHP 4.2.11 DAN MYSQL 5.6.21 PADA PT KENCANA GEMILANG.”

Merupakan dibuat dan diselesaikan sendiri, dengan menggunakan literatur hasil kuliah, survei lapangan, Dosen Pembimbing dan Asisten Dosen Pembimbing, melalui tanya jawab maupun asistensi serta buku-buku acuan yang tertera dalam referensi pada karya Tugas Akhir ini.

Jika terbukti saya tidak memenuhi apa yang telah dinyatakan di atas, maka karya Tugas Akhir saya dibatalkan.

Jakarta, 08 Agustus 2018
Yang Membuat Pernyataan,

Tria Noviana

ABSTRAK

PT Kencana Gemilang merupakan salah satu perusahaan industri manufaktur yang bergerak di bidang pembuatan alat-alat rumah tangga dan kotak pemadam kebakaran yang dilabeli *brand* Miyako. Pada bagian PPIC di PT Kencana Gemilang penerapan sistemnya masih menggunakan aplikasi Microsoft Excel, dan untuk proses pengeluaran *part* ke bagian Produksi yang terdapat di gudang untuk diproses menjadi barang jadi, bagian PPIC membuat dokumen dan menyerahkannya secara langsung ke bagian gudang untuk memberitahu *part* apa yang diperlukan. Pembuatan dokumen bergantung terhadap ketelitian pekerja, sehingga dapat menimbulkan kemungkinan dokumen mengalami salah pengetikan, hilang, ataupun rusak, dan penyerahan dokumen yang secara langsung ke bagian gudang dapat mengganggu kegiatan gudang yang sedang berjalan. Hal tersebut membuat proses pengeluaran *part* menjadi lama dan kurang efektif, sehingga dengan adanya rancang bangun sistem permintaan dan pengeluaran paket *part* berbasis web ini, dapat membuat proses pengelolaan data menjadi lebih cepat dan akurat serta dapat meminimalisir dokumen salah pengetikan, hilang, ataupun rusak. Dalam membangun sistem informasi permintaan dan pengeluaran paket *part* pengembangan yang digunakan adalah *prototype* evolusioner. Pada tahap pemodelan sistem menggunakan *Unified Modeling Language* (UML) dengan membuat *usecase diagram*, *activity diagram* dan *sequence diagram*, pemodelan data dengan *class diagram* dan kamus data, perancangan *prototype* menggunakan HIPO, *flowchart*, merancang antarmuka dalam tahap pembuatan *prototype* menggunakan PHP 4.2.11 dan MySQL 5.6.21 sebagai basis data. Sistem permintaan dan pengeluaran *part* yang dirancang dan dibangun ini dapat meminimalisir tingkat kesalahan atau salah ketik dalam pembuatan dokumen yang digunakan dalam proses permintaan paket *part* dari bagian produksi ke PPIC, serta dapat mengurangi penumpukan dokumen dan memudahkan dalam pencarian data.

Kata Kunci: Sistem permintaan dan pengeluaran paket *part*, *Evolutionary Prototype*, *Unified Modeling Language* (UML), PHP, MySQL.

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul **“Rancang Bangun Sistem Informasi Permintaan dan Pengeluaran Paket *Part* Berbasis Web Menggunakan PHP 4.2.11 dan MySql 5.6.21 Pada PT Kencana Gemilang.”**

Penulisan Tugas Akhir ini bertujuan untuk memenuhi sebagian syarat penyelesaian program Sarjana Terapan pada program studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

Terselesainya tugas akhir ini tidak terlepas dari bantuan moril maupun materil, bimbingan, dan petunjuk dari berbagai pihak. Oleh karena itu, tak salah kiranya bila penulis mengungkapkan rasa terima kasih yang sebesar-besarnya dan penghargaan kepada:

1. Allah SWT yang telah memberikan segala sesuatu serta kelancaran dalam menyelesaikan Tugas Akhir ini.
2. Bapak Muhammad Adam Abikusna dan Ibu Suyati Selaku orang tua penulis yang selalu mendoakan, memberikan motivasi, dan pengorbanannya baik segi moril, maupun materi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
3. Ibu Ika Yulianti Adam dan Ibu Dwi Siti Zuraida Selaku kakak penulis yang selalu mendoakan, memberikan motivasi, dan pengorbanannya baik dari segi moril, maupun materi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
4. Bapak Dr. Mustofa, ST, MT. Selaku Direktur Politeknik STMI Jakarta.
5. Bapak Dr. Ridzky Kramandita, S.Kom, MT. Selaku Pembantu Direktur I Politeknik STMI Jakarta.
6. Bapak Drs. Jacob Saragih, MM selaku Ketua Program Studi Sistem Informasi Industri Otomotif.

7. Ibu Fifi Lailasari Hadianastuti, S.Kom, M.Kes. Selaku Dosen Pembimbing yang telah bersedia membantu memberikan arahan, bimbingan dan penjelasan dalam penyusunan Tugas Akhir.
8. Bapak Ahmad Juniar, S.Kom, MT. Selaku asisten dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
9. Seluruh teman-teman jurusan Sistem Informasi khususnya angkatan 2011 yang telah memberikan semangat dan motivasi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
10. Bapak Suyatman dan Ibu Wartini selaku pembimbing diperusahaan yang sedia berbagi ilmu serta waktunya untuk penulis sehingga penulis bisa menyelesaikan Tugas Akhir ini.
11. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
12. Serta semua pihak yang tidak dapat disebutkan satu persatu yang telah memberikan kritik, saran, dan bantuan dalam penyusunan Tugas Akhir ini.

Semoga segala bantuan yang telah diberikan kepada penulis mendapat balasan yang setimpal dari Allah SWT. Penulis menyadari masih banyak kekurangan dalam penyusunan Tugas Akhir ini. Semoga Tugas Akhir ini dapat memberikan manfaat dan informasi yang berguna bagi kami dan para pembaca.

Jakarta, Agustus 2018

Penulis,
Tria Noviana

DAFTAR ISI

	Halaman
Halaman Judul.....	i
Lembar Persetujuan Dosen Pembimbing.....	ii
Lembar Bimbingan Tugas Akhir	iv
Lembar Pernyataan Keaslian.....	vi
Abstraksi	vii
Kata Pengantar	viii
Daftar Isi	x
Daftar Gambar.....	xiii
Daftar Tabel	xv
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Pokok Permasalahan	2
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah.....	3
1.5. Manfaat Tugas Akhir	3
1.6. Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1. Pengertian Rancang Bangun.....	6
2.2. Pengertian Sistem	6
2.2.1. Karakteristik Sistem.....	7
2.2.2. Klasifikasi Sistem.	9
2.3. Pengertian Informasi.....	10
2.3.1. Kualitas Informasi.....	11
2.3.2. Nilai Informasi.....	11
2.4. Pengertian Sistem Informasi	12
2.5. Pengertian Produksi	13
2.5.1. Kegiatan Produksi.....	13

2.5.2. Faktor-Faktor Produksi	14
2.6. Bahan Baku	14
2.7. Pengertian <i>Production Planning and Inventory Control</i>	15
2.8. Hubungan Antara Produksi dengan PPIC (<i>Production Planning and Inventory Control</i>).....	16
2.9. Konsep Dasar Permintaan Barang	16
2.10. Konsep Dasar Prosedur Pengeluaran Barang Produksi	17
2.11. SDLC (<i>System Development Life Cycle</i>).....	17
2.12. <i>Prototype</i>	19
2.13. <i>Flowchart</i>	21
2.14. <i>Unified Modeling Language (UML)</i>	24
2.15. Kamus Data.....	33
2.16. HIPO (<i>Hierarchy plus Input-Process-Output</i>).....	34
2.17. PHP (Hypertext Pre-Processor)	36
2.18. MySQL.....	37
BAB III METODOLOGI PENELITIAN	39
3.1. Metodologi Penelitian.....	39
3.2. Identifikasi Masalah.....	39
3.3. Metode Pengumpulan Data.....	40
3.4. Pengembangan Sistem	41
3.5. Kerangka Pemecahan Masalah	42
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA	45
4.1. Sekilas Tentang Perusahaan.....	45
4.2. Visi dan Misi.....	45
4.3. Struktur Organisasi dan <i>Job Description</i>	46
4.4. Produk yang Dihasilkan.....	51
4.5. Dokumen yang digunakan	54
4.6. Prosedur Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi	57
4.7. Prosedur Sistem Informasi Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi yang Sedang Berjalan	59

4.7.1. <i>Use Case Diagram</i> yang Berjalan pada Prosedur Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi.....	59
BAB V ANALISIS DAN PEMBAHASAN.....	62
5.1. Analisis Kebutuhan Sistem.....	62
5.2. Sistem Informasi Permintaan dan Pengeluaran Paket <i>Part</i> yang Diusulkan	63
5.3. Analisis Sistem	65
5.3.1. <i>Use Case Diagram</i> yang Diusulkan.....	65
5.3.2. <i>Activity Diagram</i> yang Diusulkan.....	70
5.3.3. <i>Sequence Diagram</i> yang Diusulkan.....	73
5.4. Perancangan Data	76
5.4.1. <i>Class Diagram</i> yang Diusulkan.....	76
5.4.2. Kamus Data.....	77
5.4.3. <i>Deployment Diagram</i> yang Diusulkan	79
5.5. Perancangan Sistem	81
5.5.1. <i>Hierarchy plus Input-Proses-Output (HIPO)</i> Aplikasi Permintaan dan Pengeluaran Paket <i>Part</i>	81
5.5.2. <i>Flowchart</i> Aplikasi Permintaan dan Pengeluaran Paket <i>Part</i>	83
5.5.3. Perancangan <i>Interface</i> Aplikasi Usulan.....	84
5.6. Analisis Kebutuhan <i>Software</i> dan <i>Hardware</i>	93
BAB VI KESIMPULAN DAN SARAN.....	94
6.1. Kesimpulan	94
6.2. Saran	94
DAFTAR PUSTAKA	95

DAFTAR GAMBAR

	Halaman
Gambar II.1 Pengembangan Prototipe Evolusioner.....	20
Gambar II.2 Pengembangan Prototipe <i>Requirement</i>	21
Gambar II.3 Struktur UML.....	25
Gambar III.1 Kerangka Pemecahan Masalah.....	44
Gambar IV.1 Struktur Organisasi PT Kencana Gemilang	48
Gambar IV.2 <i>Production Card</i>	54
Gambar IV.3 <i>Job Order Production</i>	55
Gambar IV.4 Rencana Produksi.....	56
Gambar IV.5 <i>Flowmap</i> Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi.....	58
Gambar IV.6 <i>Use Case Diagram</i> Prosedur Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi.....	60
Gambar V.1 <i>Flowmap</i> Permintaan dan Pengeluaran Paket <i>Part</i> yang Diusulkan	64
Gambar V.2 <i>Use Case Diagram</i> Permintaan dan Pengeluaran Paket <i>Part</i> Usulan	67
Gambar V.3 <i>Activity Diagram</i> Proses <i>Login</i>	70
Gambar V.4 <i>Activity Diagram</i> Transaksi yang Diusulkan.....	71
Gambar V.5 <i>Activity Diagram</i> Stok Data <i>Part</i> yang Diusulkan	71
Gambar V.6 <i>Activity Diagram Input</i> Permintaan yang Dilakukan Operator Produksi yang Diusulkan	72
Gambar V.7 <i>Sequence Diagram Login</i> yang Diusulkan	73
Gambar V.8 <i>Sequence Diagram</i> Transaksi JOP yang Diusulkan	74
Gambar V.9 <i>Sequence Diagram</i> Melihat Data <i>Part</i> yang Diusulkan.....	75
Gambar V.10 <i>Sequence Diagram</i> Transaksi <i>Production Card</i> yang Diusulkan.....	75

Gambar V.11 <i>Class Diagram</i> yang Diusulkan	76
Gambar V.12 <i>Deployment Diagram</i> yang Diusulkan	80
Gambar V.13 Hirarki Menu Aplikasi Permintaan dan Pengeluaran Paket <i>Part</i>	82
Gambar V.14 <i>Flowchart</i> Aplikasi Permintaan dan Pengeluaran Paket <i>Part</i> ..	83
Gambar V.15 Rancangan <i>Form Login</i>	84
Gambar V.16 Rancangan Tampilan Utama	85
Gambar V.17 Rancangan Submenu Data User	86
Gambar V.18 Rancangan Tampilan Submenu Data <i>Part</i>	87
Gambar V.19 Rancangan Tampilan Submenu Rencana Produksi.....	88
Gambar V.20 Rancangan Submenu <i>Production Card</i>	89
Gambar V.21 Rancangan <i>Form Production Card</i>	89
Gambar V.22 Rancangan Tampilan Submenu JOP	90
Gambar V.23 Rancangan <i>Form JOP</i>	91
Gambar V.24 Rancangan Laporan Permintaan dan Pengeluaran Paket <i>Part</i> ..	92
Gambar V.25 Rancangan Laporan Stok <i>Part</i>	93

DAFTAR TABEL

		Halaman
Tabel II.1	Simbol <i>Flow Direction</i>	22
Tabel II.1	Simbol <i>Flow Direction</i> (lanjutan)	23
Tabel II.2	Simbol Proses.....	23
Tabel II.3	Simbol <i>Input</i> dan <i>Output</i>	24
Tabel II.4	Simbol-simbol <i>Use Case Diagram</i>	26
Tabel II.4	Simbol-simbol <i>Use Case Diagram</i> (lanjutan)	27
Tabel II.5	Simbol-simbol <i>Activity Diagram</i>	27
Tabel II.5	Simbol-simbol <i>Activity Diagram</i> (lanjutan)	28
Tabel II.6	Simbol-simbol <i>Sequence Diagram</i>	29
Tabel II.6	Simbol-simbol <i>Sequence Diagram</i> (lanjutan)	30
Tabel II.7	Simbol-simbol <i>Class Diagram</i>	31
Tabel II.8	Simbol-simbol <i>Deployment Diagram</i>	32
Tabel II.9	Simbol-simbol <i>Component Diagram</i>	33
Tabel II.10	Bentuk Kamus Data Untuk Tabel <i>Login</i>	34
Tabel II.11	Jenis Data	38
Tabel IV.1	Produk PT Kencana Gemilang	52
Tabel IV.2	Tipe Blender	52
Tabel IV.3	Tipe Magic Com	52
Tabel IV.4	Paket <i>Part</i> Blender.....	53
Tabel IV.5	Paket <i>Part</i> Magic Com	53
Tabel IV.6	Penjelasan <i>Production Card</i>	55
Tabel IV.7	Penjelasan JOP.....	56
Tabel IV.8	Definisi Aktor Prosedur Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi.....	60
Tabel IV.8	Definisi Aktor Prosedur Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi (lanjutan)	61
Tabel IV.9	Definisi <i>Use Case</i> Prosedur Permintaan dan Pengeluaran Paket <i>Part</i> ke Bagian Produksi.....	61

Tabel V.1	Identifikasi Kebutuhan <i>User</i>	62
Tabel V.1	Identifikasi Kebutuhan <i>User</i> (lanjutan).....	63
Tabel V.2	Definisi Aktor	66
Tabel V.3	Definisi <i>Use Case</i>	66
Tabel V.4	Skenario <i>Use Case Login</i>	67
Tabel V.5	Skenario <i>Use Case Input Production Card</i>	68
Tabel V.6	Skenario <i>Use Case Input JOP</i>	68
Tabel V.7	Skenario <i>Use Case Melihat List Production Card</i>	68
Tabel V.8	Skenario <i>Use Case Melihat List JOP</i>	69
Tabel V.9	Skenario <i>Use Case Memvalidasi JOP</i>	69
Tabel V.10	Skenario <i>Use Case Membuat Laporan</i>	69
Tabel V.11	Spesifikasi Tabel <i>User</i>	77
Tabel V.12	Spesifikasi Tabel <i>Part</i>	77
Tabel V.13	Spesifikasi Tabel <i>Production Card</i>	78
Tabel V.14	Spesifikasi Tabel Rencana Produksi	78
Tabel V.15	Spesifikasi Tabel <i>JOP</i>	79

BAB I

PENDAHULUAN

1.1 Latar Belakang

Ilmu pengetahuan dan teknologi berkembang setiap hari dan penggunaannya diterapkan di berbagai bidang, salah satunya bidang industri manufaktur. Dampak dari perkembangan ini membawa kemudahan bagi dunia industri dalam menjalankan setiap kegiatannya, serta berdampak pada tingkat persaingan yang semakin tinggi. Oleh karena itu, perusahaan yang bergerak di bidang industri memanfaatkan sistem informasi agar dapat menjalankan kegiatan bisnisnya dengan baik dan dapat bersaing dengan perusahaan lainnya.

Perusahaan yang bergerak di bidang industri manufaktur memerlukan sistem informasi yang dapat mengelola segala aktivitas guna menjalankan organisasinya. Sistem informasi digunakan untuk mempermudah penyajian informasi kepada pemakainya. Sistem informasi digunakan sebagai alat untuk mendukung kegiatan produksi dengan lebih efektif dan efisien di dalam perusahaan.

PT Kencana Gemilang merupakan salah satu perusahaan industri manufaktur yang bergerak di bidang pembuatan alat-alat rumah tangga dan kotak pemadam kebakaran yang dilabeli *brand* Miyako. Produk-produk yang diproduksi PT Kencana Gemilang di antaranya *magic com*, *magic jar*, *rice cooker*, dan sebagainya. Banyaknya jenis produk membuat penanganan proses produksi dan pengendalian jumlah material serta *partnya* menjadi kompleks, sehingga PT Kencana Gemilang membangun divisi PPIC (*Production Planning and Inventory Control*).

Pada bagian PPIC di PT Kencana Gemilang penerapan sistemnya masih menggunakan aplikasi Microsoft Excel, data yang ada diolah dalam bentuk *spreadsheet*. Untuk proses pengeluaran *part* ke bagian Produksi yang terdapat di gudang untuk diproses menjadi barang jadi (*finished goods*), bagian PPIC membuat JOP (*Job Order Production*) menggunakan Microsoft Excel dan

selanjutnya dicetak menjadi dokumen. JOP yang sudah dibuat diserahkan oleh bagian PPIC secara langsung ke bagian gudang untuk memberitahu *part* apa yang diperlukan, kemudian gudang akan menyiapkan barang untuk selanjutnya diserahkan ke bagian Produksi. Pembuatan dokumen bergantung terhadap ketelitian pekerja, sehingga dapat menimbulkan kemungkinan dokumen mengalami salah pengetikan, hilang, ataupun rusak, dan penyerahan dokumen yang secara langsung ke bagian gudang dapat mengganggu kegiatan gudang yang sedang berjalan. Hal tersebut membuat proses pengeluaran *part* menjadi kurang efektif, sehingga diperlukan suatu sistem yang dapat membuat proses pengelolaan data menjadi lebih cepat dan akurat serta dapat meminimalisir dokumen salah pengetikan, hilang, ataupun rusak.

Berdasarkan latar belakang di atas, maka dibuatlah Tugas Akhir dengan judul **“RANCANG BANGUN SISTEM INFORMASI PERMINTAAN DAN PENGELUARAN PAKET *PART* BERBASIS *WEB* MENGGUNAKAN PHP 5.3.1 DAN MYSQL 5.1.41 PADA PT KENCANA GEMILANG”**.

1.2 Pokok Permasalahan

Setelah mengamati proses permintaan dan pengeluaran paket *part* ke bagian produksi yang terdapat pada bagian PPIC di PT Kencana Gemilang, ditemukan beberapa masalah yaitu sebagai berikut:

1. Bagian PPIC masih menggunakan dokumen yang dibuat menggunakan Microsoft Excel dan kemudian diserahkan secara langsung ke bagian gudang. Hal ini sangat bergantung terhadap ketelitian pekerja sehingga dapat menimbulkan dokumen salah pengetikan, hilang, ataupun rusak, dan kegiatan proses produksi yang berjalan dapat terhambat.
2. Media penyimpanan masih berupa fisik, sehingga terjadi penumpukan dokumen dan menyulitkan untuk mencari dokumen yang diperlukan karena harus melihat dokumen satu per satu.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah:

1. Merancang dan membangun sistem informasi yang dapat memfasilitasi proses pengelolaan data permintaan dan pengeluaran paket *part* ke bagian produksi menjadi terkomputerisasi.
2. Merancang dan membangun sistem informasi dengan basis data yang mampu menyimpan dan mengelola data permintaan dan pengeluaran paket *part* ke bagian produksi sehingga dapat mengurangi penumpukan dokumen dan mempermudah pencarian data.

1.4 Batasan Masalah

Batasan masalah yang diambil dalam pengamatan ini terbatas pada:

1. Penelitian hanya sebatas mengenai proses permintaan dan pengeluaran paket *part* ke bagian produksi di bagian PPIC selama 1 bulan pada PT Kencana Gemilang.
2. Paket *part* hanya mencakup produk *blender* dan *magic com*.

1.5 Manfaat Tugas Akhir

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

1. Bagi penulis
 - a. Memberikan kemampuan dalam mengaplikasikan teori yang telah diperoleh selama perkuliahan secara jelas terhadap masalah yang diamati.
 - b. Memberikan wawasan dan pengalaman dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi permasalahan.

2. Bagi perusahaan

Hasil penelitian ini agar dapat diimplementasikan di perusahaan untuk membantu kinerja perusahaan, karena dengan adanya sistem informasi permintaan dan pengeluaran paket *part* berbasis *web* menggunakan PHP

4.2.11 dan MySQL 5.6.21 ini diharapkan dapat membantu perusahaan untuk meningkatkan kinerja dalam pelaksanaan kerjanya.

3. Bagi pihak lain

Hasil penelitian ini diharapkan dapat memberikan sumbangan pemikiran dan sebagai referensi bagi peneliti lain yang melakukan penelitian serupa.

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada di dalamnya. Adapun sistematika penulisan Tugas Akhir ini diuraikan dalam enam bab adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang, pokok permasalahan, tujuan penelitian, batasan masalah, manfaat tugas akhir, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini membahas tentang berbagai teori yang diperoleh dari buku-buku literatur atau pun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan adalah seputar sistem informasi, pengertian UML (*Unified Modelling Language*) dan pengertian PHP (*Personal Home Page*).

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang jenis dan sumber data, metode pengumpulan data, serta langkah-langkah yang akan dilakukan dalam perumusan dan pemecahan masalah termasuk metodologi pengembangan sistem yang digunakan. Metode pengembangan sistem yang digunakan yakni *Prototype Evolusioner*.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini akan membahas mengenai data yang telah diperoleh berdasarkan penelitian selama melaksanakan Praktek Kerja

Lapangan di PT Kencana Gemilang, serta mengidentifikasi sistem yang sedang berjalan diperusahaan tersebut.

BAB V ANALISIS DAN PEMBAHASAN

Pada bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis sistem yang meliputi alur sistem yang diusulkan, perancangan *interface* serta pembuatan alur program.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan yang dapat diambil dan mengemukakan saran-saran yang mungkin diperlukan oleh perusahaan dan peneliti selanjutnya.

BAB II

LANDASAN TEORI

2.1 Pengertian Rancang Bangun

Kata rancang berarti mengatur segala sesuatu sebelum bertindak, mengerjakan atau melakukan sesuatu untuk merencanakan. Sedangkan kata bangun sesuatu yang didirikan. Rancang bangun berarti merancang atau mendesain sesuatu yang akan dibuat sesuai yang diinginkan oleh pengguna.

2.2 Pengertian Sistem

Sistem berasal dari bahasa Latin “*systema*” atau bahasa Yunani “*sustēma*” yang berarti suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi, atau energi. Pengertian sistem menurut para ahli dapat didefinisikan sebagai berikut:

1. Sistem adalah kumpulan elemen yang saling berhubungan dan saling berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama (Sutarman, 2009).
2. Dikutip dari buku Hartono (2013), menurut Campbel (1979) dalam buku *Understanding Information System: Foundations for Control* menegaskan bahwa sistem adalah “*any group of interrelated components or part which function together to achieve goal*” berarti sistem adalah sehimpunan bagian-bagian atau komponen yang saling berkaitan dan secara bersama-sama berfungsi atau bergerak untuk mencapai suatu tujuan.
3. Sistem dapat didefinisikan dengan pendekatan prosedur dan pendekatan komponen, sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu (Mustakini, 2009).

Terdapat dua kelompok pendekatan dalam mendefinisikan sistem, yaitu:

- a. Pendekatan sistem yang lebih menekankan pada prosedur, mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan. Berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.
- b. Pendekatan yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai suatu kumpulan dari elemen-elemen yang saling berinteraksi untuk mencapai suatu tujuan tertentu.

Berdasarkan beberapa pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa sistem adalah kumpulan elemen yang terhubung, berinteraksi dan saling mempengaruhi untuk mencapai suatu tujuan.

2.2.1 Karakteristik Sistem

Menurut Sutabri (2004) sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah yang mempunyai komponen sistem, batasan sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah, dan sasaran sistem.

1. Komponen sistem (*components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu subsistem. Setiap subsistem memiliki sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar, yang disebut dengan supra sistem.

2. Batas sistem (*boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan yang tidak dapat dipisah-pisahkan.

3. Lingkungan luar sistem (*environment*)

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut sedangkan lingkungan luar yang merugikan harus dikendalikan.

4. Penghubung sistem (*interface*)

Sebagai media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. Keluaran suatu subsistem akan menjadi masukan untuk subsistem yang lain dengan melewati penghubung. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

5. Masukan sistem (*input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

6. Keluaran sistem (*output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, yang mana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan.

7. Pengolah sistem (*process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran sistem (*objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.2.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya sebagai berikut (Sutabri, 2004):

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan. Sedangkan sistem fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penggajian, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

3. Sistem Deterministik dan Sistem Probabilistik

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

4. Sistem Terbuka dan Sistem Tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.3 Pengertian Informasi

Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi, merupakan suatu komponen yang sangat penting dalam sistem, sehingga informasi ini sangat penting di dalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil dan akhirnya berakhir, karena tanpa informasi kegiatan yang dijalankan akan mengalami kesulitan.

Sumber dari informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal atau data-data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu (Jogiyanto, 2005).

Fakta atau observasi mentah yang biasanya banyak data yang menjelaskan kegiatan tersebut. Informasi merupakan data yang telah diubah menjadi konteks yang berarti dan berguna bagi pemakai akhir tertentu (McLeod, 2004).

Pendapat lain mengenai definisi informasi yang dikemukakan oleh Abdul Kadir, informasi merupakan hasil dari data yang diproses yang berasal dari *inputan* suatu kejadian yang nyata yang berguna bagi pemakainya (Abdul Kadir, 2003).

Sedangkan menurut Jogiyanto (2005) menyatakan bahwa informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Berdasarkan pengertian diatas informasi dapat disimpulkan sebagai kumpulan data yang telah diproses, diorganisasikan ke dalam bentuk yang sesuai dengan kebutuhan dan memiliki arti bagi penggunanya.

2.3.1 Kualitas Informasi

Menurut Jogiyanto (2005) kualitas informasi yang diharapkan tergantung 4 hal pokok yaitu:

1. Akurat

Akurat mempunyai arti informasi yang dihasilkan harus bebas dari kesalahan-kesalahan yang tidak biasa, tidak menyesatkan dan mencerminkan maksudnya.

2. Tepat waktu

Tepat waktu berarti informasi yang disampaikan ke penerima tidak terlambat, karena informasi adalah landasan untuk mengambil suatu keputusan. Untuk itu diperlukan suatu teknologi untuk dan mengirim dengan cepat dan tepat agar informasi tidak usang.

3. Relevan

Berarti informasi mempunyai manfaat dan berguna bagi pemakainya. Karena batas relevansi seseorang berbeda, maka informasi bisa dikatakan berguna jika benar-benar berguna dan dibutuhkan pemakainya.

4. Aman

Aman berarti informasi harus terbebas dari penyadapan oleh pihak orang yang tidak berwenang dalam penggunaan informasi tersebut.

2.3.2 Nilai Informasi

Suatu informasi dikatakan memiliki nilai apabila manfaatnya lebih besar dibandingkan dengan biaya mendapatkannya. Akan tetapi perlu diperhatikan bahwa informasi yang digunakan dalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan. Sehingga tidak memungkinkan dan sulit untuk menghubungkan suatu bagian informasi pada suatu masalah tertentu dengan biaya untuk memperolehnya, karena sebagian besar informasi tidak hanya dinikmati oleh suatu pihak di dalam perusahaan.

Kegunaan informasi adalah untuk mengurangi hal ketidakpastian di dalam proses pengambilan keputusan tentang suatu keadaan. Informasi yang bernilai juga harus bersifat informatif, dan terstruktur agar tidak menimbulkan

kesalahpahaman terhadap penerimanya. Sebagian besar informasi ditaksir nilai kegunaannya bukan dengan uang, tetapi dengan nilai efektifitasnya (Jogiyanto, 2005).

2.4 Pengertian Sistem Informasi

Dalam buku yang ditulis oleh Sutabri (2012), sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

Sistem informasi berbasis komputer dalam suatu organisasi terdiri dari komponen-komponen yang masing-masing saling berinteraksi dengan yang lainya membentuk satu kesatuan untuk mencapai sasarnya. Komponen tersebut terdiri dari 6 blok yaitu sebagai berikut:

1. Blok Masukan

Input mewakili data yang masuk ke dalam sistem informasi. *Input* di sini termasuk metode-metode dan media untuk menangkap data yang akan dimasukan yang 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 basis 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 tingkat manajemen serta semua pemakai sistem.

4. Blok Teknologi

Teknologi merupakan kotak alat (*tool box*) dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan sekaligus mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

5. Blok Database

Database merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali

Pengendalian perlu dirancang dan diterapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat cepat langsung diatasi.

2.5 Pengertian Produksi

Produksi adalah segala kegiatan dalam menciptakan dan menambah kegunaan (*utility*) suatu barang atau jasa, untuk kegiatan mana dibutuhkan faktor-faktor produksi dalam ilmu ekonomi berupa tanah, tenaga kerja, dan *skill* (*organization, managerial, dan skills*) (Assauri, 2016). Pada dasarnya produksi dibagi menjadi 2 bagian yaitu:

1. Produksi Langsung

Produksi yang menggunakan faktor-faktor produksi alam dan tenaga kerja. Produksi langsung terbagi menjadi dua bagian:

- a. Produksi Primer
- b. Produksi Sekunder

2. Produksi Tak Langsung

Produksi yang tidak menaikkan nilai penggunaan dan bukan dari alam tetap memberikan sumbangan jasa yang sangat bermanfaat bagi perusahaan.

2.5.1 Kegiatan Produksi

Kegiatan produksi adalah salah satu bagian dari beberapa kegiatan perusahaan di samping kegiatan personalia, keuangan, dan pemasaran. Keempat kegiatan tersebut tidak bisa dipisahkan karena merupakan satu kesatuan yang menjadikan perusahaan berhasil, maju dan berkembang. Kegiatan produksi atau fungsi produksi, pelaksanaan maupun pencapaian tujuan bagi produksi menjadi tanggung jawab seorang manager produksi. Pada fungsi produksi, seorang

manager produksi akan menghadapi masalah-masalah yang berkaitan dengan perusahaan secara keseluruhan dan harus diatasinya. Masalah-masalah di bagian produksi diantaranya (Handoko, 2011):

1. Perencanaan Produksi
2. Perencanaan Fasilitas Fisik Produksi
3. Pengendalian Produksi
4. Pemeliharaan Persediaan dan Kualitas Produksi
5. Pemeliharaan Peralatan

2.5.2 Faktor-faktor Produksi

Faktor produksi adalah sumber daya yang digunakan dalam sebuah proses produksi barang dan jasa. Pada awalnya, faktor produksi dibagi menjadi empat kelompok. Namun pada perkembangannya, faktor sumber daya alam diperluas cakupannya menjadi seluruh benda *tangible*, baik langsung dari alam maupun tidak, yang digunakan oleh perusahaan, yang kemudian disebut sebagai faktor fisik (*physical resources*). Selain itu, beberapa ahli juga menganggap sumber daya informasi sebagai sebuah faktor produksi mengingat semakin pentingnya peran informasi saat ini. Menurut Griffin (2006) terdapat 5 hal yang dianggap sebagai faktor produksi, yaitu:

1. Tenaga kerja (*labor*)
2. Modal (*capital*)
3. Sumber daya fisik (*physical resources*)
4. Kewirausahaan (*entrepreneurship*)
5. Sumber daya informasi (*information resources*)

2.6 Bahan Baku

Bahan baku merupakan bahan langsung (*direct material*), yaitu bahan yang membentuk suatu kesatuan yang tidak terpisahkan dari produk jadi. Bahan baku adalah bahan utama atau bahan pokok dan merupakan komponen utama dari suatu produk. Bahan baku biasanya lebih mudah ditelusuri dalam suatu produk dan harganya relatif tinggi dibandingkan dengan bahan pembantu. Bahan

pembantu (*indirect material*) merupakan bahan pelengkap yang melekat pada suatu produk. Bahan pembantu biasanya tidak mudah ditelusuri dalam suatu produk dan harganya relatif rendah dibandingkan dengan bahan baku (Nafarin, 2007).

Bahan baku terbagi menjadi tiga bagian yaitu bahan mentah, bahan setengah jadi, dan bahan jadi.

1. Bahan mentah

Bahan mentah adalah bahan yang belum pernah mengalami proses pengolahan.

2. Bahan setengah jadi

Bahan setengah jadi adalah bahan yang sudah diolah tetapi belum menjadi produk akhir. Agar menjadi bahan siap pakai perlu pengolahan lebih lanjut. Dalam industri manufaktur, komponen atau *part* termasuk dalam bahan setengah jadi.

3. Bahan jadi

Bahan jadi adalah bahan yang siap dikonsumsi untuk memenuhi kebutuhan.

2.7 Pengertian *Production Planning and Inventory Control* (PPIC)

Production Planning and Inventory Control (PPIC) adalah sebuah proses pengendalian aliran material masuk dan keluar dari sebuah sistem kerja yang bertujuan untuk memenuhi permintaan pasar dan pendistribusian yang tepat sehingga dapat meminimalkan biaya produksi.

Perancangan dan pengendalian produksi harus dilakukan di awal proses sebelum melakukan proses produksi, yang bertujuan untuk menentukan apa saja yang harus dilakukan pada awal hingga tahap akhir. Perencanaan juga tidak boleh diberhentikan hingga proses itu selesai karena hasilnya pasti tidak sesuai dengan yang diharapkan, sehingga harus dievaluasi berkala dengan melakukan pengendalian. (Nasution, 2008).

Tugas dan tanggung jawab PPIC pada umumnya yaitu memimpin dan bertanggung jawab untuk kegiatan pengendalian persediaan, pengendalian

produksi perencanaan, dan kontrol pengiriman, serta pengendalian bahan yang masuk, penyimpanan, penyediaan, dan pengiriman.

2.8 Hubungan Antara Produksi dengan PPIC (*Production Planning and Inventory Control*)

PPIC dan produksi memiliki keterkaitan yang erat. PPIC merupakan suatu departemen dalam suatu perusahaan yang bertugas dalam membuat perencanaan produksi dan pengendalian persediaan, sedangkan produksi merupakan suatu departemen yang bertugas untuk memproses barang mentah atau setengah jadi menjadi barang jadi.

Perencanaan produksi secara umum memiliki 4 tahap yaitu *planning*, *execution*, *monitoring*, dan *evaluation*. PPIC sangat berperan penting dalam tahap *planning* dan *monitoring*, sedangkan produksi berperan penting dalam tahap *execution*. Dan untuk tahap *evaluation* dilakukan oleh setiap departemen.

Kegiatan produksi tidak akan berjalan tanpa perencanaan terlebih dahulu, dan persediaan barang tidak akan tercapai jika kegiatan produksi tidak berjalan, oleh karena itu Produksi dan PPIC memiliki hubungan yang sangat erat.

2.9. Konsep Dasar Prosedur Permintaan dan Pengeluaran Barang Produksi

Prosedur adalah suatu urutan pekerjaan, karena biasanya melibatkan beberapa orang dalam satu bagian atau lebih, disusun untuk menjamin adanya perlakuan yang seragam terhadap transaksi – transaksi perusahaan yang sering terjadi (Baridwan, 2008).

Permintaan dapat diartikan sebagai jumlah barang yang dibutuhkan. Permintaan adalah perbuatan untuk mendapatkan sesuatu, sedangkan barang adalah benda dalam berbagai bentuk dan uraian yang meliputi bahan baku, barang setengah jadi, barang jadi atau peralatan yang spesifikasinya ditetapkan oleh pengguna barang atau jasa. Permintaan barang berarti kegiatan yang dilakukan oleh Bagian Produksi untuk mendapatkan barang sesuai dengan kebutuhan.

Pengeluaran adalah perbuatan untuk membawa keluar atau memindahkan sesuatu dari sebelah dalam ke sebelah luar. Pengeluaran barang berarti kegiatan yang dilakukan untuk memindahkan barang yang berada di dalam gudang ke bagian produksi untuk diproses menjadi suatu barang jadi.

Prosedur permintaan dan pengeluaran barang produksi ini dimulai dengan pembuatan dokumen permintaan barang yang dibuat oleh bagian produksi. Dokumen ini kemudian akan digunakan sebagai acuan untuk menyiapkan barang yang dibutuhkan. Barang yang diminta oleh bagian produksi disiapkan oleh gudang sesuai dengan dokumen permintaan barang. Barang yang akan diantarkan kemudian dicek kelengkapannya apakah jumlah dan jenisnya sudah sesuai dengan dokumen permintaan barang atau tidak. Jika jumlah atau jenisnya tidak sesuai maka barang akan disiapkan kembali, dan jika barang sudah sesuai dengan dokumen permintaan barang maka dibuatkan dokumen pengeluaran barang produksi. Dokumen ini kemudian akan disesuaikan dengan dokumen permintaan barang produksi dan barangnya secara aktual, untuk kemudian diantarkan ke bagian produksi.

2.10 SDLC (*Sistem Development Life Cycle*)

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

Tahapan yang ada pada SDLC secara global adalah sebagai berikut:

1. Inisiasi (*initiation*)

Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

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

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

3. Perencanaan (*planning*)
Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.
4. Analisis kebutuhan (*requirements analysis*)
Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.
5. Desain (*design*)
Mentransformasikan kebutuhan *detail* menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
6. Pengembangan (*development*)
Mengkonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan, membuat basis data dan mempersiapkan prosedur kasus pengujian, mempersiapkan berkas atau *file* pengujian, pengkodean, pengkompilasian, memperbaiki dan membersihkan program serta peninjauan pengujian.
7. Integrasi dan pengujian (*integration and test*)
Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user* sehingga menghasilkan laporan analisis pengujian.
8. Implementasi (*implementation*)
Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan luar *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.
9. Operasi dan pemeliharaan (*operations and maintenance*)
Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

10. Disposisi (*disposition*)

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

2.11 *Prototype*

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

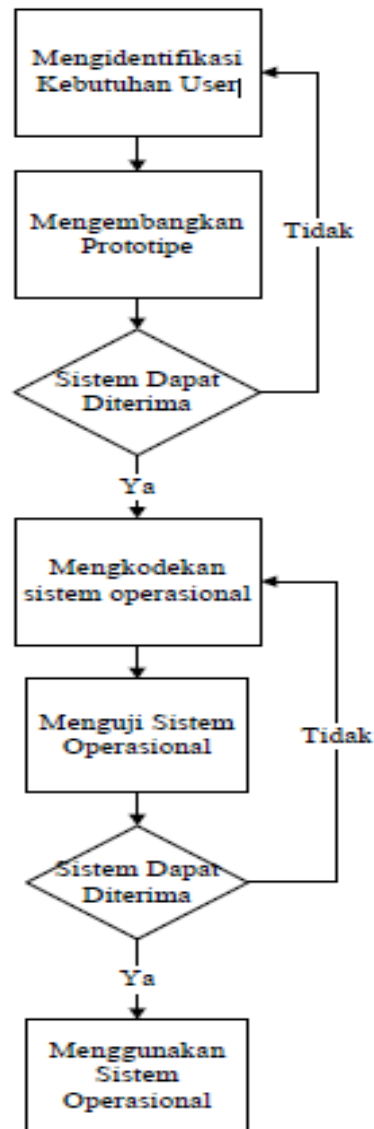
1. Prototipe evolusioner adalah prototipe yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi. Ada empat langkah yang diambil dalam mengembangkan suatu prototipe evolusioner yaitu identifikasi kebutuhan pengguna, mengembangkan prototipe, menentukan prototipe dapat diterima atau tidak, dan penggunaan prototipe.



Gambar II.1 Pengembangan Prototipe Evolusioner

(Sumber: McLeod 2011)

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



Gambar II.2 Pengembangan Prototipe *Requirement*

(Sumber: McLeod 2011)

2.12 *Flowchart*

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

Flowchart digunakan untuk alat bantu komunikasi dan dokumentasi. Pada waktu menggambar suatu diagram alir, analisis sistem dapat mengikuti ketentuan-ketentuan sebagai berikut: (Jogiyanto, 2005).

1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.
3. Harus ditunjukkan darimana kegiatan akan dimulai dan dimana akan berakhirnya.
4. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol-simbol bagan alir yang standar.

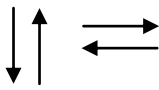

Simbol-simbol *flowchart* yang digunakan merupakan simbol-simbol *flowchart* standar yang dikeluarkan oleh ANSI dan ISO.

Berikut simbol-simbol standar yang digunakan untuk menggambarkan diagram alir sesuai kegunaan simbol: (Febriani, 2015)

1. *Flow Direction Symbols*


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

Tabel II.1 Simbol *Flow Direction*

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

(Sumber: Febriani, 2015)

Tabel II.1 Simbol *Flow Direction* (lanjutan)






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

(Sumber: Febriani, 2015)

2. Simbol Proses

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

Tabel II.2 Simbol Proses

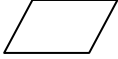


Simbol	Nama	Keterangan
	<i>Process</i>	Simbol yang menunjukkan pengolahan yang dilakukan komputer.
	<i>Decision</i>	Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban/aksi.
	<i>Predafined Process</i>	Simbol untuk mempersiapkan penyimpanan yang digunakan sebagai tempat pengolahan di dalam <i>storage</i> .
	Terminal	Simbol untuk permulaan atau akhir dari suatu program.
	<i>Manual Input</i>	Simbol untuk pemasukan data secara manual <i>on-line</i> keyboard.

(Sumber: Febriani, 2015)

3. Simbol *Input* dan *Output*

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

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

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

(Sumber: Febriani, 2015)

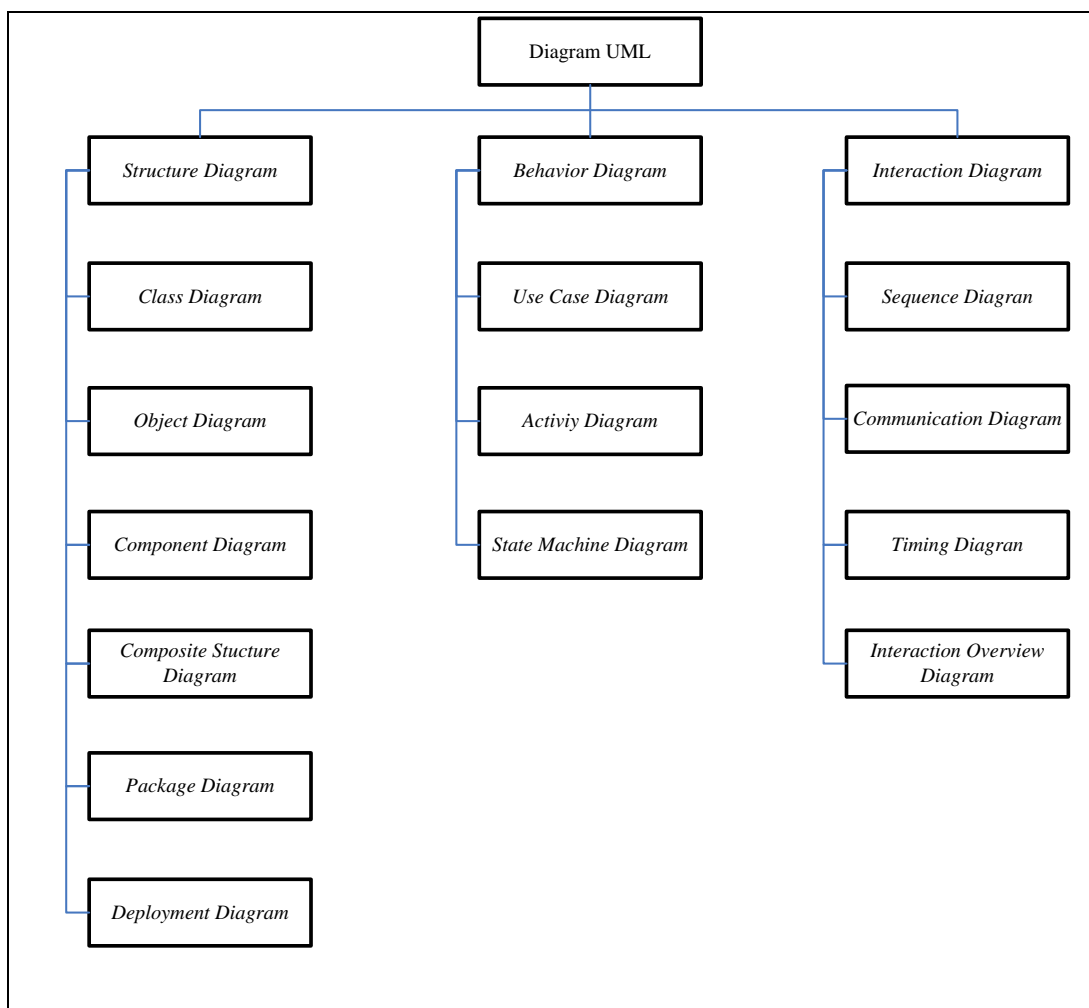
2.13 *Unified Modeling Language (UML)*

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).

Di mulai pada bulan Oktober 1994, Booch, Rumbaugh dan Jacobson yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan memelopori usaha untuk penyatuan metodologi perancangan berorientasi objek. Pada tahun 1995 dirilis pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh *Object Management Group (OMG)*. (Dharwiyanti, 2015).

UML terdiri dari 13 macam diagram yang dikelompokan dalam 3 kategori, adalah sebagai berikut:

1. *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu stuktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi antar subsistem pada suatu sistem.



Gambar II.3 Struktur UML


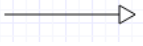
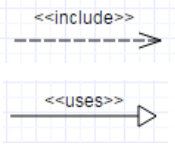
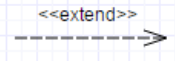
(Sumber: Rosa A.S dan Shalahuddin, 2014)

Adapun beberapa penjelasan dari macam-macam diagram tersebut adalah sebagai berikut:

1. *Use Case Diagram*



Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang berhak menggunakan fungsi-fungsi itu. Berikut ini adalah simbol dari *use case diagram*:

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

Simbol	Nama	Keterangan
	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat.
	<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum– khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
	<i>Include/uses</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.
	<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu.

(Sumber: Rosa dan Shalahuddin, 2014)

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




Simbol	Nama	Keterangan
	<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
	<i>Use Case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

(Sumber: Rosa dan Shalahuddin, 2014)

2. *Activity Diagram*

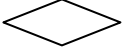

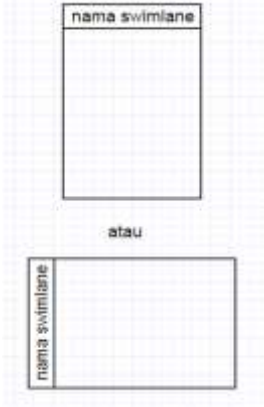
Diagram aktivitas atau activity diagram menggunakan aliran kerja (workflow) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut ini adalah simbol-simbol yang ada pada diagram aktivitas (Rosa dan Shalahuddin, 2014):

Tabel II.5 Simbol-simbol *Activity Diagram*

Simbol	Nama	Keterangan
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
	Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

(Sumber: Rosa dan Shalahuddin, 2014)

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


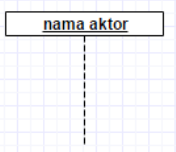

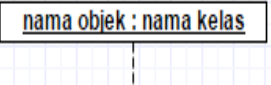
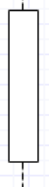
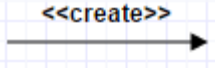
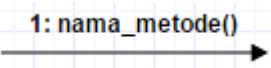
Simbol	Nama	Keterangan
	Percabangan/ <i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	Penggabungan/ <i>Join</i>	Asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu.
	<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

(Sumber: Rosa dan Shalahuddin, 2014)

3. *Sequence Diagram*

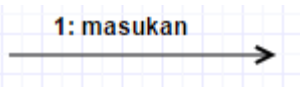
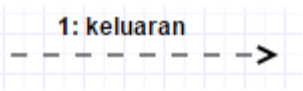
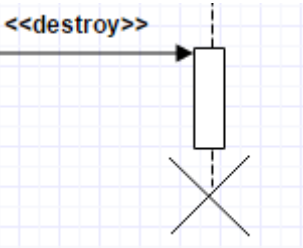
Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Berikut ini adalah simbol-simbol yang ada pada diagram sekuen (Rosa dan Shalahuddin, 2014):

Tabel II.6 Simbol-simbol *Sequence Diagram*

Simbol	Nama	Keterangan
 <p>nama aktor</p> <p>Atau</p>  <p>Tanpa waktu aktif</p>	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
	Garis hidup	Menyatakan kehidupan suatu objek.
	Objek	Menyatakan objek yang berinteraksi pesan.
	Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya. Aktor tidak memiliki waktu aktif.
	Pesan tipe <i>create</i>	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
	Pesan tipe <i>call</i>	Menyatakan suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri.

(Sumber: Rosa dan Shalahuddin, 2014)

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

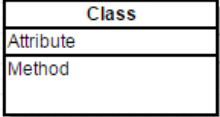






Simbol	Nama	Keterangan
	Pesan tipe send	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
	Pesan tipe return	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
	Pesan tipe destroy	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.

(Sumber: Rosa dan Shalahuddin, 2014)

4. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, sedangkan metode atau operasi adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Berikut ini adalah simbol-simbol yang ada pada *class diagram* (Rosa dan Shalahuddin, 2014):

Tabel II.7 Simbol-simbol *Class Diagram*

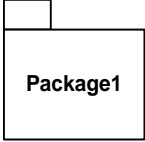



Simbol	Nama	Keterangan
	Kelas	Kelas pada struktur sistem.
	Antarmuka	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
	Asosiasi	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
	Asosiasi berarah	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i> .
	Generalisasi	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus).
	Kebergantungan	Relasi antarkelas dengan makna kebergantungan antar kelas.
	Agregasi	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>).

(Sumber: Rosa dan Shalahuddin, 2014)

5. *Deployment Diagram*

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* juga dapat digunakan untuk memodelkan sistem tambahan yang menggambarkan rancangan *device*, *node*, dan *hardware*, sistem *client/server*, sistem terdistribusi murni, dan rekayasa ulang aplikasi. Berikut ini merupakan tabel simbol dari *deployment diagram*:

Tabel II.8 Simbol-simbol *Deployment Diagram*

Simbol	Nama	Keterangan
	<i>Package</i>	Merupakan sebuah kumpulan dari satu atau lebih <i>node</i> .
	<i>Link</i>	Relasi antar <i>node</i> .
	<i>Dependency</i>	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
	<i>Node</i>	Biasanya mengacu pada perangkat keras, perangkat lunak yang tidak dibuat sendiri, jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang harus diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.

(Sumber: Rosa dan Shalahuddin, 2014)

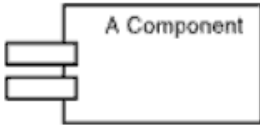

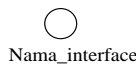

6. *Object Diagram*

Diagram objek atau *object diagram* menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Diagram objek juga berfungsi untuk mendefinisikan contoh nilai atau isi dari atribut tiap kelas.

7. *Component Diagram*

Diagram komponen dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem.

Tabel II.9 Simbol-simbol *Component Diagram*

No.	Simbol	Deskripsi
1.	<p>Komponen</p> 	Komponen sistem
2.	<p>Kebergantungan/<i>dependency</i></p> 	Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai.
3.	<p>Antarmuka/<i>interface</i></p> 	Sama dengan konsep <i>interface</i> pada pemrograman berorientasi objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen.
	<p>Link</p> 	Relasi antar komponen

Sumber: Rosa dan Shalahuddin (2014)

2.14 Kamus Data

Kamus data adalah daftar alfabetis dari nama-nama yang termasuk pada sebagai model sistem. Seperti namanya, kamus harus mencakup deskripsi yang berhubungan dengan entitas bernama tersebut dan jika nama itu merepresentasikan objek komposit, mungkin saja ada deskripsi mengenai komposisinya. Informasi lain seperti tanggal pembuatan, pembuatnya dan representasi entitas juga dapat dimasukkan, tergantung pada tipe model yang dikembangkan (Sommerville, 2003).

Keuntungan penggunaan kamus data (Sommerville, 2003):

1. Kamus data merupakan mekanisme untuk manajemen nama. Banyak orang yang harus menciptakan nama untuk entitas dan relasi ketika mengembangkan model sistem yang besar. Nama-nama ini harus dipakai secara konsisten dan tidak boleh bertentangan. Kamus data dapat memeriksa keunikan nama dan memberitahu analis persyaratan sekiranya terjadi duplikasi nama.

2. Kamus data sebagai tempat penyimpanan informasi organisasional yang dapat menghubungkan analisis, desain, implementasi, dan evolusi. Sementara sistem dikembangkan, informasi diambil untuk memberitahu perkembangan. Informasi baru ditambahkan pada sistem. Semua informasi mengenai entitas berada pada suatu tempat.

Kamus data sangat membantu analisis sistem dalam mendefinisikan data yang mengalir di dalam sistem, sehingga pendefinisian data itu dapat dilakukan dengan lengkap dan terstruktur. Pembentukan kamus data dilaksanakan dalam tahap analisis dan perancangan suatu sistem. Pada tahap analisis, kamus data merupakan alat komunikasi antara *user* dan analisis sistem tentang data yang mengalir di dalam sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh *user*.

Sementara itu, pada tahap perancangan sistem kamus data digunakan untuk merancang *input*, laporan dan *database*. Berikut ini adalah contoh bentuk kamus data:

Tabel II.10 Bentuk Kamus Data Untuk Tabel *Login*

No.	Nama Elemen	Akronim	Tipe
1.	ID <i>user</i>	ID_user	<i>String</i>
2.	<i>Password</i>	Pwd	<i>String</i>

(Sumber: Rosa dan Shalahuddin, 2011)

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

Menurut Jogiyanto (2005) *Hierarchy plus Input-Process-Output* (HIPO) merupakan metodologi yang dikembangkan dan didukung oleh IBM. HIPO sebenarnya alat dokumentasi program, akan tetapi sekarang HIPO banyak digunakan sebagai alat desain dan teknik dokumentasi dalam siklus pengembangan sistem. HIPO berbasis pada fungsi, yaitu setiap modul didalam sistem digambarkan oleh fungsi utamanya.

HIPO (*Hierarchy plus Input-Process-Output*) 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, bukannya menunjukkan statemen-statementen program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan yang jelas dari *input* yang harus digunakan dan *output* yang harus dihasilkan oleh masing-masing fungsi pada tiap-tiap tingkatan dari diagram HIPO.
4. Untuk menyediakan *output* yang tepat dan sesuai dengan kebutuhan-kebutuhan pemakai.

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

1. *Visual Table Of Contents (VTOC)*

Visual table of contents menggambarkan hubungan fungsi-fungsi di sistem secara berjenjang, *visual table of contents* menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur. Pada diagram ini nama dan nomor dari program HIPO didefinisikan. Struktur paket diagram dan hubungan fungsi juga didefinisikan dalam bentuk hirarki. Keterangan masing-masing fungsi diberikan pada bagian penjelasan yang diikutsertakan dalam diagram ini.

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.

3. *Detail Diagram*

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

2.16 PHP (Hypertext Pre-Processor)

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

Server-side scripts dijalankan ketika *browser* melakukan permintaan *file.php* dari *server*. PHP dipanggil oleh web *server*, dimana proses script perintah yang ada di suatu halaman dieksekusi mulai dari awal sampai akhir di dalam mesin PHP. Setelah *script* PHP tersebut diolah, hasilnya akan ditampilkan kepada *client* melalui *web browser* berupa tampilan *HTML*. Menurut Welling dan Thomson (2003), beberapa keunggulan PHP adalah:

1. *High Performance*

PHP sangat efisien. Dengan menggunakan *server* tunggal yang tidak mahal, *user* dapat melakukan banyak pekerjaan setiap harinya.

2. *Database Integration*

PHP mempunyai sambungan ke banyak sistem basis data, antara lain MySQL, PostgreSQL, Oracle, Informix, dan Sysbase databases.

3. *Built-in-Libraries*

PHP dirancang khusus untuk web, dan mempunyai banyak *built-in-function* untuk menampilkan banyak fungsi di dalam web.

4. Harga yang murah

PHP adalah perangkat lunak gratis.

5. Mudah dalam pembelajaran dan penggunaan

Sintaks PHP berdasarkan bahasa pemrograman lainnya, terutama C dan Java.

6. *Portability*

PHP dapat digunakan di banyak sistem operasi yang berbeda.

7. Ketersediaan *Source Code*

Kode PHP dapat langsung diakses dan dimodifikasi secara bebas.

2.17 MySQL

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

MySQL adalah salah satu database server yang menggunakan SQL (Structured Query Language) adalah bahasa pertanyaan (*query language*) yang distandarisasi untuk menanyakan informasi dari sebuah basis data (Welling dan Thomson, 2003).

MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Terdapat tiga kategori tipe data yang didukung oleh MySQL, yaitu tipe data numerik, *string*, serta penanggalan dan waktu. Sebuah tipe data yang akan disimpan harus sesuai dengan tipe data kolom yang bersangkutan. (Wahana Komputer, 2010).

MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam merancang tabel. Beberapa jenis data yang tersedia pada MySQL adalah sebagai berikut:

Tabel II.11 Jenis Data

Jenis Data	Keterangan
CHAR	Sebuah <i>string</i> dengan panjang tetap. Sisa Jumlah karakter yang belum terisi akan diisi dengan spasi, akan tetapi spasi ini dibuang jika data dipanggil. Jangkauan nilai M adalah 1-255 karakter
VARCHAR	<i>String</i> dengan panjang berupa variabel. M bisa mencapai 65535
DATE	Data berupa tanggal. Format tanggal dalam bentuk 'YYYY-MM-DD'
TIME	Data berupa waktu. Format waktu dalam bentuk 'HH:MM:SS'
TINYINT	Bilangan antara -128 sampai dengan +127
SMALLINT	Bilangan antara -32768 sampai dengan +32767
INT	Bilangan antara -2147683648 sampai dengan +2147683647
FLOAT	Bilangan <i>floating point</i> yang kecil (presisi tunggal). Jangkauan nilai yang diizinkan adalah -3.402823466E+38 s/d -1.175494351e-38,0 dan 1.175494351E-38 s/d 3.402823466E+38
DOUBLE	Bilangan <i>floating point</i> dengan ukuran normal (presisi ganda). Jangkauan nilai yang diizinkan adalah -1.7976931348623157E+308 s/d -2.225073858507201E-308,0 dan 2.225073858507201E-308 s/d 1.7976931348623157E+308
ENUM	Sebuah <i>enumeration</i> . Sebuah obyek string yang hanya boleh memiliki satu nilai, yang terambil dari 'value1', 'value2', ..., NULL atau nilai spesial "" <i>error</i> . Sebuah <i>enum</i> dapat menampung 65535 pilihan nilai
TEXT, BLOB	Sebuah TEXT atau BLOB dengan panjang karakter maksimum 65535 karakter

(Sumber: Sutaji, 2012)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, yang memiliki langkah-langkah sistematis. Dengan adanya metodologi penelitian, kegiatan penelitian yang akan dilakukan menjadi lebih terarah sesuai dengan urutan yang telah ditetapkan, namun tidak menutup kemungkinan akan terjadi perubahan urutan penelitian yang disebabkan oleh permasalahan yang sedang diteliti.

Metode penelitian yang dipakai yaitu metode penelitian kualitatif. Metode ini merupakan metode yang dipakai untuk mendapatkan data dengan cara yang lebih subjektif dan data yang diperoleh berasal dari pandangan seseorang mengenai fenomena (keadaan, proses, dan kejadian) tertentu. Metode inilah yang dipakai dalam pembuatan Tugas Akhir.

3.2 Identifikasi Masalah

Identifikasi masalah adalah salah satu proses penelitian yang boleh dikatakan paling penting di antara proses lain. Masalah penelitian akan menentukan kualitas dari penelitian, bahkan juga menentukan apakah sebuah kegiatan bisa disebut penelitian atau tidak.

Identifikasi masalah dalam proses penelitian secara umum dapat dilakukan dengan:

1. Studi Kepustakaan

Studi kepustakaan dilakukan dengan membaca buku dan literatur yang berkaitan dengan judul tugas akhir, sehingga dapat mengidentifikasi permasalahan yang muncul dalam proses penelitian.

2. Studi Lapangan

Studi lapangan dilakukan dengan melakukan observasi atau pengamatan secara langsung kegiatan yang terdapat di bagian PPIC pada PT Kencana Gemilang dan melakukan wawancara kepada beberapa karyawan di bagian PPIC, sehingga masalah yang ada dalam proses penelitian dapat diidentifikasi.

3.3 Metode Pengumpulan Data

Secara sederhana, pengumpulan data diartikan sebagai proses atau kegiatan yang dilakukan penulis untuk mengungkap atau menjangkau berbagai fenomena, informasi atau kondisi lokasi penelitian sesuai dengan lingkup

penelitian. Betapa pentingnya pengumpulan data dalam proses penelitian, karena tanpa data lapangan, proses analisis data dan kesimpulan hasil penelitian tidak dapat dilaksanakan.

Adapun cara yang digunakan untuk mengumpulkan data dan informasi dalam penelitian ini adalah:

1. Studi Lapangan

Studi lapangan adalah usaha pengumpulan data yang pada objek dan dilakukan dengan cara sebagai berikut:

- a. Observasi, yaitu usaha untuk mengumpulkan data dengan cara mengamati secara langsung kegiatan di PT Kencana Gemilang, kemudian dilakukan pencatatan informasi terhadap objek yang dianggap perlu. Observasi ini dilakukan di Bagian PPIC (*Production Planning and Inventory Control*).
- b. Wawancara, yaitu penelitian secara langsung yang dilakukan melalui tanya jawab terhadap segala hal yang diperlukan untuk penyusunan Tugas Akhir berdasarkan tujuan penelitian. Wawancara ini dilakukan kepada beberapa karyawan PT Kencana Gemilang di Bagian PPIC.

2. Studi Kepustakaan

Studi kepustakaan dilakukan dengan membaca buku dan literature 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 Pengembangan Sistem

Pengembangan sistem yaitu menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah berjalan. Pada penelitian ini, untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat sebuah *prototype* pengembangan sistem dari pengeluaran *part* ke bagian produksi yang telah berjalan. Sistem dikembangkan menjadi lebih sesuai dengan kebutuhan. Metodologi pengembangan sistem yang digunakan pada penelitian ini adalah metodologi *prototype* evolusioner.

Tahapan-tahapan dalam metodologi *prototype* evolusioner adalah sebagai berikut:

1. Mengidentifikasi kebutuhan pengguna

Pada tahap ini dilakukan perencanaan semua kebutuhan baik kebutuhan pengguna maupun kebutuhan sistem yang akan digunakan dalam membangun aplikasi sistem agar pengembangan dapat dibuat dengan mudah. Hasil identifikasi kebutuhan dibuat menjadi sebuah gambaran sistem. Setelah itu dibuat daftar tugas agar pengembangan sistem dapat dibuat lebih terkonsep.

2. Membuat *prototype*

Prototype dibuat dengan menggunakan *modeling system UML (unified modeling language)*, *engine* program dibuat dengan bahasa pemrograman PHP dan *database* menggunakan MySQL 5.6.20. Pembuatan *interface* dirancang sebaik mungkin agar memberikan kenyamanan pada pengguna.

3. Menyesuaikan dan evaluasi *prototype* dengan keinginan pengguna

Tahap ini menentukan apakah *prototype* dapat diterima atau tidak, *prototype* pertama yang telah dibuat ditunjukkan dan digunakan oleh pengguna agar diketahui kelayakan *prototype* tersebut harus dikembangkan lagi sampai benar-benar memenuhi kebutuhan pengguna.

4. Menggunakan *prototype*

Tahap ini dilakukan ketika pengguna sudah memutuskan untuk menggunakan sistem dan aplikasi yang telah dibuat beberapa kali perbaikan pengembangan.

3.5 Kerangka Pemecahan Masalah

Kerangka pemecahan masalah akan menggambarkan langkah-langkah penelitian yang akan dilakukan. Penjelasan dan penerapan *prototype* langkah-langkah dalam pemecahan masalah adalah sebagai berikut:

1. Studi pendahuluan

Studi pendahuluan yang dilakukan adalah studi lapangan dan studi pustaka. Tujuan melakukan studi pendahuluan adalah untuk mendapatkan pengetahuan mengenai system apa yang sedang diteliti.

2. Permasalahan

Proses identifikasi masalah dilakukan agar dapat diketahui seperti apa sistem yang telah berjalan untuk mencari apa yang harus diperbaiki dari sistem lama tersebut dan membuat aplikasinya.

3. Tujuan Penelitian

Tujuan penelitian dibuat untuk memperjelas target yang akan dicapai dan dibuat dalam penelitian.

4. Batasan Masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya.

5. Menentukan metodologi yang digunakan

Metodologi *prototype* evolusioner dipilih karena metodologi ini memiliki banyak kelebihan dan dapat menghasilkan sistem yang baik karena pengguna turut serta dalam pengembangan. Adapun tahapan pengembangan evolusioner itu mengidentifikasi kebutuhan pengguna, membuat *prototype*, dan menggunakan *prototype*.

6. Mengidentifikasi kebutuhan pengguna

Identifikasi dilakukan agar dapat membuat sebuah gambaran sistem dan kebutuhan aplikasi yang akan dibuat. Data yang diperlukan dikumpulkan untuk kemudian diolah sehingga menghasilkan informasi yang dibutuhkan.

7. Membuat *prototype*

Merancang *prototype* sesuai dengan analisis dan perancangan yang dilakukan. Pemodelan sistem menggunakan *use case diagram*, *activity diagram*, dan *sequence diagram*. Pemodelan data menggunakan *class diagram* dan kamus data. Desain sistem menggunakan HIPO (*Hierarchy plus Input-Process-Output*), *flowchart*, dan *interface*.

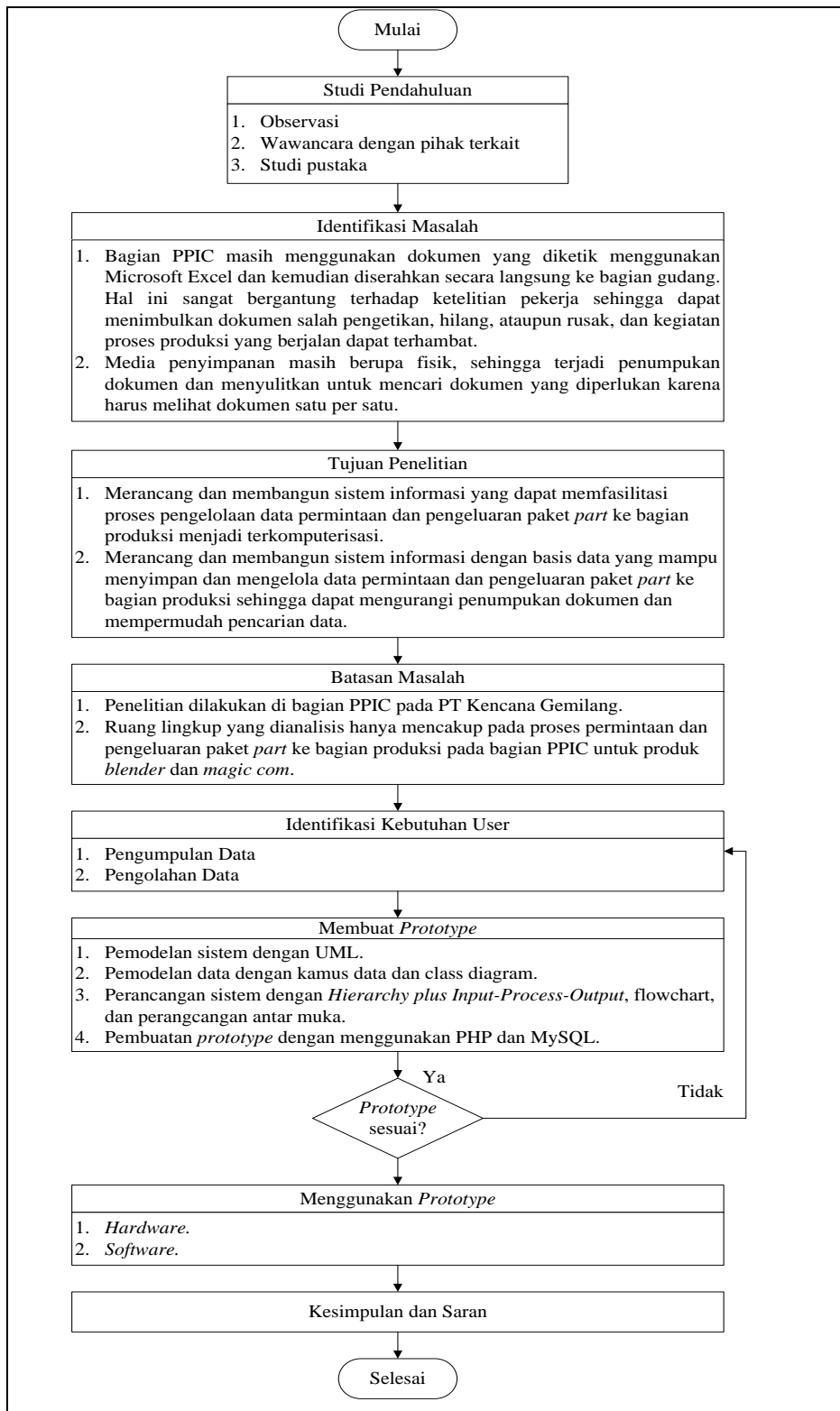
8. Menggunakan *prototype*

Prototype digunakan setelah beberapa kali dilakukan perbaikan. Penerapan *prototype* pertama ditunjukkan kepada pengguna untuk dinilai kelayakan. Jika belum layak, penilaian dari pengguna digunakan untuk menganalisis, mendesain, dan mengimplementasikan kembali *prototype* berikutnya yang lebih baik sampai *prototype* diterima.

9. Kesimpulan dan saran

Setelah *prototype* sistem usulan diterima maka tahap selanjutnya adalah menarik kesimpulan dari penelitian yang telah dilakukan dan memberikan saran yang membangun bagi perusahaan.

Berikut ini adalah gambar III.1 *flowchart* kerangka pemecahan masalah dalam Tugas Akhir:



Gambar III.1 Kerangka Pemecahan Masalah
(Sumber: Hasil Pengolahan Data, 2016)

BAB III

METODOLOGI PENELITIAN

3.6 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, yang memiliki langkah-langkah sistematis. Dengan adanya metodologi penelitian, kegiatan penelitian yang akan dilakukan menjadi lebih terarah sesuai dengan urutan yang telah ditetapkan, namun tidak menutup kemungkinan akan terjadi perubahan urutan penelitian yang disebabkan oleh permasalahan yang sedang diteliti.

Metode penelitian yang dipakai yaitu metode penelitian kualitatif. Metode ini merupakan metode yang dipakai untuk mendapatkan data dengan cara yang lebih subjektif dan data yang diperoleh berasal dari pandangan seseorang mengenai fenomena (keadaan, proses, dan kejadian) tertentu. Metode inilah yang dipakai dalam pembuatan Tugas Akhir.

3.7 Identifikasi Masalah

Identifikasi masalah adalah salah satu proses penelitian yang boleh dikatakan paling penting di antara proses lain. Masalah penelitian akan menentukan kualitas dari penelitian, bahkan juga menentukan apakah sebuah kegiatan bisa disebut penelitian atau tidak.

Identifikasi masalah dalam proses penelitian secara umum dapat dilakukan dengan:

3. Studi Kepustakaan

Studi kepustakaan dilakukan dengan membaca buku dan literatur yang berkaitan dengan judul tugas akhir, sehingga dapat mengidentifikasi permasalahan yang muncul dalam proses penelitian.

4. Studi Lapangan

Studi lapangan dilakukan dengan melakukan observasi atau pengamatan secara langsung kegiatan yang terdapat di bagian PPIC pada PT Kencana Gemilang dan melakukan wawancara kepada beberapa karyawan di bagian PPIC, sehingga masalah yang ada dalam proses penelitian dapat diidentifikasi.

3.8 Metode Pengumpulan Data

Secara sederhana, pengumpulan data diartikan sebagai proses atau kegiatan yang dilakukan penulis untuk mengungkap atau menjangkau berbagai fenomena, informasi atau kondisi lokasi penelitian sesuai dengan lingkup

penelitian. Betapa pentingnya pengumpulan data dalam proses penelitian, karena tanpa data lapangan, proses analisis data dan kesimpulan hasil penelitian tidak dapat dilaksanakan.

Adapun cara yang digunakan untuk mengumpulkan data dan informasi dalam penelitian ini adalah:

3. Studi Lapangan

Studi lapangan adalah usaha pengumpulan data yang pada objek dan dilakukan dengan cara sebagai berikut:

- c. Observasi, yaitu usaha untuk mengumpulkan data dengan cara mengamati secara langsung kegiatan di PT Kencana Gemilang, kemudian dilakukan pencatatan informasi terhadap objek yang dianggap perlu. Observasi ini dilakukan di Bagian PPIC (*Production Planning and Inventory Control*).
- d. Wawancara, yaitu penelitian secara langsung yang dilakukan melalui tanya jawab terhadap segala hal yang diperlukan untuk penyusunan Tugas Akhir berdasarkan tujuan penelitian. Wawancara ini dilakukan kepada beberapa karyawan PT Kencana Gemilang di Bagian PPIC.

4. Studi Kepustakaan

Studi kepustakaan dilakukan dengan membaca buku dan literature 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.

3.9 Pengembangan Sistem

Pengembangan sistem yaitu menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah berjalan. Pada penelitian ini, untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat sebuah *prototype* pengembangan sistem dari pengeluaran *part* ke bagian produksi yang telah berjalan. Sistem dikembangkan menjadi lebih sesuai dengan kebutuhan. Metodologi pengembangan sistem yang digunakan pada penelitian ini adalah metodologi *prototype* evolusioner.

Tahapan-tahapan dalam metodologi *prototype* evolusioner adalah sebagai berikut:

5. Mengidentifikasi kebutuhan pengguna

Pada tahap ini dilakukan perencanaan semua kebutuhan baik kebutuhan pengguna maupun kebutuhan sistem yang akan digunakan dalam membangun aplikasi sistem agar pengembangan dapat dibuat dengan mudah. Hasil identifikasi kebutuhan dibuat menjadi sebuah gambaran sistem. Setelah itu dibuat daftar tugas agar pengembangan sistem dapat dibuat lebih terkonsep.

6. Membuat *prototype*

Prototype dibuat dengan menggunakan *modeling system* UML (*unifed modeling language*), *engine* program dibuat dengan bahasa pemograman PHP dan *database* menggunakan MySQL 5.6.20. Pembuatan *interface* dirancang sebaik mungkin agar memberikan kenyamanan pada pengguna.

7. Menyesuaikan dan evaluasi *prototype* dengan keinginan pengguna

Tahap ini menentukan apakah *prototype* dapat diterima atau tidak, *prototype* pertama yang telah dibuat ditunjukkan dan digunakan oleh pengguna agar diketahui kelayakan *prototype* tersebut harus dikembangkan lagi sampai benar-benar memenuhi kebutuhan pengguna.

8. Menggunakan *prototype*

Tahap ini dilakukan ketika pengguna sudah memutuskan untuk menggunakan sistem dan aplikasi yang telah dibuat beberapa kali perbaikan pengembangan.

3.10 Kerangka Pemecahan Masalah

Kerangka pemecahan masalah akan menggambarkan langkah-langkah penelitian yang akan dilakukan. Penjelasan dan penerapan *prototype* langkah-langkah dalam pemecahan masalah adalah sebagai berikut:

10. Studi pendahuluan

Studi pendahuluan yang dilakukan adalah studi lapangan dan studi pustaka. Tujuan melakukan studi pendahuluan adalah untuk mendapatkan pengetahuan mengenai system apa yang sedang diteliti.

11. Permasalahan

Proses identifikasi masalah dilakukan agar dapat diketahui seperti apa sistem yang telah berjalan untuk mencari apa yang harus diperbaiki dari sistem lama tersebut dan membuat aplikasinya.

12. Tujuan Penelitian

Tujuan penelitian dibuat untuk memperjelas target yang akan dicapai dan dibuat dalam penelitian.

13. Batasan Masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya.

14. Menentukan metodologi yang digunakan

Metodologi *prototype* evolusioner dipilih karena metodologi ini memiliki banyak kelebihan dan dapat menghasilkan sistem yang baik karena pengguna turut serta dalam pengembangan. Adapun tahapan pengembangan evolusioner itu mengidentifikasi kebutuhan pengguna, membuat *prototype*, dan menggunakan *prototype*.

15. Mengidentifikasi kebutuhan pengguna

Identifikasi dilakukan agar dapat membuat sebuah gambaran sistem dan kebutuhan aplikasi yang akan dibuat. Data yang diperlukan dikumpulkan untuk kemudian diolah sehingga menghasilkan informasi yang dibutuhkan.

16. Membuat *prototype*

Merancang *prototype* sesuai dengan analisis dan perancangan yang dilakukan. Pemodelan sistem menggunakan *use case diagram*, *activity diagram*, dan *sequence diagram*. Pemodelan data menggunakan *class diagram* dan kamus data. Desain sistem menggunakan HIPO (*Hierarchy plus Input-Process-Output*), *flowchart*, dan *interface*.

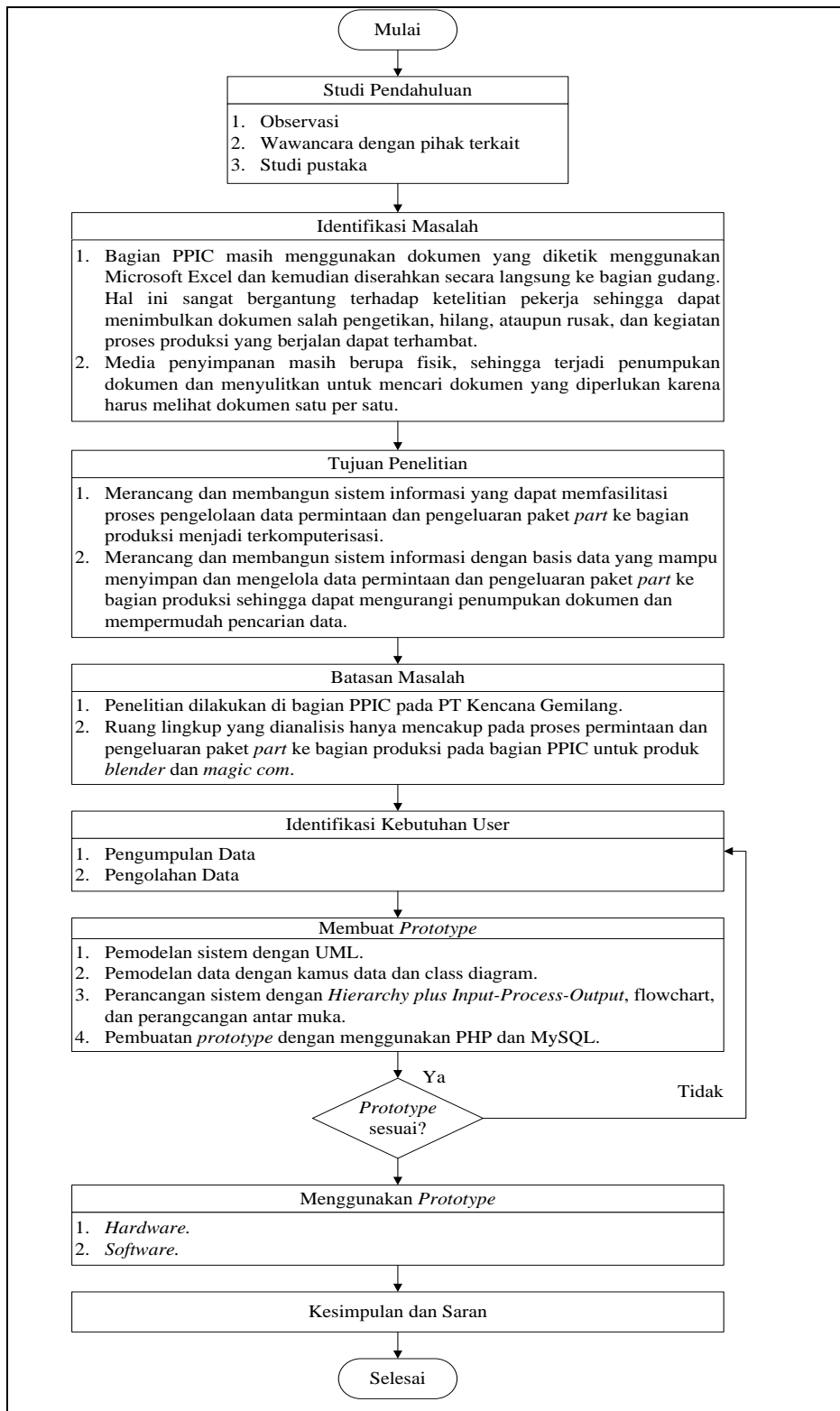
17. Menggunakan *prototype*

Prototype digunakan setelah beberapa kali dilakukan perbaikan. Penerapan *prototype* pertama ditunjukkan kepada pengguna untuk dinilai kelayakan. Jika belum layak, penilaian dari pengguna digunakan untuk menganalisis, mendesain, dan mengimplementasikan kembali *prototype* berikutnya yang lebih baik sampai *prototype* diterima.

18. Kesimpulan dan saran

Setelah *prototype* sistem usulan diterima maka tahap selanjutnya adalah menarik kesimpulan dari penelitian yang telah dilakukan dan memberikan saran yang membangun bagi perusahaan.

Berikut ini adalah gambar III.1 *flowchart* kerangka pemecahan masalah dalam Tugas Akhir:



Gambar III.1 Kerangka Pemecahan Masalah
(Sumber: Hasil Pengolahan Data, 2016)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Tentang Perusahaan

Cepatnya perkembangan zaman dan kemajuan teknologi terutama dalam bidang industri yang merupakan penggerak sektor ekonomi negara, maka sangat dibutuhkan perusahaan yang memiliki prospek handal, salah satunya adalah PT Kencana Gemilang yang bergerak di bidang pembuatan alat-alat rumah tangga dan kotak pemadam kebakaran.

Pendiri dari PT Kencana Gemilang merupakan orang-orang yang memiliki pemikiran dan pandangan ke depan akan adanya suatu kesuksesan serta pengetahuan yang luas akan dunia industri. PT Kencana Gemilang berdiri pada tanggal 7 Februari 1987 dengan nama PT Tri Telaga Teguh dan terletak di salah satu kawasan industri bagian barat Kota Tangerang tepatnya di jalan Raya Serang Km 16,8, Desa Talaga, Cikupa, Kabupaten Tangerang, Banten.

4.2 Visi dan Misi

Visi dari PT Kencana Gemilang yaitu, “Sebaik mungkin, semurah mungkin”, dan memiliki misi sebagai berikut:

1. Mengutamakan kepuasan pelanggan di semua lapisan masyarakat.
2. Menciptakan inovasi terus-menerus dalam pengembangan dan perbaikan produk guna menghasilkan produk yang berkualitas dan terjangkau di seluruh lapisan masyarakat.
3. Meningkatkan kompetensi sumber daya manusia menuju era globalisasi.
4. Meningkatkan sarana dan prasarana pendukung untuk mewujudkan mutu produk yang diharapkan pelanggan.
5. Menetapkan, menerapkan, memelihara, dan melakukan perbaikan terus-menerus terhadap sistem manajemen mutu.

PT Kencana Gemilang memiliki kebijakan mutu dan lingkungan yang dibuat untuk menjamin kesesuaian dan keberhasilan dalam mencapai tujuan

perusahaan. Kebijakan mutu perusahaan menunjukkan komitmen perusahaan untuk:

1. Menerapkan sistem manajemen mutu ISO 9001:2008, agar menjadi perusahaan yang berorientasi mutu pada semua kegiatannya.
2. Memberikan pelayanan terbaik kepada pelanggan pada semua tingkat.
3. Selalu berfikir dan bertindak secara efektif, efisien, dan berusaha untuk mencapai yang terbaik dalam segala kegiatan maupun kesempatan.
4. Saling menghargai dan membina kerja sama yang baik di lingkungan internal maupun eksternal.
5. Meningkatkan kompetensi karyawan di semua tingkatan agar lebih produktif, sesuai dengan tuntutan perusahaan.
6. Meningkatkan sarana dan prasarana pendukung untuk mewujudkan mutu produk yang diharapkan pelanggan.

Setiap karyawan bertanggung jawab atas kualitas pada setiap aktifitas kerjanya untuk menghasilkan produk yang kompetitif dan lingkungan kerja yang sehat. Kebijakan mutu ini dikomunikasikan untuk dipahami dan diimplementasikan kepada semua karyawan, supplier, rekanan, serta pihak lain yang berkepentingan dan secara periodik akan ditinjau peningkatannya.

4.3 Struktur Organisasi dan *Job Description*

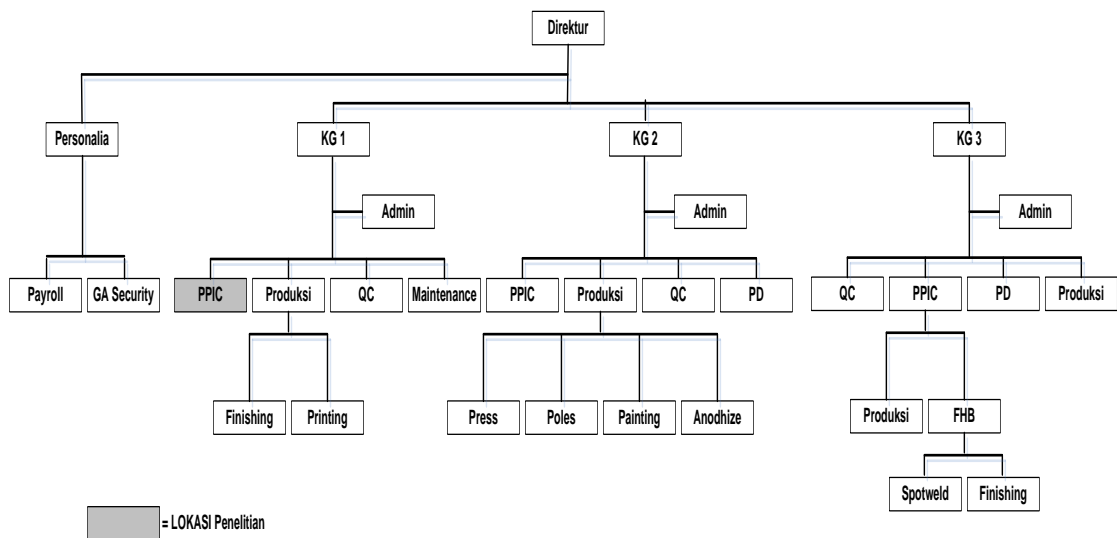
Setiap perkumpulan yang memiliki tujuan bersama pasti memiliki suatu struktur yang membagi antar anggotanya. Struktur tersebut bertujuan sebagai dasar dari penyusunan aktifitas agar dapat bekerja secara harmonis, begitu juga dengan perusahaan yang merupakan perkumpulan aktivitas usaha yang besar, harus memiliki struktur organisasi agar proses manajemen atau pengelolaan usahanya dapat berjalan dengan baik. Struktur organisasi tersebut sangat berpengaruh terhadap sukses tidaknya proses pengelolaan usaha, karena menentukan pedoman kerja perusahaan, mengatur kerja sama, serta mencegah kesimpang-siuran, kelambatan, dan kesulitan dalam bekerja.

Organisasi merupakan suatu kesatuan yang diatur oleh sekelompok orang yang telah sepakat untuk mencapai tujuan bersama. Prinsip yang harus

diperhatikan dalam menyusun organisasi perusahaan yaitu perangkaian pekerjaan yang menjadi satu sistem kesatuan dalam perintah, pemeliharaan, kemampuan pengawasan, dan penyusunan tatanan organisasi atau skema jabatan.

Penyusunan organisasi perusahaan harus berdasarkan asas-asas, yaitu sebagai berikut:

1. Perumusan tujuan setiap kegiatan harus memiliki tujuan yang benar dan dapat dimengerti serta harus dijiwai oleh setiap pekerja, untuk itu para pekerja harus melaksanakan tujuan dengan baik agar dapat mencapai penghematan material dan waktu.
2. Pemilihan wewenang berdasarkan kepercayaan dan keyakinan akan kemampuan pelaksanaan pekerjaan dengan sebaik-baiknya serta diimbangi dengan rasa tanggung jawab.
3. Perincian tugas dan tanggung jawab anggota organisasi dalam pencapaian tujuan.
4. Tanggung jawab merupakan tingkatan wewenang dari atas sampai bawah pada suatu organisasi.
5. Kesatuan komando dalam organisasi pada setiap pekerjaan harus diterima dan dipertanggungjawabkan kepada pimpinan.
6. Batas kemampuan pengawas yaitu batas kemampuan yang dimiliki untuk mengoreksi dengan baik.



Gambar IV.1 Struktur Organisasi PT Kencana Gemilang
(Sumber: PT Kencana Gemilang, 2014)

PT Kencana Gemilang memiliki struktur organisasi yang cukup jelas garis kekuasaannya, dan memiliki fleksibilitas kerja serta disiplin kerja yang sangat baik sehingga tercipta keteraturan dalam kegiatan bisnisnya.

Divisi atau departemen di PT Kencana Gemilang dibentuk untuk menggerakkan jalannya proses produksi agar sistem kerja dan pengelolaan semua komponen dapat dicapai dengan baik, serta agar setiap departemen dapat menjalankan tugasnya masing-masing sesuai prosedur kerja yang telah ditetapkan. Adapun departemen yang terdapat di PT Kencana Gemilang serta fungsi kerjanya masing-masing adalah sebagai berikut:

1. Product Development

Fungsi kerja:

Membuat produk baru dan mengembangkannya sampai produk tersebut dapat diproduksi dalam skala besar serta melakukan proses perbaikan. Dalam pembuatan produk baru selalu diawali dengan pembuatan *sample* kemudian dikembangkan.

2. Planning Production and Inventory Control

Fungsi kerja:

Membuat perencanaan produksi dan melakukan pengendalian terhadap *part*, material, hingga *finish goods* tanpa terjadi kekurangan atau kelebihan stok barang yang tidak sesuai standar.

3. Production

Fungsi kerja:

Melakukan proses *trial* produksi sampai produksi berskala besar dimana dalam setiap menjalankan prosesnya selalu memperhatikan kualitas dan kuantitas barang sesuai dengan standar.

4. Quality Assurance

Fungsi kerja:

Melakukan proses pengendalian mutu terhadap suatu barang mulai dari material, *part*, sampai *finish goods* pada setiap produk yang dihasilkan, sehingga penjualan produk kepada konsumen tidak mengalami penurunan mutu.

5. Mould Shop

Fungsi kerja:

Membuat *mould* dan melakukan proses perbaikan serta membuat alat bantu produksi (bila diperlukan) seperti *jig*, khususnya dalam pembuatan *mould* metal, sedangkan untuk *mould* plastik hanya melakukan proses perbaikan saja.

6. Maintenance

Fungsi kerja:

Melakukan proses perbaikan dan pemeliharaan terhadap semua fasilitas yang terdapat di perusahaan.

7. Accounting

Fungsi kerja:

Melakukan proses perhitungan pada aspek finansial perusahaan seperti biaya, pemasukan, aset, dan hal lain terkait dengan keuangan perusahaan untuk pembuatan laporan neraca, rugi laba, dan sebagainya.

8. Purchasing dan Admin

Fungsi kerja:

Melakukan pembuatan PO (*purchase order*) dan pemesanan material atau *part* kepada *supplier*.

9. Personalia dan GA

Fungsi kerja:

Melakukan proses rekrutmen karyawan baru, pengangkatan karyawan tetap, membuat peraturan perusahaan, membuat jadwal kerja serta sistem penggajian.

Setiap departemen harus menjalankan fungsi kerjanya sesuai prosedur yang telah ditetapkan oleh perusahaan agar tujuan dari perusahaan ini dapat berjalan dengan maksimal. Departemen dalam PT Kencana Gemilang memiliki bagian yang tugas dan wewenangnya berbeda satu sama lain sesuai kebutuhan organisasi. Bagian-bagian tersebut adalah:

1. Direktur

Bertugas mengawasi, mengarahkan, serta memberi perintah agar perusahaan dapat berjalan dengan baik sesuai dengan tujuan yang telah ditentukan sebelumnya.

2. Kepala bagian Produksi

Bertugas mengelola seluruh jalannya operasi produksi serta pekerja yang terdapat pada departemen produksi untuk memenuhi target dari perusahaan baik dari segi kualitas maupun kuantitas produk.

3. Kepala *shift*

Bertugas mengelola jalannya proses serta pekerja yang terdapat di departemennya dalam ruang lingkup kelompok yang dipimpin.

4. Operator

Bertugas menjalankan atau mengoperasikan mesin untuk proses yang terdapat di departemennya dan bertanggung jawab pada proses tersebut.

5. Kepala bagian PPIC (*Planning Production and Inventory Control*)



Bertugas melakukan perencanaan produksi yang menyangkut pengadaan bahan baku serta pendukung dan menjadwalkan produksi.

6. Kepala bagian QA (*Quality Assurance*)
Bertugas melakukan pengecekan barang, baik yang masuk maupun yang keluar.
7. Kepala bagian gudang material
Bertugas menerima, menyimpan, dan mendistribusikan material.
8. Kepala bagian gudang *part* metal
Bertugas menerima, menyimpan, dan mendistribusikan *part* metal.
9. Kepala bagian gudang *part*
Bertugas menerima, menyimpan, dan mendistribusikan *part* dan barang-barang yang bersifat umum.
10. Kepala bagian gudang *part* jadi
Bertugas menerima *part* jadi dari *part* setengah jadi dan melakukan penyetokan dengan spesifikasi *part*.
11. Kepala bagian gudang *finish goods*
Bertugas menerima, menyimpan, dan mendistribusikan *finish goods*.
12. Kepala bagian Personalia
Bertanggung jawab atas penyediaan, pembinaan, serta pelepasan karyawan, menjaga ketertiban umum, keamanan, kenyamanan, serta kebersihan perusahaan, dan menjaga nama baik perusahaan dengan cara menjalin hubungan yang harmonis dengan instansi yang lain.
13. Kepala bagian Marketing
Bertugas mencari *order* produk sesuai target perusahaan, melakukan perencanaan terhadap kecepatan dan kemampuan produksi, serta mengevaluasi stok bahan baku dan barang jadi.

4.4 Produk yang Dihasilkan

Produk yang dihasilkan PT Kencana Gemilang merupakan alat-alat rumah tangga dan kotak pemadam kebakaran. Produk tersebut dipasarkan secara lokal maupun impor dan dilabeli dengan *brand* “Miyako”. Dalam penulisan ini yang dibahas hanya produk yang ditangani oleh bagian PPIC KG 1, yaitu:

Tabel IV.1 Produk PT Kencana Gemilang

Gambar	Nama Produk
	<i>Magic Com</i>
	<i>Blender</i>

(Sumber: PT Kencana Gemilang, 2016)

Tabel IV.2 Tipe Blender

BL 101 GS	BL 211 PLY
BL 102 PL	BL 301 GSY
BL 151 GF	BL 302 GSY
BL 152 GF	

(Sumber: PT Kencana Gemilang, 2016)

Tabel IV.3 Tipe Magic Com

MCM 507	MCM 638 SCC
MCM 508	MCM 686 BCE
MCM 509	MCM 706
MCM 606 A	MCM 707
MCM 608	MCM 838

(Sumber: PT Kencana Gemilang, 2016)

Untuk membuat 1 unit blender atau magic com memerlukan 1 paket *part*, berikut merupakan paket *part* dari blender dan magic com:

Tabel IV.4 Paket *Part* Blender

No	<i>Part</i>	Jumlah	No	<i>Part</i>	Jumlah
1	SMALL COVER	1	11	VARIABLE SPEED CONTROLS	1
2	COVER A	1	12	PULSE SWITCH	1
3	CUP	1	13	SHOCK PROOF PLASTIC	1
4	BLADE	1	14	MIXED STICK	1
5	COVER C	1	15	DRIVER	1
6	MAT (PLASTIC)	1	16	MOTOR (1500W)	1
7	MAT (S/S)	1	17	PART D	1
8	COVER B (ABS BASE)	1	18	PC BOARD	1
9	COVER B (CASTING BASE)	1	19	P BASE	1
10	ON/OFF SWITCH	1			

(Sumber: PT Kencana Gemilang, 2016)

Tabel IV.5 Paket *Part* Magic Com

No	<i>Part</i>	Jumlah	No	<i>Part</i>	Jumlah
1	TOP COVER	1	8	SPOON	1
2	CLAMP KNOB	1	9	SWITCH	1
3	HANDLE	1	10	MEASURING CUP	1
4	DEW DISH	1	11	CLAMP HOOK	1
5	OUT CASE	1	12	PAN	1
6	PILOT LAMP	1	13	OVEN COVER	1
7	POWER CORD	1			1

(Sumber: PT Kencana Gemilang, 2016)

4.5 Dokumen yang digunakan

Pengamatan dokumen dilakukan untuk mengetahui dokumen yang digunakan dalam prosedur permintaan paket *part* ke bagian produksi pada PT Kencana Gemilang, untuk mengetahui kegunaan dokumen tersebut, berikut ini

adalah dokumen yang terkait dalam prosedur permintaan paket *part* ke bagian produksi:

1. *Production Card* dari bagian produksi

Merupakan dokumen dari bagian produksi ke bagian PPIC untuk melakukan permintaan paket *part* ke bagian produksi. Dokumen ini akan menjadi dasar dari pembuatan dokumen JOP (*Job Order Production*) yang dibuat oleh bagian PPIC. Contoh *production card* dari bagian produksi terdapat pada gambar IV.2 di bawah ini:



Gambar IV.2 *Production Card*
Sumber: PT Kencana Gemilang (2015)

Penjelasan mengenai *Production Card* diatas dapat dilihat pada Tabel IV.6 dibawah ini:

Tabel IV.6 Penjelasan *Production Card*

No	Nama	Keterangan
1.	Notes	Keterangan dari <i>Production Card</i>

2.	Date	Tanggal <i>Production Card</i> dibuat
3.	Card No	Nomer dari <i>Production Card</i>
4.	Part ID	ID dari <i>part</i> yang diminta
5.	Part Name	Nama dari ID <i>part</i> yang diminta
6.	Qty Job	Jumlah <i>part</i> yang diminta

(Sumber: Hasil Pengolahan Data, 2016)

2. JOP (*Job Order Production*)

Merupakan dokumen yang dibuat berdasarkan *production card* oleh bagian PPIC ke bagian gudang untuk mengeksekusi permintaan *part* ke bagian produksi. Contoh dokumen JOP terdapat pada gambar IV.3 di bawah ini:

PT. KENCANA GEMILANG

JOB ORDER PRODUCTION

Tanggal : 23/10/2015
Part ID : BLXX-FG151-GF
Part Name : BL 151-GF
No. Bukti Job : JOB/ 201510/00599
Qty Job : 768
Keterangan : LOKAL LINE 13

PPIC _____ PRODUCTION _____

Gambar IV.3 *Job Order Production*
Sumber: PT Kencana Gemilang (2015)

Penjelasan mengenai JOP di atas dapat dilihat pada Tabel IV.7 di bawah ini:

Tabel IV.7 Penjelasan JOP

No	Nama	Keterangan
1.	Tanggal	Tanggal JOP dibuat

2.	Part ID	ID dari <i>part</i> yang diminta
3.	Part Name	Nama dari ID <i>part</i> yang diminta
4.	No. Bukti Job	Nomer dari JOP
5.	Qty Job	Jumlah <i>part</i> yang diminta
6.	Keterangan	Keterangan dari JOP

(Sumber: Hasil Pengolahan Data, 2016)

3. Rencana Produksi

Merupakan dokumen perencanaan produksi harian. Contoh dokumen rencana produksi terdapat pada gambar IV.4 di bawah ini:

TANGGAL		26-Oct-15		
NO	MESIN /LINE	DAILY PLAN		
		PART NAME	PART NO	QTY
1	1	MCM 507	MCMX-FG507	1900
2	2	MCM 508	MCMX-FG508	1800
3	3	MCM 509	MCMX-FG509	1000
4	4	MCM 606 A	MCMX-FG606-A	1200
5	5	MCM 608	MCMX-FG608	1200
6	6	MCM 638 SCC	MCMX-FG638-SCC	1200
7	7	MCM 686 BCE	MCMX-FG686-BCE	1200
8	8	MCM 706	MCMX-FG706	2000
9	9	MCM 707	MCMX-FG707	2000
10	10	MCM 838	MCXX-FG838	1400
TOTAL MCM				14,900
11	11	BL 101GS	BLXX-FG101-GS	1000
12	12	BL 102 PL	BLXX-FG102-PL	1200
13	13	BL 151GF	BLXX-FG151-GF	1500
14	14	BL 152 GF	BLXX-FG152-GF	580
15	15	BL 211PLY	BLXX-FG211-PLY	1500
16	16	BL 301GSY	BLXX-FG301-GSY	510
17	17	BL 302 GSY	BLXX-FG302-GSY	1200
TOTAL BL				7,490

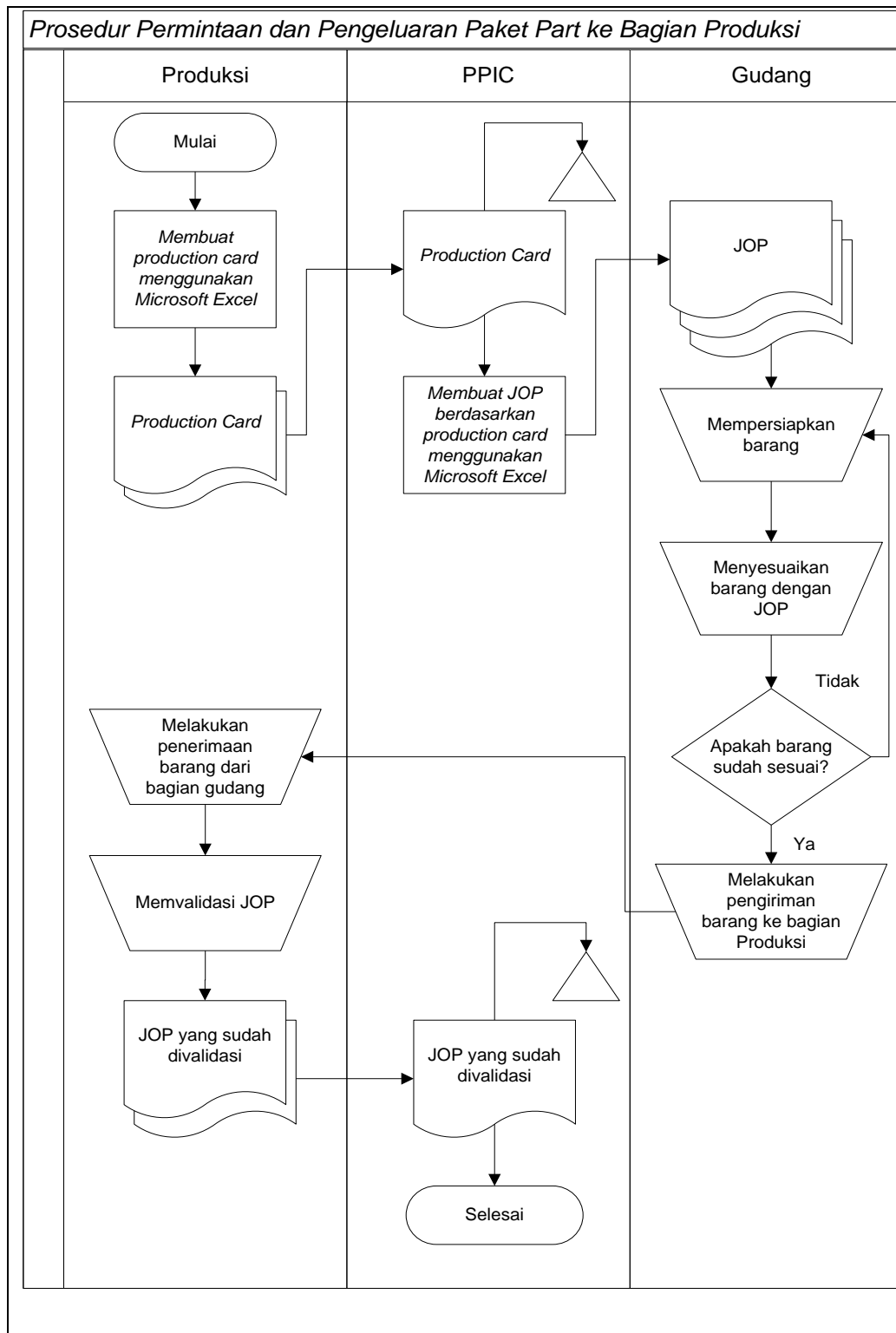
Gambar IV.4 Rencana Produksi
Sumber: PT Kencana Gemilang (2015)

4.6 Prosedur Permintaan dan Pengeluaran Paket *Part* ke Bagian Produksi

Prosedur permintaan dan pengeluaran paket *part* ke bagian produksi dapat dilihat melalui Gambar IV.5. Berikut tahapannya:

1. Bagian PPIC menerima *production card* dari bagian produksi.
2. Berdasarkan *production card* yang diterima, dibuatlah JOP (*Job Order Production*) yang merupakan perintah pengeluaran *part* ke area produksi.

3. Kepala Gudang menerima JOP, kemudian operator gudang persiapan barang menyiapkan barang sesuai dengan JOP, lalu diserahkan ke operator gudang bagian *Supply* untuk memeriksa kelengkapan barang.
4. Jika barang tidak sesuai dengan JOP, maka operator gudang bagian *Supply* memberikan informasi ke gudang persiapan barang untuk melakukan persiapan ulang.
5. Jika barang sudah sesuai dengan JOP, maka operator gudang bagian *Supply* melakukan penyerahan barang ke bagian produksi.



Gambar IV.5 Flowmap Permintaan dan Pengeluaran Paket Part ke Bagian Produksi
(Sumber: Hasil Pengolahan, 2016)

4.7 Prosedur Sistem Informasi Permintaan dan Pengeluaran Paket *Part* ke Bagian Produksi yang Sedang Berjalan

Pengolahan data dan analisis terhadap sistem yang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah apa yang sedang dihadapi sistem untuk dapat dijadikan dasar perancangan sistem yang baru.

Kegiatan pengolahan data dan analisis sistem yang berjalan menggunakan analisis sistem yang berorientasi pada objek-objek sangat diperlukan oleh sistem yang akan dirancang. Analisis sistem berorientasi objek bertujuan untuk menitikberatkan kepada fungsionalitas sistem yang berjalan dengan tidak terlalu menitikberatkan pada alur proses dari sistem.

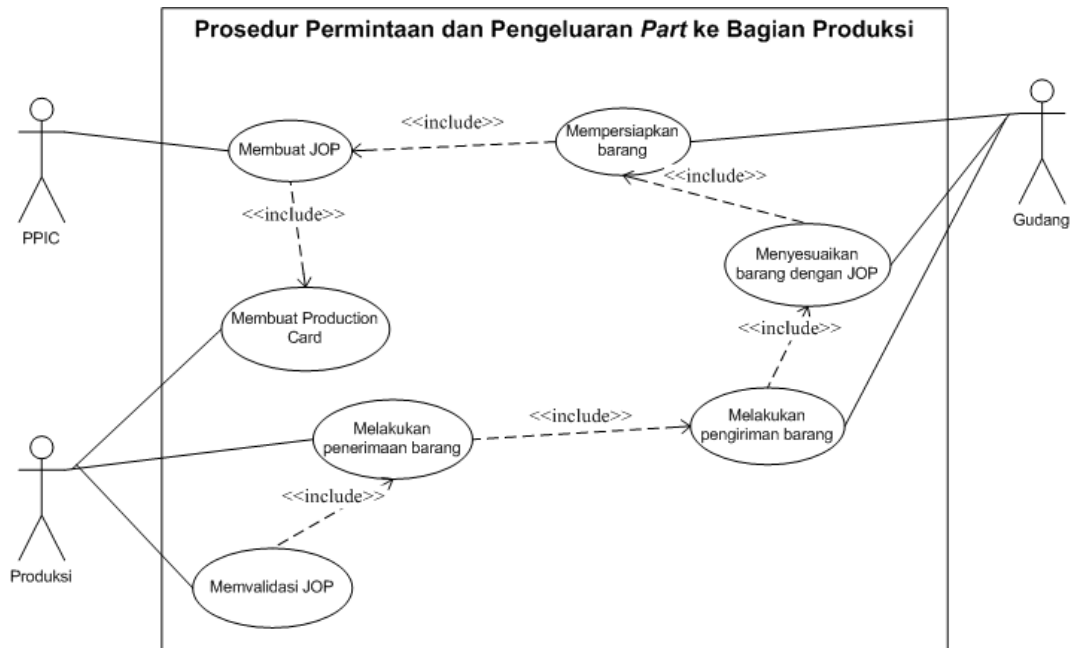
Selanjutnya, dari hasil analisis ini divisualisasi dan didokumentasi dengan *Unified Modeling Language* (UML) melalui diagram *use case*, skenario *use case*, dan *activity diagram*. Pertimbangan dari diagram ini dapat mewakili secara keseluruhan sistem yang berjalan dan dapat dimengerti oleh *user*.

4.7.1 *Use Case Diagram* yang Berjalan pada Prosedur Permintaan dan Pengeluaran Paket *Part* ke Bagian Produksi

Use case diagram menggambarkan siapa saja *actor* yang melakukan prosedur dalam sistem serta fungsi-fungsi (proses) yang terlibat dalam transformasi pada sistem tersebut.

Pada dasarnya *actor* bukanlah bagian dari *use case diagram*, namun untuk terciptanya suatu *use case diagram* diperlukan beberapa *actor*, dimana *actor* tersebut menggambarkan seseorang atau sesuatu seperti perangkat, atau sistem lain yang berinteraksi dengan sistem. *Actor* adalah seseorang atau sistem lain yang berinteraksi mengirim atau menerima pesan dari dan ke dalam sistem dengan sistem.

Use case diagram prosedur permintaan dan pengeluaran paket *part* ke bagian produksi pada PT Kencana Gemilang dapat dilihat pada Gambar IV.6.



Gambar IV.6 Use Case Diagram Prosedur Permintaan dan Pengeluaran Paket Part ke Bagian Produksi
Sumber : Hasil Pengolahan Data (2016)

Penjelasan *use case diagram* prosedur permintaan dan pengeluaran paket *part* ke bagian produksi yang sedang berjalan di PT Kencana Gemilang adalah sebagai berikut:

a. Definisi Aktor

Pendefinisian aktor pada *use case diagram* prosedur permintaan dan pengeluaran paket *part* ke bagian produksi yang sedang berjalan pada PT Kencana Gemilang dapat dilihat pada Tabel IV.8 berikut:

Tabel IV.8 Definisi Aktor Prosedur Permintaan dan Pengeluaran Paket Part ke Bagian Produksi

No.	Aktor	Definisi
1.	PPIC	Membuat JOP dengan menggunakan Microsoft Excel.

(Sumber: Hasil Pengolahan Data, 2016)

Tabel IV.8 Definisi Aktor Prosedur Permintaan dan Pengeluaran Paket *Part* ke Bagian Produksi (lanjutan)

No.	Aktor	Definisi
2.	Gudang	Mempersiapkan barang serta mengecek kelengkapan barang, dan melakukan pengiriman barang ke bagian produksi.
3.	Produksi	Melakukan penerimaan barang dari bagian gudang.

(Sumber: Hasil Pengolahan Data, 2016)

b. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* prosedur permintaan dan pengeluaran Paket *part* ke bagian produksi yang sedang berjalan di PT Kencana Gemilang dapat dilihat pada Tabel IV.9 berikut:

Tabel IV.9 Definisi *Use Case* Prosedur Permintaan dan Pengeluaran Paket *Part* ke Bagian Produksi

No.	<i>Use Case</i>	Definisi
1.	Membuat JOP	Proses pembuatan JOP
2.	Mempersiapkan barang	Proses persiapan barang sesuai dengan JOP
3.	Menyesuaikan barang	Proses penyesuaian barang dengan JOP.
4.	Melakukan pengiriman barang	Proses pengiriman barang ke bagian produksi.
5.	Membuat PC	Proses pembuatan <i>production card</i>
6.	Melakukan penerimaan barang	Proses penerimaan barang dari bagian gudang.
7.	Memvalidasi JOP	Proses memvalidasi JOP

(Sumber: Hasil Pengolahan Data, 2016)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, dibutuhkan sistem informasi permintaan dan pengeluaran paket *part* agar mempermudah proses permintaan dan pengeluaran paket *part* dari gudang ke bagian produksi. Untuk mewujudkan hal tersebut penulis berusaha membuat aplikasi yang *user friendly*, sehingga lebih mudah pengoperasiannya bagi pengguna yang baru menggunakan aplikasi permintaan dan pengeluaran paket *part*.

Kebutuhan program aplikasi dapat dijelaskan melalui proses-proses yang dibutuhkan pemakai dan kebutuhan sistem berdasarkan usulan solusi yang dibuat. Identifikasi kebutuhan sistem komputerisasi menjelaskan proses-proses yang dilakukan oleh program aplikasi, dimana proses-proses tersebut dibutuhkan oleh pemakai. Berikut adalah penjelasan identifikasi kebutuhan user aplikasi permintaan dan pengeluaran paket *part* pada PT Kencana Gemilang.

Tabel V.1 Identifikasi Kebutuhan *User*

No	Identifikasi Kebutuhan	Uraian
1	Proses Sistem Berjalan	1. Membuat <i>production card</i>
2	Data Masukan (<i>input</i>)	1. Data rencana produksi 2. Data <i>production part</i>
3	Data Keluaran (<i>output</i>)	1. JOP (<i>Job Order Production</i>) 2. Laporan permintaan dan pengeluaran paket <i>part</i> dan Laporan stok <i>part</i>

Sumber : Hasil Analisis Data(2018)

Tabel V.1 Identifikasi Kebutuhan *User* (lanjutan)

No	Identifikasi Kebutuhan	Uraian
4	Model sistem informasi	1. Input : <i>production card</i> Proses : Menginput <i>production card</i> Output : JOP 2. Input : JOP Proses : Menginput JOP Output : Laporan permintaan dan Pengeluaran paket <i>part</i> dan laporan stok

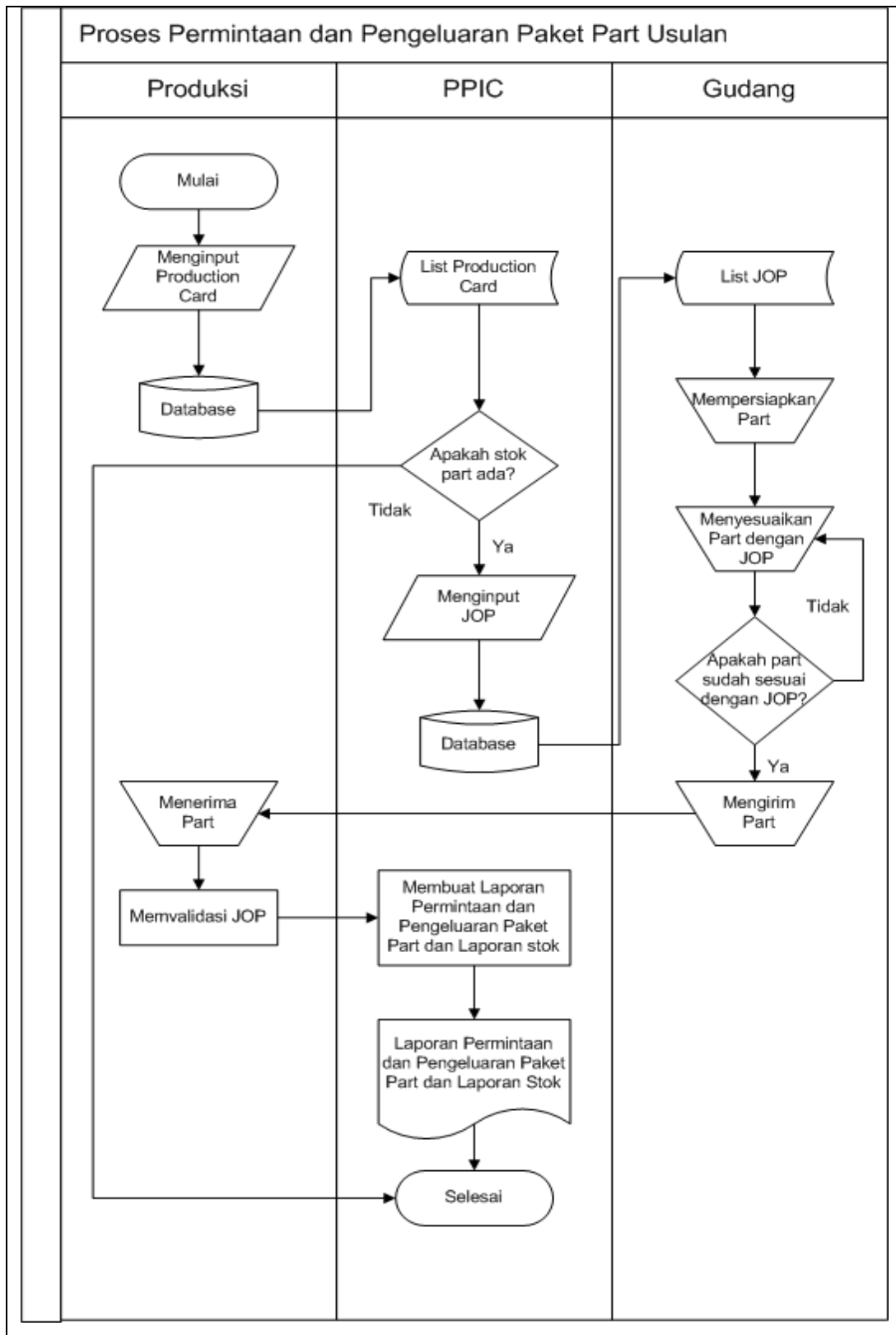
Sumber : Hasil Analisis Data(2018)

5.2 Sistem Informasi Permintaan dan Pengeluaran Paket *Part* yang Diusulkan

Sistem informasi yang diusulkan bertujuan untuk mengembangkan prosedur dan proses suatu data agar tujuan dari suatu organisasi dapat tercapai. Perancangan dibuat untuk meminimalkan kekurangan dari sistem lama yang kurang efektif dan efisien ke sistem baru yang lebih terprogram dalam permintaan dan pengeluaran paket *part* pada PT Kencana Gemilang. Berikut adalah prosedur perancangan sistem informasi permintaan dan pengeluaran paket *part* usulan:

- a. Bagian Produksi memasukan PC (*production card*) ke dalam sistem untuk meminta *part* berdasarkan rencana produksi.
- b. PPIC membuatkan perintah pengeluaran *part* atau JOP (*Job Order Production*) berdasarkan PC dari bagian Produksi.

- c. Bagian gudang melihat JOP, kemudian mempersiapkan barang sesuai dengan JOP untuk selanjutnya memeriksa kelengkapan barang. Jika barang sesuai dengan JOP maka barang diantar ke bagian Produksi.
- d. Bagian produksi melakukan validasi JOP setelah menerima barang.
- e. Bagian PPIC membuat laporan permintaan dan pengeluaran paket *part* dan Laporan stok.



Gambar V.1 *Flowmap* Permintaan dan Pengeluaran Paket *Part* yang Diusulkan

Sumber : Hasil Analisis Data(2018)

5.3 Analisis Sistem

Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan. Pada tahap ini dilakukan upaya memperbaiki sistem atau membangun sistem dengan memanfaatkan teknologi baru untuk mengurangi dan mengatasi berbagai permasalahan yang telah terjadi pada sistem yang lama sehingga dapat meningkatkan efektifitas dan efisiensi. Aplikasi yang dirancang diharapkan memberi solusi alternatif baru yang memberi kemudahan terhadap proses permintaan dan pengeluaran paket *part* pada PT Kencana Gemilang.

Tujuan analisis sistem adalah untuk memberikan penjelasan dan mendefinisikan kepada *user* yang akan menggunakan program. Pembuatan program ini diharapkan dapat mengatasi kekurangan yang ada dan membantu *user* dalam menyelesaikan pekerjaannya. Adapun perancangan yang diusulkan merupakan langkah untuk lebih mengefektifkan dan mengefisienkan sistem yang sudah terkomputerisasi dengan menggunakan sistem yang terprogram.

5.3.1 Use Case Diagram Yang Diusulkan

Use case merupakan pemodelan untuk sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Alasan memilih diagram *use case* untuk menggambarkan sistem yang diusulkan, agar mempermudah dalam menggambarkan dan membaca sistem informasi yang akan diusulkan.

Berikut adalah definisi *use case* dan definisi aktor dari *use case* diagram permintaan dan pengeluaran paket *part* yang diusulkan:

Tabel V.2 Definisi Aktor

No	Aktor	Deskripsi
1	Produksi	Aktor yang melakukan permintaan paket <i>part</i> .
2	PPIC	Aktor yang membuat JOP untuk mengeluarkan paket <i>part</i> dari gudang ke bagian Produksi.
3.	Gudang	Aktor yang melakukan pengiriman paket <i>part</i> .

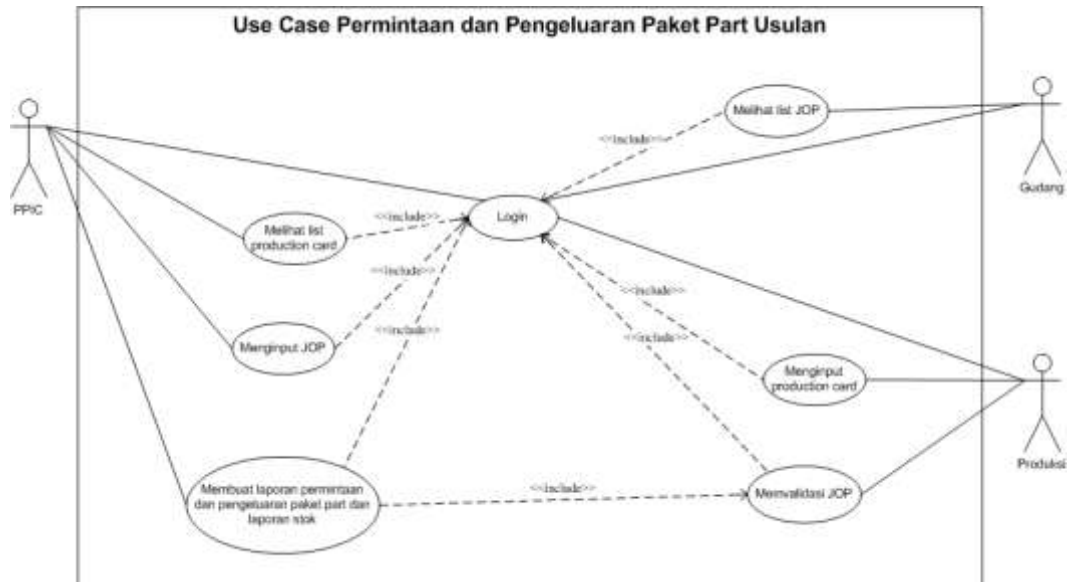
Sumber: Hasil Analisis Data (2018)

Tabel V.3 Definisi *Use Case*

No	<i>Use case</i>	Definisi
1	Login	Verifikasi <i>username</i> dan <i>password</i> untuk admin dan <i>user</i>
2	<i>Input Production Card</i>	Memasukkan permintaan paket <i>part</i> ke dalam sistem
3	<i>Input JOP</i>	Memasukan perintah pengeluaran paket <i>part</i> sesuai dengan <i>production card</i> ke dalam sistem
4	Melihat <i>list production card</i>	Proses melihat <i>list production card</i>
5	Melihat <i>list JOP</i>	Proses melihat <i>list JOP</i>
6	Memvalidasi JOP	Proses memvalidasi JOP
7	Membuat Laporan	Membuat laporan permintaan dan pengeluaran paket <i>part</i> dan laporan stok

Sumber: Hasil Analisis Data (2018)

Berdasarkan analisis yang telah dilakukan, maka *use case* diagram yang diusulkan dapat dilihat pada Gambar V.2 berikut ini:



Gambar V.2 *Use Case Diagram* Permintaan dan Pengeluaran Paket *Part Usulan*

Sumber: Hasil Analisis Data(2018)

Berikut adalah skenario *use case* diagram proses permintaan dan pengeluaran paket *part* yang diusulkan:

1. Skenario *Use Case Login*

Tabel V.4 Skenario *Use Case Login*

<i>Use case name</i>	Login
<i>Actor</i>	Produksi, PPIC, Gudang
<i>Description</i>	Memberikan hak akses yang berbeda-beda
<i>Normal flow</i>	1. Memasukkan <i>username</i> dan <i>password</i> 2. Menampilkan halaman utama aplikasi
<i>Alternate flow</i>	Salah memasukkan <i>username</i> dan <i>password</i> tidak dapat masuk ke halaman utama

Sumber: Hasil Analisis Data (2018)

2. Skenario Use Case Input Production Card

Tabel V.5 Skenario Use Case input Production Card

<i>Use case name</i>	<i>input production card</i>
<i>Actor</i>	Produksi
<i>Description</i>	Meng- <i>input</i> permintaan paket <i>part</i>
<i>Normal flow</i>	Memasukkan permintaan paket <i>part</i> yang dibutuhkan Produksi
<i>Alternate flow</i>	

Sumber: Hasil Analisis Data (2018)

3. Skenario Use Case Input JOP

Tabel V.6 Skenario Use Case Input JOP

<i>Use case name</i>	<i>Input JOP</i>
<i>Actor</i>	PPIC
<i>Description</i>	Memasukan data pengeluaran paket <i>part</i> berdasarkan <i>production card</i>
<i>Normal flow</i>	Memasukan JOP sesuai dengan <i>production card</i>
<i>Alternate flow</i>	

Sumber: Hasil Analisis Data (2018)

4. Skenario Use Case Melihat List Production Card

Tabel V.7 Skenario Use Case Melihat List Production Card

<i>Use case name</i>	Melihat <i>List Production Card</i>
<i>Actor</i>	PPIC
<i>Description</i>	Melihat <i>list production card</i>
<i>Normal flow</i>	Melihat <i>list</i>
<i>Alternate flow</i>	

Sumber: Hasil Analisis Data (2018)

5. Skenario *Use Case* Melihat *List JOP*

Tabel V.8 Skenario *Use Case* Melihat *List JOP*

<i>Use case name</i>	Melihat <i>List JOP</i>
<i>Actor</i>	Gudang
<i>Description</i>	Melihat <i>list JOP</i>
<i>Normal flow</i>	Melihat <i>list</i>
<i>Alternate flow</i>	

Sumber: Hasil Analisis Data (2018)

6. Skenario *Use Case* Memvalidasi *JOP*

Tabel V.9 Skenario *Use Case* Memvalidasi *JOP*

<i>Use case name</i>	Memvalidasi <i>JOP</i>
<i>Actor</i>	Produksi
<i>Description</i>	Memvalidasi bahwa barang sudah diterima oleh bagian produksi
<i>Normal flow</i>	Memvalidasi <i>JOP</i>
<i>Alternate flow</i>	

Sumber: Hasil Analisis Data (2018)

7. Skenario *Use Case* Membuat Laporan

Tabel V.10 Skenario *Use Case* Membuat Laporan

<i>Use case name</i>	Membuat Laporan
<i>Actor</i>	PPIC
<i>Description</i>	Membuat laporan permintaan dan pengeluaran paket <i>part</i> dan laporan stok
<i>Normal flow</i>	Membuat laporan
<i>Alternate flow</i>	

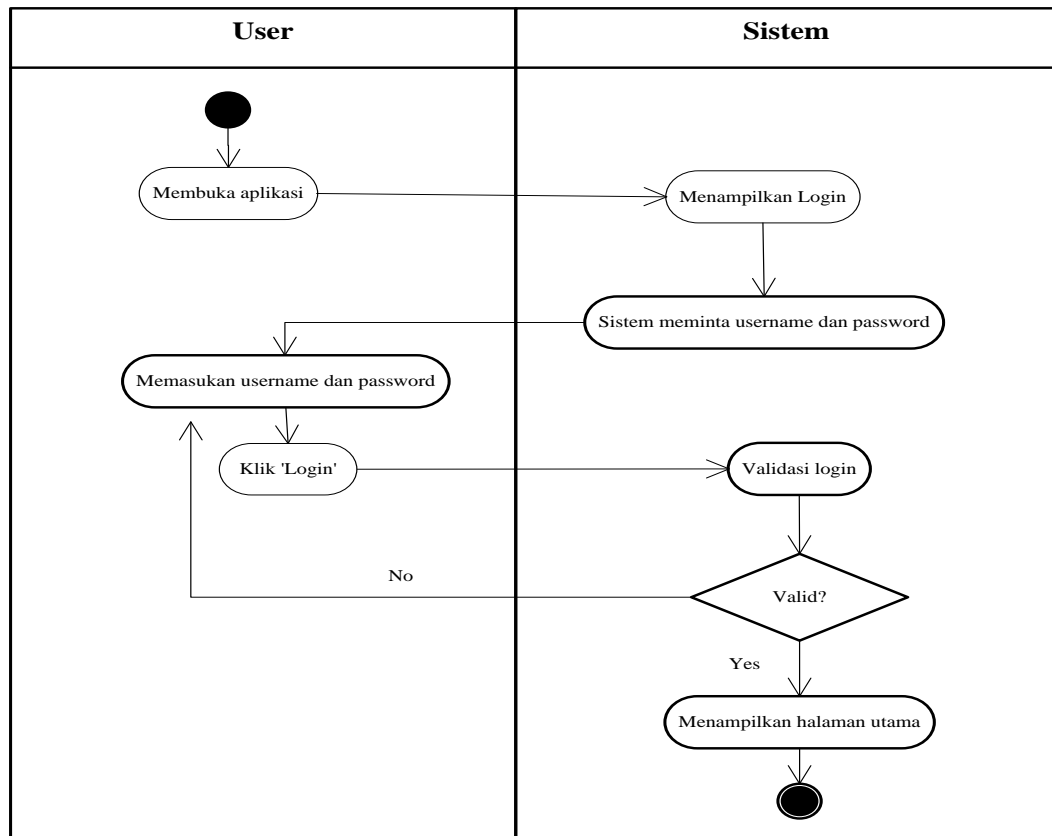
Sumber: Hasil Analisis Data (2018)

5.3.2 Activity Diagram Yang Diusulkan

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar lebih memahami sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem informasi permintaan dan pengeluaran paket *part*.

1. Activity Diagram Proses Login

Diagram berikut merupakan diagram aktivitas yang menjelaskan kegiatan *login* terhadap sistem, seperti yang dapat dilihat pada gambar V.3 sebagai berikut:

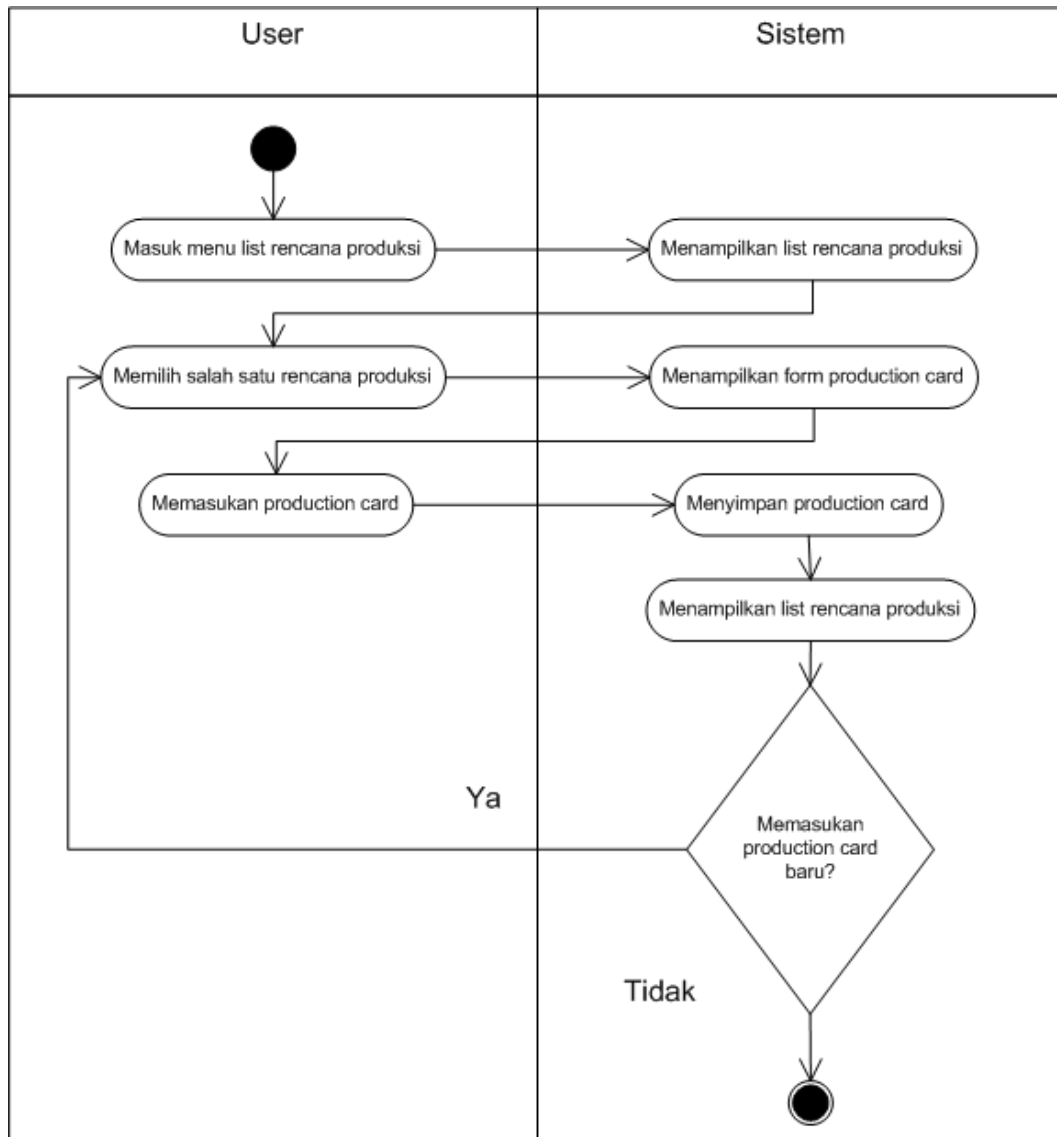


Gambar V.3 Activity Diagram Proses Login

Sumber : Hasil Analisa Data (2018)

2. Activity Diagram input production card

Diagram berikut merupakan diagram aktivitas yang menjelaskan kegiatan memasukan *production card* seperti yang dapat dilihat pada Gambar V.4 sebagai berikut:

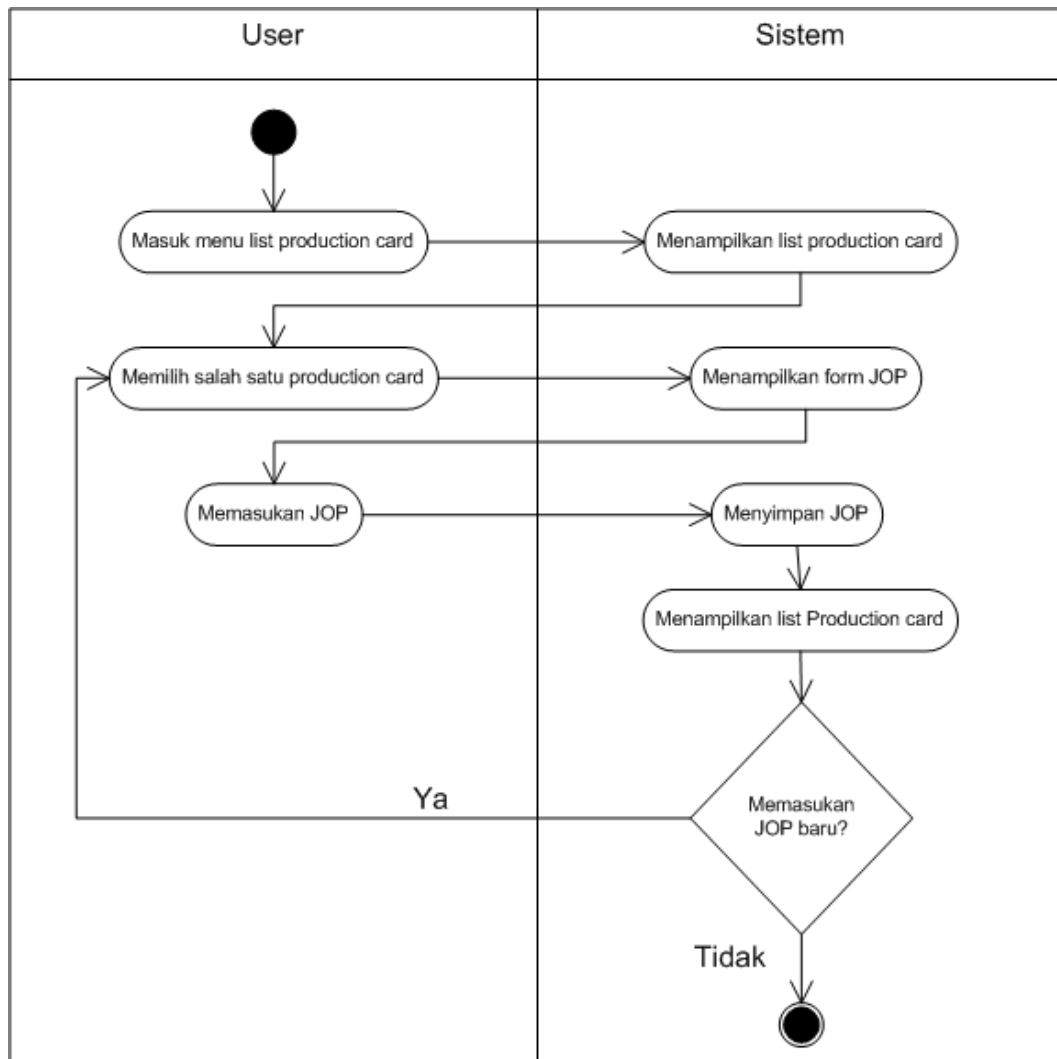


Gambar V.4 Activity Diagram input production card yang Diusulkan

Sumber : Hasil Analisa Data (2018)

3. Activity Diagram input JOP

Diagram berikut merupakan diagram aktivitas yang menjelaskan kegiatan memasukan JOP, seperti yang dapat dilihat pada Gambar V.5 sebagai berikut:

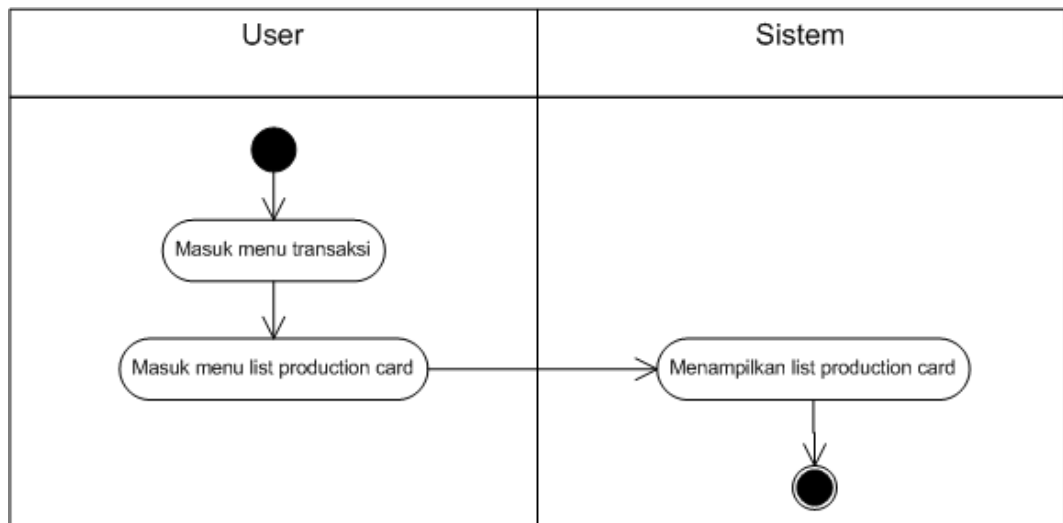


Gambar V.5 Activity Diagram input JOP yang Diusulkan

Sumber : Hasil Analisa Data (2018)

4. Activity Diagram Melihat List Production Card

Diagram berikut merupakan diagram aktivitas yang menjelaskan kegiatan melihat *list production card*, seperti yang dapat dilihat pada Gambar V.6 sebagai berikut:

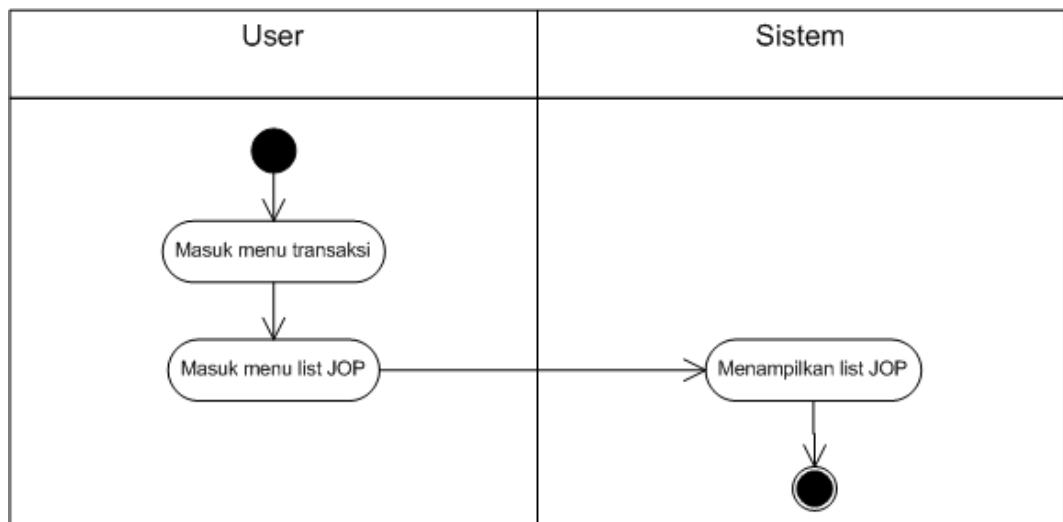


Gambar V.6 *Activity Diagram* Melihat *List Production Card* yang Diusulkan

Sumber : Hasil Analisa Data (2018)

5. *Activity Diagram* Melihat *List JOP*

Diagram berikut merupakan diagram aktivitas yang menjelaskan kegiatan melihat *list JOP*, seperti yang dapat dilihat pada Gambar V.7 sebagai berikut:

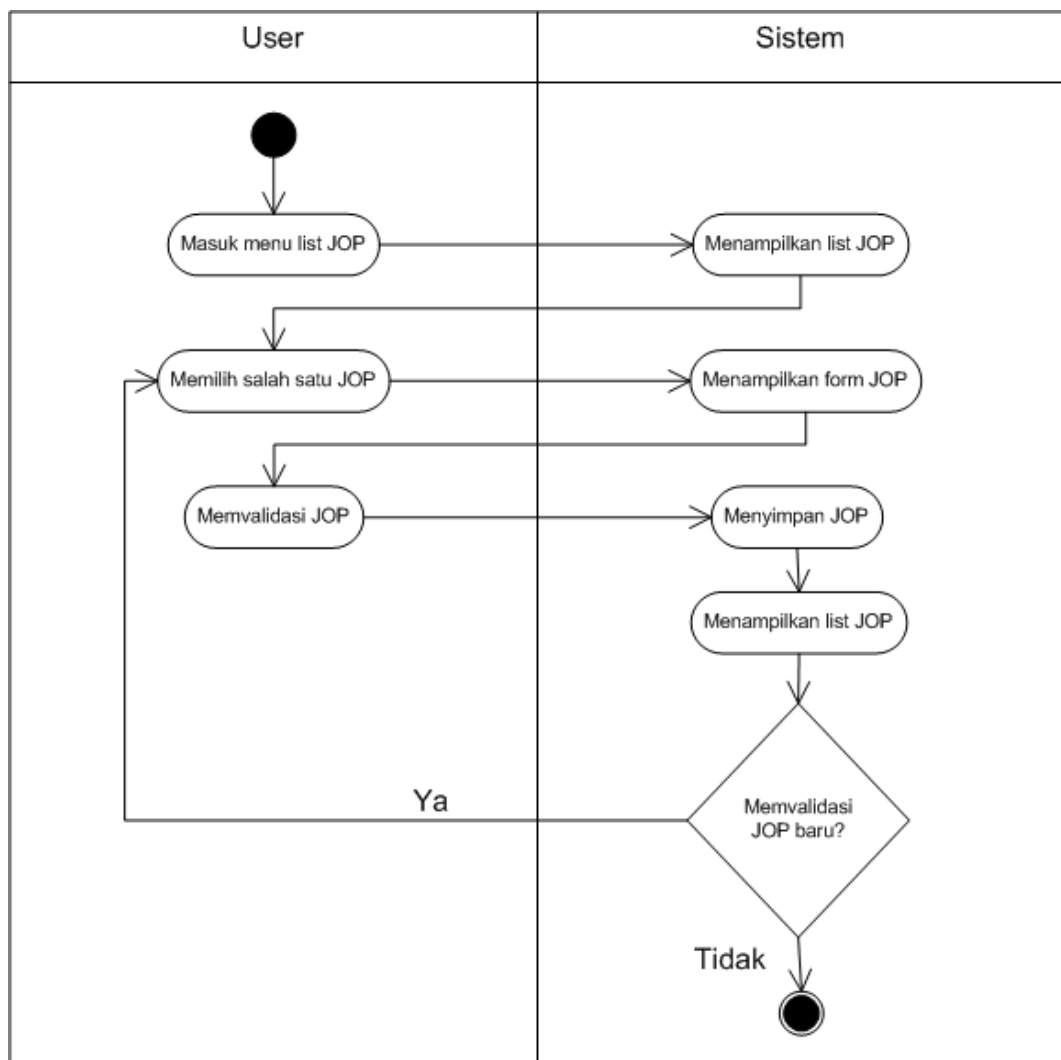


Gambar V.7 *Activity Diagram* Melihat *List JOP* yang Diusulkan

Sumber : Hasil Analisa Data (2018)

6. Activity Diagram Memvalidasi JOP

Diagram berikut merupakan diagram aktivitas yang menjelaskan kegiatan memvalidasi JOP, seperti yang dapat dilihat pada Gambar V.8 sebagai berikut:

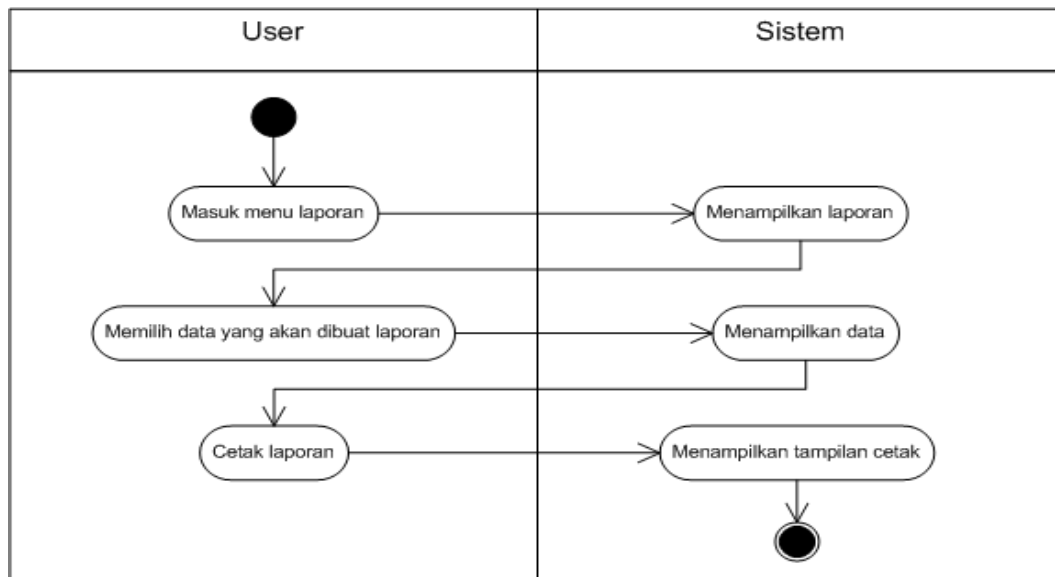


Gambar V.8 Activity Diagram memvalidasi JOP yang Diusulkan

Sumber : Hasil Analisa Data (2018)

7. Activity Diagram Membuat Laporan

Diagram berikut merupakan diagram aktivitas yang menjelaskan kegiatan membuat laporan, seperti yang dapat dilihat pada Gambar V.9 sebagai berikut:



Gambar V.9 Activity Diagram Membuat Laporan yang Diusulkan

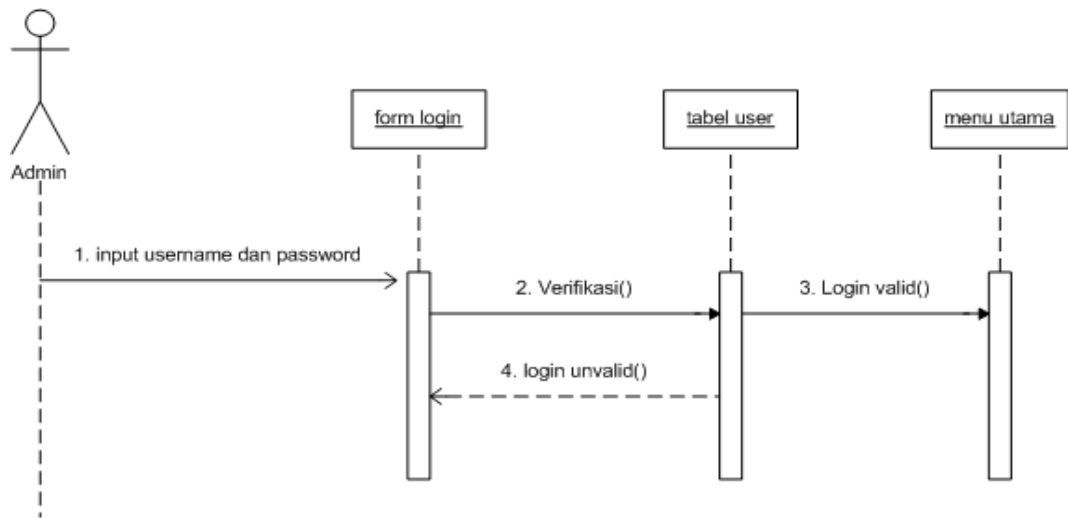
Sumber : Hasil Analisa Data (2018)

5.3.3 Sequence Diagram Yang Diusulkan

Sequence diagram menjelaskan secara detail urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*. *Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (pengguna, *display* dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi *vertical* (waktu) dan dimensi *horizontal* (*object-object* yang terkait). *Sequence diagram* digunakan untuk memodelkan pengiriman *message* antar *objects*.

1. Sequence Diagram Proses Login

Sequence Diagram login menggambarkan proses yang sedang terjadi dalam *login* admin. Prosesnya dimulai dengan admin masuk kedalam *form login* terlebih dahulu, kemudian selanjutnya admin dapat melakukan *login* dengan memasukkan *username* dan *password* untuk dapat masuk kedalam sistem. Proses *login* itu sendiri diikuti juga validasi yang dilakukan oleh sistem dimana terdapat proses verifikasi *username* dan *password* apakah sesuai atau tidak. Apabila berhasil admin tersebut dapat masuk dan menggunakan sistem, seperti yang dapat dilihat pada gambar V.10 sebagai berikut:

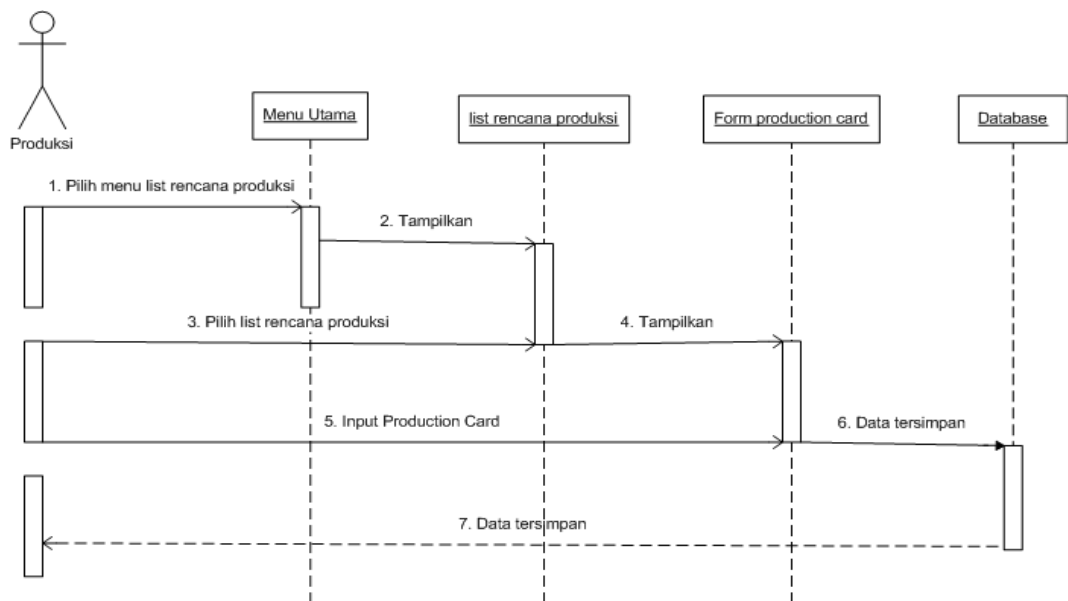


Gambar V.10 *Sequence Diagram Login* yang Diusulkan

Sumber : Hasil Analisa Data(2018)

2. *Sequence Diagram Input Production Card*

Sequence Diagram yang merupakan proses *input production card*. Diasumsikan bahwa admin sudah melakukan proses *login*, setelah itu admin akan masuk ke halaman utama dan memilih menu transaksi untuk *input production card*.

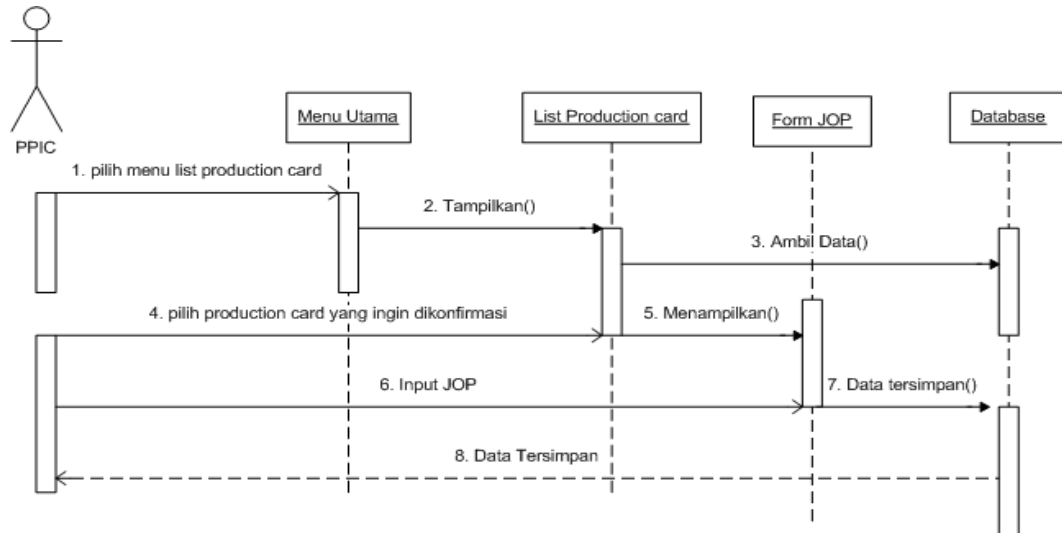


Gambar V.11 *Sequence Diagram Input Production Card* yang Diusulkan

Sumber : Hasil Analisa Data (2018)

3. *Sequence Diagram Input JOP*

Sequence Diagram yang merupakan proses *input JOP*. Diasumsikan bahwa admin sudah melakukan proses *login*, setelah itu admin akan masuk ke halaman utama dan memilih menu transaksi untuk *input JOP*.

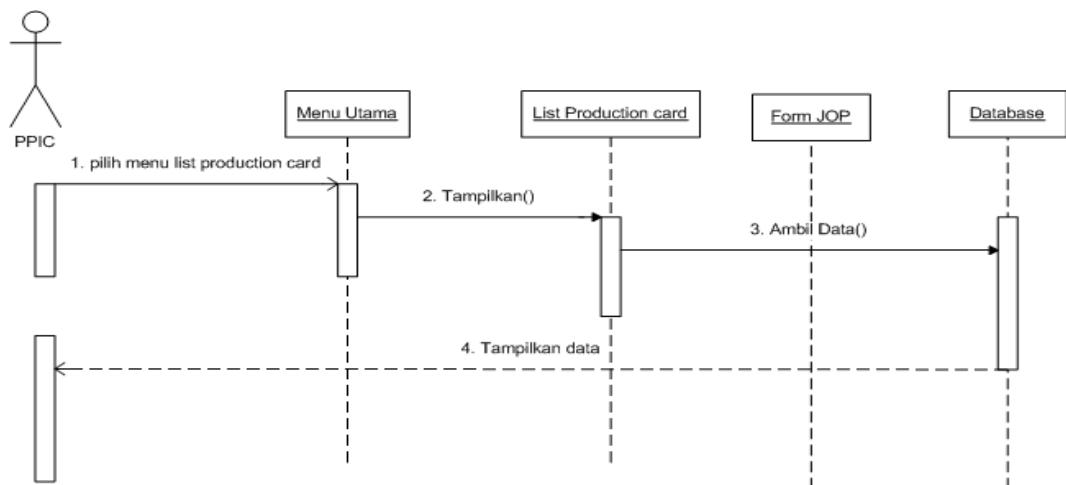


Gambar V.11 *Sequence Diagram Input JOP* yang Diusulkan

Sumber : Hasil Analisa Data (2018)

4. *Sequence Diagram Melihat List Production Card*

Sequence Diagram Melihat List Production Card yang merupakan proses melihat *list production card*. Diasumsikan bahwa admin sudah melakukan proses *login*, setelah itu admin akan memilih menu *List Production Card*.

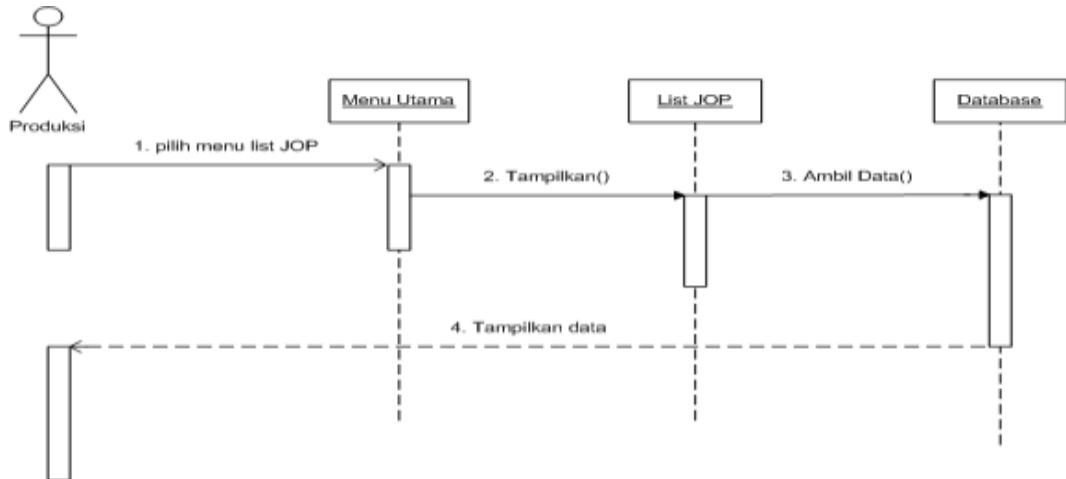


Gambar V.12 *Sequence Diagram Melihat List Production Card* yang Diusulkan

Sumber : Hasil Analisa Data (2018)

5. Sequence Diagram Melihat List JOP

Sequence Diagram Melihat List JOP yang merupakan proses melihat list JOP. Diasumsikan bahwa admin sudah melakukan proses login, setelah itu admin akan memilih menu List JOP.

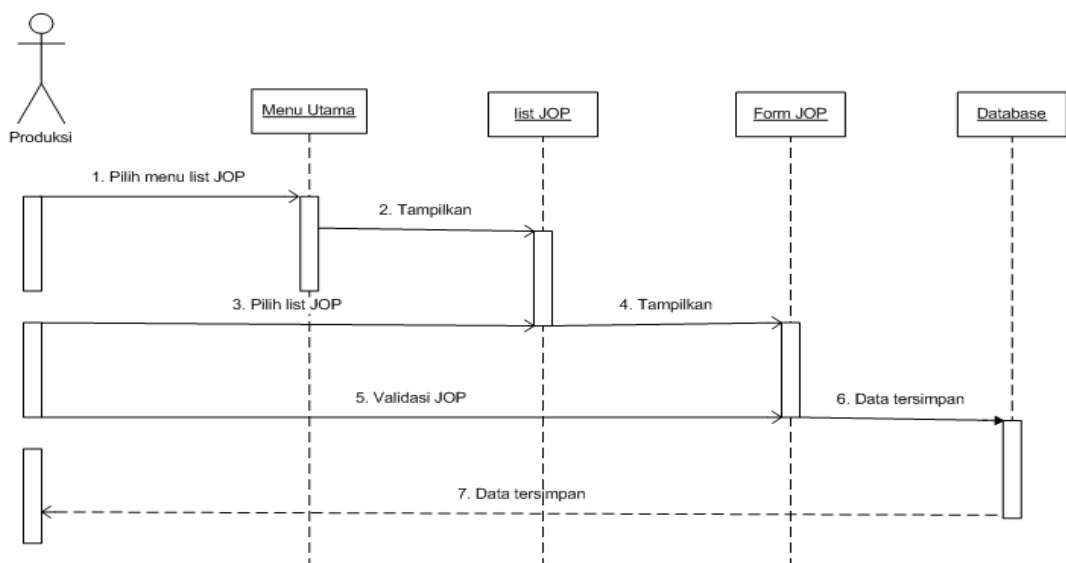


Gambar V.13 Sequence Diagram Melihat List JOP yang Diusulkan

Sumber : Hasil Analisa Data (2018)

6. Sequence Diagram Memvalidasi JOP

Sequence Diagram Memvalidasi JOP yang merupakan proses memvalidasi JOP. Diasumsikan bahwa admin sudah melakukan proses login.

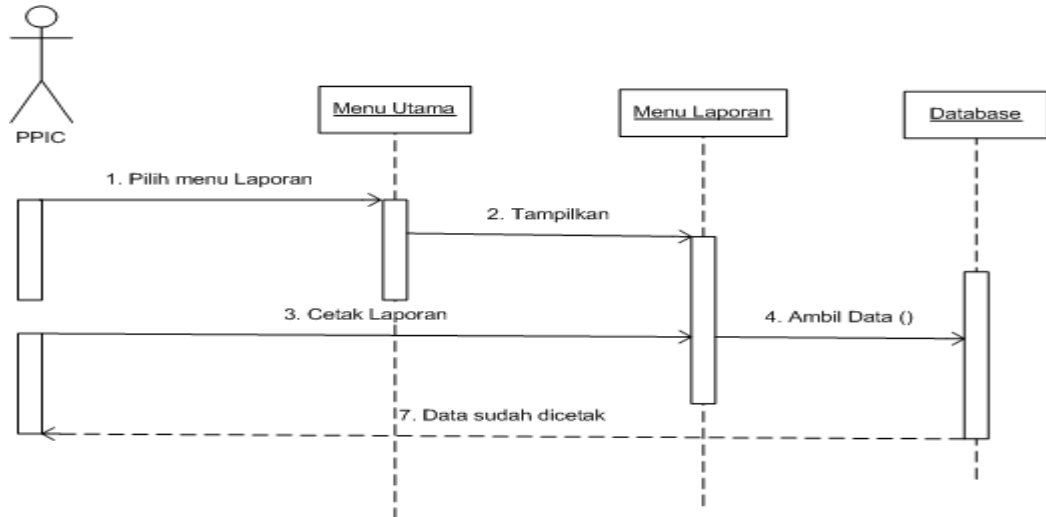


Gambar V.14 Sequence Diagram Memvalidasi JOP yang Diusulkan

Sumber : Hasil Analisa Data (2018)

7. *Sequence Diagram* Membuat Laporan

Sequence Diagram Membuat Laporan yang merupakan proses Membuat Laporan. Diasumsikan bahwa admin sudah melakukan proses *login*.



Gambar V.15 *Sequence Diagram* Membuat Laporan yang Diusulkan

Sumber : Hasil Analisa Data (2018)

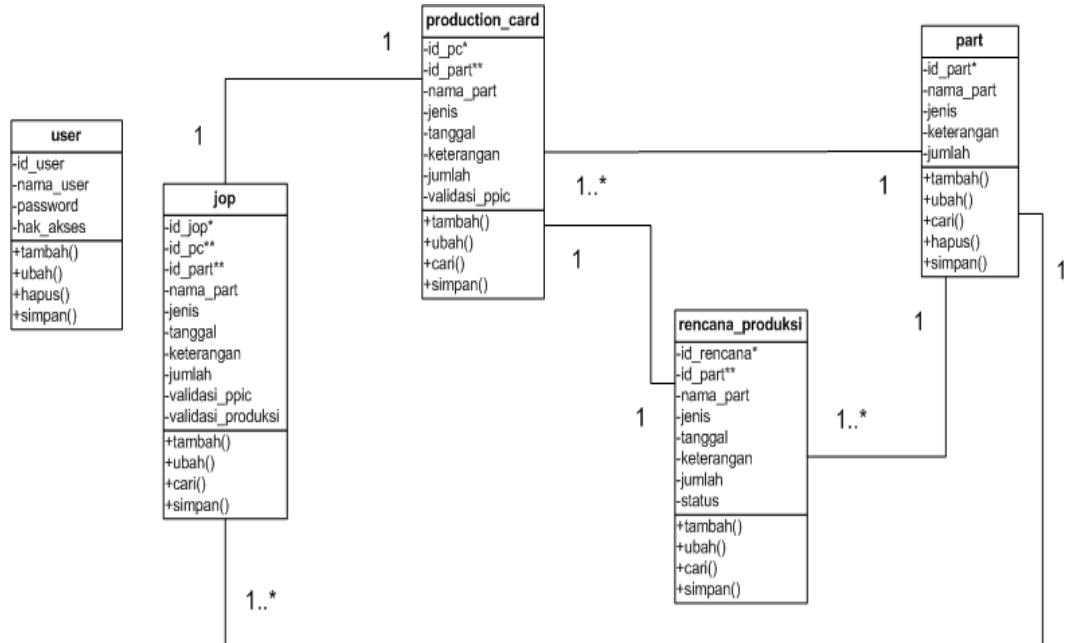
5.4 Perancangan Data

Pemodelan data adalah metode yang digunakan untuk menentukan dan menganalisis persyaratan data yang diperlukan untuk mendukung proses bisnis suatu organisasi. Data yang dibutuhkan adalah dicatat sebagai data model konseptual dengan definisi data yang terkait.

5.4.1 *Class Diagram* yang Diusulkan

Class diagram membantu dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model desain dari suatu sistem. Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam menggambarkan struktur dari semua kelas yang membentuk arsitektur yang

dibuat, seperti yang dapat dilihat *class diagram* aplikasi permintaan dan pengeluaran paket *part* adalah sebagai berikut:



Gambar V.16 *Class Diagram* yang Diusulkan

Sumber: Hasil Analisis Data (2018)

5.4.2 Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan dan basis data. Berikut adalah kamus data sistem permintaan dan pengeluaran paket *part*:

1. Spesifikasi Tabel User

- Nama Tabel : user
- Fungsi : Untuk menyimpan, melihat dan mengubah data user
- Tipe : File Data Master

Tabel V.11 Spesifikasi Tabel User

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID user	id_user	Varchar	15	Primary Key
2	Nama user	nama_user	Varchar	15	

3	Password	password	<i>Varchar</i>	15	
4	Hak akses	hak_akses	<i>Varchar</i>	15	

Sumber: Hasil Analisis Data (2018)

2. Spesifikasi Tabel *Part*

Nama Tabel : *part*

Fungsi : Untuk menyimpan, melihat dan mengubah data *part*

Tipe : File Data Master

Tabel V.12 Spesifikasi Tabel *Part*

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID <i>Part</i>	id_part	<i>Varchar</i>	20	<i>Primary Key</i>
2	Nama <i>Part</i>	nama_part	<i>Varchar</i>	20	
3	Jenis <i>Part</i>	Jenis	<i>Varchar</i>	20	
4	Keterangan <i>Part</i>	keterangan	<i>Varchar</i>	100	
5	Jumlah <i>Part</i>	jumlah	<i>Integer</i>	11	

Sumber: Hasil Analisis Data (2018)

3. Spesifikasi Tabel JOP

Nama Tabel : *jop*

Fungsi : Untuk menampilkan jop dari bagian PPIC

Tipe : File Data Transaksi

Tabel V.13 Spesifikasi Tabel JOP

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID JOP	id_jop	<i>Varchar</i>	20	<i>Primary Key</i>
2	ID Production Card	id_pc	<i>Varchar</i>	20	<i>Foreign key</i>
3	ID Part	id_part	<i>Varchar</i>	20	
4	Nama Part	nama_part	<i>Varchar</i>	20	
5	Jenis Part	jenis	<i>Varchar</i>	20	
6	Tanggal	tanggal	<i>Date</i>		

7	Keterangan Part	keterangan	<i>Varchar</i>	100	
8	Jumlah Part	jumlah	<i>Integer</i>	11	
9	Validasi PPIC	validasi_ppic	<i>Integer</i>	1	
10	Validasi Produksi	validasi_produksi	<i>Integer</i>	1	

Sumber: Hasil Analisis Data (2018)

4. Spesifikasi Tabel *Production_Card*

Nama Tabel : *production_card*

Fungsi : Untuk menampilkan *production card* dari bagian produksi

Tipe : File Data Transaksi

Tabel V.14 Spesifikasi Tabel *Production Card*

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID <i>Production Card</i>	id_pc	<i>Varchar</i>	20	<i>Primary Key</i>
2	ID <i>Part</i>	id_part	<i>Varchar</i>	20	<i>Foreign key</i>
3	Nama <i>Part</i>	nama_part	<i>Varchar</i>	20	
4	Jenis <i>Part</i>	jenis	<i>Varchar</i>	20	
5	Tanggal	tanggal	<i>Date</i>		
6	Keterangan <i>Part</i>	keterangan	<i>Varchar</i>	100	
7	Jumlah <i>Part</i>	jumlah	<i>Integer</i>	11	
8	Validasi PPIC	Validasi_ppic	<i>Integer</i>	1	

Sumber: Hasil Analisis Data (2018)

5. Spesifikasi Tabel Rencana Produksi

Nama Tabel : rencana_produksi

Fungsi : Untuk menampilkan rencana produksi dari bagian PPIC

Tipe : File Data Master

Tabel V.15 Spesifikasi Tabel Rencana Produksi

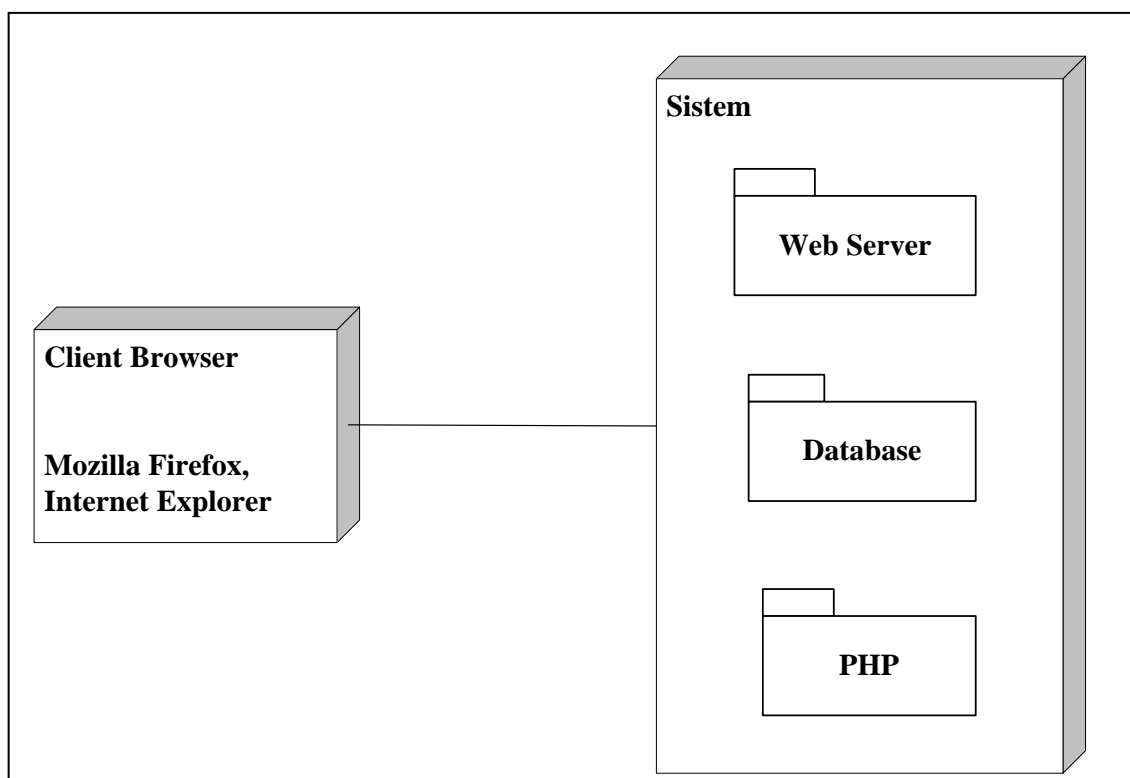
No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID Rencana	id_rencana	<i>Integer</i>	11	<i>Primary Key</i>
2	ID Part	id_part	<i>Varchar</i>	20	<i>Foreign Key</i>
3	Nama Part	nama_part	<i>Varchar</i>	20	

4	Jenis Part	Jenis	<i>Varchar</i>	20	
5	Tanggal	Tanggal	<i>Date</i>		
6	Keterangan Part	keterangan	<i>Varchar</i>	100	
7	Jumlah Part	jumlah	<i>Integer</i>	11	
8	Status	status	<i>Integer</i>	1	

Sumber: Hasil Analisis Data (2018)

5.4.3 *Deployment Diagram Yang Diusulkan*

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Seperti yang dapat dilihat pada gambar V.17 sebagai berikut:



Gambar V.17 *Deployment Diagram* yang Diusulkan

Sumber: Hasil Analisis Data (2018)

Deployment diagram di atas menggambarkan pembuatan aplikasi permintaan dan pengeluaran *part* berbasis *web* yang memanfaatkan PHP sebagai

desain sistemnya. Berikut adalah penjelasan *deployment diagram* aplikasi permintaan dan pengeluaran paket *part* berbasis *web*:

1. PHP

PHP adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah aplikasi berbasis web dan bisa digunakan bersamaan dengan HTML.

2. *Database*

Database adalah program komputer yang menyediakan layanan data lainnya ke komputer atau program komputer, seperti yang ditetapkan oleh model *client-server*. Istilah ini juga merujuk kepada sebuah komputer yang didedikasikan untuk menjalankan program *server database*. Beberapa sistem manajemen *database* yang sering menyediakan fungsi server adalah MySQL yang dipakai untuk *database server* aplikasi permintaan dan pengeluaran paket *part* berbasis *web*.

3. *Web Server*

Suatu program sekaligus mesin yang menjalankan program komputer yang mengerti protokol HTTP dan dapat menanggapi permintaan-permintaan dari *web browser*. Salah satu *web server* yaitu apache yang digunakan untuk membuat aplikasi permintaan dan pengeluaran paket *part* berbasis *web*.

4. *Web Browser*

Dalam Bahasa Indonesia *web browser* memiliki arti sebagai penjelajah *web*. Fungsi *web browser* itu sendiri adalah untuk menampilkan dan melakukan interaksi dengan dokumen-dokumen yang disediakan oleh *web server*.

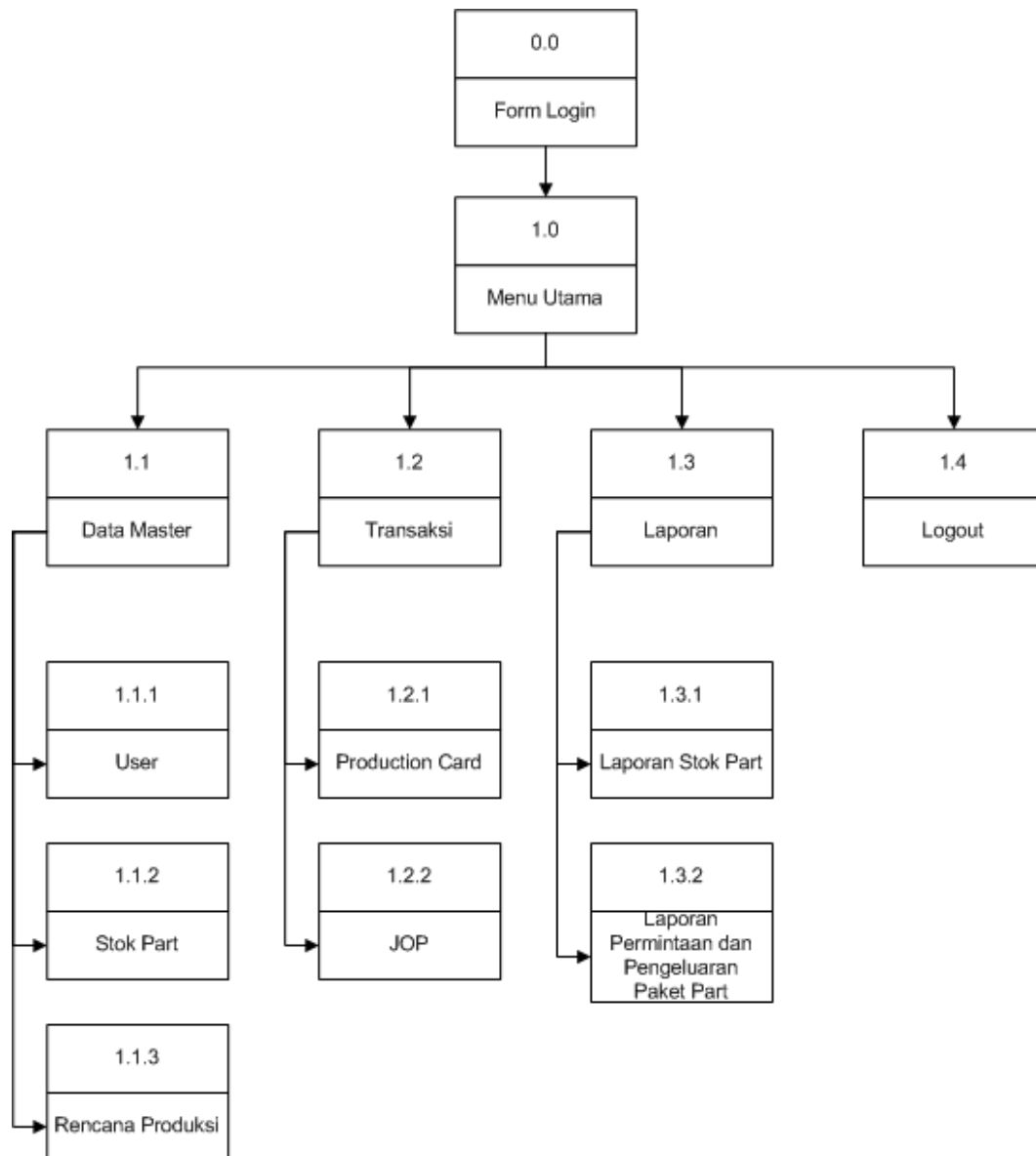
5.5 Perancangan Sistem

Perancangan sistem merupakan suatu kegiatan pengembangan serta perbaikan terhadap sebuah sistem yang berjalan. Pada tahap ini dilakukan upaya memperbaiki sistem atau membangun sistem dengan memanfaatkan teknologi

baru untuk mengurangi dan mengatasi berbagai permasalahan yang telah terjadi pada sistem yang lama sehingga dapat meningkatkan efektifitas dan efisiensi.

5.5.1 *Hierarchy plus Input-Process-Output (HIPO) Aplikasi Permintaan dan Pengeluaran Paket Part*

Pembuatan aplikasi dilakukan menggunakan bahasa pemrograman PHP dan MySQL sebagai perangkat lunak basis data. Berikut adalah struktur menu hirarki program yang digambarkan dalam *Hierarchy plus Input-Process-Output (HIPO)*:

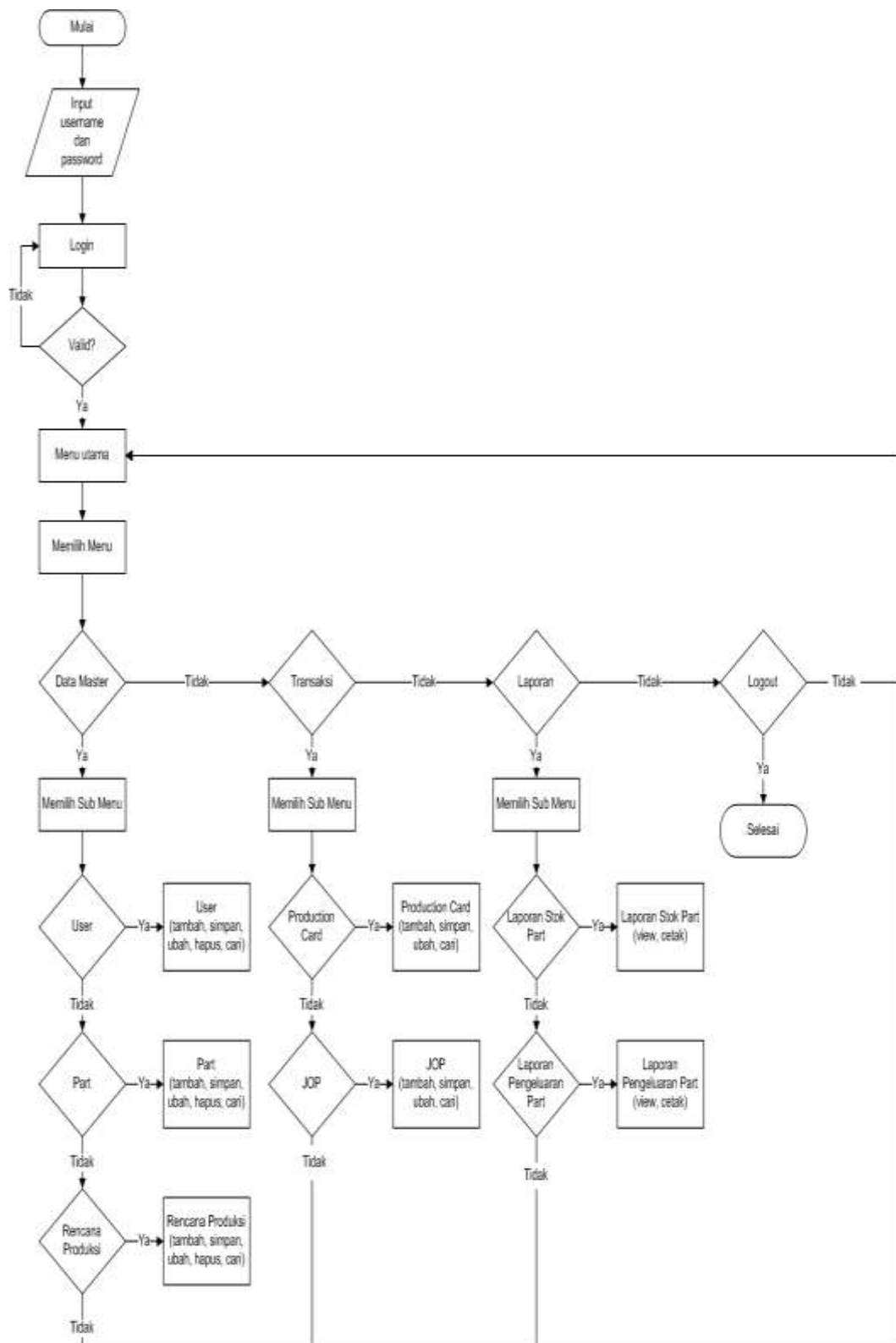


Gambar V.18 Hirarki Menu Aplikasi Permintaan dan Pengeluaran *Part*

Sumber: Hasil Analisis Data (2018)

5.5.2 *Flowchart* Aplikasi Permintaan dan Pengeluaran Paket *Part*

Flowchart aplikasi permintaan dan pengeluaran paket *part* dapat menggambarkan alur logika yang sebenarnya. Bagian ini juga memperjelas urutan prosedur sistem dan spesifikasi proses. Berikut adalah *flowchart* aplikasi permintaan dan pengeluaran paket *part*:



Gambar V.19 Flowchart Aplikasi Permintaan dan Pengeluaran Paket Part

Sumber: Hasil Analisis Data (2018)

5.5.3 Perancangan *Interface* Aplikasi Usulan

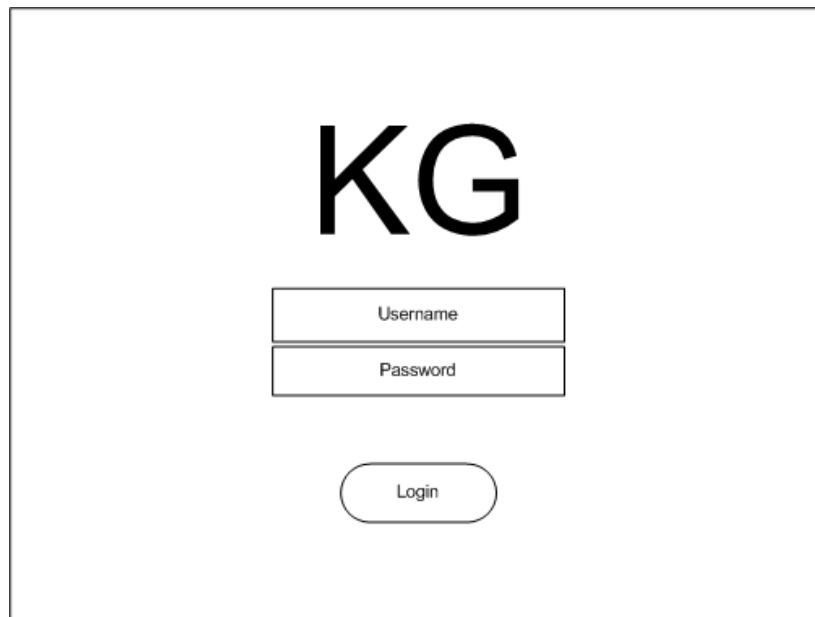
Perancangan *interface* merupakan tahapan untuk membuat tampilan atau desain dari sistem informasi permintaan dan pengeluaran paket *part*. Rancangan tampilan yang dibuat meliputi rancangan *input* dan rancangan *output* dari sistem yang akan dibuat.

1. Perancangan *Input*

Perancangan *input* merupakan gambaran *interface* tempat memasukan data-data ke dalam sistem, berikut ini *form-form* utama untuk *input* data:

a. Rancangan *Form Login*

Form Login digunakan untuk membedakan hak akses pengguna. Melalui *Form Login* ini pengguna yang boleh masuk sistem adalah pengguna yang mengetahui *username* dan *password* atau pengguna yang berwenang untuk menggunakan sistem yaitu .



Gambar V.20 Rancangan *Form Login*

Sumber: Hasil Analisis Data (2018)

b. Rancangan Tampilan Utama

Menu utama ini dirancang untuk memudahkan *user* dalam mengakses aplikasi dengan pilihan menu-menu yang sudah disediakan sesuai kebutuhan.

PT KENCANA GEMILANG	
Beranda	<h1>LOGO</h1>
Data Master	
User Part Rencana Produksi	
Transaksi	
Production Card JOP	
Laporan	
Laporan Stok Laporan Permintaan dan Pengeluaran Paket Part	
Keluar	

Gambar V.21 Rancangan Tampilan Utama

Sumber: Hasil Analisis Data (2018)

Rancangan tampilan utama ini dibuat untuk memasukkan dan melihat menu apa saja yang tersedia pada aplikasi ini, dimana ada beberapa menu yang dapat dipilih yaitu:

- 1) Beranda yang berfungsi untuk ke halaman utama.
- 2) Data Master terdapat 2 submenu yaitu sebagai berikut:
 - a) Data user berfungsi untuk memasukan, menghapus, dan mengubah data user yang dapat mengakses aplikasi permintaan dan pengeluaran *part*.
 - b) Data *part* berfungsi untuk memasukan, menghapus, dan mengubah data *part*.
- 3) Transaksi terdapat 2 submenu yaitu sebagai berikut:
 - a) *Production Card* berfungsi untuk melihat list permintaan dari bagian produksi.

- b) JOP berfungsi untuk mengeksekusi atau memproses permintaan.
 - 4) Laporan terdapat 2 submenu yaitu sebagai berikut:
 - a) Laporan Pengeluaran *Part* berfungsi untuk melihat jumlah *part* yang keluar dari gudang.
 - b) Laporan Stok *Part* berfungsi untuk melihat stok akhir *part* yang ada di gudang.
 - 5) Keluar yang berfungsi untuk keluar dari aplikasi.
 - c. Rancangan Form Data User
- Rancangan ini berfungsi untuk memasukkan data user. Dimana ada beberapa tombol yang dapat digunakan, yaitu sebagai berikut:
- 1) Tambah berfungsi untuk memasukan data user baru.
 - 2) Edit berfungsi untuk mengubah data yang ada.
 - 3) Hapus berfungsi untuk menghapus data yang telah disimpan.

PT KENCANA GEMILANG				
Beranda				
	Data User			
Data Master	Tambah Data	Edit Data	Hapus Data	
User Part Rencana Produksi		ID User	Nama User	Password
				Hak Akses
Transaksi				
Production Card JOP List Order				
Laporan				
Laporan Stok Part Laporan Pengeluaran Part				
Keluar				

Gambar V.22 Rancangan Submenu Data User

Sumber: Hasil Analisis Data (2018)

d. Rancangan Form Data *Part*

Rancangan ini berfungsi untuk memasukkan data *part*. Dimana ada beberapa tombol yang dapat digunakan, yaitu sebagai berikut:

- 1) Tambah berfungsi untuk memasukan data *part* baru.
- 2) Edit berfungsi untuk mengubah data yang ada.
- 3) Hapus berfungsi untuk menghapus data yang telah disimpan.

PT KENCANA GEMILANG						
Beranda						
	Data Part					
Data Master	Tambah Data Edit Data Hapus Data					
User Part Rencana Produksi		ID Part	Nama Part	Jenis	Keterangan	Jumlah
Transaksi						
Production Card JOP List Order						
Laporan						
Laporan Stok Part Laporan Pengeluaran Part						
Keluar						

Gambar V.23 Rancangan Tampilan Submenu Data *Part*

Sumber: Hasil Analisis Data (2018)

e. Rancangan Form Rencana Produksi

Rancangan ini berfungsi untuk memasukkan rencana produksi. Dimana ada beberapa tombol yang dapat digunakan, yaitu sebagai berikut:

- 1) Tambah berfungsi untuk memasukan data baru.
- 2) Edit berfungsi untuk mengubah data yang ada.
- 3) Hapus berfungsi untuk menghapus data yang telah disimpan.

PT KENCANA GEMILANG							
Beranda							
	Rencana Produksi						
Data Master	Tambah Data Edit Data Hapus Data						
User Part Rencana Produksi	ID Rencana	ID Part	Nama Part	Jenis	Keterangan	Tanggal	Jumlah
Transaksi							
Production Card JOP List Order							
Laporan							
Laporan Stok Part Laporan Pengeluaran Part							
Keluar							

Gambar V.24 Rancangan Tampilan Submenu Rencana Produksi

Sumber: Hasil Analisis Data (2018)

f. Rancangan Form *Production Card*

Rancangan ini berfungsi untuk memasukkan permintaan *part* dari bagian produksi. Dimana ada beberapa tombol yang dapat digunakan, yaitu sebagai berikut:

- 1) Tambah berfungsi untuk memasukan data baru.
- 2) Edit berfungsi untuk mengubah data yang ada.
- 3) Hapus berfungsi untuk menghapus data yang telah disimpan.

PT KENCANA GEMILANG																				
Beranda																				
	Production Card																			
Data Master	Tambah Data Edit Data Hapus Data																			
User Part Rencana Produksi	<table border="1"> <thead> <tr> <th>Id PC</th> <th>Id Part</th> <th>Nama Part</th> <th>Jenis</th> <th>Tanggal</th> <th>Keterangan</th> <th>Jumlah</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>						Id PC	Id Part	Nama Part	Jenis	Tanggal	Keterangan	Jumlah							
Id PC	Id Part	Nama Part	Jenis	Tanggal	Keterangan	Jumlah														
Transaksi																				
Production Card JOP List Order																				
Laporan																				
Laporan Stok Part Laporan Pengeluaran Part																				
Keluar																				

Gambar V.25 Rancangan *Submenu Production Card*

Sumber: Hasil Analisis Data (2018)

PT KENCANA GEMILANG																				
Beranda																				
	Production Card																			
Data Master	Tambah Data																			
User Part Rencana Produksi	<table border="1"> <thead> <tr> <th>Id PC</th> <th>Id Part</th> <th>Nama Part</th> <th>Jenis</th> <th>Tanggal</th> <th>Keterangan</th> <th>Jumlah</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>						Id PC	Id Part	Nama Part	Jenis	Tanggal	Keterangan	Jumlah							
Id PC	Id Part	Nama Part	Jenis	Tanggal	Keterangan	Jumlah														
Transaksi																				
Production Card JOP List Order																				
Laporan																				
Laporan Stok Part Laporan Pengeluaran Part																				
Keluar																				

FORM PRODUCTION CARD						
Id Production Card	<input type="text"/>					
Id Part	<input type="text"/>					ngan
Nama Part	<input type="text"/>					Jumlah
Jenis	<input type="text"/>					
Tanggal	<input type="text"/>					
Keterangan	<input type="text"/>					
Jumlah	<input type="text"/>					
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>						

Gambar V.26 Rancangan *Form Production Card*

Sumber: Hasil Analisis Data (2018)

g. Rancangan Form JOP

Rancangan ini berfungsi untuk *menggenerate production card* oleh bagian PPIC. Dimana ada beberapa tombol yang dapat digunakan, yaitu sebagai berikut:

- 1) Tambah berfungsi untuk memasukan data baru.
- 2) Edit berfungsi untuk mengubah data yang ada.
- 3) Hapus berfungsi untuk menghapus data yang telah disimpan.

PT KENCANA GEMILANG										
Beranda										
	JOP									
Data Master	Tambah Data Edit Data Hapus Data									
User Part Rencana Produksi	Id JOP	Id PC	Id Part	Nama Part	Jenis	Tanggal	Keterangan	Jumlah	PPIC	Produksi
Transaksi										
Production Card JOP List Order										
Laporan										
Laporan Stok Part Laporan Pengeluaran Part										
Keluar										

Gambar V.27 Rancangan Tampilan *Submenu JOP*

Sumber: Hasil Analisis Data (2018)

PT KENCANA GEMILANG																					
Beranda																					
	JOP																				
Data Master	Tambah Data																				
User Part Rencana Produksi	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">FORM JOP</th> </tr> </thead> <tbody> <tr> <td style="width: 20%;">Id JOP</td> <td><input type="text"/></td> </tr> <tr> <td>Id Production Card</td> <td><input type="text"/></td> </tr> <tr> <td>Id Part</td> <td><input type="text"/></td> </tr> <tr> <td>Nama Part</td> <td><input type="text"/></td> </tr> <tr> <td>Jenis</td> <td><input type="text"/></td> </tr> <tr> <td>Tanggal</td> <td><input type="text"/></td> </tr> <tr> <td>Keterangan</td> <td><input type="text"/></td> </tr> <tr> <td>Jumlah</td> <td><input type="text"/></td> </tr> <tr> <td colspan="2" style="text-align: center;"> <input type="button" value="Simpan"/> <input type="button" value="Batal"/> </td> </tr> </tbody> </table>	FORM JOP		Id JOP	<input type="text"/>	Id Production Card	<input type="text"/>	Id Part	<input type="text"/>	Nama Part	<input type="text"/>	Jenis	<input type="text"/>	Tanggal	<input type="text"/>	Keterangan	<input type="text"/>	Jumlah	<input type="text"/>	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	
FORM JOP																					
Id JOP	<input type="text"/>																				
Id Production Card	<input type="text"/>																				
Id Part	<input type="text"/>																				
Nama Part	<input type="text"/>																				
Jenis	<input type="text"/>																				
Tanggal	<input type="text"/>																				
Keterangan	<input type="text"/>																				
Jumlah	<input type="text"/>																				
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>																					
	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">Id JOP</th> <th style="width: 5%;">J</th> <th style="width: 5%;">ngan</th> <th style="width: 5%;">Jumlah</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	Id JOP	J	ngan	Jumlah																
Id JOP	J	ngan	Jumlah																		
Transaksi																					
Production Card JOP List Order																					
Laporan																					
Laporan Stok Part Laporan Pengeluaran Part																					
Keluar																					

Gambar V.28 Rancangan *Form JOP*

Sumber: Hasil Analisis Data (2018)

2. Perancangan *Output*

Perancangan *output* merupakan gambaran *interface* tempat mencetak data ke dalam kertas, berikut ini *form-form* untuk *output* data:

a. Rancangan Laporan Permintaan dan Pengeluaran *Part*

Rancangan laporan permintaan dan pengeluaran *part* dibuat untuk mengetahui jumlah *part* yang keluar setiap harinya. Dimana ada tombol cetak pada laporan tersebut, berfungsi untuk mencetak laporan yang sudah dibuat.

PT KENCANA GEMILANG													
Beranda													
	Laporan Pengeluaran Part												
Data Master													
User Part Rencana Produksi	Tanggal Awal <input type="text"/> Tanggal Akhir <input type="text"/> <input type="button" value="OK"/>												
	<table border="1"> <thead> <tr> <th>Id JOP</th> <th>Tanggal</th> <th>Id Part</th> <th>Nama Part</th> <th>Jenis</th> <th>Jumlah</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Id JOP	Tanggal	Id Part	Nama Part	Jenis	Jumlah						
Id JOP	Tanggal	Id Part	Nama Part	Jenis	Jumlah								
Transaksi													
Production Card JOP List Order													
	<input type="button" value="CETAK"/>												
Laporan													
Laporan Stok Part Laporan Pengeluaran Part													
Keluar													

Gambar V.29 Rancangan Laporan Permintaan dan Pengeluaran Paket *Part*

Sumber: Hasil Analisis Data (2018)

b. Rancangan Laporan Stok Barang

Rancangan laporan stok *part* dibuat untuk mengetahui jumlah stok *part* yang ada di gudang. Dimana ada tombol cetak yang berfungsi untuk mencetak laporan yang sudah dibuat. Dan tombol cari yang berfungsi untuk mencari data yang akan dicetak.

PT KENCANA GEMILANG											
Beranda											
	Laporan Stok Part										
Data Master											
User Part Rencana Produksi	Jenis <input type="text"/> <input type="button" value="Cari"/>										
Transaksi											
Production Card JOP List Order	<table border="1"> <thead> <tr> <th>ID Part</th> <th>Nama Part</th> <th>Jenis</th> <th>Keterangan</th> <th>Jumlah</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	ID Part	Nama Part	Jenis	Keterangan	Jumlah					
ID Part	Nama Part	Jenis	Keterangan	Jumlah							
Laporan	<input type="button" value="CETAK"/>										
Laporan Stok Part Laporan Pengeluaran Part											
Keluar											

Gambar V.30 Rancangan Laporan Stok *Part*

Sumber: Hasil Analisis Data (2018)

5.6 Analisis Kebutuhan *Software* Dan *Hardware*

Kebutuhan *software* dan *hardware* yang harus dipenuhi agar aplikasi dapat berjalan dengan lancar adalah sebagai berikut:

1. *Hardware* yang dibutuhkan:
 - a. *Processor* : *Pentium 2,08GHz*
 - b. Kebutuhan memori utama: 2 GB
 - c. *Harddisk*: 20 GB
2. *Software* yang dibutuhkan:
 - a. Sistem operasi *Windows 7*
 - b. XAMPP versi 3.0.12
 - c. *Browser* Mozilla Firefox dan Google Chrome

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil penelitian dengan mengumpulkan data, mengolah data, dan melakukan perancangan, maka dapat disimpulkan bahwa:

1. Dengan adanya sistem permintaan dan pengeluaran paket part dapat meminimalisir tingkat kesalahan dalam pembuatan dokumen yang digunakan dalam proses permintaan paket part dari bagian produksi ke PPIC.
2. Dengan adanya aplikasi ini data yang ada disimpan ke dalam sistem sehingga dapat mengurangi penumpukan dokumen dan mempermudah dalam pencarian data.

6.2 Saran

Beberapa saran yang diharapkan agar aplikasi permintaan dan pengeluaran paket part ini menjadi lebih baik adalah sebagai berikut:

1. Diperlukan pelatihan khusus terhadap pengguna aplikasi ini.
2. Adanya pemeliharaan terhadap sistem yang telah dibuat agar sistem tetap terjaga dengan baik, dengan cara melakukan perbaikan pada sistem apabila terjadi kesalahan atau error pada aplikasi tersebut.
3. Untuk pengembangan sistem selanjutnya diharapkan dapat membahas rancangan interface yang lebih menarik agar user dapat lebih mudah menggunakannya.

DAFTAR PUSTAKA

- Assauri, Sofjan., 2016, *Manajemen Operasi Produksi*, Edisi ke-5, PT. Raja Grafindo Persada, Jakarta.
- Gaspersz, Vincent, 2012, *All In One: Production and Inventori Management*, Edisi ke-8, Vinchristo Publication, Bogor.
- Griffin, Ricky W., 2004, *Manajemen*, Edisi ke-5, Jilid 2, Terjemahan Gina Gania, Erlangga, Jakarta.
- Handoko, T. Hani, 2011, *Dasar-dasar Manajemen Produksi dan Operasi*, BPFYogyakarta, Yogyakarta.
- Hartono, Bambang, 2013, *Sistem Informasi Manajemen Berbasis Komputer*, Rineka Cipta, Jakarta.
- Jogiyanto, HM, 2005, *Analisis & Desain Sistem Informasi Pendekatan. Terstruktur Teori dan Praktik Aplikasi Bisnis*, Andi, Yogyakarta.
- Kadir, Abdul, 2003, *Pengenalan Sistem Infromasi*, Andi, Yogyakarta.
- Mc leod, Raymond; Schell, George P, 2008, *Sistem Informasi Manajemen*, 10th ed., Salemba Empat, Jakarta.
- Munawar, 2005, *Pemodelan Visual menggunakan UML*, Graha Ilmu, Yogyakarta.
- Mustakini, Hartono, Jogiyanto, 2009, *Sistem Informasi Teknologi*, Andi Offset, Yogyakarta.
- Nafarin, 2007, *Penganggaran Perusahaan*, Penerbit Salemba Empat, Jakarta.
- Nasution, 2008, *Perencanaan dan Pengendalian Proses Produksi*, PT Gramedia Pustaka Utama, Jakarta.
- Purbadian, Yenda, 2015, *Aplikasi Penjualan Web Base Dengan PHP untuk Panduan Skripsi*, CV Asfa Solution, Cirebon.
- Rama, Dasaratha V dan Jones, Frederick L., 2008, *Sistem Informasi Akuntansi*, Salemba Empat, Jakarta.
- Riyanto, 2015, *Membuat Aplikasi Apotek Integrasi Barcode Scanner Dengan PHP & MYSQL*, Penerbit Gava Media, Yogyakarta.

- Rosa, A.S. & Shalahuddin, M, 2011, *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*, Cetakan Pertama, Modula, Bandung.
- Sutabri, Tata, 2004, *Analisa Sistem Informasi*, ANDI Yogyakarta, Yogyakarta.
- Sutaji, Deni, 2012, *Sistem Inventory Mini Market dengan PHP & JQuery*, Lokomedia, Yogyakarta.
- Sutarman, 2009, *Pengantar Teknologi Informasi*, Bumi Aksara, Jakarta.
- Tim EMS, 2012, *Web Programming for Beginners*, PT Elex Media Komputindo, Jakarta.
- Wahana Komputer, 2010, *Panduan belajar MySQL Database Server*, Media Kita, Jakarta.

LAMPIRAN

1. Menu Login

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class login extends CI_Controller {
    public $session;
    public function __construct(){
        parent::__construct();
        $this->load->library('session');
        $this->load->model('Crud_m');
    }
    public function index()
    {
        $this->form_validation-
>set_rules('nama_user','nama','required|alpha_numeric');
        $this->form_validation-
>set_rules('password','Password','required|alpha_numeric');
        if($this->form_validation->run()===FALSE)
        {
            $this->load->view('Login/login_v');
        } ELSE {
            $this->load->model('Crud_m');
            $valid_user = $this->Crud_m->check_credential();
            if($valid_user == FALSE)
            {
                $this->session->set_flashdata('error', 'Nama Pengguna /
Kata Sandi Anda Salah');
                redirect('Login');
            } else
            {
                $this->session-
>set_userdata('nama_user',$valid_user->nama_user);
                $this->session-
>set_userdata('password',$valid_user->password);
                $this->session-
>set_userdata('hak_akses',$valid_user->hak_akses);
                switch($valid_user->hak_akses){
                    case 1 : //admin
                        redirect('Jop');
                        break;
                    case 2 : // member
                        redirect('rencanaProduksi');
                        break;
                    case 3 : // member
                        redirect('listPC');
```

```

        break;
        default : break;
        print_r($_POST);
    }
    exit();
}
}
}
// Berfungsi untuk menghapus session atau logout
function logout() {
    session_destroy();
    redirect('Login');
}
}
?>

```

2. Menu Utama

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class beranda extends CI_Controller {
    public function __construct(){
        parent::__construct();
    }
    public function index()
    {
        $this->load->view('header/header_v');
        $this->load->view('beranda/beranda_v');
    }
    public function produksi()
    {
        $this->load->view('header/header_produksi');
        $this->load->view('beranda/beranda_produksi');
    }
    public function gudang()
    {
        $this->load->view('header/header_gudang');
        $this->load->view('beranda/beranda_gudang');
    }
}
?>

```

```

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
</head>

```

```

<body>
<div id="wrapper">
<div id="page-wrapper">
<div class="graphs">
<div class="col_3">
<div class="col-md-10 widget widget1">
<div class="r3_counter_box">
<h3><i class="fa fa-user"></i>SELAMAT DATANG DI SISTEM INFORMASI
PERMINTAAN DAN PENGELUARAN PART</h3>
<div class="panel panel-success">
<table class="table table-responsive">
<tr>
<td>Status</td>
<td>:</td>
<td>PPIC</td>
</tr>
</table>
</div>
</div>
</div>
<div class="clearfix"></div>
</div>
<div class="copy">
<p>&copy; 2017 <a href="#" target="_blank">PT KENCANA
GEMILANG</a></p>
</div>
</div>
</div>
</div>
<script src="<?php echo base_url('assets/js/bootstrap.min.js');?>"></script>
</body>
</html>

```

3. Menu Data Master

```

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<script type="text/javascript" language="javascript"
src="//code.jquery.com/jquery-1.10.2.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/1.10.4/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.js"></script>

```

```

<link rel="stylesheet" type="text/css" href="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.css">
</head>
<body>
<div id="wrapper">
<div id="page-wrapper">
<div class="graphs">
<div class="col_3">
<div class="panel panel-success">
<div class="panel-heading">
<h3>Data Users</h3>
</div>
<div class="panel-body" style="background-color: honeydew;">
<div class="table-responsive">
<table id="myTable" class="table table-bordered table-hover" >
<thead class="success">
<tr>
<th>Id User</th>
<th>Nama User</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<?php $no=0;
        foreach($menunya as $row){
            $no++; ?>
<tr>
<td><?=$row['id_user']?></td>
<td><?=$row['nama_user']?></td>
<td><?=$row['password']; ?></td>
</tr>
<?php $no++;
        } ?>
</tbody>
</table>
<script type="text/javascript">
        $(document).ready(function() {
            $('#myTable').DataTable();
        });
</script>
</div>
</div>
</div>
</div>
</div>
<div class="clearfix"></div>
<div class="copy" >

```

```
<p>&copy; 2017 <a href="" target="_blank">PT KENCANA  
GEMILANG</a></p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script type="text/javascript" src="<?php  
base_url('assets/js/bootstrap.min.js');?>"></script>
```

```
</body>
```

```
</html>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title></title>
```

```
<script type="text/javascript" language="javascript"  
src="//code.jquery.com/jquery-1.10.2.min.js"></script>
```

```
<script type="text/javascript" language="javascript"  
src="//cdn.datatables.net/1.10.4/js/jquery.dataTables.min.js"></script>
```

```
<script type="text/javascript" language="javascript"  
src="//cdn.datatables.net/plug-  
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.js"></script>
```

```
<link rel="stylesheet" type="text/css" href="//cdn.datatables.net/plug-  
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.css">
```

```
</head>
```

```
<body>
```

```
<div id="wrapper">
```

```
<div id="page-wrapper">
```

```
<div class="graphs">
```

```
<div class="col_3">
```

```
<div class="panel panel-success">
```

```
<div class="panel-heading">
```

```
<h3>Data Part List</h3>
```

```
</div>
```

```
<div class="panel-body" style="background-color: honeydew;">
```

```
<div class="table-responsive">
```

```
<table id="myTable" class="table table-bordered table-hover" >
```

```
<thead class="success">
```

```
<tr>
```

```
<th>Id Part</th>
```

```
<th>Nama Part</th>
```

```
<th>Jenis</th>
```

```
<th>Jumlah</th>
```

```
</tr>
```

```

</thead>
<tbody>
<?php $no=0;
        foreach($menunya as $row){
            $no++; ?>
<tr>
<td><?=$row['id_part']?></td>
<td><?=$row['nama_part']?></td>
<td><?=$row['jenis']; ?></td>
<td><?=$row['jumlah']; ?></td>
</tr>
<?php $no++;
            } ?>
</tbody>
</table>
<script type="text/javascript">
    $(document).ready(function() {
        $('#myTable').DataTable();
    });
</script>
</div>
</div>
</div>
</div>
</div>
<div class="clearfix"></div>
<div class="copy" >
<p>&copy; 2017 <a href="" target="_blank">PT KENCANA
GEMILANG</a></p>
</div>
</div>
</div>
</div>
</div>
<script type="text/javascript" src="<?php
base_url('assets/js/bootstrap.min.js');?>"></script>
<script type="text/javascript">
    $(document).ready(function() {
        $('#example').DataTable( {
            dom: 'Bftrtp',
            buttons: [
                'copy', 'csv', 'excel', 'pdf', 'print'
            ]
        } );
    } );
</script>
</body>

```

```

</html>

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<script type="text/javascript" language="javascript"
src="//code.jquery.com/jquery-1.10.2.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/1.10.4/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.js"></script>
<link rel="stylesheet" type="text/css" href="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.css">
</head>
<body>
<div id="wrapper">
<div id="page-wrapper">
<div class="graphs">
<div class="col_3">
<div class="panel panel-success">
<div class="panel-heading">
<h3>JOP</h3>
</div>
<div class="panel-body" style="background-color: honeydew;">
<div class="table-responsive">
<table id="myTable" class="table table-bordered table-hover" >
<thead class="success">
<tr>
<th>No.</th>
<th>Id Barang</th>
<th>Nama Barang</th>
<th>Jenis</th>
<th>Tanggal</th>
<th>Jumlah</th>
<th>Keterangan</th>
<th>Validasi</th>
</tr>
</thead>
<tbody>
<?php $no=0;
foreach($menunya->result() as $row){
$no++; ?>
</tr>

```

```

<td><?php echo $no; ?></td>
<td name="id_pc"><?php echo $row->id_pc; ?></td>
<td name="id_part"><?php echo $row->id_part; ?></td>
<td name="nama_part"><?php echo $row->nama_part; ?></td>
<td name="jenis"><?php echo $row->jenis; ?></td>
<td name="tanggal"><?php echo $row->tanggal; ?></td>
<td name="jumlah"><?php echo $row->jumlah; ?></td>
<td name="keterangan"><?php echo $row->keterangan; ?></td>
<td><a href="<?php echo site_url('Ppic/aksi_insert_jop'); ?>">Ya</a>
<a href="<?php echo site_url('Ppic/aksi_insert_jop'); ?>">Tidak</a>
</td>
</tr>
<?php }?>
</tbody>
</table>
<script type="text/javascript">
    $(document).ready(function() {
        $('#myTable').DataTable();
    });
</script>
</div>
</div>
</div>
</div>
</div>
<div class="clearfix"></div>
<div class="copy" >
<p>&copy; 2017 <a href="" target="_blank">PT KENCANA
GEMILANG</a></p>
</div>
</div>
</div>
</div>
</div>
</div>
<script type="text/javascript" src="<?php
base_url('assets/js/bootstrap.min.js');?>"></script>
</body>
</html>

```

4. Menu Data Transaksi

```

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<script type="text/javascript" language="javascript"
src="//code.jquery.com/jquery-1.10.2.min.js"></script>

```

```

<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/1.10.4/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.js"></script>
<link rel="stylesheet" type="text/css" href="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.css">
</head>
<body>
<div id="wrapper">
<div id="page-wrapper">
<div class="graphs">
<div class="col_3">
<div class="panel panel-success">
<div class="panel-heading">
<h3><i class="fa fa-plus-circle"></i>Production Card</h3>
</div>
<div class="panel-body" style="background-color: honeydew;">
<?php
                $menuT = $menuX->result();
                //print_r($menuT);
                //echo $menuT[0]->id; //karena hasilnya adalah array dan cuma 1
--> bisa juga di foreach
                ?>
<form class="form-horizontal" action="<?php echo
site_url('rencanaProduksi/aksi_insert');?>" method="post">
<div class="form-group">
<label for="focusedinput" class="col-sm-2 control-label">ID Rencana <label
style="color:red;">*</label></label>
<div class="col-sm-8">
<input type="text" class="form-control" id="focusedinput" required=""
name="id_rencana" value="<?php echo $menuT[0]->id_rencana ?>">
</div>
</div>
<div class="form-group">
<label class="col-md-2 control-label">ID Part <label
style="color:red;">*</label></label>
<div class="col-sm-8">
<input type="text" class="form-control" id="focusedinput" required=""
name="id_part" value="<?php echo $menuT[0]->id_part ?>">
</div>
</div>
<div class="form-group">
<label for="focusedinput" class="col-sm-2 control-label">Nama Part <label
style="color:red;">*</label></label>
<div class="col-sm-8">

```

```

<input type="text" class="form-control" id="focusedinput" required=""
name="nama_part" value="<?php echo $menuT[0]->nama_part ?>">
</div>
</div>
<div class="form-group">
<label for="focusedinput" class="col-sm-2 control-label">Jenis <label
style="color:red;">*</label></label>
<div class="col-sm-8">
<input type="text" class="form-control" id="focusedinput" required=""
name="jenis" value="<?php echo $menuT[0]->jenis ?>">
</div>
</div>
<div class="form-group">
<label for="focusedinput" class="col-sm-2 control-label">Tanggal <label
style="color:red;">*</label></label>
<div class="col-sm-8">
<input type="date" class="form-control" id="focusedinput" required=""
name="tanggal" value="<?php echo $menuT[0]->tanggal ?>">
</div>
</div>
<div class="form-group">
<label for="focusedinput" class="col-sm-2 control-label">Jumlah <label
style="color:red;">*</label></label>
<div class="col-sm-8">
<input type="text" class="form-control" id="focusedinput" required=""
name="jumlah" value="<?php echo $menuT[0]->jumlah ?>">
</div>
</div>
<div class="form-group">
<label for="focusedinput" class="col-sm-2 control-label">Keterangan <label
style="color:red;">*</label></label>
<div class="col-sm-8">
<input type="text" class="form-control" id="focusedinput" required="readonly"
name="keterangan" value="<?php echo $menuT[0]->keterangan ?>">
</div>
</div>
<div class="row">
<div class="col-sm-8 col-sm-offset-2">
<button type="submit" class="btn-info btn">Simpan</button>
<button type="reset" class="btn-primary btn">Reset</button>
<a href="<?php echo site_url('listPC/listPC_tambah');?>" class="btn btn-danger
">Batal</a>
</div>
</div>
</form>
</div>

```

```

</div>
<div class="clearfix"></div>
</div>
<div class="copy" >
<p>&copy; 2017 <a href="#" target="_blank">PT KENCANA
GEMILANG</a></p>
</div>
</div>
</div>
</div>
</div>
<script type="text/javascript">
function isNumberKey(evt){
var charCode = (evt.which) ? evt.which : event.keyCode
if (charCode > 31 && (charCode < 48 || charCode > 57)){
$('#thumbnail_alert').html('interger only');
return false;
}
$('#thumbnail_alert').html("");
$('#durasi').keyup(function(){
var lent = $(this).val();
if(lent.length == 2){
$('#input[name=thumbnail_time]').val(lent+':');
}
});
return true;
}
</script>
<script type="text/javascript" src="<?php echo base_url('assets/js/jquery-
1.11.2.js');?>"></script>
<script src="<?php echo base_url('assets/js/bootstrap.min.js');?>"></script>
</body>
</html>

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<script type="text/javascript" language="javascript"
src="//code.jquery.com/jquery-1.10.2.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/1.10.4/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.js"></script>

```

```

<link rel="stylesheet" type="text/css" href="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.css">
</head>
<body>
<div id="wrapper">
<div id="page-wrapper">
<div class="graphs">
<div class="col_3">
<div class="panel panel-success">
<div class="panel-heading">
<h3>Data List Product Card</h3>
</div>
<div class="panel-body" style="background-color: honeydew;">
<div class="table-responsive">
<table id="myTable" class="table table-bordered table-hover" >
<thead class="success">
<tr>
<th>Id Production Card</th>
<th>Id Part</th>
<th>Nama Part</th>
<th>Jenis</th>
<th>Tanggal</th>
<th>Jumlah</th>
<th>Keterangan</th>
<th>Aksi</th>
</tr>
</thead>
<tbody>
<?php $no=0;
        foreach($menunya as $row){
            $no++; ?>
<tr>
<td name="id_pc" value="id_pc"><?=$row['id_pc']?></td>
<td name="id_part" value="id_part"><?=$row['id_part']?></td>
<td name="nama_part" value="nama_part"><?=$row['nama_part']; ?></td>
<td name="jenis" value="jenis"><?=$row['jenis']?></td>
<td name="tanggal" value="tanggal"><?=$row['tanggal']?></td>
<td name="jumlah" value="jumlah"><?=$row['jumlah']?></td>
<td name="keterangan" value="keterangan"><?=$row['keterangan']; ?></td>
<td><a href="<?php echo site_url('jop/update_ya/'.$row['id_pc']); ?>" class="btn-
info btn">YA</a>
<a href="<?php echo site_url('jop/update_tidak/'.$row['id_pc']); ?>" class="btn
btn-danger ">TIDAK</a>
</td>
</tr>
<?php $no++;

```

```

        } ?>
</tbody>
</table>
<script type="text/javascript">
    $(document).ready(function() {
        $('#myTable').DataTable();
    });
</script>
</div>
</div>
</div>
</div>
</div>
<div class="clearfix"></div>
<div class="copy" >
<p>&copy; 2017 <a href="" " target="_blank">PT KENCANA
GEMILANG</a></p>
</div>
</div>
</div>
</div>
</div>
</div>
<script type="text/javascript" src="<?php echo
base_url('assets/js/bootstrap.min.js');?>"></script>
</body>
</html>

```

5. Menu Laporan

```

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<script type="text/javascript" language="javascript"
src="//code.jquery.com/jquery-1.10.2.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/1.10.4/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.js"></script>

<link rel="stylesheet" type="text/css" href="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.css">
</head>
<body>
<div id="wrapper">

```

```

<div id="page-wrapper">
<div class="graphs">
<div class="col_3">
<div class="panel panel-success">
<div class="panel-heading">
<h3>Data Part List</h3>
</div>
<div class="panel-body" style="background-color: honeydew;">
<div class="table-responsive">
<table id="myTable" class="table table-bordered table-hover" >
<thead class="success">
<tr>
<th>Id Part</th>
<th>Nama Part</th>
<th>Jenis</th>
<th>Jumlah</th>
</tr>
</thead>
<tbody>
<?php $no=0;
        foreach($menunya as $row){
            $no++; ?>
<tr>
<td><?=$row['id_part']?></td>
<td><?=$row['nama_part']?></td>
<td><?=$row['jenis']; ?></td>
<td><?=$row['jumlah']; ?></td>
</tr>
<?php $no++;
        } ?>
</tbody>
</table>
<script type="text/javascript">
        $(document).ready(function() {
            $('#myTable').DataTable();
        });
</script>
</div>
</div>
</div>
</div>
<div class="clearfix"></div>
<div class="copy" >
<p>&copy; 2017 <a href="" " target="_blank">PT KENCANA
GEMILANG</a></p>
</div>

```

```

</div>
</div>
</div>
</div>
<script type="text/javascript" src="<?php echo
base_url('assets/js/bootstrap.min.js');?>"></script>
<script type="text/javascript">
$(document).ready(function() {
$('#example').DataTable( {
dom: 'Bfrtip',
buttons: [
'copy', 'csv', 'excel', 'pdf', 'print'
]
} );
});
</script>
</body>
</html>

```

```

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<script type="text/javascript" language="javascript"
src="//code.jquery.com/jquery-1.10.2.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/1.10.4/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" language="javascript"
src="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.js"></script>
<link rel="stylesheet" type="text/css" href="//cdn.datatables.net/plug-
ins/9dcbeed42ad/integration/bootstrap/3/dataTables.bootstrap.css">
</head>
<body>
<div id="wrapper">
<div id="page-wrapper">
<div class="graphs">
<div class="col_3">
<div class="panel panel-success">
<div class="panel-heading">
<h3>Data List JOP</h3>
</div>
<div class="panel-body" style="background-color: honeydew;">
<div class="table-responsive">
<table id="myTable" class="table table-bordered table-hover" >

```

```

<thead class="success">
<tr>
<th>Id JOP</th>
<th>Id Production Card</th>
<th>Id Part</th>
<th>Nama Part</th>
<th>Jenis</th>
<th>Tanggal</th>
<th>Jumlah</th>
<th>Keterangan</th>
<th>Validasi PPIC</th>
<th>Validasi Produksi</th>
</tr>
</thead>
<tbody>
<?php $no=0;
        foreach($jop as $row){
            $no++; ?>
<tr>
<td name="id_jop" value="id_pc"><?=$row['id_jop']?></td>
<td name="id_pc" value="id_pc"><?=$row['id_pc']?></td>
<td name="id_part" value="id_part"><?=$row['id_part']?></td>
<td name="nama_part" value="nama_part"><?=$row['nama_part']; ?></td>
<td name="jenis" value="jenis"><?=$row['jenis']?></td>
<td name="tanggal" value="tanggal"><?=$row['tanggal']?></td>
<td name="jumlah" value="jumlah"><?=$row['jumlah']?></td>
<td name="keterangan" value="keterangan"><?=$row['keterangan']; ?></td>
<td name="keterangan" value="keterangan"><?=$row['validasi_ppic']; ?></td>
<td name="keterangan" value="keterangan"><?=$row['validasi_produksi'];
?></td>
<td>
</td>
</tr>
<?php $no++;
        } ?>
</tbody>
</table>
<script type="text/javascript">
        $(document).ready(function() {
            $('#myTable').DataTable();
        });
</script>
</div>
</div>
</div>
</div>

```

```
<div class="clearfix"></div>
<div class="copy" >
<p>&copy; 2017 <a href="" target="_blank">PT KENCANA
GEMILANG</a></p>
</div>
</div>
</div>
</div>
</div>
<script type="text/javascript" src="<?php echo
base_url('assets/js/bootstrap.min.js');?>"></script> </body> </html>
```