No. Dok. 1345
Cpy : 1

D3
629.280 285
Has
R.

RANCANG BANGUN SISTEM INFORMASI PENGAWASAN PART KE LINE PRODUKSI MENGGUNAKAN PHP 5.6.12 DAN MYSQL 5.6.26 PADA PT KRAMA YUDHA RATU MOTOR

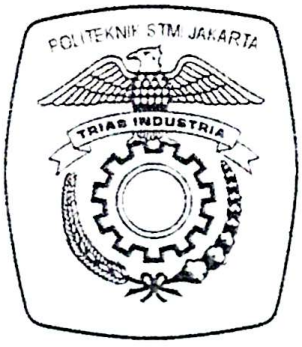
TUGAS AKHIR

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

OLEH

MARTINA HASIAN

1310036



POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2017

BADAN BUKU PERPUSTAKAAN
Tgl. Terima 20/01/19
No. Induk Buku 50/511/SB-TA/19

POLITEKNIK STMI JAKARTA

KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir	: RANCANG BANGUN SISTEM PENGAWASAN PART KE LINE PRODUKSI MENGGUNAKAN PHP 5.6.12 DAN MYSQL 5.6.26 PADA PT KRAMA YUDHA RATU MOTOR
Disusun Oleh	:
Nama	: Martina Hasian
Nim	: 1310036
Program Studi	: Sistem Informasi Industri Otomotif

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta Kementerian Republik Indonesia pada hari Jumat tanggal 22 November 2017.

Jakarta, 27 November 2017

Dosen Pembimbing	Ketua Penguji
Drs. Jacob Saragih, MM	**Dedy Trisanto, S. Kom, MMSI**
NIP. 195404281986031002	**NIP. 197805052005021002**

Dosen Penguji	Dosen Penguji
Dr. Ridzky Kramanandita, S. Kom, MT	**Ahmad Juniar, S. Kom, MT**
NIP. 197403022002121001	**NIP. 197906052006041002**

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

TANDA PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir	:	**RANCANG BANGUN SISTEM PENGAWASAN PART KE LINE PRODUKSI MENGGUNAKAN PHP 5.6.12 DAN MYSQL 5.6.26 PADA PT KRAMA YUDHA RATU MOTOR**

Disusun Oleh

Nama	:	Martina Hasian
NIM	:	1310036
Program Studi	:	Sistem Informasi Industri Otomotif
Tanggal Seminar	:	25 September 2017
Tanggal Sidang	:	22 November 2017
Tanggal Lulus	:	22 November 2017

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

Jakarta, 22 November 2017

Dosen Pembimbing



Drs. Jacob Saragih, MM

NIP : 195404281986031002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

TANDA PERSETUJUAN ASISTEN PEMBIMBING

Judul Tugas Akhir : **RANCANG BANGUN SISTEM PENGAWASAN PART KE LINE PRODUKSI MENGGUNAKAN PHP 5.6.12 DAN MYSQL 5.6.26 PADA PT KRAMA YUDHA RATU MOTOR**

Disusun Oleh

Nama	:	Martina Hasian
NIM	:	1310036
Program Studi	:	Sistem Informasi Industri Otomotif
Tanggal Seminar	:	25 September 2017
Tanggal Sidang	:	22 November 2017
Tanggal Lulus	:	22 November 2017

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

Jakarta, 22 November 2017

Asisten Pembimbing



Fifi Lailasari Hadianastuti, S.Kom, M. Kes

NIP: 197310162005022001

Kementerian Perindustrian
REPUBLIK INDONESIA

POLITEKNIK STMI JAKARTA
d.h. SEKOLAH TINGGI MANAJEMEN INDUSTRI
Jl. Letjen Suprapto No. 26 Cempaka Putih, Jakarta 10510
Telp: (021) 42886064 Fax: (021) 42888206
www.stmi.ac.id

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Martina Hasian
NIM : 1310036
Judul Tugas Akhir : Rancang Bangun Sistem Informasi Pengawasan Part ke Line Produksi pada PT Krama Yudha Ratu Motor dengan menggunakan PHP 5.6.12 dan MySql 5.6.26
Pembimbing : Drs. Jacob Saragih, MM
Asisten Pembimbing : Fifi Lailasari Hadianastuti, S.Kom, M. Kes

Tanggal	BAB	Keterangan	Paraf
24/05/2017	BAB I	Bab I Latar Belakang Masalah, Tujuan Penelitian	
05/06/2017	BAB II	Bab II Landasan Teori	
12/06/2017	BAB III	Bab III Metodologi Penelitian, kerangka penelitian	
19/06/2017	BAB III	Bab III Koreksi	
10/07/2017	BAB IV	Bab IV Pengumpulan dan Pengolahan Data	
17/07/2017	BAB IV	Bab IV Koreksi	
24/07/2017	BAB V	Bab V Membuat Prototype	
31/07/2017	BAB V	Bab V Koreksi	
07/08/2017	BAB I – V	Bab I - Bab V Koreksi	
10/09/2017	BAB I - V	Bab I - V	

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif

Drs. Jacob Saragih, MM
NIP : 195404281986031002

Pembimbing

Drs. Jacob Saragih, MM
NIP : 195404281986031002

SAI GLOBAL CERTIFICATION SERVICES Pty Ltd Registration ISO 9001:2008 No Reg QEC 264727

Kementerian Perindustrian
REPUBLIK INDONESIA

POLITEKNIK STMI JAKARTA
d.h. SEKOLAH TINGGI MANAJEMEN INDUSTRI
Jl. Letjen Suprapto No. 26 Cempaka Putih, Jakarta 10510
Telp: (021) 42886064 Fax: (021) 42888206
www.stmi.ac.id

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Martina Hasian
NIM : 1310036
Judul Tugas Akhir : Rancang Bangun Sistem Informasi Pengawasan Part ke Line Produksi pada PT Krama Yudha Ratu Motor dengan menggunakan PHP 5.6.12 dan MySql 5.6.26
Pembimbing : Drs. Jacob Saragih, MM
Asisten Pembimbing : Fifi Lailasari Hadianastuti, S.Kom, M. Kes

Tanggal	BAB	Keterangan	Paraf
24/07/2017	BAB I	Bab I Latar Belakang Masalah, Tujuan Penelitian	
2/08/2017	BAB I	Bab I Perbaikan Latar Belakang Masalah	
4/08/2017	BAB I - III	Bab I Latar belakang masalah, bab III kerangka penelitian	
7/08/2017	BAB I	Bab I Tujuan penelitian	
10/08/2017	BAB IV	Bab IV Pengumpulan dan Pengolahan Data	
14/08/2017	BAB IV - V	Bab IV – V Koreksi	
24/08/2017	BAB IV - V	Bab Perbaikan Usecase	
31/8/2017	BAB V	Bab V Perbaikan Flowmap dan Usecase usulan	
04/09/2017	BAB I – V	Bab I - Bab V Koreksi	
05/9/2017	BAB I - V	Bab I - V	

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif

Drs. Jacob Saragih, MM
NIP : 195404281986031002

Asisten Pembimbing

Fifi Lailasari Hadianastuti, S.Kom, M. Kes
NIP: 197310162005022001

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Martina Hasian

Nim : 1310036

Program Studi : Sistem Informasi

Berstatus sebagai mahasiswa Program Studi Sistem Informasi di Politeknik STMI Jakarta Kementerian Perindustrian RI. Dengan ini menyatakan bahwa hasil karya Tugas Akhir yang saya buat dengan judul:

"RANCANG BANGUN SISTEM INFORMASI PENGAWASAN PART KE LINE PRODUKSI PADA PT KRAMA YUDHA RATU MOTOR DENGAN MENGGUNAKAN PHP 5.6.12 DAN MYSQL 5.6.26"

- **Dibuat** dan diselesaikan sendiri, dengan menggunakan literature hasil kuliah, survey lapangan, dibantu oleh dosen pembimbing maupun asisten dosen pembimbing, serta buku-buku maupun jurnal-jurnal ilmiah yang menjadi bahan acuan yang tertera dalam referensi karya Tugas Akhir ini.
- **Bukan** merupakan hasil duplikasi karya tulis yang sudah dipublikasikan atau yang pernah dipakai sebelumnya untuk mendapatkan gelar sarjana di Universitas/Perguruan Tinggi lain, kecuali yang telah disebutkan sumbernya dan dicantumkan pada referensi karya Tugas Akhir ini.
- **Bukan** merupakan karya tulis hasil terjemahan dari kumpulan buku atau jurnal acuan yang tertera dalam referensi karyaTugas Akhir ini.

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

Jakarta, 15 November 2017

Yang Membuat Pernyataan,



Martina Hasian

ABSTRAK

PT Krama Yudha Ratu Motor (PT KRM) merupakan sebuah perusahaan perseroan terbatas yang bergerak dalam bidang perakitan kendaraan otomatis jenis niaga bermerek Mitsubishi. PT Krama Yudha Ratu Motor memproduksi dua jenis kendaraan, yaitu kendaraan Niaga dan *Passenger*. Pada PT Krama Yudha Ratu Motor terdapat bagian *Part Control* yang bertugas mengirim *part* ke *line* produksi dan juga menangani penggantian *part* yang rusak saat diproduksi, sehingga setiap *part* cacat (*defect*) yang ditemui dapat diganti untuk diproduksi ke depannya. Sistem pemberitahuan *part* rusak dari produksi ke *Part Control* yang masih menggunakan kertas dokumen dirasakan kurang efektif karena proses informasi mengenai data *part* rusak memakan waktu yang cukup lama. Admin produksi harus mengisi *form Line Damage Card* yang berisi data *part* seperti: jenis, nama, dan *type part* yang rusak serta penyebab dari kerusakannya. Kemudian *Foreman* Produksi membawa *form Line Damage Card* kepada *manager* produksi, dan kepala bagian produksi untuk ditandatangani. Setelah itu, *form Line Damage Card* akan dibawa ke bagian *Part Control* untuk distempel sebagai persetujuan penggantian *part* baru dan dikirim kembali ke *line* produksi. Semua prosesnya dilakukan secara manual sehingga hal ini sangat memakan banyak waktu dalam penyampaian informasi. Untuk mendukung proses pengawasan *part*, disusunlah sebuah sistem informasi dengan metode pengembangan sistem *evolutionary prototype*. Sistem informasi ini dirancang dengan menggunakan *Unified Modelling Language* (UML) untuk pemodelan sistem, PHP 5.6.12 sebagai bahasa pemrograman dan MySQL5.6.26 sebagai basis datanya. Dengan dibuatnya Sistem Informasi Pengawasan *Part* ke *Line* Produksi memudahkan penginputan data *part* serta pelaporan data *part* rusak lebih cepat dan terkomputerisasi.

Kata Kunci: *Unified Modelling Language* (UML), PHP, MySQL, Sistem Informasi Pengawasan *Part*.

KATA PENGANTAR

Puji Syukur kepada Allah Bapa yang telah memberikan penyertaan, kekuatan dan kemampuan sehingga dapat menyelesaikan Tugas Akhir ini dengan baik dan lancar.

Tugas Akhir ini disusun untuk memenuhi persyaratan akademik yang telah ditetapkan oleh Politeknik STMI dengan bobot 4 SKS. Penulis mengambil pokok bahasan untuk laporan dengan judul "RANCANG BANGUN SISTEM INFORMASI PENGAWASAN *PART* KE *LINE* PRODUKSI MENGGUNAKAN PHP 5.6.12 DAN MySQL 5.6.26". Penulisan ini juga merupakan penerapan atau latihan untuk mengaplikasikan teori-teori yang pernah penulis dapatkan pada mata kuliah yang dipelajari di bangku kuliah dalam kenyataan di lapangan.

Dalam penulisan laporan ini, penulis banyak mendapatkan bantuan, bimbingan, petunjuk dan dukungan dari berbagai pihak. Dengan ketulusan dan kerendahan hati, perkenankanlah penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Bapak Dr. Mustofa ST, MT selaku Direktur Politeknik STMI Jakarta
2. Bapak Drs. Jacob Saragih, MM selaku Ketua Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta dan selaku Dosen Pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
3. Ibu Noveriza Yuliasari, SSi, MT selaku Sekretaris Program Studi Sistem Informasi Industri Otomotif di Politeknik STMI Jakarta.
4. Ibu Fifi Lailasari Hadianastuti, S. Kom, M. Kes selaku Asisten Dosen Pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
5. Bapak Robby Cahyadi selaku *Foreman Departmen Part Control* dan seluruh staff pada PT Krama Yudha Ratu Motor yang telah membantu dalam melaksanakan penelitian di PT Krama Yudha Ratu Motor.
6. Seluruh jajaran dosen dan staf karyawan Politeknik STMI Jakarta yang telah membantu dalam kelancaran pembuatan penulisan ini.

7. Kedua Orang tua, Bapak Pinondang Tampubolon dan Ibu Mindo Panjaitan, serta kakak dan adik saya, Desmond Ary, Gian Giovedi, dan Reinaldo Asido yang telah memberikan doa, dukungan, pengorbanan, semangat dan kasih sayang hingga saat ini.
8. Teman-teman seperjuangan dari kelas SA02 angkatan 2010, atas dukungan dalam penyusunan Tugas Akhir.
9. Rekan-rekan mahasiswa Politeknik STMI Jakarta Program Studi Sistem Informasi dengan masukan dan motivasinya selama penulisan ini.

Dengan segala kemampuan dan keterbatasan, penulis menyadari segala kekurangan yang dalam penulisan, karena itu penulis sangat mengharapkan segala kritik atau saran yang dapat membangun dari semua pihak dan juga berharap penulisan ini dapat berguna bagi diri pribadi maupun pihak-pihak lain yang membacanya.

Jakarta, September 2017

Penulis

DAFTAR ISI

Halaman

Halaman Judul i

Lembar Persetujuan Dosen Pembimbing ii

Lembar Persetujuan Asisten Dosen Pembimbing iii

Lembar Bimbingan Tugas Akhir iv

Lembar Pernyataan Keaslian vi

Abstrak vii

Kata Pengantar viii

Daftar Isi x

Daftar Gambar xiv

Daftar Tabel xvi

BAB I PENDAHULUAN 1

1.1 Latar Belakang 1

1.2 Pokok Permasalahan 2

1.3 Tujuan Penelitian 2

1.4 Batasan Masalah 3

1.5 Manfaat Penelitian 3

1.6 Sistematika Penulisan 5

BAB II LANDASAN TEORI 6

2.1 Konsep Dasar Sistem Informasi 6

2.1.1 Sistem 6

2.1.2 Karakteristik Sistem 6

2.1.3 Pengertian Informasi 9

2.1.4 Siklus Informasi 9

2.1.5 Kualitas Informasi 10
2.1.6 Sistem Informasi 10
2.2 Pengembangan Sistem 13
2.2.1 Model *Prototype* secara umum 14
2.2.2 *Evolutionary Prototype* 17
2.3 Pengertian Bahan Baku 18
2.4 Pengertian Pengawasan 19
2.5 Pengertian *Part Control* 19
2.5.1 Fungsi *Part Control* 19
2.6 Bagan Alir *Flowchart* 20
2.7 *Unified Modeling Language* 21
2.7.1 *Use Case Diagram* 23
2.7.2 *Class Diagram* 24
2.7.3 *Activity Diagram* 25
2.7.4 *Sequence Diagram* 26
2.7.5 *Deployment Diagram* 28
2.8 Kamus Data 29
2.9 *Hierarchy Plus Input-Output* (HIPO) 30
2.10 *Hypertext Preprocessor* (PHP) 32
2.11 *My Structure Query Language* (MySQL) 33
2.12 XAMPP 34
BAB III METODOLOGI PENELITIAN 36
3.1 Jenis dan Sumber Data 36
3.2 Metode Pengumpulan Data 36
3.3 Metode Pendekatan dan Pengembangan Sistem 37
3.3.1 Metode Pendekatan Sistem 37
3.3.2 Metode Pengembangan Sistem 38

3.4 Kerangka Penelitian 40

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA 43

4.1 Hubungan Kerja Sama PT Krama Yudha Ratu Motor 43

4.2 Struktur Organisasi 44

4.3 Sistem Manajemen 46

4.4 Jenis-Jenis Produk 49

4.5 *Part Control* 50

4.5.1 Sistem *Part Control* 50

4.5.2 Jenis *Part* 51

4.6 Proses Pengiriman *Part* yang Sedang Berjalan 54

4.6.1 Dokumen yang Terkait 56

4.7 *Use Case* Sistem yang Berjalan di PT KRM 59

BAB V ANALISIS DAN PEMBAHASAN 64

5.1 Analisis Kebutuhan Sistem 64

5.2 Perancangan Sistem Usulan 65

5.2.1 *Flowmap* Usulan 65

5.2.2 *Use Case* Usulan 67

5.2.3 *Use Case Description* 68

5.2.4 *Activity Diagram* Usulan 75

5.2.5 *Sequence* yang Diusulkan 83

5.3 *Class Diagram* 93

5.3.1 Kamus Data 93

5.4 *Deployment Diagram* 96

5.4.1 HIPO 97

5.4.2 Perancangan Interface Program 98

5.5 Implementasi Sistem 102

BAB VI KESIMPULAN DAN SARAN **103**

6.1 Kesimpulan 103

6.2 Saran 103

DAFTAR PUSTAKA **104**

LAMPIRAN

DAFTAR GAMBAR

Halaman

Gambar II.1 Karakteristik Sistem 8

Gambar II.2 Siklus Informasi 10

Gambar II.3 Blok Sistem Informasi yang Berinteraksi 11

Gambar II.4 Mekanisme Pengembangan Sistem dengan Prototipe 16

Gambar II.5 *Evolutionary Prototype* Model 18

Gambar II.6 Klasifikasi Diagram *UML* 22

Gambar II.7 *Visual Table of Content* 31

Gambar II.8 *Overview Diagram* 32

Gambar III.1 Kerangka Penelitian 40

Gambar IV.1 Hubungan Kerjasama PT KRM 44

Gambar IV.2 Struktur Organisasi 45

Gambar IV.3 Dokumen SMML PT KRM 48

Gambar IV.4 Bagian *Part Control* 51

Gambar IV.5 *Flowmap* proses pengawasan *part* yang sedang berjalan 55

Gambar IV.6 Dokumen Rencana Produksi PC/PPC 56

Gambar IV.7 Dokumen Pengiriman *Part* Rawan Hilang/Rusak 57

Gambar IV.8 *Form Line Damage Card* 58

Gambar IV.9 *Use Case* proses pengawasan *part* yang sedang berjalan 60

Gambar V.1 *Flowmap* Sistem Informasi Pengawasan *Part* usulan 66

Gambar V.2. *Use Case* Sistem Informasi Pengawasan *Part* usulan 67

Gambar V.3 *Activity Diagram* Proses *Login* 75

Gambar V.4 *Activity Diagram* Mengelola Master User 76

Gambar V.5 *Activity Diagram* Mengelola Master Barang 77

Gambar V.6 *Activity Diagram* Mengelola Master Kendaraan 78

Gambar V.7 *Activity Diagram* Membuat Data Rencana Produksi PC/PPC 79
Gambar V.8 *Activity Diagram* Mencetak Data Rencana Produksi PC/PPC..... 80
Gambar V.9 *Activity Diagram* Membuat Laporan Pengiriman Part..... 81
Gambar V.10 *Activity Diagram* Membuat Line Damage Card 82
Gambar V.11 *Activity Diagram* Menyetujui Line Damage Card 83
Gambar V.12 *Sequence Diagram Login*..... 84
Gambar V.13 *Sequence Diagram* Data User..... 84
Gambar V.14 *Sequence Diagram* Data Barang..... 85
Gambar V.15 *Sequence Diagram* Data Kendaraan 86
Gambar V.16 *Sequence Diagram* Membuat Data Rencana Produksi 86
Gambar V.17 *Sequence Diagram* Mencetak Data Rencana Produksi..... 87
Gambar V.18 *Sequence Diagram* Membuat Laporan Pengiriman *Part*..... 88
Gambar V.19 *Sequence Diagram* Membuat *Line Damage Card*..... 89
Gambar V.20 *Sequence Diagram* Menyetujui LDC oleh Kepala Bagian..... 90
Gambar V.21 *Sequence Diagram* Menyetujui LDC oleh Manager..... 91
Gambar V.22 *Sequence Diagram* Menyetujui LDC oleh *Part Control*..... 92
Gambar V.23 *Class Diagram* 93
Gambar V.24 *Deployment Diagram* yang Diusulkan..... 97
Gambar V.25 HIPO Sistem Informasi Pengawasan *Part*..... 97
Gambar V.26 Rancangan Tampilan *Form Login* 98
Gambar V.27 Rancangan Tampilan Halaman Menu Utama 99
Gambar V.28 Rancangan Tampilan Master *User*..... 100
Gambar V.29 Rancangan Tampilan Master Barang..... 100
Gambar V.30 Rancangan Tampilan Menu *Line Damage Card* 101
Gambar V.31 Rancangan TampilanMenyetujui *Line Damage Card* 101

DAFTAR TABEL

Halaman

Tabel II.1 Simbol Bagan Alir Sistem *(System Flowchart)* 20

Tabel II.2 Simbol-Simbol *Use Case Diagram* 23

Tabel II.3 Simbol-Simbol Class *Diagram* 25

Tabel II.4 Simbol-Simbol *Activity Diagram* 26

Tabel II.5 Simbol-Simbol *Sequence Diagram* 27

Tabel II.6 Simbol-Simbol *Deployment Diagram* 29

Tabel II.7 Beberapa Tipe Data Pada MYSQL 33

Tabel IV.1 Jenis-Jenis Produk 49

Tabel IV.2 Jenis *Part* Kendaraan CJM 51

Tabel IV.3 Jenis *Part* Kendaraan SL 52

Tabel IV.4 Jenis *Part* Kendaraan Truck Diesel 52

Tabel IV.5 Jenis *Part* Kendaraan Fuso 53

Tabel IV.6 Jenis *Part* Kendaraan Outlander Sport 53

Tabel IV.7 Defini Aktor dalam Proses Pengawasan *Part* 61

Tabel IV.8 Defini *Use Case* dalam Proses Pengawasan *Part* 61

Tabel V.1 Kebutuhan Sistem 64

Tabel V.2 Definisi *Actor Use Case* Sistem Pengawasan *Part* 68

Tabel V.3 *Use Case Description Login* 68

Tabel V.4 *Use Case Description* Mengelola Data Master 69

Tabel V.5 *Use Case Description* Mater *User* 69

Tabel V.6 *Use Case Description* Mater Barang 70

Tabel V.7 *Use Case Description* Master Kendaraan 70

Tabel V.8 *Use Case Description* Membuat Data Rencana Produksi 71

Tabel V.9 *Use Case Description* Mencetak Data Rencana Produksi 72

Tabel V.10 *Use Case Description* Membuat Laporan Pengiriman *Part* 72
Tabel V.11 *Use Case Description* Mengecek Laporan Pengiriman *Part* 73
Tabel V.12 *Use Case Description* Membuat *Line Damage Card* 73
Tabel V.13 *Use Case Description* Mengecek *Line Damage Card* 74
Tabel V.14 *Use Case Description* Menyetujui *Line Damage Card* 74
Tabel V.15 Tabel *User* 94
Tabel V.16 Tabel Barang 94
Tabel V.17 Tabel Kendaraan 95
Tabel V.18 Tabel Membuat Rencana Produksi PC/PPC 95
Tabel V.19 Tabel Membuat Laporan Pengiriman *Part* 96
Tabel V.20 Tabel Membuat *Line Damage Card* 96

BAB I
PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi informasi yang saat ini semakin pesat mendukung banyak perkembangan di dunia bisnis dan industri. Perkembangan tersebut meliputi banyaknya aktifitas-aktifitas bisnis yang dahulu dilakukan secara manual kini dapat dilakukan secara otomatis. Data yang dahulu dibuat dan disimpan dalam *hardcopy* kini dapat dikelola secara digital.

Sistem informasi pada perusahaan digunakan untuk pengolahan data dan pelaporan yang terkomputerasi karena lebih cepat, lengkap, tepat dan akurat, sehingga dapat dipakai sebagai alat pendukung pengambilan keputusan secara efektif.

PT Krama Yudha Ratu Motor (PT KRM) merupakan sebuah perusahaan perseroan terbatas yang bergerak dalam bidang perakitan kendaraan bermerek Mitsubishi. Pada PT Krama Yudha Ratu Motor terdapat bagian *Part Control* yang bertugas mengirim *part* ke *line* produksi dan juga menangani penggantian *part* yang rusak saat diproduksi, sehingga setiap *part* cacat (*defect*) yang ditemui dapat diganti untuk diproduksi ke depannya.

Ada tiga jenis dokumen pada Sistem Pengawasan *Part*, yaitu dokumen Rencana Produksi PC/PPC, dokumen Pengiriman *Part* Rawan Hilang/Rusak, dan dokumen *Line Damage Card*. Dokumen Rencana Produksi PC/PPC merupakan dokumen yang digunakan *Part Control* sebagai acuan jumlah *part* yang akan dikirim ke *line* produksi. Dokumen Pengiriman *Part* Rawan Hilang/Rusak merupakan dokumen yang berisi data pengiriman *part* yang diketahui sering hilang atau rusak. Sedangkan dokumen *Line Damage Card* merupakan data *part* yang rusak saat sedang diproduksi.

Sistem pengawasan *part* pada *Part Control* yang masih menggunakan dokumen kertas dirasakan kurang efektif karena proses informasi mengenai data *part* rusak memakan waktu yang cukup lama. Admin produksi harus mengisi *form*

Line Damage Card yang berisi data *part* seperti: jenis, nama, dan tipe *part* yang rusak serta penyebab dari kerusakannya. Kemudian *Foreman* Produksi membawa *form Line Damage Card* kepada Kepala Bagian produksi, dan Manager produksi untuk ditandatangani. Setelah itu, *form Line Damage Card* akan dibawa ke bagian *Part Control untuk* distempel sebagai persetujuan penggantian *part* baru dan mengirim kembali ke *line* produksi. Semua prosesnya dilakukan secara manual sehingga hal ini sangat memakan banyak waktu dalam penyampaian informasi. Dengan latar belakang permasalahan inilah maka penulis Tugas Akhir mengambil judul **"Rancang Bangun Sistem Informasi Pengawasan *Part* ke *Line* Produksi pada PT Krama Yudha Ratu Motor dengan menggunakan PHP 5.6.12 dan MySQL 5.6.26".**

1.2. Permasalahan

Adapun permasalahan yang akan dibahas pada penelitian Tugas Akhir ini adalah sebagai berikut:

1. Pembuatan laporan Pengiriman Part Rawan Hilang/Rusak dan *form Line Damage Card* masih berupa dokumen kertas sehingga rentan terjadi kerusakan atau hilang data serta pencarian data pengiriman *part* dan data *part* rusak pada dokumen kertas cukup memakan banyak waktu.
2. Sistem penyampaian data dari Produksi ke *Part Control* masih menggunakan cara pemberitahuan manual seperti datang ke tempat tujuan.
   Foreman produksi juga harus membawa *Line Damage Card* ke berbagai pihak untuk meminta persetujuan sehingga memperlambat proses penggantian *part* yang baru.

1.3. Tujuan Penelitian

Adapun tujuan dari Penetilian Tugas Akhir ini adalah:

1. Merancang bangun Sistem Informasi Pengawasan *Part* untuk memudahkan pembuatan laporan Pengiriman Part Rawan Hilang/Rusak dan *Line Damage Card* yang terkomputerisasi tanpa harus selalu menggunakan kertas sebagai

sarana informasi sehingga pencarian data pengiriman *part* dan *part* rusak lebih mudah dilakukan.

2. Membangun Sistem Informasi Pengawasan *Part* untuk memudahkan *Part Control* dan Produksi saling berinteraksi dalam penyampaian data laporan pengiriman *part* dan *part* rusak. Memudahkan Kepala Bagian Produksi dan Manager Produksi dalam memberikan persetujuan pada *Line Damage Card* dengan waktu yang lebih singkat.

1.4. Pembatasan Masalah

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

1. Penelitian dilakukan selama 1 bulan di PT Krama Yudha Ratu Motor bagian *Part Control* pada tanggal 1 -- 31 Mei 2014
2. Jenis kendaraan yang dibahas adalah kendaraan niaga yaitu jenis CJM, SL, TD, dan Fuso.
3. Sistem informasi yang dibangun hanya mencakup proses pengawasan *part* yang rusak saat sedang diproduksi.
4. Sistem informasi yang dibangun hanya menangani masalah penginputan data, pembuatan data *Line Damage Card* oleh *Foreman* Produksi serta proses persetujuan *Line Damage Card* oleh Kepala Bagian Produksi dan Manager Produksi.

1.5. Manfaat Tugas Akhir

Beberapa manfaat yang diperoleh dalam pelaksanaan penelitian dan penyusunan Tugas Akhir adalah:

1. Bagi perusahaan

   Rancang Bangun Sistem Informasi Pengawasan *Part* diharapkan dapat mempermudah kegiatan penginputan data *part* khususnya pada bagian *Part Control* di PT Krama Yudha Ratu Motor serta dapat memberikan kontribusi untuk perbaikan dan pengembangan sistem informasi *part* pada perusahaan itu ataupun pada perusahaan sejenis lainnya.

2. Bagi penulis

   Rancang Bangun Sistem Informasi Pengawasan *Part* diharapkan dapat menjadi saran untuk kebenaran teori selama perkuliahan terhadap aplikasinya pada perusahaan sehingga penulis dapat merumuskan dan mencari solusi penyelesaian masalah yang terjadi di dalam perusahaan secara praktis dan menerapkannya di dalam dunia kerja.

3. Bagi pihak lain

   Pihak lain dalam hal ini adalah para pengambil keputusan sehingga Rancang Bangun Sistem Informasi Pengawasan *Part* diharapkan dapat digunakan sebagai sumbangan pemikiran dan juga sebagai referensi bagi peneliti lain yang mengadakan penelitian serupa.

1.6. Sistematika Penulisan

Sitematika penulisan ini ditujukan untuk memberikan gambaran sistematis yang terdiri dari:

BAB I PENDAHULUAN

Bab ini merupakan bagian pendahuluan yang membahas tentang latar belakang masalah, permasalahan, tujuan penelitian, pembatasan masalah, manfaat penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menjelaskan berbagai teori yang diperoleh dari buku-buku literatur ataupun berbagai macam referensi. Teori yang dipaparkan pada Rancang Bangun Sistem Informasi Pengawasan *Part* seperti pengertian sistem informasi, pengembangan sistem informasi, *Part Control*, bagan alir (flowchart), Unified Modelling Languange (UML), kamus data, HIPO, konsep dasar *Hypertext Preprocessing (PHP), MySQL* dan XAMPP.

BAB III METODOLOGI PENELITIAN

Bab ini berisikan tentang jenis dan sumber data serta kerangka penelitian. Adapun kerangka penelitian ini sebagai berikut:

identifikasi masalah, studi pendahuluan, penggunaan *prototype evolutionary* serta kesimpulan dan saran.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini membahas tentang profil perusahaan, jenis produk, proses pengawasan *part* yang sedang berjalan menggunakan *flowchart* dan UML

BAB V ANALISIS DAN PEMBAHASAN

Bab ini menguraikan analisis yang dirinci tentang proses sistem pelaporan *part* rusak yang diusulkan menggunakan *flowchart, Unified Modelling Languange* (UML), *Hierarchy plus Input-Process-Output (HIPO),* dan perancangan *interface* program.

BAB VI KESIMPULAN DAN SARAN

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

BAB II
LANDASAN TEORI

2.1 Konsep dasar Sistem Informasi

2.1.1 Sistem

Kata sistem awalnya berasal dari bahasa Yunani (sustēma) dan bahasa Latin (systēma). Berikut pengertian system menurut beberapa ahli antara lain sebagai berikut:

1. Dalam buku Hartono (2005), sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegitan atau untuk mnyelesaikan suatu sasaran yang tertentu FitzGerald et al, (1981)
2. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (Hartono, 2005)

2.1.2 Karakteristik Sistem

Menurut Hartono (2005), suatu sistem mempunyai sifat atau karakteristik tertentu, yaitu antara lain sebagai berikut:

1. Komonen-komponen (*components)*

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari system yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu yang lebih besar lagi yang disebut *supra system.*
2. Batasan sistem (*boundary*)

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini

memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem (*environments*)

   Lingkungan luar dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung (*interface*)

   Penghubung merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran dari satu subsistem akan menjadi masukan untuk subsistem yang lainnya melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan (*input*)

   Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh di dalam sistem komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran (*output*)

   Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna

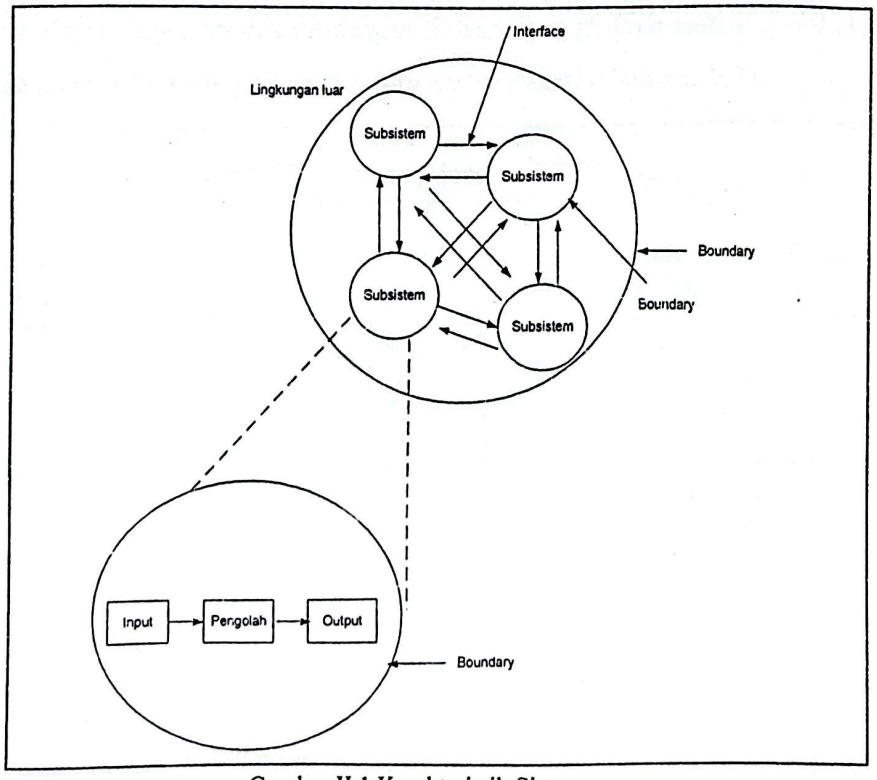
merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

7. Pengolah (*process*)

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

8. Sasaran (*objectives*) atau tujuan (*goal*)

   Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.



Gambar II.1 Karakteristik Sistem

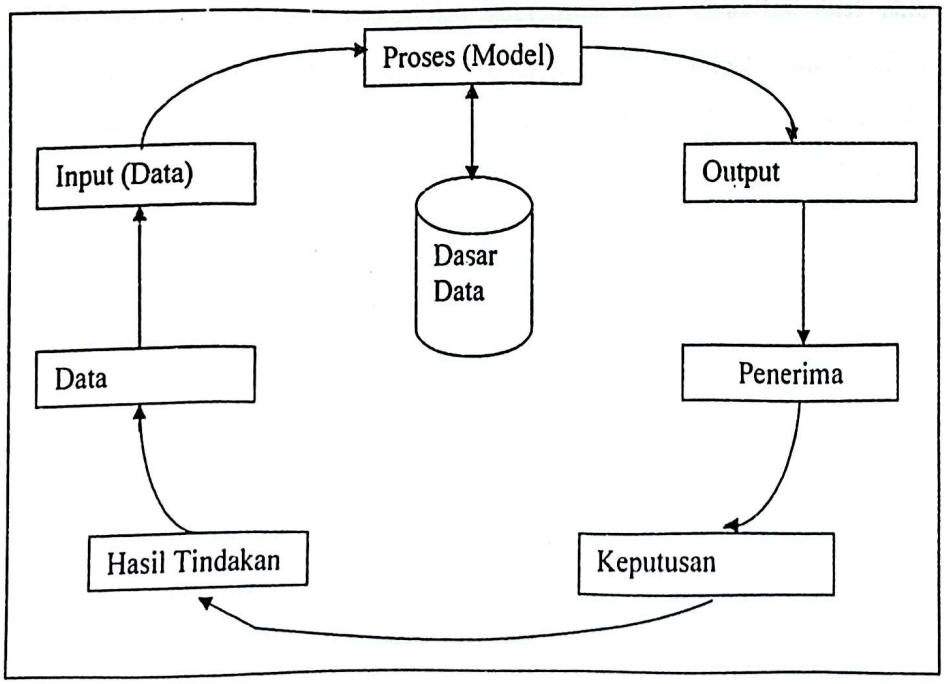
(Sumber: Hartono, 2005)

2.1.3 Informasi

Menurut Hartono (2005) informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.

2.1.4 Siklus Informasi

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



Gambar II.2 Siklus Informasi

(Sumber: Hartono, 2005)

2.1.5 Kualitas informasi

Kualitas informasi sangat dipengaruhi oleh 3 hal, yaitu (Hartono, 2005):

1. Relevan *(Relevancy)*

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

2. Akurat *(Accuracy)*

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

3. Tepat Waktu *(timeliness)*

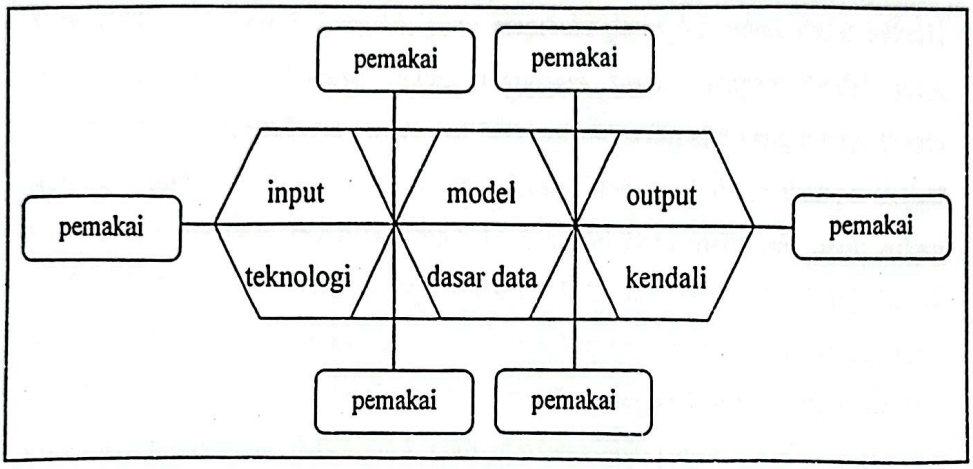
   Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat. Informasi yang terlambat tidak mempunyai nilai yang baik, sehingga kalau digunakan sebagai dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan dalam keputusan dan tindakan. Kondisi demikian menyebabkan mahalnya suatu nilai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

2.1.6 Sistem Informasi

Dalam buku Hartono (2005), sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Robert dan Roscoe, 1983)

Sistem informasi dapat terdiri dari beberapa komponen yang disebut blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), dan blok teknologi (*technology block*), blok dasar data (*database block*), dan blok kendali (*control block*). Sebagai suatu sistem,

keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarannya.



Gambar II.3 Blok Sistem Informasi yang Berinteraksi

(Sumber: Hartono, 2005)

Keterangan:

Blok Masukan

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

Blok Model

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

Blok Keluaran

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

Blok Teknologi

Teknologi merupakan "kotak alat" (*tool-box*) dari pekerjaan sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu

pengendalian dari sistem keseluruhan. Teknologi terdiri dari 2 bagian utama, yaitu perangkat lunak (*software*) dan perangkat keras (*hardware*).

Perangkat lunak berupa program yang membuat perangkat keras dapat bekerja dengan mengintruksikannya untuk memroses sesuai dengan model yang diterapkan. Perangkat keras terdiri dari bermacam-macam alat yang menyediakan dukungan phisik untuk blok-blok lainnya. Untuk blok *input*, disediakan perangkat keras untuk memasukkan data. Untuk blok model guna memroses berdasarkan model dibutuhkan alat proses yang disebut CPU (*Central Processing Unit*). Untuk blok keluaran diperlukan alat untuk menampilkan hasil pengolahan. Untuk blok basis data digunakan teknologi *diskette, harddisk, magnetic tape* maupun *optical disc* untuk menyimpan data guna keperluan proses lebih lanjut.

Blok Basis Data

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

Blok Kendali

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

2.2 Pengembangan Sistem

Menurut Hartono (2005), pengembangan sistem dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diperbaiki oleh karena beberapa hal, yaitu:

1. Munculnya masalah pada sistem yang lama.
2. Untuk meraih kesempatan.
3. Adanya instruksi.

Proses perancangan atau pengembangan sistem informasi, mulai dari konsep sampai dengan implementasinya disebut dengan istilah *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* (SDLC) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik) (Rosa A. S. dan M. Shalahuddin, 2013).

Menurut Rosa A. S. dan M. Shalahuddin (2013), tahapan-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*.

2.2.1 Prototipe

Prototipe merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat sesuatu program dengan cepat dan

bertahap sehingga segera dapat dievaluasi oleh pemakai. Hal ini berbeda dengan pendekatan SDLC tradisional (konvensional) yang lebih banyak menghabiskan waktu untuk menghasilkan spesifikasi yang sangat rinci sebelum pemakai dapat mengevaluasi sistem. Mengingat kebanyakan pemakai mengalami kesulitan dalam memahami spesifikasi sistem berakibat bahwa pemakai tidak begitu paham sampai pengujian dilakukan. Selain itu, prototipe membuat proses pegembangan sistem informasi menjadi lebih mudah, terutama pada keadaan kebutuhan pemakai sulit untuk diidentifikasi (Kadir, 2009).

Mekanisme pengembangan sistem dengan prototipe menurut Kadir (2009) antara lain sebagai berikut:

1. Identifikasi kebutuhan pemakai

   Tahap ini merupakan tahap dimana pengembang dan pemakai bertemu. Pada tahap ini juga pemakai akan menjelaskan kebutuhan sistemnya.

2. Membuat prototipe

   Pada tahap ini, pengembang akan mulai membuat prototipe sesuai dengan kebutuhan pemakai.

3. Menguji prototipe

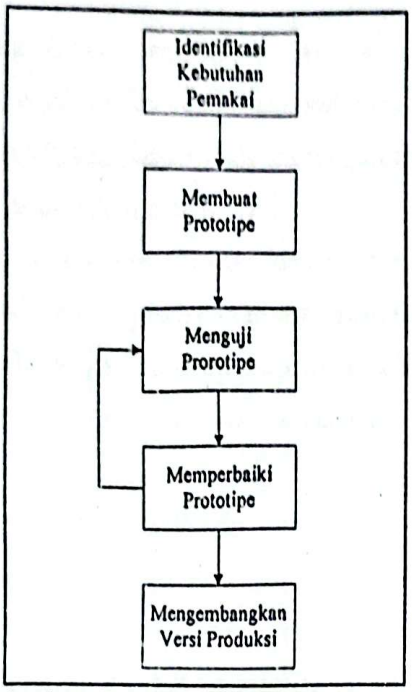
   Pada tahap ini, pemakai akan menguji prototipe ydan memberikan kritikan atau saran.

4. Memperbaiki prototipe

   Pada tahap ini, pengembang akan melakukan modifikasi sesuai dengan masukan pemakai. Setelah modifikasi selesai dilakukan, maka akan dilanjutkan ke tahap menguji prototipe. Tahap ini akan terus berulang sampai prototipe yang dibuat sesuai dengan kebutuhan pemakai.

5. Mengembangkan versi produksi

   Tahap ini merupakan tahap terakhir dalam pembuatan prototipe dimana pengembang merampungkan sistem sesuai dengan masukan terakhir dari pemakai.



Gambar II.4 Mekanisme Pengembangan Sistem Dengan Prototipe

(Sumber: Kadir, 2009)

Kelebihan model prototipe adalah sebagai berikut (Kadir, 2009):

1. Pendefinisian kebutuhan pemakai menjadi lebih baik karena keterlibatan pemakai yang lebih intensif.
2. Meningkatkan kepuasan pemakai dan mengurangi risiko pemakai tidak menggunakan sistem mengingat keterlibatan mereka yang sangat tinggi sehingga sistem memenuhi kebutuhan mereka dengan lebih baik.
3. Mempersingkat waktu pengembangan.
4. Memperkecil kesalahan disebabkan pada setiap versi prototipe, kesalahan segera terdeteksi oleh pemakai.
5. Pemakai memiliki kesempatan yang lebih banyak dalam meminta perubahan-perubahan.
6. Menghemat biaya (menurut penelitian, biaya pengembangan dapat mencapai 10% hingga 20% dibandingkan kalau menggunakan SDLC tradisional).

Kelemahan model prototipe adalah sebagai berikut (Kadir, 2009):

1. Prototipe hanya bisa berhasil jika pemakai bersungguh-sungguh dalam menyediakan waktu dan pikiran untuk menggarap prototipe.

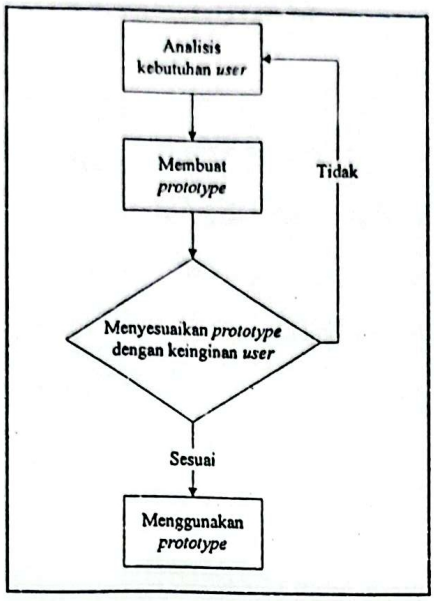
2. Kemungkinan dokumentasi terabaikan karena pengembang lebih berkonsentrasi pada pengujian dan pembuatan prototipe.
3. Mengingat target waktu yang pendek, ada kemungkinan sistem yang dibuat tidak lengkap dan bahkan sistem kurang teruji.
4. Jika terlalu banyak proses pengulangan dalam membuat prototipe, ada kemungkinan pemakai menjadi jenuh dan memberikan reaksi yang negatif.
5. Apabila tidak terkelola dengan baik, prototipe menjadi tak pernah berakhir. Hal ini disebabkan pemintaan terhadap perubahan terlalu mudah untuk dipenuhi.

2.2.2 *Evolutionary Prototype*

Evolutionary prototype yaitu, *prototype* yang secara terus menerus dikembangkan hingga *prototype* tersebut memenuhi fungsi dan prosedur yang dibutuhkan oleh sistem. Pada pendekatan evolusioner, suatu *prototype* dibangun berdasarkan pada kebutuhan dan pemahaman secara umum. *Prototype* kemudian diubah dan dievolusikan dari pada dibuang. *Prototype* yang dibuang biasanya digunakan dengan aspek sistem yang dimengerti secara luas dan dibangun atas kekuatan tahapan *evolutionary prototype* (McLeod dan Scholl, 2008).

Tahapan-tahapan model *evolutionary prototype* menurut McLeod dan Scholl (2008) adalah sebagai berikut:

1. Analisis kebutuhan *user*, pengembang dan *user* atau pemilik sistem melakukan diskusi dimana *user* atau pemilik sistem menjelaskan kepada pengembang tentang kebutuhan sistem yang mereka inginkan.
2. Membuat *prototype*, pengembang membuat *prototype* dari sistem yang telah dijelaskan oleh *user* atau pemilik sistem.
3. Menyesuaikan *prototype* dengan keinginan *user* atau pemilik sistem, pengembang menanyakan kepada *user* atau pemilik sistem tentang *prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.
4. Menggunakan *prototype*, sistem mulai dikembangkan dengan *prototype* yang sudah dibuat.



Gambar II.5 *Evolutionary Prototype* Model
(Sumber: McLeod dan Scholl, 2008)

2.3 Bahan Baku

Part Bahan baku merupakan salah satu faktor pembentuk terjadinya barang jadi sehhingga segala sesuatu yang menyangkut bahan baku harus benar-benar diperhatikan. Masalah tersebut diantaranya:

- Bagaimana jumlah bahan baku yang tersedia tidak kurang karena akan mengganggu jalannya proses produksi.
- Bagaimana bahan baku agar jangan terlalu berlebihan karena merupakan pemborosan kalau terlalu lama.
- Bagaiman agar biaya ekstra yang digunakan untuk memesan bahan baku yang kurang (karena mengejar target jumlah produksi dan kapasitas mesin yang terpakai) tidak terlalu merugikan dan sebagainya.

Dengan adanya pengendalian bahan baku maka perusahaan akan berusaha untuk menyediakan bahan baku yang diperlukan dalam proses produksi sedemikian rupa agar berjalan dengan lancar tanpa terjadi kekuarangan persediaan atau kelebihan persediaan.

2.4 Pengawasan

Pengawasan adalah proses untuk memastikan bahwa segala aktifitas yang terlaksana sesuai dengan apa yang telah direncanakan.

Pengawasan merupakan suatu usaha agar suatu pekerjaan dapat dilaksanakan sesuai dengan rencana yang telah ditentukan, dan dengan adanya pengawasan dapat memperkecil timbulnya hambatan, sedangkan hambatan yang telah terjadi dapat segera diketahui untuk kemudian dapat dilakukan tindakan perbaikan.

2.5 *Part Control*

Part Control adalah berbagai kegitan yang digunakan oleh manajemen perusahaan untuk mengelola, mengatur, mengkoordinir, dan mengarahkan proses pengadaan dan pengiriman *part* dengan waktu yang tepat.

Pada dasarnya *Part Control* dimaksudkan untuk membantu kelancaran proses produksi, melayani kebutuhanperusahaan akan bahan baku dari waktu ke waktu.

Part Control yang dilaksanakan pada perusahaan yang satu dengan yang lain akan berbeda-beda tergantung pada sistem kebijaksanaan perusahaan yang digunakan pada perusahaan tersebut.

2.5.1 Fungsi *Part Control*

Fungsi utama *Part Control* adalah sebagai wadah untuk menangani proses keluar masuknya *part.*

Part Control melayani kebutuhan perusahaan akan bahan baku berupa *part* dari waktu ke waktu.

Fungsi tersebut di atas ditentukan oleh berbagai kondisi seperti:

1 Apabila jangka waktu pengiriman *part* relatif lama maka perusahaan perlu persediaan *part* yang cukup untuk memenuhi kebutuhan perusahaan selama jangka waktu produksi.

2 Sering kali *part* yang sedang diproduksi mengalami kerusakan sehingga dibutuhkan dengan segera penggantian *part* yang baru agar tidak mengganggu proses berjalannya produksi.

3. Selain untuk memenuhi permintaan langganan, pengendalian juga diperlukan apabila biaya untuk mencari *part* atau biaya kehabisan *part* atau bahan relatif besar.

Fungsi Part Control pada suatu perusahaan antara lain adalah:

a. Menghindari keterlambatan pengiriman *part*
b. Menghindari ada material/part yang rusak
c. Menjamin kelangsungan produksi

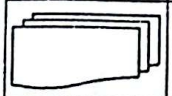
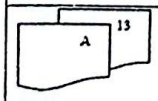
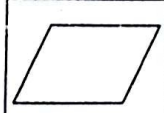
Salah satu tujuan dari pengendalian persediaan adalah meminimalkan biaya-biaya yang timbul akibat dari adanya persediaan tersebut

2.6 Bagan Alir (*Flowchart*)

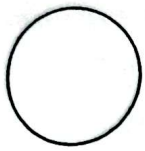
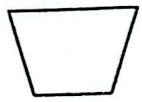
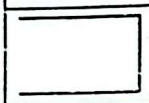
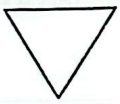
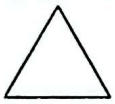
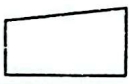
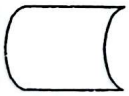
Flowchart adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi (Hartono, 2005).

Berikut ini merupakan simbol standar yang biasanya digunakan oleh analis dalam membuat bagan alir atau *flowchart* sistem, antara lain:

Tabel II.1 Simbol-Simbol *Flowchart*

Simbol	Nama	Keterangan
	Dokumen	Digunakan untuk semua jenis dokumen yang merupakan formulir untuk merekam transaksi
	Dokumen rangkap	Menggambarkan dokumen asli dan tembusannya
	Berbagai dokumen	Menggambarkan berbagai jenis dokumen yang digabungkan bersama dalam satu paket
	Catatan	Menggambarkan catatan akuntansi yang digunakan untuk mencatat data vang direkam sebelumnya di dalam dokumen
	Penghubung pada halaman yang berbeda	Untuk menggambarkan bagan alir dokumen suatu sistem diperlukan lebih dari satu

Tabel II.1 Simbol-Simbol *Flowchart* (Lanjutan)

Simbol	Nama	Keterangan
		halaman.
	Penghubung pada halaman yang sama	Menggambarkan alir dokumen dibuat mengalir dari atas ke bawah dan dari kiri kekanan. Simbol penghubung yang memungkinkan aliran dokumen berhenti di suatu lokasi pada halaman tertentu dan kembali berjalan pada halaman yang sama.
	Kegiatan manual	Untuk menggambarkan kegiatan manual seperti: menerima order, mengisi formulir, membandingkan dll
	Keterangan/komentar	Untuk menambahkan komentar agar pesan yang disampaikan lebih jelas
	Arsip sementara	Menunjukkan tempat penyimpanan dokumen
	Arsip permanen	Menunjukkan tempat penyimpanan dokumen secara permanen yang tidak akan diproses lagi
	*Online computer process*	Menggambarkan pengolahan komputer secara *online*
	*Keying, typing*	Menggambarkan pemasukan data ke dalam komputer melalui *online* terminal
	Pita *magnetic*	Menggambarkan arsip komputer yang berbentuk pita magnetic
	*Online storage*	Menggambarkan arsip komputer yang berbentuk *online* (di dalam memori komputer)

(Sumber: Hartono, 2005)

2.7 *Unified Modelling Language* (UML)

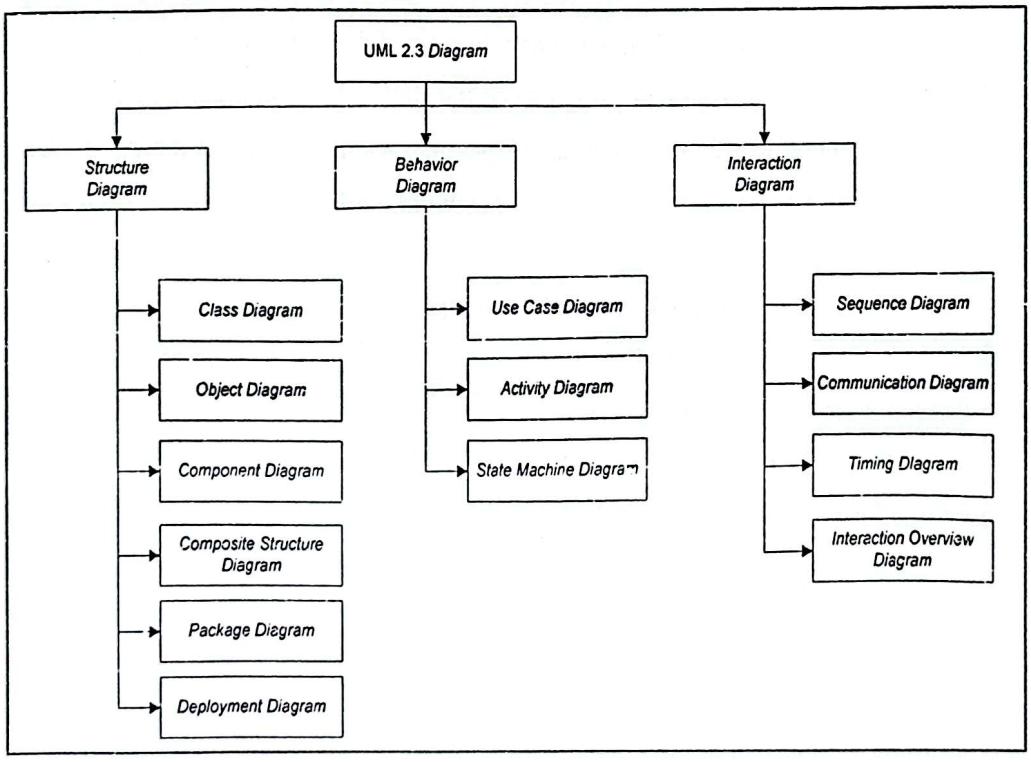
Menurut Munawar (2005), UML (*Unified Modeling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini dikarenakan UML menyediakan bahasa pemodelan

visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan mereka dengan yang lain.

Tujuan dari pemodelan UML adalah sebagai berikut:

1. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
2. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
3. Menyatukan praktik-praktik terbaik yang terdapat dalam pemodelan.

UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori tersebut dapat dilihat pada gambar berikut ini (Rosa A. S. dan M. Shalahuddin, 2013):



Gambar II.6 Klasifikasi Diagram UML

(Sumber: Rosa dan Shalahuddin, 2013)

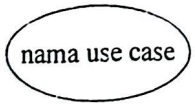
Berikut ini penjelasan singkat dari pembagian kategori tersebut (Rosa A. S. dan M. Shalahuddin, 2013):

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

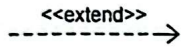
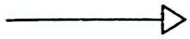
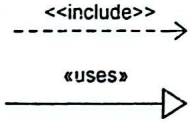
2.7.1 *Use Case Diagram*

Use case atau diagram *use case* 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. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa dan Shalahuddin, 2013)

Tabel II.2 Simbol-simbol *Use Case Diagram*

Simbol	Deskripsi
Use case 	Fungsionalitas yang disediakan oleh sistem sebagai unit-unit yang salling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama *use case.*
Aktor/*actor* 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambaran orag, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.

Tabel II.2 Simbol-simbol *Use Case Diagram* (Lanjutan)

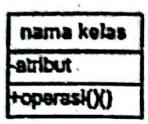
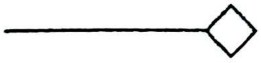
Simbol	Keterangan
Asosiasi/*association*	Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan aktor.
Ekstensi/*extend* 	Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri tanpa *use case* tambahan itu; mirip dengan prinsip *inheritance* pada pemrograman berorientasi objek. Arah panah mengarah pada *use case* yang ditambahkan.
Generalisasi/ *generalization* 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. Arah panah mengarah pada *use case* yang menjadi generalisasinya (umum).
Menggunakan/*include/ uses* 	Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini.

(Sumber: Rosa dan Shalahuddin, 2013)

2.7.2 *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

Tabel II.3 Simbol-simbol *Class Diagram*

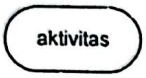
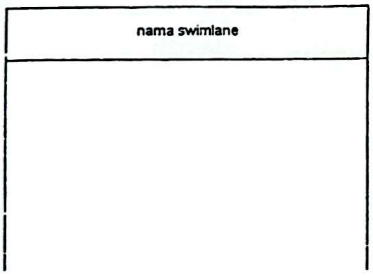
Simbol	Keterangan
Kelas 	Kelas pada struktur sistem.
Antarmuka/*interface* nama_interface	Sama dengan konsep *interface* dalam pemrograman berorientasi objek.
asosiasi/*association* 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity*
Asosiasi berarah/ *directed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

(Sumber: Rosa dan Shalahuddin, 2013)

2.7.3 *Activity Diagram*

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2013)

Tabel II.4 Simbol-simbol *Activity Diagram*

Simbol	Keterangan
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/*decision* 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan/*join* 	Asosiasi penggabungan dimana lebih dari satu aktivitas yang digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swimlane 	Memisahkan organisasi bisnis yang bertanggungjawab terhadap aktivitas yang terjadi.

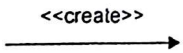
(Sumber: Rosa dan Shalahuddin, 2013)

2.7.4 *Sequence Diagram*

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

sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case* (Rosa A. S. dan M. Shalahuddin, 2013).

Tabel II.5 Simbol-simbol *Sequence Diagram*

Simbol	Keterangan
Aktor/*actor*  nama aktor Atau nama aktor tanpa waktu aktif	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambaran orag, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
Garis hidup/*lifeline* 	Menyatakan kehidupan suatu objek.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. Aktor tidak memiliki waktu aktif.
Pesan tipe *create*  	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe *call* 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri. Arah panah mengarah pada objek yang memiliki operasi/metode.

Tabel II.5 Simbol-simbol *Sequence Diagram* (Lanjutan)

Simbol	Keterangan
Pesan tipe *send* 1 : masukan	Menyatakan bahwa suatu objek mengirimkan data/masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi.
Pesan tipe *return* 1 : keluaran	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* <<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, 2013)

2.7.5 *Deployment Diagram*

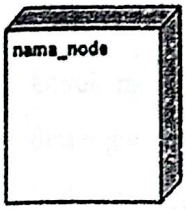
Diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut:

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device, node*, dan *hardware*.
2. Sistem *client/server*
3. Sistem terdistribusi murni
4. Rekayasa ulang aplikasi

Tabel II.6 Simbol-simbol *Deployment Diagram*

Simbol	Keterangan
Package	Merupakan sebuah bungkusan dari satu atau lebih *node*

Tabel II.6 Simbol-simbol *Deployment Diagram* (Lanjutan)

Simbol	Keterangan
Node 	Biasanya mengacu pada perangkat keras (*hardware*), perangkat lunak yang tidak dibuat sendiri (*software*), jika di dalam *node* disertakan komponen untuk mengkonsistensikan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan pada diagram komponen.
Kebergantungan/ *dependency* 	Kebergantungan antar *node*, arah panah mengarah pada *node* yang dipakai.
Link	Relasi antar *node*.

(Sumber: Rosa dan Shalahuddin, 2013)

2.8 Kamus Data

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

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

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

2.9 *Hierarchy plus Input-Proses-Output* (HIPO)

Hierarchy plus Input-Proses-Output (HIPO) merupakan metodologi yang dikembangkan dan didukung oleh IBM. HIPO sebenarnya adalah alat dokumentasi program. Tetapi sekarang, HIPO juga banyak digunakan sebagai alat disain dan teknik dokumentasi dalam siklus pengembangan sistem. HIPO berbasis pada fungsi, yaitu setiap modul didalam sistem digambarkan oleh fungsi utamanya. HIPO mempunyai sasaran utama sebagai berikut:

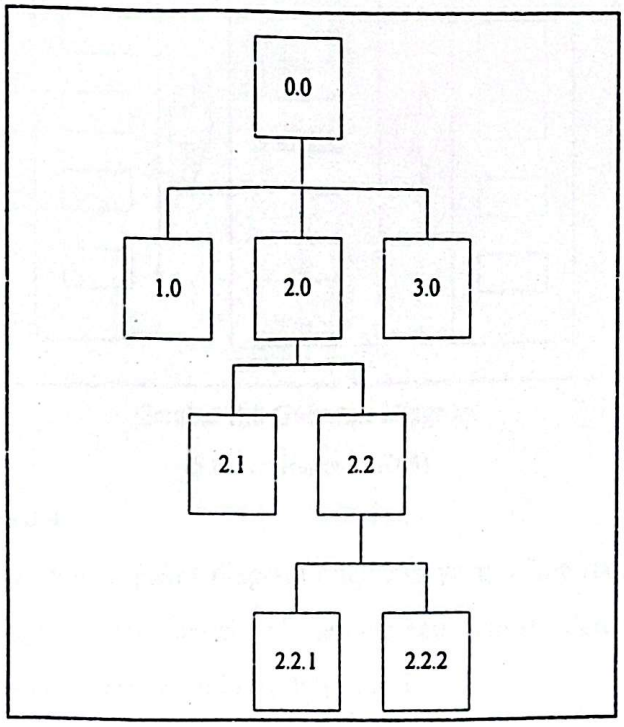
1. Untuk memberikan struktur yang memungkinkan fungsi suatu sistem dapat dimengerti.
2. Untuk menguraikan fungsi-fungsi yang akan dikerjakan oleh suatu program, bukan untuk mengkhususkan pernyataan program yang dipakai untuk melaksanakan fungsi-fungsi tersebut.
3. Untuk memberikan deskripsi visual dari *input* yang akan dipakai serta *output* yang akan dihasilkan oleh masing-masing fungsi pada tiap-tiap tingkat diagram.
4. Tujuan HIPO yang paling penting adalah untuk menghasilkan *output* yang benar dan dapat memenuhi kebutuhan *user*.

Menurut Hartono (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 sebagi 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 diitentifikasikan. Struktur paket diagram dan

hubungan fungsi juga diidentifikasikan dalam bentuk hirarki. Keterangan masing-masing fungsi diberikan pada bagian penjelasan yang diikutsertakan dalam diagram ini. *Visual table of contents* ini dapat digambarkan sebagai berikut:

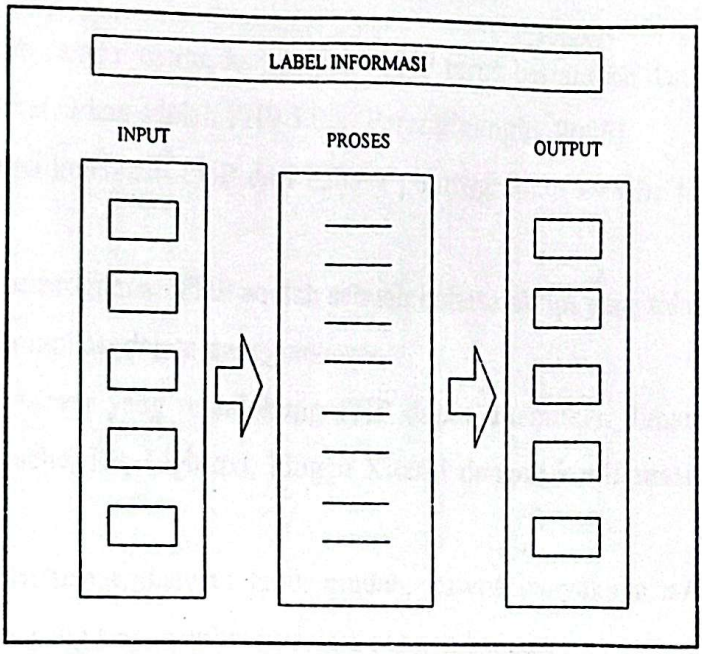


Gambar II.7 *Visual Table Of Contents*

(Sumber: Hartono, 2005)

2. *Overview diagram*

Overview diagram menunjukan secara garis besar hubungan dari *Input*, proses dan *Output*. Bagian *Input* menunjukan item-item data yang akan digunakan oleh bagian proses. Bagian proses berisi sejumlah langkah-langkah yang menggambarkan kerja dari fungsi. Bagian *Output* berisi dengan item-item data yang dihasilkan atau dimodifikasi oleh langkah-langkah proses. *Overview diagram* ini dapat dilihat pada Gambar II.8.



Gambar II.8 *Overview Diagram*

(Sumber: Hartono, 2005)

3. *Detail Diagram*

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

2.10 *Hypertext Preprocessing* (PHP)

Personal Home Page (PHP) diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, PHP digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Rasmus Lerdorf adalah salah seorang pendukung *open source*. Oleh karena itu, ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis kemudian menambah kemampuan PHP dan meluncurkan PHP 2.0 (Peranginangin, 2006).

Pada tahun 1996, PHP telah banyak digunakan dalam *website* di dunia. Sebuah kelompok pengembang *software* yang terdiri dari Rasmus, Zeew Suraski, Andi Gutman, Stig Bakken, Shane Caraveo dan Jim Winstead bekerja sama untuk menyempurnakan PHP 2.0. Akhirnya, pada tahun 1998, PHP 3.0. diluncurkan dan singkatan PHP diubah menjadi akronim berulang yaitu *Hypertext Prepocessing*. Penyempurnaan terus dilakukan sehingga pada tahun 2000 dikeluarkan PHP 4.0.

Tidak berhenti sampai disitu, kemampuan PHP terus bertambah dan versi terbaru yang telah dikeluarkan adalah PHP 5.0.x (Peranginangin, 2006).

Beberapa kelebihan PHP dari bahasa pemrograman *website* lainnya antara lain:

1. Bahasa pemrograman PHP adalah sebuah bahasa skript yang tidak melakukan sebuah kompilasi dalam penggunaanya.
2. *Website* Server yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya *milis-milis* dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahamanan, PHP adalah bahasa skripting yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah system.

2.11 MySQL

MySQL pertama kali dirintis oleh seorang *programmer database* bernama Michael Widenius. MySQL adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multi user*. MySQL memiliki dua bentuk lisensi, yaitu *free software* dan *shareware* (Wahana Komputer, 2010).

Berikut ini merupakan beberapa tipe data pada MySQL antara lain sebagai berikut:

Tabel II.7 Beberapa Tipe Data pada MySQL

Tipe Data	Ukuran	Keterangan
CHAR	M	Menampung maksimal M karakter (kombinasi huruf, angka, dan simbol-simbol). Jumlah memori yang dibutuhkan selalu M byte. M terbesar adalah 255.

Tabel II.7 Beberapa Tipe Data Pada MySQL (Lanjutan)

Tipe Data	Ukuran	Keterangan
VARCHAR	M	Karakter yang disimpan maksimal M karakter. Jumlah memori yang dibutuhkan tergantung jumlah karakter. M bisa mencapai 65535.
DATE	8 byte	Menyatakan tanggal.
TIME	8 byte	Menyatakan waktu (jam:menit:detik).
TINYINT	1 byte	Bilangan antara -128 sampai dengan +127.
SMALLINT	2 byte	Bilangan antara -32768 sampai dengan +32768.
INT	4 byte	Bilangan antara -2147683648 sampai dengan +2147683647.
FLOAT	4 byte	Bilangan pecahan.
DOUBLE	8 byte	Bilangan pecahan dengan presisi tinggi.
BOOL	1 byte	Untuk menampung nilai *true* (benar) dan *false* (salah). Identik dengan TINYINT.
ENUM	-	Menyatakan suatu tipe yang nilainya tertentu (disebutkan dalam pendefinisian).
TEXT	-	Menyimpan teks yang ukurannya sangat panjang.
BLOB	-	Untuk menyipmpan data biner (misalnya gambar atau suara).

(Sumber: Kadir, 2008)

2.12 XAMPP

XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolahan data MySQL di komputer lokal. XAMPP berperan sebagai *web server* pada komputer. XAMPP juga dapat disebut sebagai sebuah *CPanel server virtual*, yang dapat membantu melakukan *preview* sehingga dapat memodifikasi *website* tanpa harus *online* atau terakses dengan internet (Wicaksono, 2008).

Ada beberapa bagian XAMPP yang biasa digunakan pada umumnya yaitu sebagai berikut:

1. *htdocs*, adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan *script* lainnya.
2. *phpMyAdmin*, merupakan bagian untuk mengelola basis data MySQL yang ada di komputer.
3. *Control Panel*, yang berfungsi untuk mengelola layanan (*service*) XAMPP. Layanan (service) ini seperti menghentikan (*stop*) layanan, ataupun memulai (*start*) layanan.

BAB III
METODOLOGI PENELITIAN

Metodologi penelitian adalah sekumpulan peraturan, kegiatan dan prosedur yang digunakan dalam melakukan penyelidikan yang sistematis untuk meningkatkan sejumlah pengetahuan, juga merupakan suatu usaha yang sistematis dan terorganisasi untuk menyelidiki masalah tertentu yang memerlukan jawaban oleh pelaku dari suatu disiplin ilmu.

3.1 Jenis dan Sumber Data

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

1. Data Primer

   Data yang diperoleh melalui pengamatan langsung terhadap sistem yang sedang berjalan dan wawancara dengan pegawai sebagai sumber informasinya. Dalam penelitian ini data tersebut berupa alur pengawasan *part* dari bagian *Part Control* ke *line* produksi pada PT Krama Yudha Ratu Motor.

2. Data Sekunder

   Data sekunder merupakan sumber data penelitian yang diperoleh secara tidak langsung melalui media perantara. Dalam penelitian ini, data tersebut diperoleh dari data jumlah produksi.

3.2 Metode Pengumpulan Data

Metode yang digunakan dalam pengumpulan data adalah sebagai berikut:

1. Studi Lapangan
   a. Observasi

      Tahap ini merupakan tahap untuk mengumpulkan data dengan cara mengamati secara langsung keadaan maupun kegiatan dalam perusahaan untuk kemudian dilakukan pencatatan informasi terhadap objek yang

dianggap perlu. Observasi ini dilakukan di bagian Part Control CKD Departemen Part Control dan di bagian Produksi PT Krama Yudha Ratu Motor. Pada tahap ini juga dilakukan pengamatan langsung terhadap pengguna sistem dan pengamatan langsung terhadap sistem yang sedang berjalan.

b. Wawancara

Pada tahap ini penulis melakukan wawancara dengan para pengguna sistem tersebut untuk memperoleh keterangan mengenai alur penyampaian laporan data *part* rawan hilang rusak dalam menghasilkan data *part* kendaraan niaga dan laporan-laporannya serta mengenai kebutuhan sistem yang akan dikembangkan selanjutnya.

2. Studi Literatur

Pada tahap ini dilakukan studi literatur yaitu dengan membaca dan mempelajari sumber-sumber seperti buku yang berhubungan dengan analisis dan desain sistem informasi, buku pemrograman PHP, buku pemrograman MySQL, buku pembahasan Part Control, buku pengembangan perangkat lunak, dan juga mencari data tambahan yang diperlukan seperti konsep PHP, beberapa contoh *coding* program melalui media buku dan *internet*.

3.3 Metode Pendekatan dan Pengembangan Sistem

Metode pendekatan dan desain yang digunakan yaitu perancangan berorientasi objek yang merupakan alat bantu dalam OOD *(Object-Orienterd Design)* dan OOA *(Object-Oriented Analysis)*. Adapun metode pengembangan perangkat lunak yang digunakan adalah *prototype evolutionary*.

3.3.1 Metode Pendekatan Sistem

Metode pendekatan yang digunakan adalah metode pendekatan berorientasi objek (*object-oriented)*. Pendekatan berorientasi objek merupakan cara berfikir baru serta berlogika dalam menghadapi masalah-masalah yang akan di atasi dengan bantuan komputer. OOP mencoba melihat permasalahan lewat pengamatan dunia nyata dimana setiap objek adalah entitas tunggal yang memiliki

kombinasi struktur data dan fungsi. Pendekatan berorientasi objek terdiri dari analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD).

Analisis berorientasi objek (OOA) dimulai dengan menyatakan suatu masalah, analisis membuat suatu model situasi dari dunia nyata, menggambarkan sifat yang penting. Sedangkan desain berorientasi objek (OOD) merupakan tahap lanjutan setelah OOA, dimana tujuan sistem diorganisasikan ke dalam subsistem berdasarkan struktur analisis dan arsitektur yang dibutuhkan.

3.3.2 Metode Pengembangan Sistem

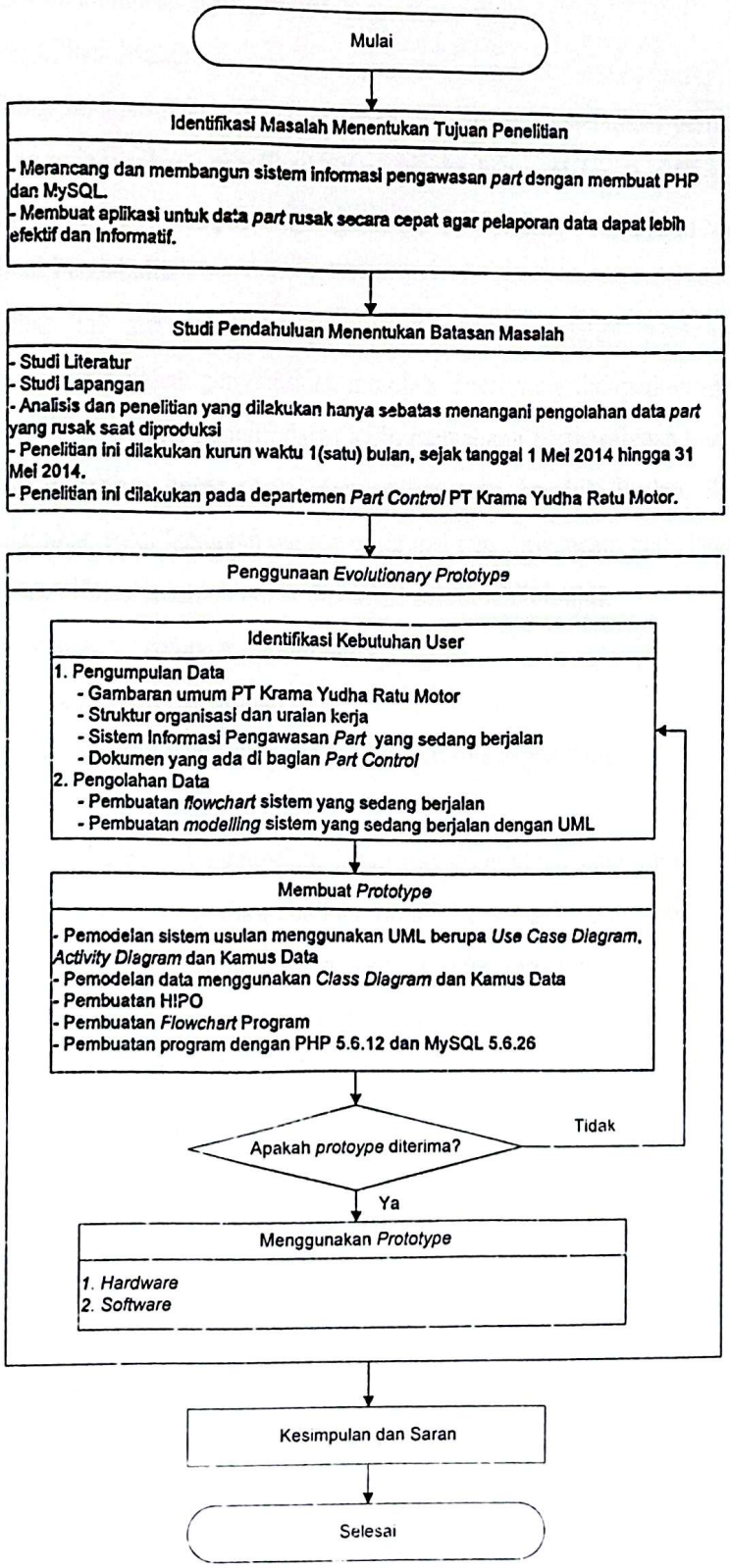
Metode pengembangan Sistem Informasi Pengawasan Part ini menggunakan metode model *prototype* jenis *evolutionary*. Model *prototype* cocok digunakan untuk menjabarkan kebutuhan *user* secara lebih detail karena *user* sering kali kesulitan menyampaikan kebutuhannya secara *detail* tanpa melihat gambaran yang jelas.

Tahapan-tahapan yang dilakukan penulis dalam metode pengembangan *protoype evolutionary* adalah sebagai berikut:

1. Mengidentifikasi kebutuhan *user* terhadap sistem yang akan dibuat. Penulis melakukan observasi bagaimana Sistem Pengawasan Part yang sedang berjalan dan mencoba menggunakan sistem tersebut untuk mengetahui apa yang diminta dan dibutuhkan dari SistenPengawasanPart tersebut.
2. Membuat *prototype* sistem usulan yang diajukan penulis. Dalam membuat *prototype* tersebut, penulis menggunakan *Unified Modelling Language* (UML) dalam merancang sistem usulan tersebut.
3. Menentukan apakah *prototype* dapat diterima. Penulis mendemonstrasikan *prototype* kepada pengguna untuk mengetahui apakah telah sesuai dengan kebutuhan pengguna. Jika iya, akan dilakukan langkah selanjutnya dan jika tidak, *prototype* akan direvisi dari tahap analisis data untuk menganalisis kembali kebutuhan pengguna terhadap sistem yang akan dibangun.
4. Setelah *prototype* diterima, maka tahap selanjutnya akan diserahkan ke pengguna, apakah *prototype* akan diterapkan atau tidak untuk perbaikan

sistem lama. Penulis melakukan pengembangan sistem hanya sampai pada tahap perancangan sistem dan tidak melakukan tahap implementasi sistem.

3.4. Kerangka Peneltian



Gambar III.1 Kerangka Penelitian

(Sumber: Hasil Pengolahan Data, 2015)

Beberapa tahapan yang penulis lakukan dalam melakukan penelitian kualitatif ini adalah sebagai berikut:

1. Identifikasi Masalah

   Tahap ini merupakan tahap mengidentifikasi dan menetapkan permasalahan yang terjadi selama penelitian sistem pemasokan *part* dari Part Control CKD ke line produksi yang sedang berjalan pada PT Krama Yudha Ratu Motor.

2. Studi Pendahuluan

   Tahap ini merupakan tahap awal dalam mempelajari teori-teori yang mendukung dalam penyelesaian masalah. Teori yang didapatkan merupakan langkah awal agar peneliti dapat lebih memahami permasalahan yang sedang diteliti dengan benar sesuai dengan kerangka berpikir ilmiah. Selain itu, dilakukan studi lapangan berupa observasi dan wawancara pada bagian yang teridentifikasi masalah untuk dipelajari secara lebih lanjut.

3. Penggunaan *Prototype Evolutionary*

   a. Identifikasi Kebutuhan *User*

      Identifikasi kebutuhan *user* terdiri dari dua aspek yaitu:

      1) Pengumpulan Data

         Tahap ini merupakan tahap mengumpulkan data yang diperlukan. Data yang dikumpulkan berkaitan dengan gambaran umum PT Krama Yudha Ratu Motor, struktur organisasi dan uraian kerja, sistem Pemasokan Part yang sedang berjalan dan dokumen yang ada pada proses Pemasokan Part.

      2) Pengolahan Data

         Tahap ini merupakan tahap mengolah data yang telah dikumpulkan. Pengolahan data yang dilakukan meliputi pembuatan *flowchart* sistem yang sedang berjalan dan pembuatan *modelling* sistem dengan menggunakan UML.

   b. Membuat *Prototype*

      Tahap ini merupakan tahap pembuatan *prototype*, dimana yang dilakukan adalah pembuatan *modelling* sistem dengan menggunakan UML, pembuatan Kamus Data, pembuatan HIPO, pembuatan program dengan

menggunakan PHP dan MySQL serta pembuatan spesifikasi *hardware* dan *software* program. Program diberi nama Sistem Informasi Pemasokan Part. Pada tahap ini akan dilihat apakah *protoype* yang dibuat dapat diterima oleh *user* atau tidak. Jika tidak, akan dilakukan identifikasi kebutuhan *user* kembali, tetapi jika ya, tahap selanjutnya adalah menggunakan *prototype* tersebut.

4. Kesimpulan dan Saran

Setelah *prototype* sistem usulan diterima maka tahap selanjutnya adalah menarik kesimpulan dari penelitian yang telah dilakukan di PT Krama Yudha Ratu Motor dan memberikan saran yang membangun bagi perusahaan tersebut.



BAB IV
PENGUMPULAN DAN PENGOLAHAN DATA

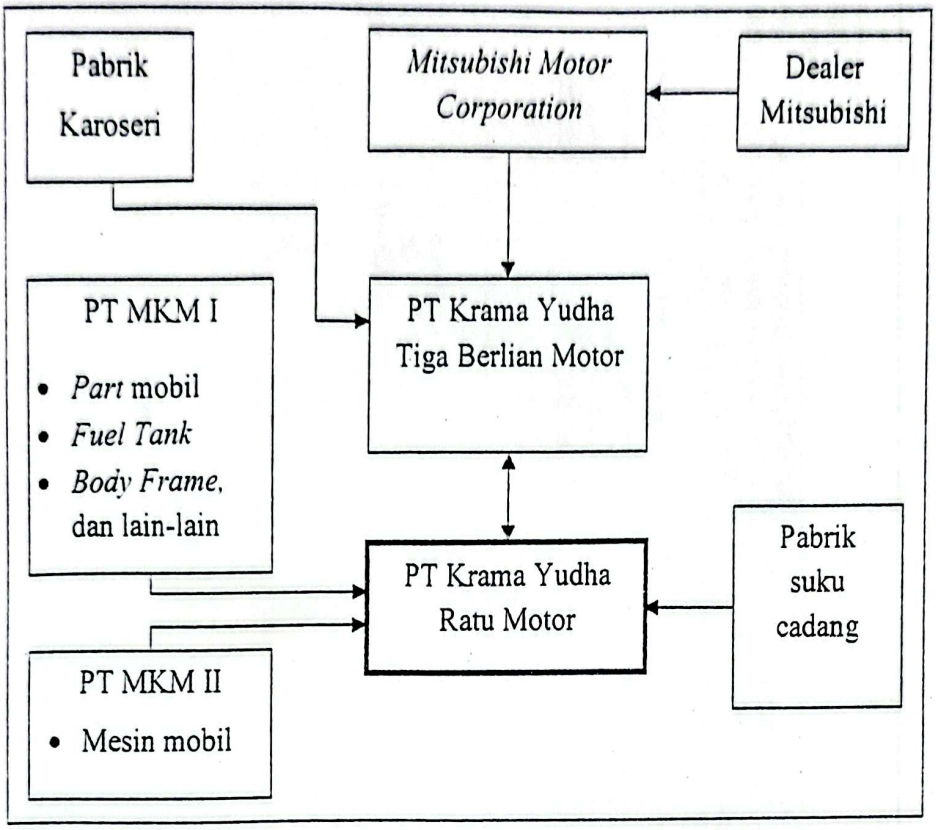
4.1. Hubungan Kerja Sama PT Krama Yudha Ratu Motor

PT Krama Yudha Tiga Berlian yang tergabung dengan *Mitsubishi Motor Corporation (MMC)* merupakan perusahaan induk *(Holding Company)* PT Krama Yudha Ratu Motor.

PT Krama Yudha Ratu Motor terikat kontrak kerja dengan PT Krama yudha Tiga Berlian. PT KTB melakukan pembelian bahan baku berupa *part* kepada *supplier.* Komponen-komponen yang telah di terima oleh PT KTB kemudian diserahkan kepada PT KRM untuk dirakit menjadi kendaraan.

Dalam proses produksinya, PT KRM juga memerlukan bahan-bahan penunjang seperti cat, amplas, bahan bakar minyak, kawat las, dan lain-lain. Pembelian bahan-bahan penunjang dilakukan langsung oleh PT KRM kepada *supplier* yang bersangkutan.

Setelah proses perakitan dan pengujian selesai dilakukan, kendaraan diserahkan kepada PT KTB untuk didistribusikan kepada para *dealer* Mitsubishi. Adapun hubungan kerjasama dapat dijabarkan pada Gambar IV.1 berikut ini.

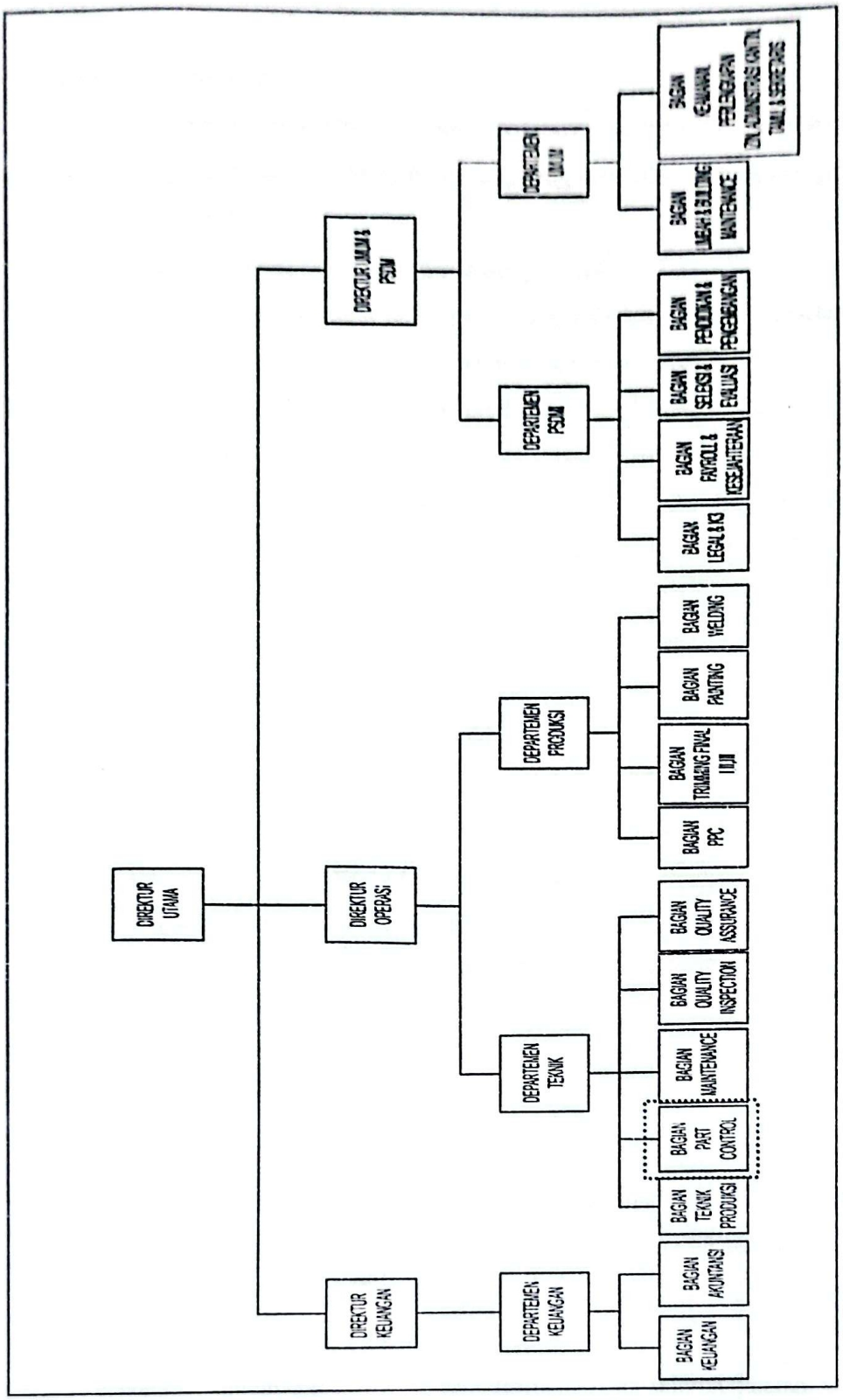


Gambar IV.1 Hubungan Kerja PT Krama Yudha Ratu Motor di dalam dan di luar Krama Yudha Mitsubishi *Group*

(Sumber: PT Krama Yudha Ratu Motor, 2014)

4.2. Struktur Organisasi

Struktur organisasi memegang peranan yang sangat penting dalam mendukung pencapaian tujuan suatu perusahaan. Karena merupakan cetak biru perusahaan, didalamnya menunjukkan dan menjelaskan pembagian tugas, tanggung jawab maupun wewenang setiap bagian yang ada dalam perusahaan, serta sekaligus sebagai alat koordinasi antar bagian, baik secara vertikal maupun secara horizontal. Struktur organisasi dapat dilihat pada Gambar IV.2.



Gambar IV.2 Struktur Organisasi PT Krama Yudha Ratu Motor

(Sumber: PT Krama Yudha Ratu Motor, 2014)

Keterangan Gambar:

Berdasarkan tempat penelitian, yang menjadi fokus penelitian ini adalah bagian Operasi yang dipimpin oleh Direktur Operasi, yang memiliki tugas dan tanggung jawab sebagai berikut:

1. Mengkoordinir kelancaran jalannya proses produksi
2. Mengesahkan rencana kerja masing-masing kepala bagian pada departemen teknik, departemen *quality Control* dan de*part*emen produksi.
3. Bertanggung jawab atas hasil jumlah produksi yang telah ditetapkan.

Dalam melaksanakan tugasnya Direktur Operasi dibantu oleh:

1. De*part*emen Teknik yang membawahi 3 bagian, yaitu Bagian Teknik produksi, Bagian *Maintenance*, Bagian *Material* (*PC Local dan PC CKD*), Bagian *Production Planning and Inventory Control (PPIC).*
2. De*part*emen *Quality Control (QC)* yang membawahi 2 bagian, yaitu Bagian *Quality Assurance*, Bagian *Quality Inspection.*
3. De*part*emen Produksi yang membawahi:
   a. Bagian *Trimming final 1 (Production Unit SL & TD)*
   b. Bagian *Trimming final 2 (Production Unit TD & FMN)*
   c. Bagian *Trimming final 3 (Production Unit CJM)*
   d. Bagian *Trimming final 4 (Production Unit ZC)*
   e. Bagian *Trimming final 5 (Production Unit Mirage/City Car)*
   f. Bagian *Welding*
   g. Bagian *Painting*

4.3. Sistem Manajemen

Aktivitas yang dilakukan di lingkungan PT Krama Yudha Ratu Motor mengacu pada Sistem Manajemen Mutu dan Lingkungan, dengan tata urutannya sebagai berikut:

1. Persyaratan Umum

   Persyaratan umum yang diikuti dan dipenuhi oleh seluruh jajaran yang menjalani kegiatan produksi PT KRM dalam hal sistem manajemen mutu dan lingkungan adalah sebagai berikut:

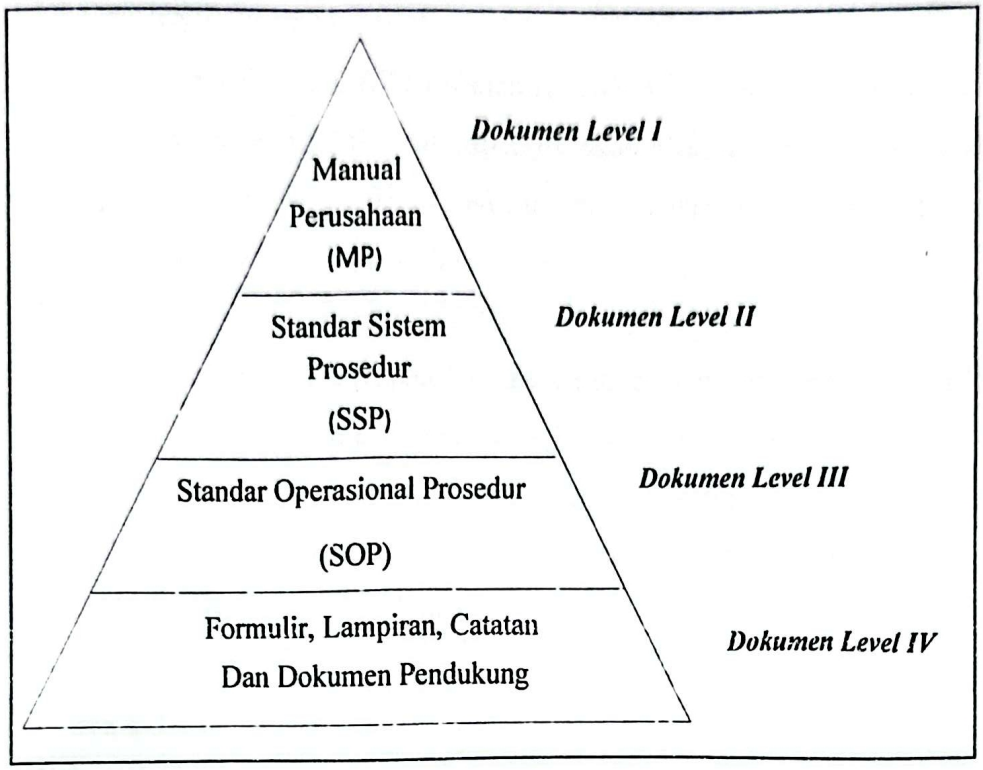
a. Manajemen PT Krama Yudha Ratu Motor (PT KRM) mengembangkan dan mendokumentasikan semua kegiatan sistem manajemen mutu dan lingkungan serta meningkatkan efektivitasnya.

b. Dalam rangka penerapan sistem manajemen mutu dan lingkungan maka manajemen PT KRM:

   1) Membuat proses bisnis perusahaan yang terdiri atas proses bisnis utama dan proses bisnis pendukung.
   2) Menyediakan peralatan, metode tata cara dan penanggung jawab pengendalian proses, serta pengolahan lingkungan.
   3) Menyediakan kebutuhan sumber daya dan informasi.
   4) Menyediakan peralatan, metode tata cara dan penanggung jawab pemantauan, pengukuran dan analisa proses lingkungan.
   5) Melakukan tindakan pencegahan, perbaikan dan peningkatan berkesinambungan (*Continuous Improvement*) dari pelaksanaan proses dan pengendalian lingkungan.

2. Persyaratan Dokumentasi

   Dokumentasi Sistem Manajemen Mutu dan Lingkungan (SMML) yang diterapkan di PT Krama Yudha Ratu Motor terstruktur seperti pada Gambar IV.3 berikut ini.



Gambar IV.3 Dokumen SMML PT Krama Yudha Ratu Motor

(Sumber: PT Krama Yudha Ratu Motor, 2014)

Keterangan Gambar:

1. Manual Perusahaan (MP)

   Manual Perusahaan adalah Dokumen Utama (Level I) yang memberikan informasi mengenai penerapan Sistem Manajemen Mutu dan Lingkungan (SMML) yang dikembangkan dan dipelihara oleh PT Krama Yudha Ratu Motor mencakup:

   a. Ruang lingkup penerapan SMML termasuk keterangan dan pembenaran dari pengecualian penerapan persyaratan ISO 9001:2008 dan ISO 14001:2004.
   b. Prosedur terdokumentasi yang dikembangkan untuk SMML atau Referensinya.
   c. Visi, Misi dan Kebijakan Mutu dan Lingkungan.
   d. Penjelasan dari interaksi antar proses-proses SMML dan Proses Bisnis Perusahaaan.

2. Standar Sistem Prosedur (SSP)

   Standar Sistem Prosedur (SSP) adalah Dokumen yang memberikan informasi mengenai aktivitas SMML yang dipersyaratkan oleh ISO 9001:2008 dan ISO 14001:2004. Standar Sistem Prosedur berisi Diagram Alir Aktivitas dan Penanggung Jawab setiap aktivitas.

3. Standar Operasional Prosedur (SOP)

   Adalah Dokumen yang menjelaskan mengenai cara dan urutan kerja serta hal-hal penting yang harus diperhatikan dan dilakukan *(Key Point)*.

4. Formulir, Lampiran, Catatan, dan Dokumen Pendukung

   Adalah Dokumen yang digunakan untuk membuktikan pelaksanaan aktivitas telah dilaksanakan sesuai SSP dan SOP.

4.4. Jenis-Jenis Produk

Ada beberapa jenis kendaraan niaga dan mobil *passanger* yang di rakit di PT Krama Yudha Ratu Motor yang kemudian diklasifikasikan menjadi beberapa *type* model kendaraan antara lain sebagai berikut:

Tabel IV.1 Jenis-Jenis Produk PT Krama Yudha Ratu Motor

Model Kendaraan	*Type*	Spesifikasi	Berat Kotor Kendaraan (Ton)
CJM	CJ-M G-PU	78 PS/ 1343 cc	1,7
	CJ-M G-MB	78 PS/ 1343 cc	1,7
SL	SL D-PU	72 PS/ 2500 cc	2,5
	SL D-MB	72 PS/ 2500 cc	2,5
Truck Diesel (TD)	*Colt Diesel* FE-304	100 PS/ 3298 cc	5
	Colt Diesel FE-334	100 PS/ 3298 cc	7
	Colt Diesel FE-334 HD	100 PS/ 3298 cc	7
	Colt Diesel FE-347	135 PS/ 4214 cc	8
	Colt Diesel FE-349	120 PS/ 3907 cc	7,5
	Colt Diesel FE-3349 HD	120 PS/ 3907 cc	7,5
	Colt Diesel FE-447	135 PS/ 4214 cc	8

(Sumber: Data Historis PT Krama Yudha Ratu Motor, 2014)

Tabel IV.1 Jenis-Jenis Produk PT Krama Yudha Ratu Motor (Lanjutan)

Model Kendaraan	*Type*	Spesifikasi	Berat Kotor Kendaraan (Ton)
Fuso	*Fuso* FM-517H	190 PS/ 7545 cc	14
	Fuso FM-517	190 PS/ 7545 cc	21
	Fuso FN-527	220 PS/ 7545 cc	23,5
	Fuso FN-527 HD	220 PS/ 7545 cc	23,5
Outlander Sport	*Outlander Sport* GLX	150 PS/2000 cc	1,405

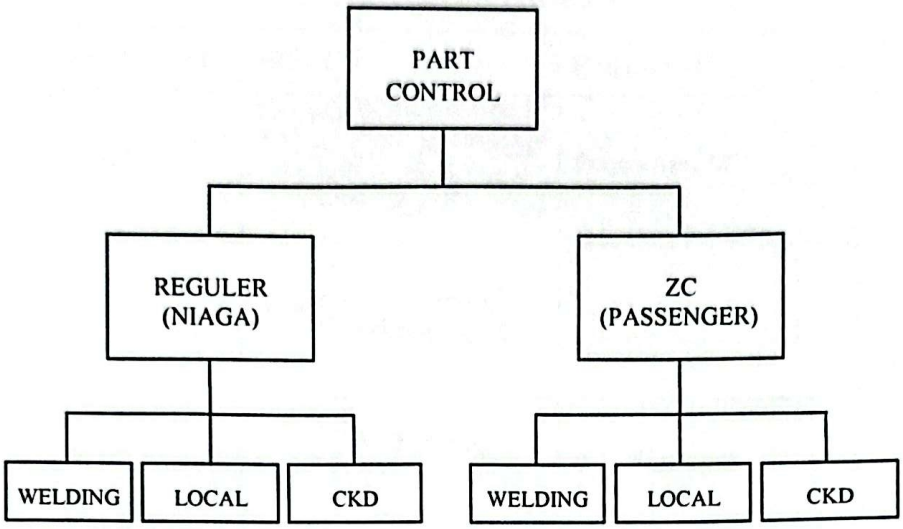
(Sumber: Data Historis PT Krama Yudha Ratu Motor, 2014)

4.5. *Part Control*

Part Control merupakan kegiatan menerima barang yang datang dari *supplier* dan mengirim kembali secara bertahap ke *line* produksi sesuai dengan jumlah dan yang *type* yang sudah ditentukan untuk dirakit menjadi kendaraan.

4.5.1. Sistem *Part Control*

Sistem *Part Control* pada PT Krama Yudha Ratu Motor dibagi atas dua tipe yaitu *Part Control* tipe *regular* (niaga) dan *Part Control* tipe ZC (*passenger*). *Part Control* bagian *regular* menangani segala kebutuhan dan pengiriman *part* untuk kendaraan-kendaraan niaga jenis CJM, SL, *Truck Diesel,* dan Fuso. Sedangkan *Part Control* pada tipe ZC menangani untuk kendaraan-kendaraan pribadi seperti *Outlander Sport.*



Gambar IV.4 Bagian *Part Control*

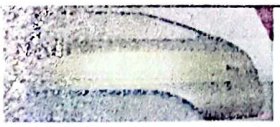
(Sumber: PT Krama Yudha Ratu Motor, 2014)

4.5.2. Jenis *Part*

Part yang terdapat pada masing-masing bagian *Part Control* berbeda sesuai dengan jenis kendaraan yang diproduksi. Contoh *part* dapat dilihat pada tabel berikut ini:

1. CJM

Tabel IV.2 Jenis *Part* Kendaraan CJM

PART CODE	SKETCH	*PART* NAME
TB-018		**Panel A Syy Front Door Rh**
TB-021		**Bracket Front Bumper Ctr**
TB-023		**Reinf Front Pilar Pedal**

(Sumber: PT Krama Yudha Ratu Motor, 2014)

2. SL

Tabel IV.3 Jenis *Part* Kendaraan SL

PART CODE	SKETCH	*PART* NAME
D-003		Frame Assy MB
D-010		Post Side Lh
D-080		Bek. Bummper

(Sumber: PT Krama Yudha Ratu Motor, 2014)

3. Truck Diesel

Tabel IV.4 Jenis *Part* Kendaraan Truck Diesel

PART CODE	SKETCH	*PART* NAME
TDP-002		MUFLER
TDP-008		PIPE AIR
TDP-0[illegible]		BRACKET FUEL TANK REAR

(Sumber: PT Krama Yudha Ratu Motor, 2014)

4. Fuso

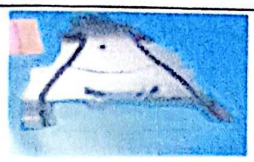
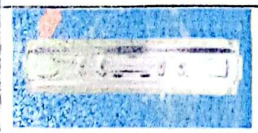
Tabel IV.5 Jenis *Part* Kendaraan Fuso

PART CODE	SKETCH	*PART* NAME
U-02		Dash Board Assy
U-031		Panel Back inner Lh
U-036		Fender.Sd.Lower Assy Lh

(Sumber: PT Krama Yudha Ratu Motor, 2014)

5. Outlander Sport

Tabel IV.6 Jenis *Part Outlander Sport*

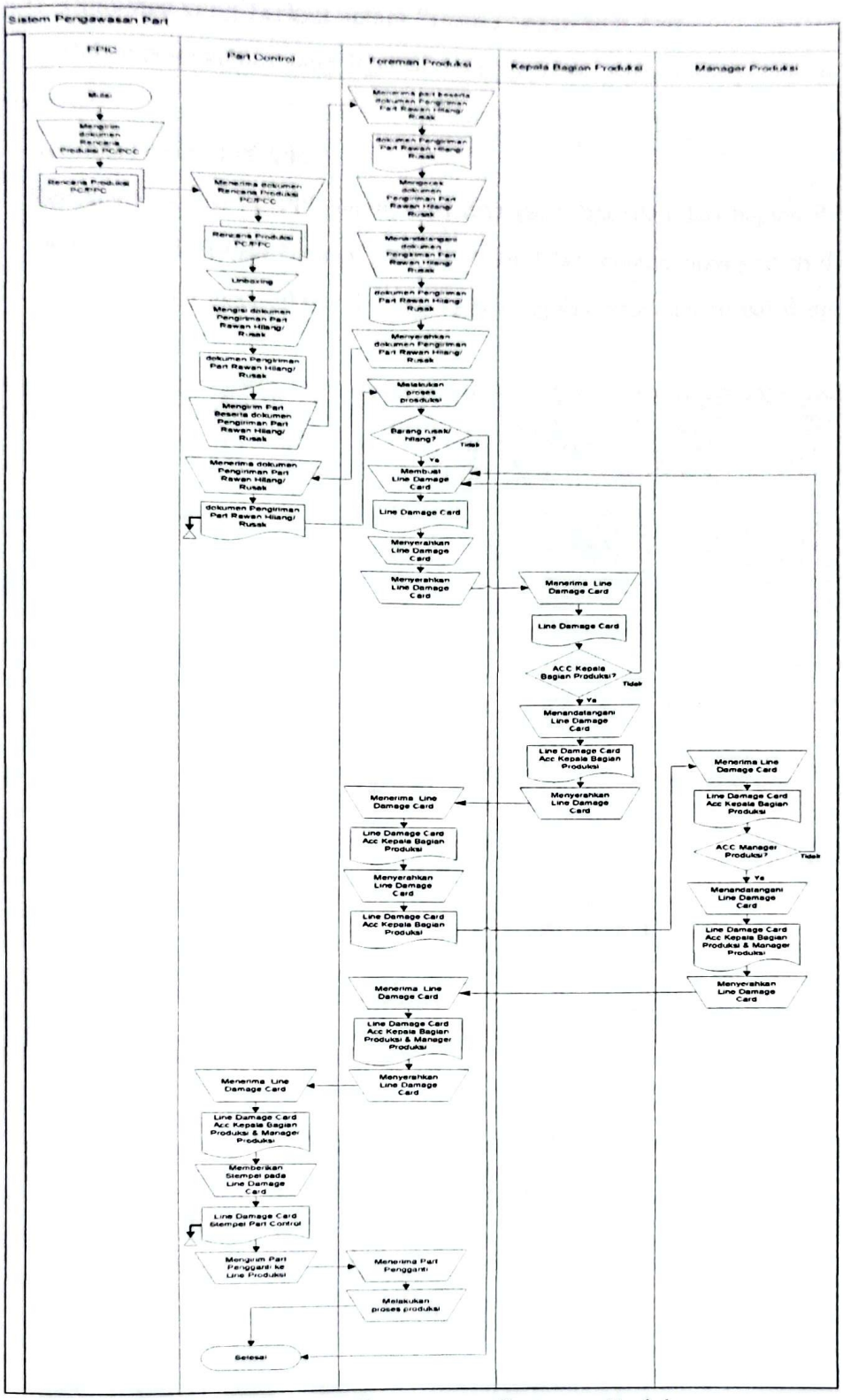
PART CODE	SKETCH	*PART* NAME
5256B082		Stay Assy, Hood Lock
52909[illegible]		Bow Assy, Roof, CTR
MN15[illegible]		BRKT, Battery, FR

(Sumber: PT Krama Yudha Ratu Motor, 2014)

4.6. Proses Pengawasan *Part* yang Sedang Berjalan

Pada PT Krama Yudha Ratu Motor proses yang digunakan untuk pengawasan *part* diuraikan sebagai berikut:

1. *Part Control* menentukan jumlah dan tujuan *part* yang akan dikirim sesuai dengan acuan data Rencana Produksi PC/PPC yang diberikan oleh PPIC.
2. *Part Control* mengisi dokumen Pengiriman *Part* berupa nama, nomor,kode, yang telah dikirimkan ke *line* produksi.
3. Bagian produksi membuat *form Line Damage Card* yang berisi data *part* yang rusak atau hilang saat diproduksi.
4. Bagian produksi membuat meminta tanda tangan Kepala Bagian Produksi dan Manager Produksi pada *form Line Damage Card* sebagai tanda persetujuan.
5. Kepala Bagian Produksi dan Manager Produksi memberikan tanda tangan pada *form Line Damage Card* sebagai tanda persetujuan.
6. *Part Control* menerima *form Line Damage Card* dan mengkonfirmasi dengan memberikan stempel untuk kemudian dapat mengirim kembali *part* pengganti ke *line* produksi.



Gambar IV.5 Proses Pengawasan *Part* yang Berjalan

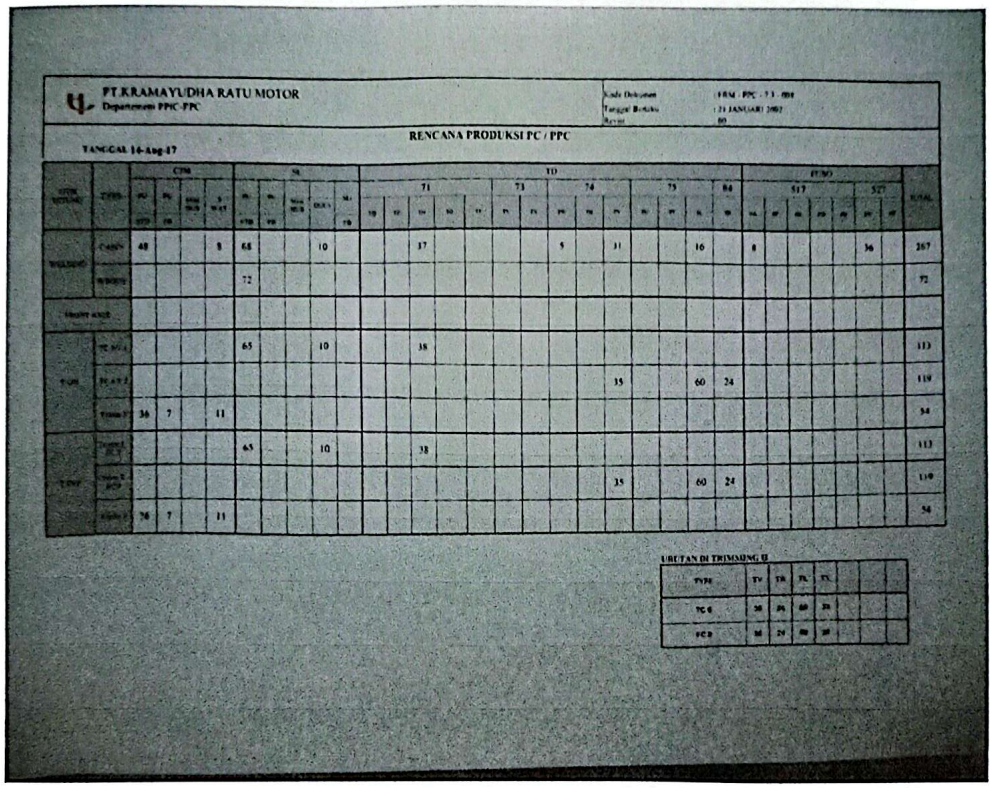
(Sumber: Pengolahan Data, 2017)

4.6.1. Dokumen yang Terkait dalam Proses Pengawasan *Part*

Dokumen-dokumen yang digunakan dalam proses pengawasan *part* ke *line* produksi:

1. Rencana Produksi PC/PPC

   Rencana Produksi PC/PPC merupakan data yang diperoleh dari bagian PPC untuk digunakan *Part Control* sebagai acuan dalam menentukan jumlah dan jenis *part* yang akan dikirim ke masing-masing *line* produksi sesuai dengan tipe kendaraan.



PT.KRAMAYUDHA RATU MOTOR
Departemen PPIC-PPC

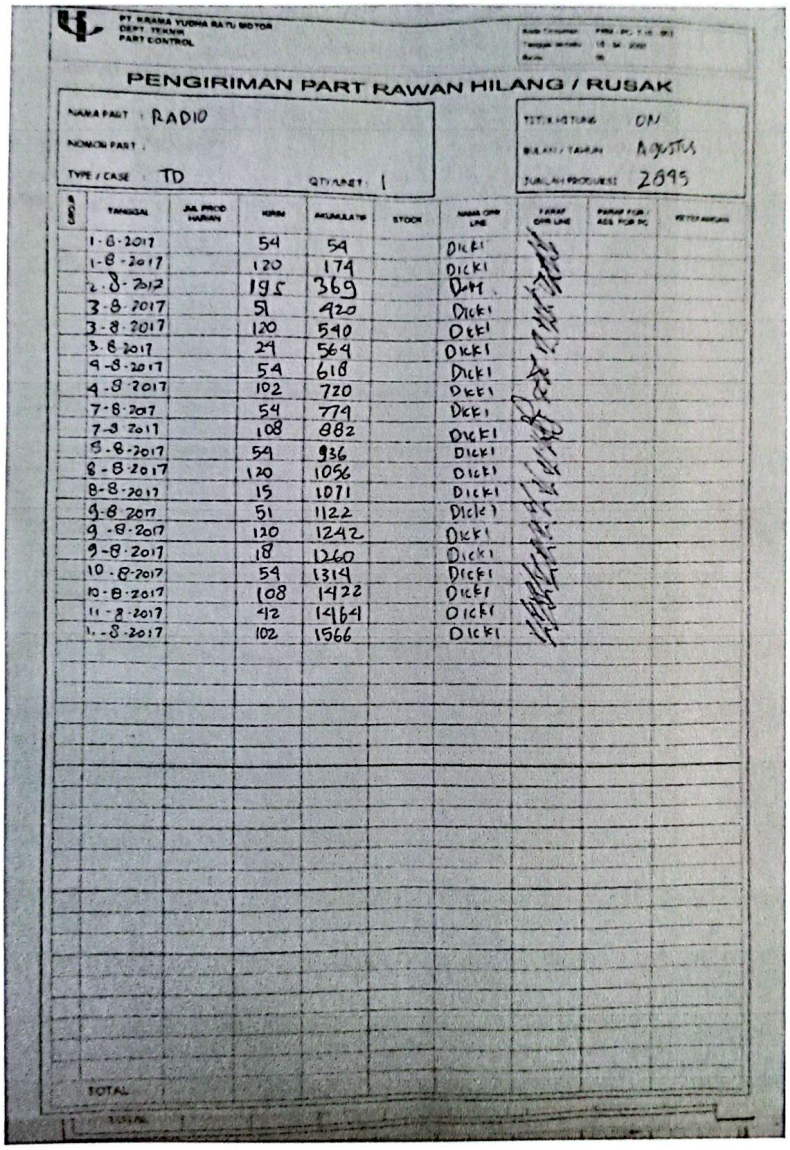
RENCANA PRODUKSI PC / PPC

TANGGAL 16-Aug-17

| | | CPM | | | | SL | | | | | TD | | | | |

2. Pengiriman *Part* Rawan Hilang/Rusak

   Pengiriman *Part* Rawan Hilang/Rusak merupakan dokumen yang berisi data-data pengiriman *part* yang diketahui sering hilang atau rusak pada saat diproduksi.



PT KRAMA YUDHA RATU MOTOR
DEPT. TEKNIK
PART CONTROL

PENGIRIMAN PART RAWAN HILANG / RUSAK

NAMA PART : RADIO
NOMOR PART :
TYPE / CASE : TD QTY/SET : 1

TITIK HITUNG : ON
BULAN / TAHUN : AGUSTUS
JUMLAH PRODUKSI : 2895

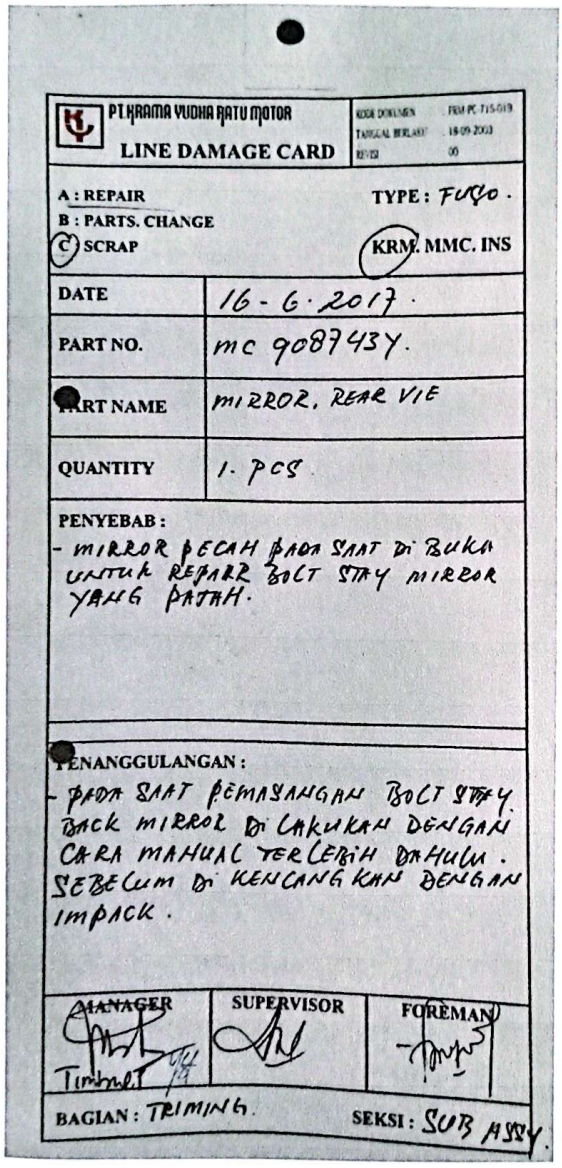
NO	TANGGAL	JML PROD HARIAN	KIRIM	AKUMULATIF	STOCK	NAMA CHIP LINE	PARAF CHIP LINE	PARAF PCH / ASS. PCH PC	KETERANGAN
	1-8-2017		54	54		DICKI			
	1-8-2017		120	174		DICKI			
	2-8-2017		195	369		DICKI			
	3-8-2017		51	420		DICKI			
	3-8-2017		120	540		DICKI			
	3-8-2017		24	564		DICKI			
	4-8-2017		54	618		DICKI			
	4-8-2017		102	720		DICKI			
	7-8-2017		54	774		DICKI			
	7-8-2017		108	882		DICKI			
	8-8-2017		54	936		DICKI			
	8-8-2017		120	1056		DICKI			
	8-8-2017		15	1071		DICKI			
	9-8-2017		51	1122		DICKI			
	9-8-2017		120	1242		DICKI			
	9-8-2017		18	1260		DICKI			
	10-8-2017		54	1314		DICKI			
	10-8-2017		108	1422		DICKI			
	11-8-2017		42	1464		DICKI			
	11-8-2017		102	1566		DICKI			
	TOTAL								

Gambar IV.7 Dokumen Pengiriman *Part* Rawan Hilang/Rusak

(Sumber: PT Krama Yudha Ratu Motor, 2014)

3. *Line Damage Card*

Form Line Damage Card merupakan data *part* yang rusak saat sedang diproduksi berupa nomor, nama, tipe, dan jenis kerusakan *part* serta cara penggulangan yang dilakukan sesuai dengan jenis kerusakan yang terjadi pada *part* tersebut.



PT KRAMA YUDHA RATU MOTOR

LINE DAMAGE CARD

KODE DOKUMEN : FRM PC-TLS-019
TANGGAL BERLAKU : 18-09-2013
REVISI : 00

A : REPAIR
B : PARTS. CHANGE
C SCRAP

TYPE : FUSO .
KRM. MMC. INS

DATE	16 - 6 . 2017 .
PART NO.	mc 908743Y.
PART NAME	MIRROR, REAR VIE
QUANTITY	1. PCS

PENYEBAB :
- MIRROR PECAH PADA SAAT DI BUKA UNTUK REPAIR BOLT STAY MIRROR YANG PATAH.

PENANGGULANGAN :
- PADA SAAT PEMASANGAN BOLT STAY BACK MIRROR DI LAKUKAN DENGAN CARA MANUAL TERLEBIH DAHULU. SEBELUM DI KENCANGKAN DENGAN IMPACK .

MANAGER	SUPERVISOR	FOREMAN
Timbul		

BAGIAN : TRIMING. SEKSI : SUB ASSY.

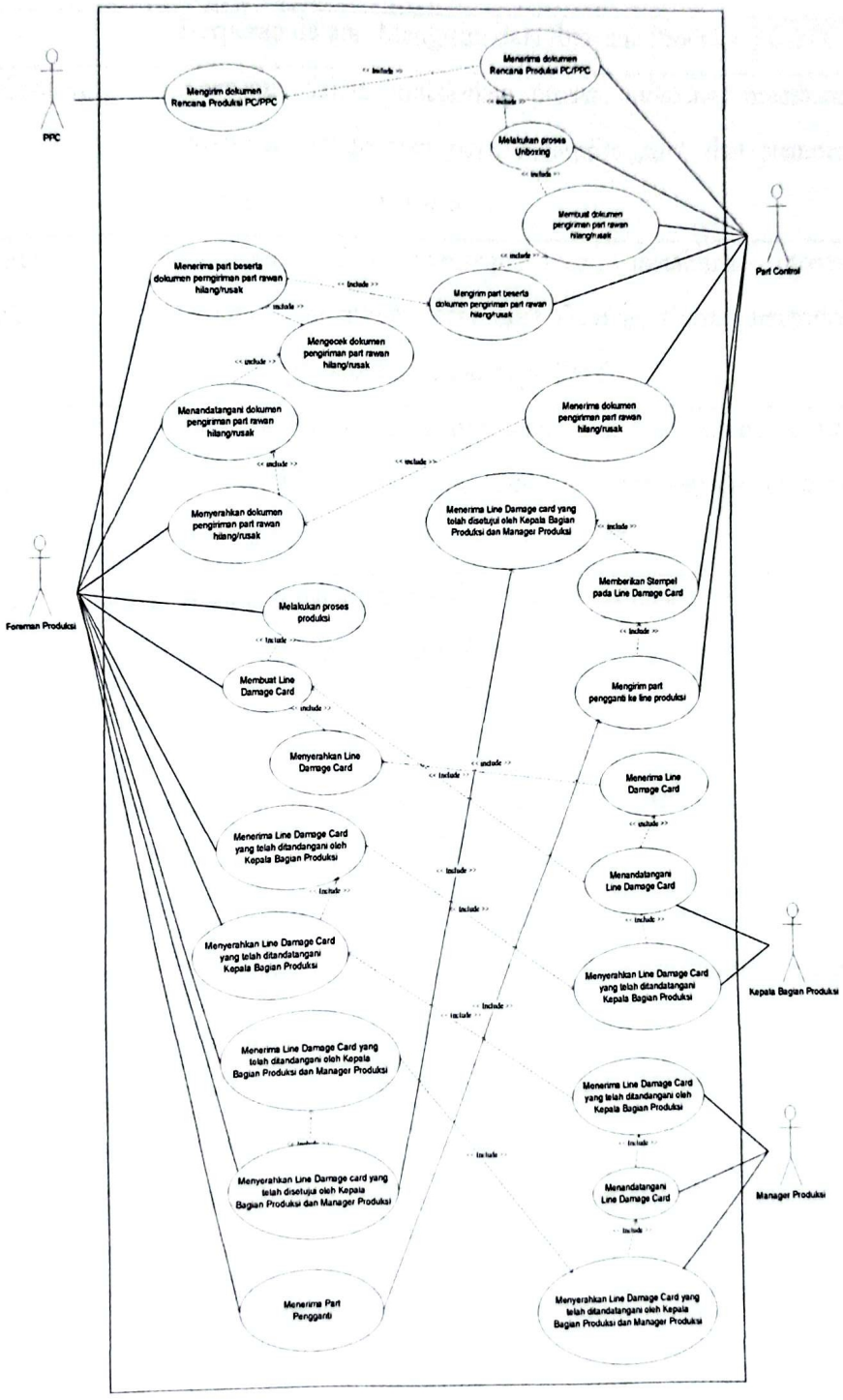
Gambar IV.8 *Form Line Damage Card*

(Sumber: PT Krama Yudha Ratu Motor, 2014)

4.7. Sistem Pengawasan *Part* dengan *Unified Modelling Language* (UML)

Penggambaran *Unified Modelling Language* (UML) Sistem Pengawasan *Part* pada PT Krama Yudha Ratu Motor dengan menggunakan *Use Case Diagram* dan *Activity Diagram* adalah sebagai berikut :

1. *Use Case Diagram*



Gambar IV.9 Use Case Diagram SistemPengawasan *Part* yang Berjalan

(Sumber: Pengolahan Data, 2017)

Tabel IV.7 Definisi *Actor* dalam Proses Pengawasan *Part*

Actor	Keterangan
PPIC	Berperan dalam Mengirim data Rencana Produksi PC/PPC
Part Control	Berperan dalam melakukan proses *unboxing*, membuat dokumen pengiriman *part*, mengirim *part*, dan stempel *form Line Damage Card.*
Foreman Produksi	Berperan dalam menerima *part*, melakukan proses produksi, membuat *form Line Damage Card*, menerima dan menyerahkan *Line Damage Card*
Kepala Bagian Produksi	Berperan dalam menandatangani *form Line Damage Card*, menyerahkan *form Line Damage Card* kepada *foreman* produksi.
Manager Produksi	Berperan dalam menandatangani *form Line Damage Card*, menyerahkan *form Line Damage Card*

(Sumber: Hasil Pengolahan Data, 2017)

Tabel IV.8 Definisi *Use Case* dalam Proses Pengawasan *Part*

Use Case	Keterangan
Mengirim Dokumen Rencana Produksi PC/PPC	Merupakan kegiatan pengiriman dokumen Rencana Produksi PC/PPC sebagai acuan *Part Control* dalam pengiriman *part*. Kegiatan ini dilakukan oleh *Actor* PPIC.
Menerima Dokumen Rencana Produksi PC/PPC	Merupakan kegiatan penerimaan dokumen Rencana Produksi PC/PPC sebagai acuan *Part Control* dalam pengiriman *part*. Kegiatan ini dilakukan oleh *Actor Part Control.*
Unboxing	Merupakan kegiatan membongkar *part* dari gudang *Part Control* untuk dikelompokan sesuai jumlah dan jenis *part* sebelum dikirim ke *line* produksi. Kegiatan ini dilakukan oleh *Actor Part Control.*

(Sumber: Hasil Pengolahan Data, 2017)

Tabel IV.8 Definisi *Use Case* dalam Proses Pengawasan *Part* (Lanjutan I)

Use Case	Keterangan
Membuat Laporan Pengiriman Part/Rawan Hilang atau Rusak	Merupakan kegiatan membuat laporan data *part* yang dikirim ke line produksi. Kegiatan ini dilakukan oleh *Actor Part Control.*
Mengirim *part* beserta Laporan Pengiriman *Part* Rawan Hilang/Rusak	Merupakan kegiatan pengiriman *part* ke line produksi beserta Laporan Pengiriman Part Rawan Hilang/Rusak. Kegiatan ini dilakukan oleh *Actor Part Control.*
Menerima *part* beserta Laporan Pengiriman Part Rawan Hilang/Rusak	Merupakan kegiatan penerimaan *part* beserta Laporan Pengiriman Part Rawan Hilang/Rusak yang dikirim dari *Part Control.* Kegiatan ini dilakukan oleh *Actor Part Control*
Menandatangani Laporan Pengiriman Part Rawan Hilang/Rusak	Merupakan kegiatan menandatangani Laporan Pengiriman Part Rawan Hilang/Rusak. Kegiatan ini dilakukan oleh *Actor Foreman* Produksi.
Menyerahkan Laporan Pengiriman Part Rawan Hilang/Rusak	Merupakan kegiatan mengembalikan Laporan Pengiriman Part Rawan Hilang/Rusak yang telah ditandatangani *Foreman* Produksi kepada *Part Control.* Kegiatan ini dilakukan oleh *Actor Foreman* Produksi.
Menerima Laporan Pengiriman Part Rawan Hilang/Rusak	Merupakan kegiatan menerima Laporan Pengiriman *Part* Rawan Hilang/Rusak yang telah ditandatangani *Foreman.* Kegiatan ini dilakukan oleh *Actor Foreman* Produksi.

(Sumber: Hasil Pengolahan Data, 2017)

Tabel IV.8 Definisi *Use Case* dalam Proses Pengawasan *Part* (Lanjutan II)

Use Case	Keterangan
Melakukan Proses Produksi	Merupakan kegiatan melakukan perakitan *part* menjadi kendaraan. Kegiatan ini dilakukan oleh *Foreman* Produksi.
Membuat *Line Damage Card*	Merupakan kegiatan pembuatan *form* informasi temuan *part* yang rusak saat dipoduksi berupa jenis dan cara penanggulangan kerusakan. Kegiatan ini dilakukan oleh *Actor Foreman* Produksi..
Menyerahkan *Line Damage Card*	Merupakan kegiatan penyerahan form *Line Damage Card* untuk ditandatangani. Kegiatan ini dilakukan oleh *Actor Foreman* Produksi, Kepala Bagian Produksi, Manager Produksi.
Menerima *Line Damage Card*	Merupakan kegiatan menerima form *Line Damage Card* yang telah ditandatangani. Kegiatan ini dilakukan oleh *Actor Foreman* Produksi, Kepala Bagian Produksi, Manager Produksi.
Menandatangani *Line Damage Card*	Merupakan kegiatan penyetujuan form Line Damage Card yang dibuat oleh foreman produksi. Kegiatan ini dilakukan oleh *Actor* Kepala Bagian Produksi dan Manager Produksi.
Stempel *Line Damage Card*	Merupakan kegiatan pemberian stempel pada *form Line Damage Card* sebagai persetujuan untuk mengeluarkan *part* baru sebagai pengganti *part* rusak. Kegiatan ini dilakukan oleh *Actor Part Control.*
Mengirim *part* pengganti	Merupakan kegiatan pengiriman *part* pengganti ke line produksi Kegiatan ini dilakukan oleh *Actor Part Control.*
Menerima *part pengganti*	Merupakan kegiatan penerimaan *part* pengganti yang dikirim dari *Part Control.* Kegiatan ini dilakukan oleh *Actor Foreman* Produksi.

(Sumber: Hasil Pengolahan Data, 2017)

BAB V
ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, dibutuhkan perancangan sistem informasi pengawasan *part* yang terintegrasi sebagai sarana untuk memberikan informasi data *part* sebagai bahan baku produksi PT Krama Yudha Ratu Motor. Perancangan aplikasi sistem informasi pengawasan *part* menampilkan data yang berhubungan dengan informasi data pengiriman *part* serta data *part* yang rusak.

Berikut kebutuhan sistem yang diidentifikasi untuk sistem informasi pengawasan *part* pada PT Krama Yudha Ratu Motor adalah sebagai berikut:

Tabel V.1 Identifikasi Kebutuhan Sistem

Kebutuhan Sistem	
Nama Proyek	Sistem Informasi Pengawasan *Part*
Proses Sistem Berjalan	1. Terwujudnya sebuah sistem yang membantu proses *Part Control* dalam mengawasi pengiriman maupun penggantian *Part.*
Data Masukan (*input*)	1. Data User 2. Data *Part* 3. Data Kendaraan 4. Dokumen Perencanaan Produksi PC/PPC
Data Keluaran *(output)*	1. Laporan Pengiriman *Part* Rawan Hilang/Rusak
Model Sistem	Input : Data Master Proses : Pengolahan Data Master Output : Laporan Pengiriman *Part*

(Sumber: Hasil Pengolahan Data 2017)

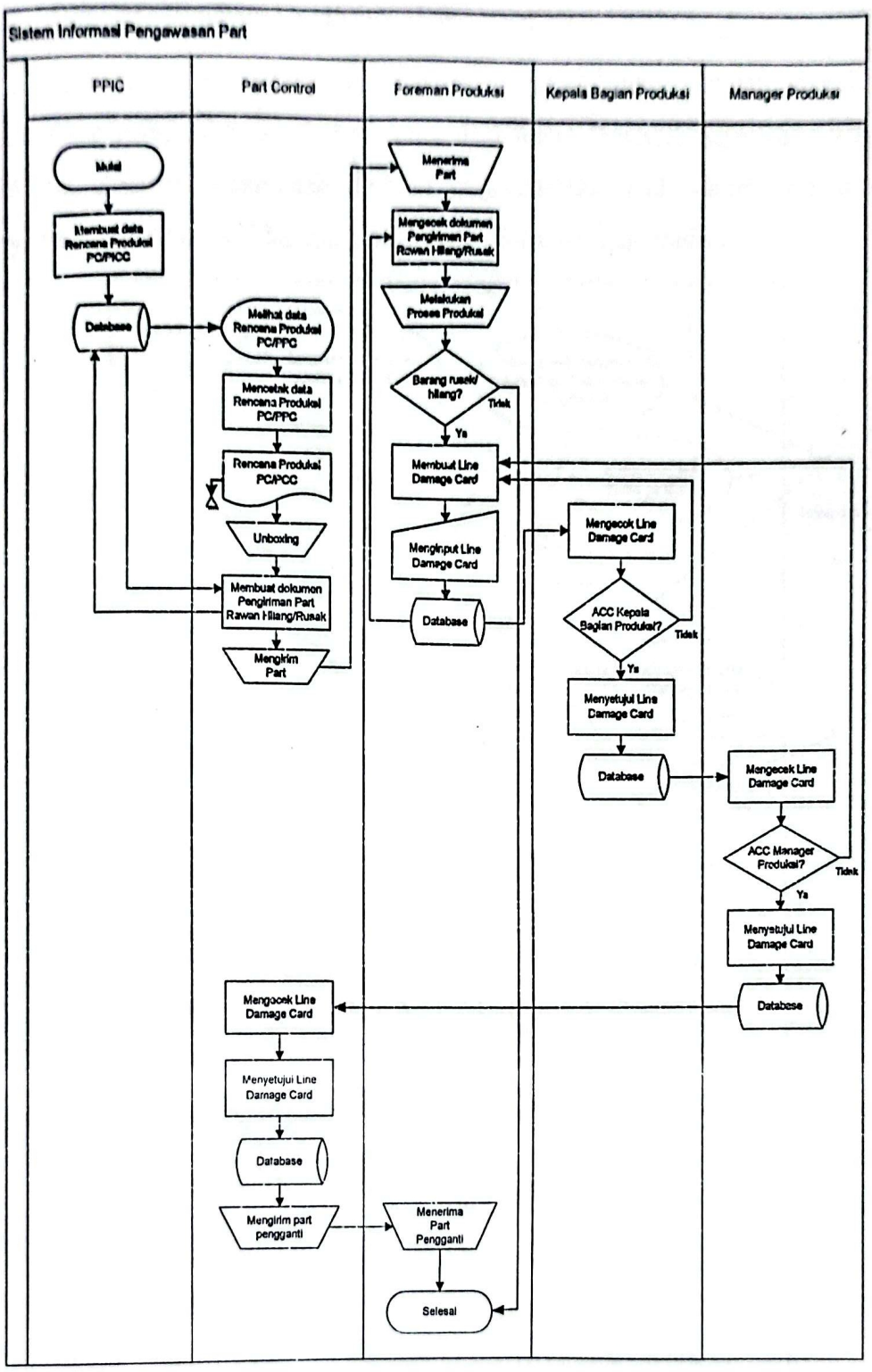
5.2 Perancangan Sistem

Tahap ini adalah tahap kedua pada metodologi *prototype evolutioner* yaitu membuat *prototype* berupa prototipe sistem setelah sebelumnya melakukan tahap mengidentifikasi kebutuhan. Perancangan proses sistem informasi pengawasan *part* menggunakan *modeling system* UML (*unified modeling language*) yang meliputi pembuatan *use case diagram, activity diagram, sequence diagram, class diagram, deployment diagram* dan juga pembuatan kamus data. Tahapan perancangan ini akan memberikan gambaran mengenai aliran informasi dan proses perancangan sistem informasi pengawasan *part* yang diusulkan. Prototipe sistem informasi pengawasan *part* yang telah selesai akan digunakan oleh pengguna untuk diketahui kekurangannya agar dapat diperbaiki.

5.2.1 Perancangan Sistem *Flowmap* Usulan

Alur proses sistem yang diusulkan berisi perancangan proses sistem. Perancangan dibuat untuk meminimalkan kekurangan dari sistem lama yang kurang efektif dan efisien ke sistem baru yang lebih terprogram dan cepat.

Berikut adalah perancangan alur proses sistem informasi pengawasan *part* yang diusulkan:

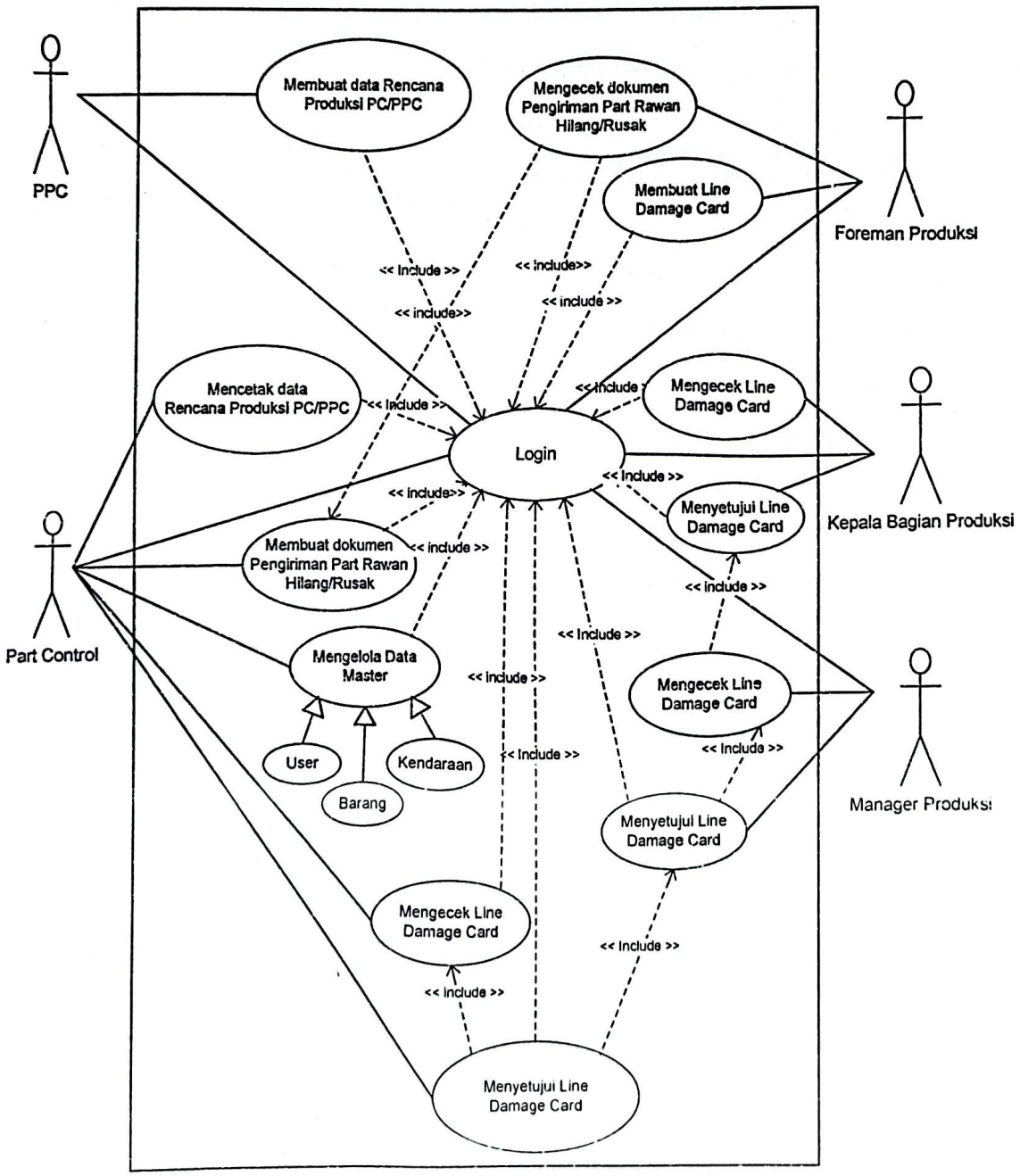


Gambar V.1 *Flowmap* Sistem Informasi Pengawasan *Part* yang Diusulkan

(Sumber: Hasil Pengolahan Data 2017)

5.2.2 *Use Case* Diagram yang Diusulkan

Use case diagram digunakan untuk mendeskripsikan sebuah interaksi antara aktor dengan sistem informasi yang akan dibuat. Berdasarkan analisis yang telah dilakukan maka *use case diagram* yang diusulkan pada sistem informasi pengawasan *part* pada PT Krama Yudha Ratu Motor sebagai berikut:



Gambar V.2 *Use Case* Diagram yang Diusulkan

(Sumber: Hasil Pengolahan Data (2017)

5.2.3 *Use Case Description* Sistem Informasi Pengawasan *Part* yang Diusulkan

Use Case Description merupakan gambaran *Use Case* yang berisi nama *Use Case, primary actor, brief description, relationship, normal flow of events,* dan *precondition.* Berikut ini adalah *Use Case Description* dari *Use Case diagram* dalam Gambar V.2:

Tabel V.2 Definisi *Actor Use Case* Sistem Pengawasan *Part*

Actor	Keterangan
PPIC	Berperan dalam Membuat data Rencana Produksi PC/PPC
Part Control	Bertugas dalam mencetak dokumen pengiriman *part,* mengelola data master, dan meyetujui *Line Damage Card.*
Foreman Produksi	Bertugas dalam mengecek dokumen pengiriman *part,* membuat *Line Damage Card* yang berisi data *part* rusak.
Kepala Bagian Produksi	Berperan dalam mengecek dan memberikan persetujuan pada data *part* rusak
Manager Produksi	Berperan dalam mengecek dan memberikan persetujuan pada data *part* rusak

(Sumber: Hasil Pengolahan Data, 2017)

Tabel V.3 *Use Case Description Login*

Nama *Use Case*	***Login***
Primary Actor	*PPC, Part Control, Foreman* Produksi, Kepala Bagian Produksi, *Manager* Produksi.
Brief Description	*Use Case* ini menggambarkan proses aktor masuk ke dalam sistem.
Relationship	-
Normal Flow of Events	1. Aktor membuka menu *Login.* 2. Aktor memasukkan *user name* dan *password* pada *form login.*

(Sumber: Hasil Pengolahan Data 2017)

Tabel V.3 *Use Case Description Login* (lanjutan)

Nama *Use Case*	***Login***
Normal Flow of Events	3. Sistem mengecek ke *database*, apakah *user name* dan *password* sudah benar. 4. Jika *user name* dan *password* benar (valid), akan mucul tampilan menu utama sistem. 5. Jika *user name* dan *password* salah (tidak valid), maka muncul *message* "Maaf, *user name* dan *password* Anda SALAH!" dan kembali ke *Login Form*.

(Sumber: Hasil Pengolahan Data 2017)

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

Nama *Use Case*	**Mengelola Data Master**
Primary Actor	*Part Control*
Brief Description	Proses pengolahan data *User*, Barang, dan Kendaraan seperti menambah, mengubah, dan menghapus data.
Relationship	-
Normal Flow of Events	*1. Part Control* membuka menu Data Master 2. Pilih Data Master 3. Klik Tambah Data untuk menambah data. 4. Klik Ubah untuk mengubah data. 5. Klik Hapus untuk menghapus data.
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data 2017)

Tabel V.5 *Use Case Description* Master *User*

Nama *Use Case*	**Master *User***
Primary Actor	*Part Control*

(Sumber: Hasil Pengolahan Data, 2017)

Tabel V.5 *Use Case Description* Master *User* (lanjutan)

Nama *Use Case*	**Master *User***
Brief Description	*Part Control* dapat melakukan Tambah, Hapus dan Ubah terhadap data Master *User*.
Relationship	-
Normal Flow of Events	1. *Part Control* membuka menu Data User 2. PilihTambah untuk menambah data *user*. 3. Pilih Ubah untuk mengubah data *user*. 4. Pilih Hapus untuk menghapus data *user*. 5. Klik tombol Simpan untuk menyimpan data.
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data, 2017)

Tabel V.6 *Use Case Description* Master Barang

Nama *Use Case*	**Master Barang**
Primary Actor	*Part Control*
Brief Description	*Part Control* dapat melakukan Tambah, Hapus, dan Ubah terhadap data Master Barang
Relationship	-
Normal Flow of Events	1. *Part Control* membuka menu Data *Part* 2. Pilih Tambah untuk menambah data *part*. 3. Pilih Ubah untuk mengubah data *part*. 4. Pilih Hapus untuk menghapus data *part*. 5. Klik tombol Simpan untuk menyimpan data.
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data, 2017)

Tabel V.7 *Use Case Description* Master Kendaraan

Nama *Use Case*	**Master Kendaraan**
Primary Actor	*Part Control*

(Sumber: Hasil Pengolahan Data, 2017)

Tabel V.7 *Use Case Description* Master Kendaraan (lanjutan)

Nama *Use Case*	**Master Kendaraan**
Primary Actor	*Part Control*
Brief Description	*Part Control* dapat melakukan Tambah, Hapus, dan Ubah terhadap data Master Kendaraan.
Relationship	-
Normal Flow of Events	1. *Part Control* membuka menu Data Kendaraan. 2. PilihTambah untuk menambah data kendaraan. 3. Pilih Ubah untuk mengubah data kendaraan. 4. Pilih Hapus untuk menghapus data kendaraan. 5. Klik tombol Simpan untuk menyimpan data.
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data, 2017)

Tabel V.8 *Use Case Description* Membuat Data Rencana Produksi PC/PPC

Nama *Use Case*	**Membuat Data Rencana Produksi PC/PPC**
Primary Actor	PPIC
Brief Description	PPIC membuat data rencana produksi sebagai data acuan untuk menentukan *part* yang akan dikirim ke *line* produksi
Relationship	-
Normal Flow of Events	1. PPIC membuka menu Rencana Produksi 2. Mencari data kendaraan yang akan diproduksi dengan menggunakan id kendaraan 3. Menginput jumlah kendaraan yang akan diproduksi sesuai tipe kendaraan dan *line* produksi. 4. Klik tombol Simpan untuk menyimpan data.
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data2017)

Tabel V.9 *Use Case Description* Mencetak Data Rencana Produksi PC/PPC

Nama *Use Case*	Membuat Data Rencana Produksi PC/PPC
Primary Actor	*Part Control*
Brief Description	*Part Control* mencetak data Rencana Produksi PC/PPC yang telah dibuat oleh PPIC
Relationship	*Inlcude:* Membuat Data Rencana Produksi PC/PPC
Normal Flow of Events	1. *Part Control* membuka menu Rencana Produksi 2. Mencetak data menjadi bentuk dokumen kertas
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data2017)

Tabel V.10 *Use Case Description* Membuat Laporan Pengiriman *Part* Rawan Hilang/Rusak

Nama *Use Case*	**Membuat Laporan Pengiriman *Part* Rawan Hilang/Rusak**
Primary Actor	*Part Control*
Brief Description	*Part Control* membuat laporan data *part* yang dikirim ke *line* produksi.
Relationship	-
Normal Flow of Events	1. *Part Control* membuka menu Data *Part* 2. Menginput data *part* yang dikirim ke *line* produksi. 3. Klik tombol Simpan untuk menyimpan data.
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data2017)

Tabel V.11 *Use Case Description* Mengecek Laporan Pengiriman *Part* Rawan Hilang/Rusak

Nama *Use Case*	**Mengecek Laporan Pengiriman *Part* Rawan Hilang/Rusak**
Primary Actor	*Foreman* Produksi
Brief Description	*Part Control* membuat laporan data *part* yang dikirim ke *line* produksi.
Relationship	-
Normal Flow of Events	1. *Part Control* membuka menu Laporan Pengiriman *Part* 2. Mengecek data *part* yang diterima
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data2017)

Tabel V.12 *Use Case Description* Membuat *Line Damage Card*

Nama *Use Case*	**Membuat *Line Damage Card***
Primary Actor	*Foreman* Produksi
Brief Description	*Foreman produksi* menginput data yang berakitan dengan *part* rusak.
Relationship	-
Normal Flow of Events	1. *Foreman Produksi* membuka menu *Line Damage Card.* 2. Menginput data yang berkaitan dengan *part* rusak. 3. Klik tombol Simpan untuk menyimpan data.
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data2017)

Tabel V.13 *Use Case Description* Mengecek *Line Damage Card*

Nama *Use Case*	**Menyetujui *Line Damage Card***
Primary Actor	Kepala Bagian Produksi, *Manager* Produksi, *Part Control.*
Brief Description	Kepala Bagian Produksi, *Manager* Produksi, dan *Part Control* mengecek data *part* rusak pada *Line Damage Card*
Relationship	*Include*: Membuat *Line Damage Card*
Normal Flow of Events	1. Kepala Bagian Produksi, *Manager* Produksi, dan *Part Control* membuka menu *Line Damage Card.* 2. Mengecek data *part* rusak
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

(Sumber: Hasil Pengolahan Data 2017)

Tabel V.14 *Use Case Description* Menyetujui *Line Damage Card*

Nama *Use Case*	**Menyetujui *Line Damage Card***
Primary Actor	Kepala Bagian Produksi, *Manager* Produksi, *Part Control.*
Brief Description	Menggambarkan proses memberi persetujuan pada data *part* rusak yang dilakukan oleh Kepala Bagian Produksi, *Manager* Produksi, dan *Part Control.*
Relationship	*Include*: Membuat *Line Damage Card*
Normal Flow of Events	1. Kepala Bagian Produksi, *Manager* Produksi, dan *Part Control* membuka menu *Line Damage Card.* 2. Pilih Acc untuk menyetujui atau Tolak untuk menolak persetujuan. 3. Pilih Simpan
Precondition	*Login* ke dalam sistem informasi pengawasan *part*

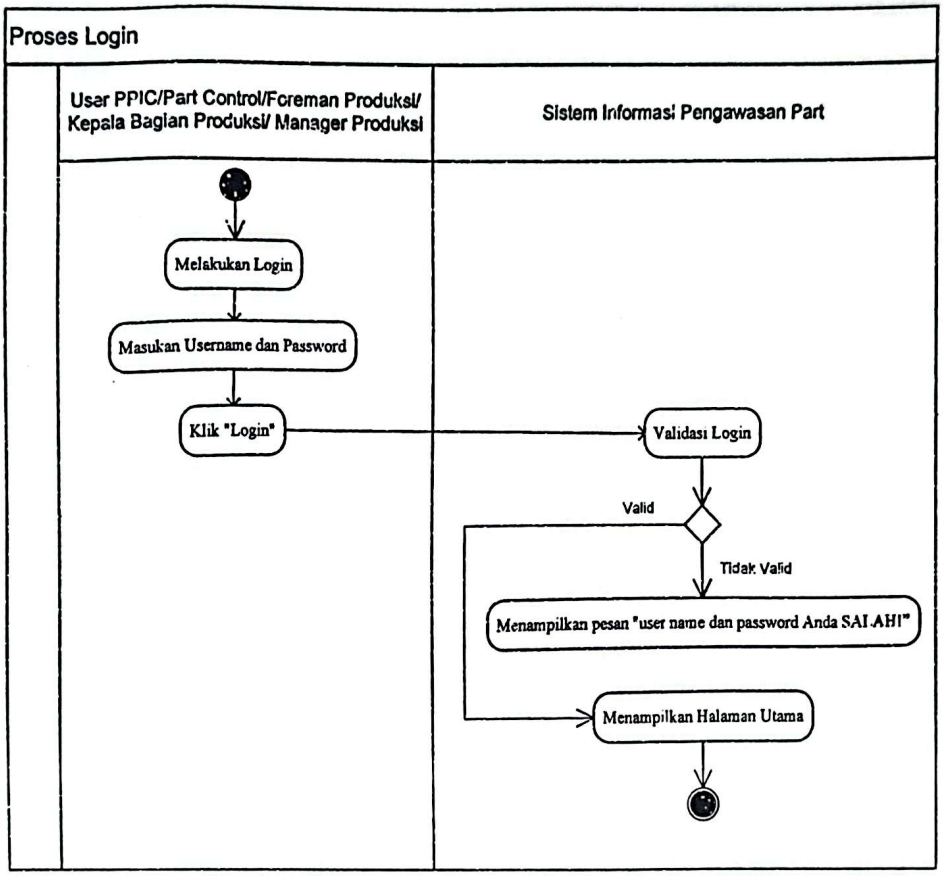
(Sumber: Hasil Pengolahan Data 2017)

5.2.4 *Activity Diagram* Sistem Informasi Pengawasan *Part*

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. *Activity diagram* bertujuan agar lebih memahami sistem yang akan dibuat.

1. *Activity Diagram* Proses *Login*

   Diagram ini menggambarkan aktivitas kegiatan akses *login* terhadap sistem. Berikut adalah kegiatan proses *login* yang diusulkan:

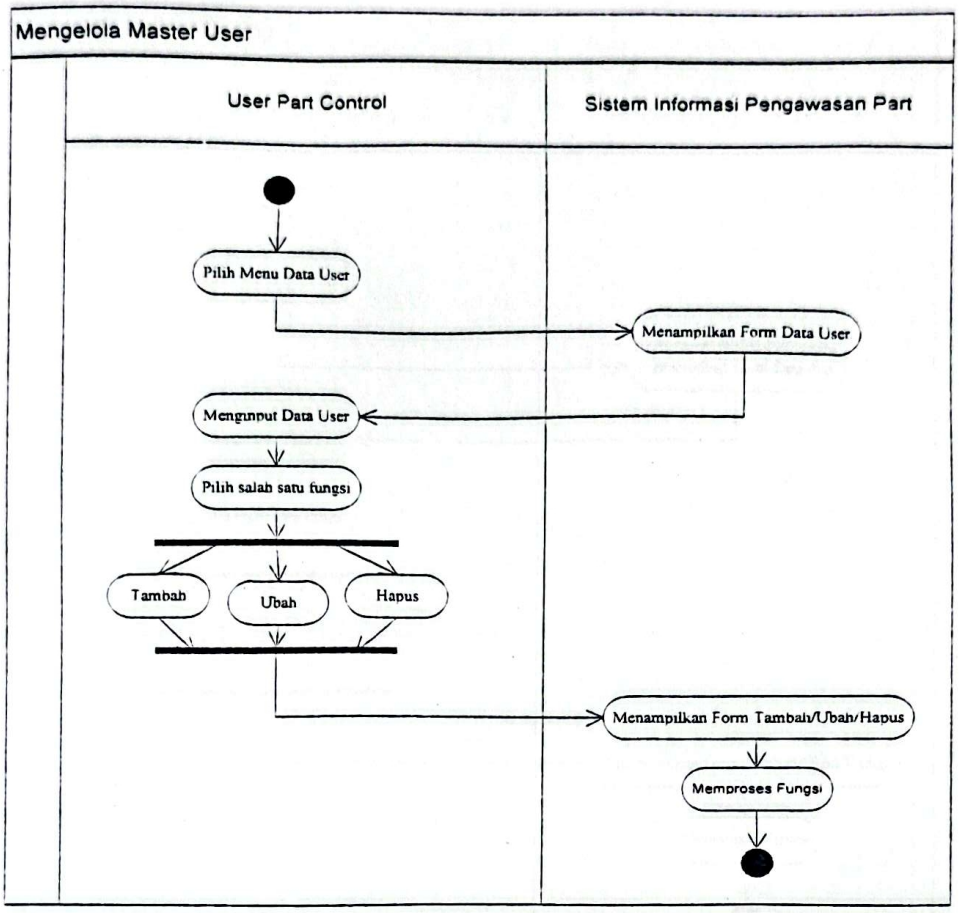


Gambar V.3 *Activity Diagram* Proses Login

(Sumber: Hasil Pengolahan Data 2017)

2. *Activity Diagram* Mengelola Master *User*

   Diagram ini menggambarkan kegiatan pengelolaan data pengguna sistem (*user*) pada usulan sistem informasi pengawasan *part*. Berikut kegiatan pengelolaan data *user* yang diusulkan:

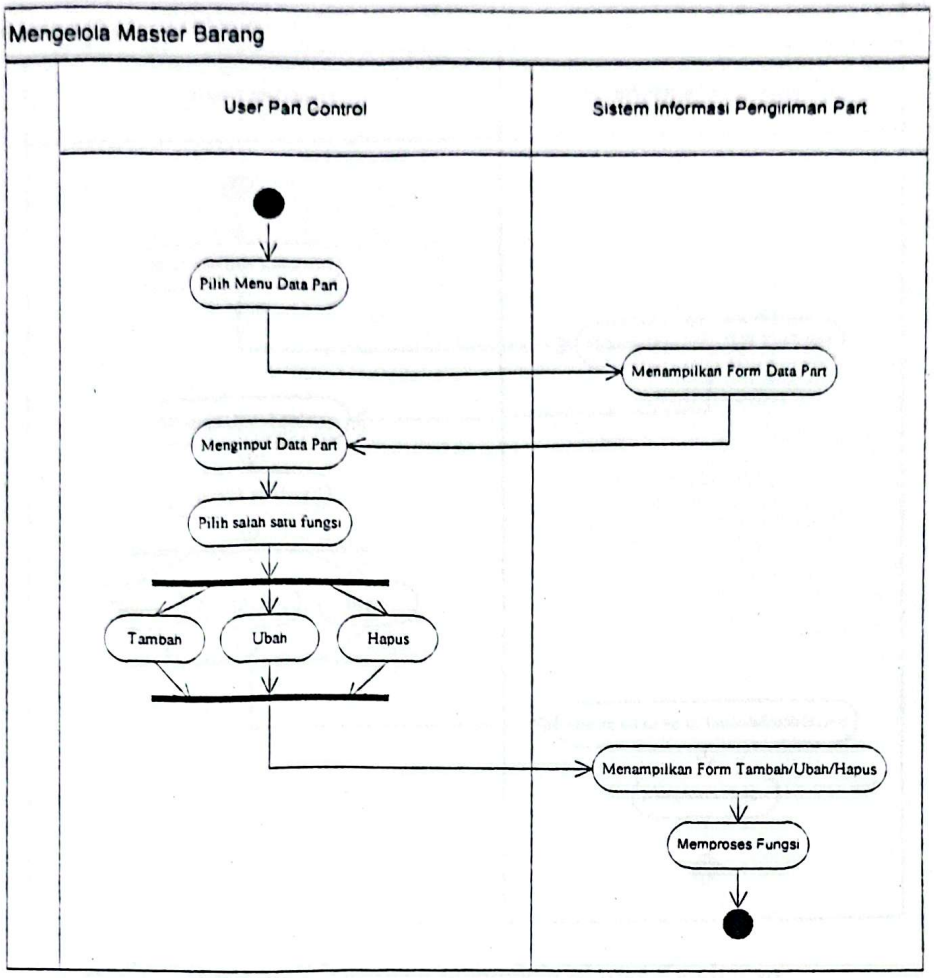


Gambar V.4 *Activity Diagram* Mengelola Master *User*

(Sumber: Hasil Pengolahan Data 2017)

3. *Activity Diagram* Mengelola Master Barang

Diagram ini menggambarkan kegiatan pengelolaan data barang pada usulan sistem informasi pengawasan *part*. Kegiatan pengelolaan data barang merupakan interaksi antara bagian user dan sistem. Berikut kegiatan pengelolaan data barang yang diusulkan:

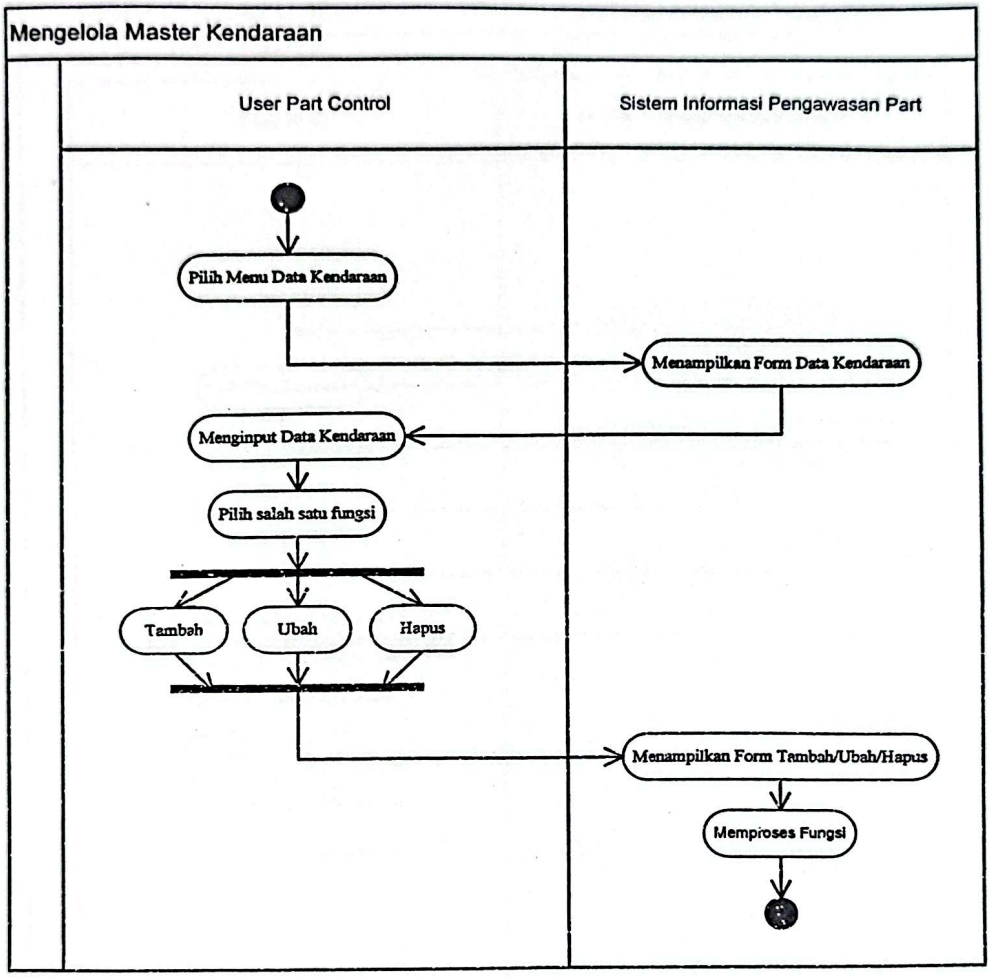


Gambar V.5 *Activity Diagram* Mengelola Master Barang

(Sumber: Hasil Pengolahan Data 2017)

4. *Activity Diagram* Mengelola Master Kendaraan

Diagram ini menggambarkan kegiatan pengelolaan data kendaraan pada usulan sistem informasi pengawasan *part*. Kegiatan pengelolaan data kendaraan merupakan interaksi antara bagian user dan sistem. Berikut kegiatan pengelolaan data kendaraan yang diusulkan:

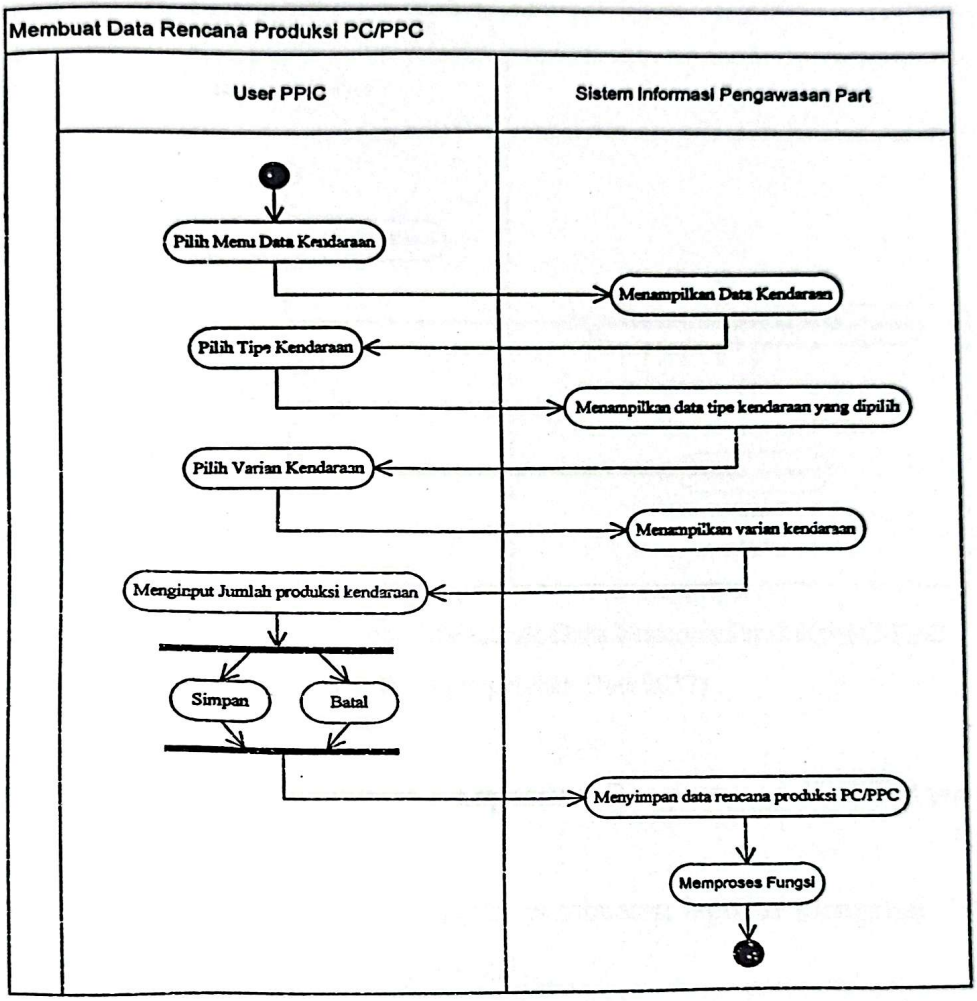


Gambar V.6 *Activity Diagram* Mengelola Master Kendaraan

(Sumber: Hasil Pengolahan Data 2017)

5. *Activity Diagram*Membuat Data Rencana Produksi PC/PPC

   Diagram ini menggambarkan kegiatan pembuatan data jumlah, tipe,dan varian kendaraan yang akan diproduksi.

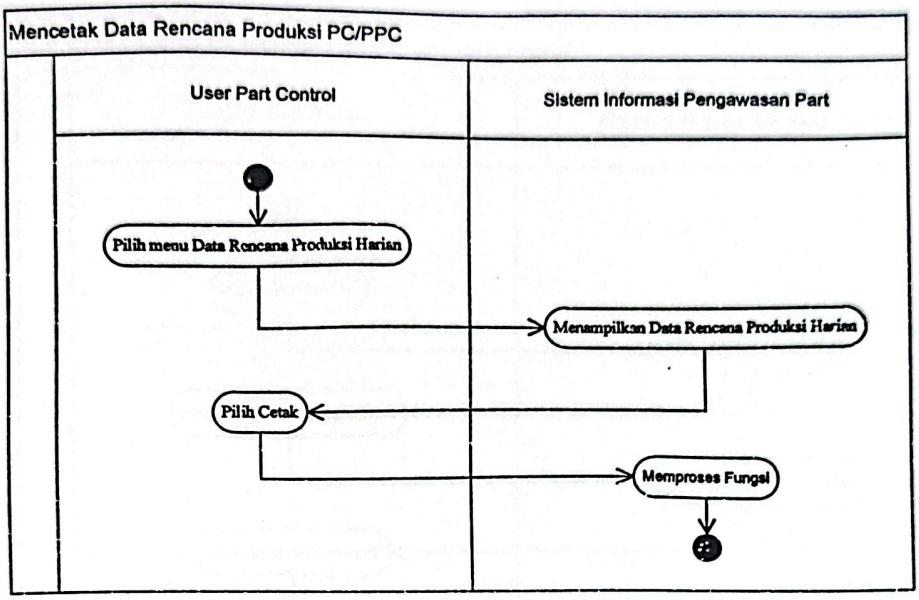


Gambar V.7 *Activity Diagram* Membuat Data Rencana Produksi PC/PPC

(Sumber: Hasil Pengolahan Data 2017)

6. *Activity Diagram* Mencetak Data Rencana Produksi PC/PPC

   Diagram ini menggambarkan kegiatan *Part Control* mencetak data Rencana Produksi yang telah dibuat oleh PPIC

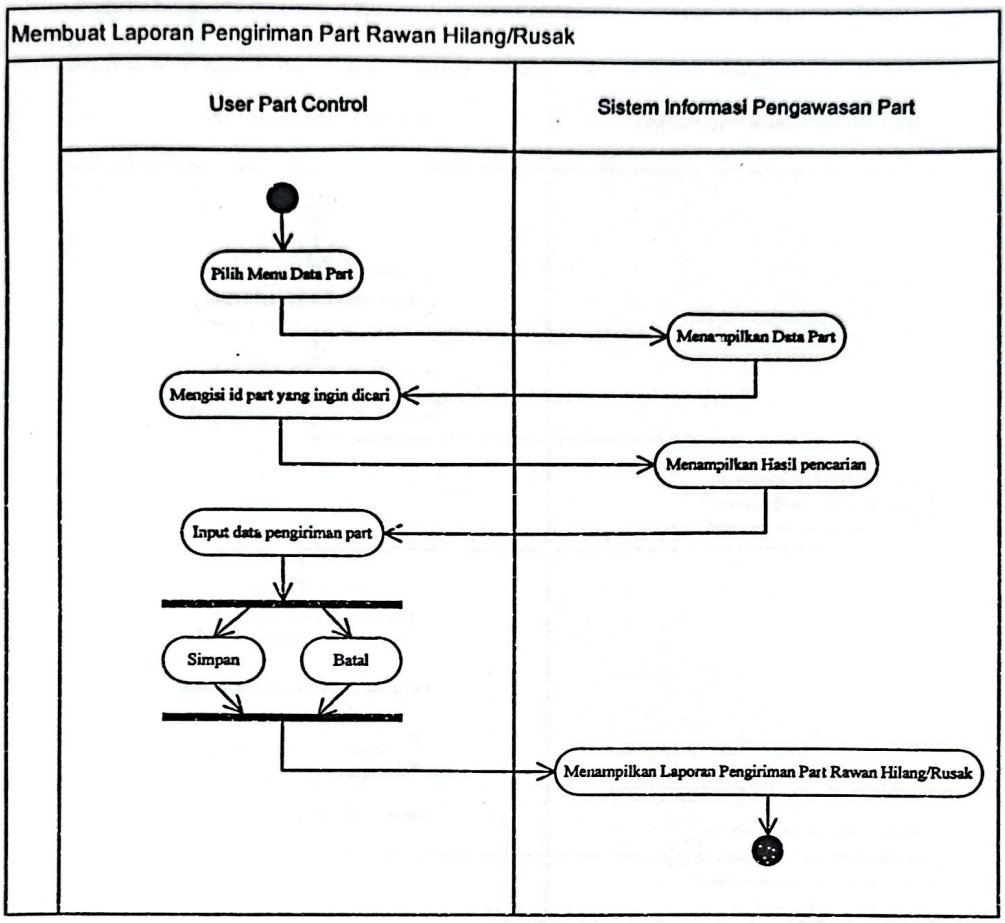


Gambar V.8 *Activity Diagram* Mencetak Data Rencana Produksi PC/PPC

(Sumber: Hasil Pengolahan Data 2017)

7. *Activity Diagram* Membuat Laporan Pengiriman *Part* Rawan Hilang/Rusak

   Diagram ini menggambarkan kegiatan pembuatan laporan mengenai data *part* yang akan dikirim ke *line* produksi.

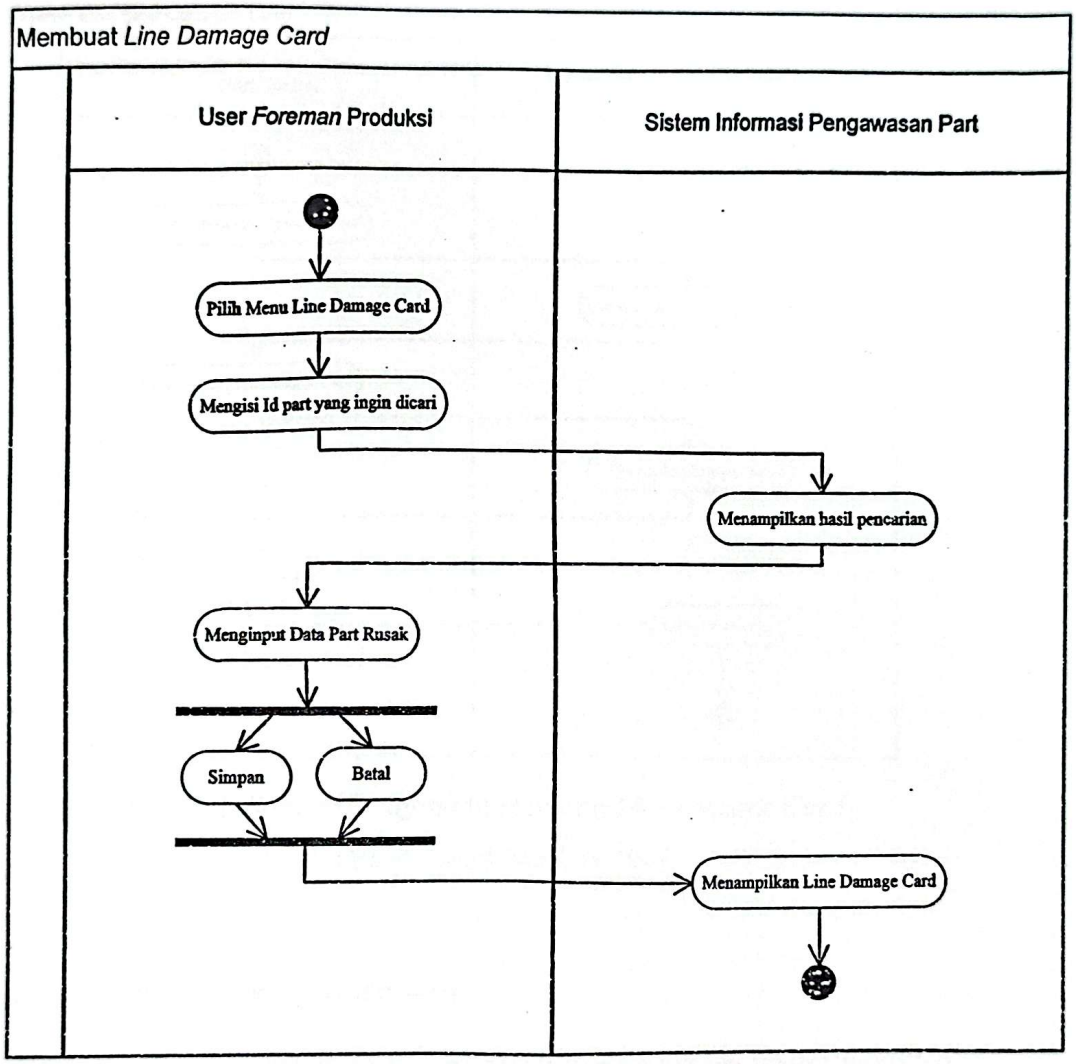


Gambar V.9 *Activity Diagram* Membuat Laporan Pengiriman Part Rawan Hilang/Rusak

(Sumber: Hasil Pengolahan Data 2017)

8. *Activity Diagram* Membuat *Line Damage Card*

   Diagram ini menggambarkan kegiatan pembuatan data *part* yang rusak seperti nomor, tipe,dan nama *part* serta penyebab kerusakan *part* dan cara penanggulangan yang harus dilakukan pada kerusakan part tersebut.

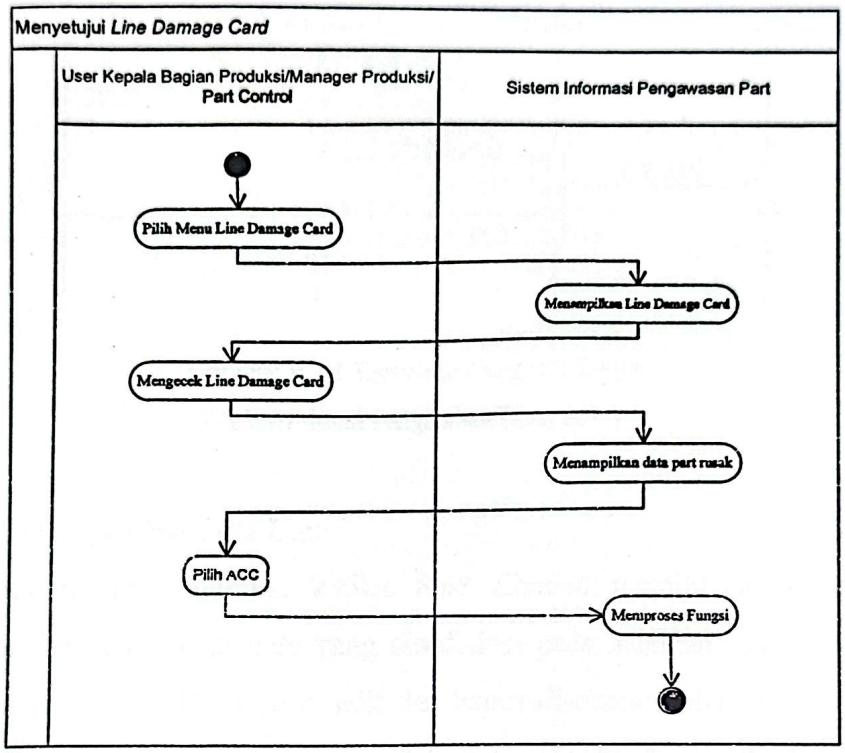


Gambar V.10 *Activity Diagram* Membuat *Line Damage Card*

(Sumber: Hasil Pengolahan Data 2017)

9. *Activity Diagram* Menyetujui *Line Damage Card*

   Diagram ini menggambarkan kegiatan memberikan persetujuan pada *Line Damage Card* sebagai tahapan proses untuk penggantian *part*. Kegiatan ini dilakukan oleh Kepala Bagian Produksi, Manager Produksi, dan *Part Control*.



Gambar V.11 *Activity Diagram* Menyetujui *Line Damage Card*

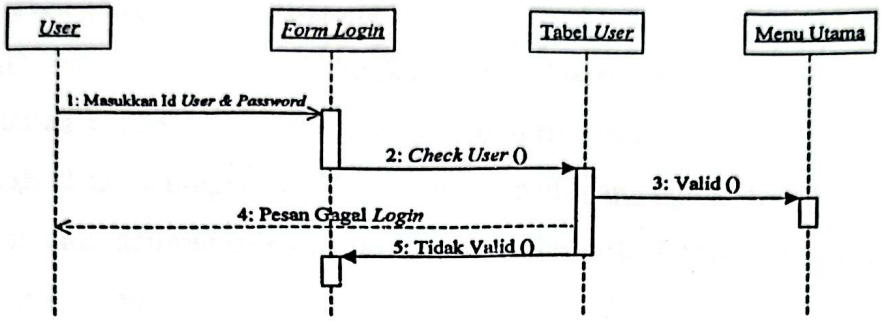
(Sumber: Hasil Pengolahan Data 2017)

5.2.5 *Sequence Diagram* Yang Diusulkan

Sequence diagram menjelaskan secara detail urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*. Berikut ini merupakan beberapa gambar *sequence diagram* dari Sistem informasi pengawasan part.

1. *Sequence Diagram Login*

   User memulai login dengan mengakses *form login*, kemudian menginputkan *username* dan *password*. Sistem melakukan validasi dengan mengecek hasil inputan *User* apakah sesuai dengan yang tersimpan didalam basis data *User*. Hasil pengecekan dapat menghasilkan data valid atau tidak. Sistem akan menampilkan *display menu* atau pesan jika inputan tidak valid kepada pengguna sistem.

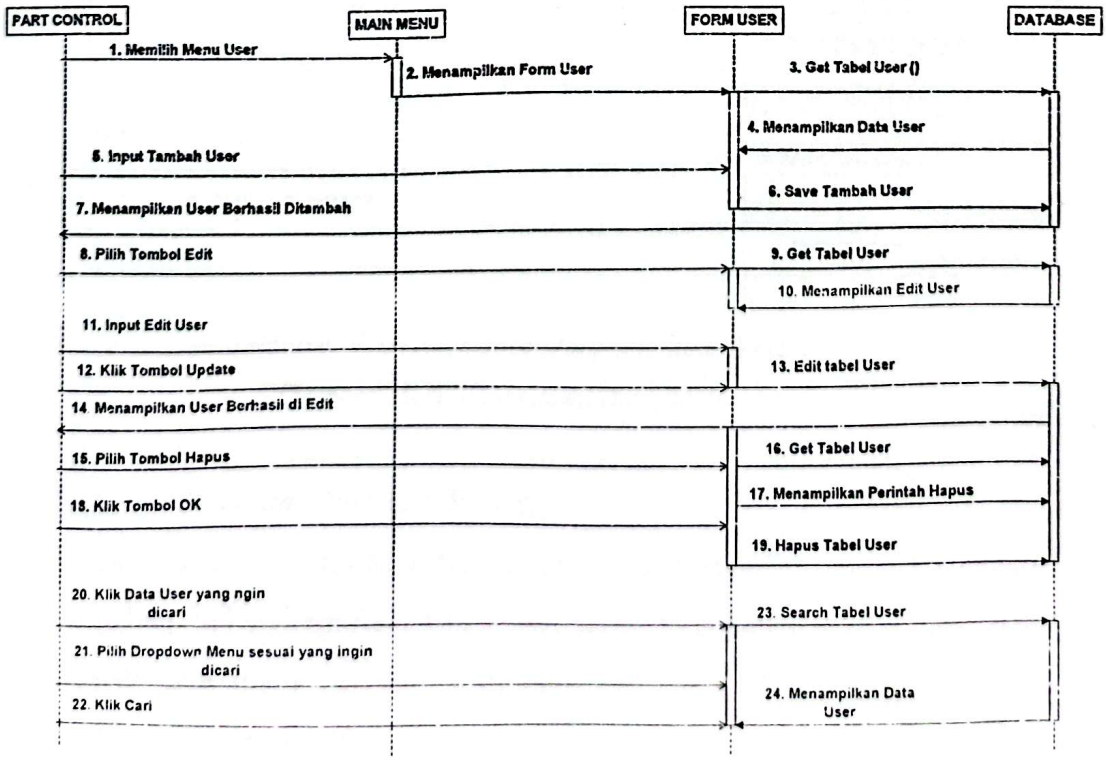


Gambar V.12 *Sequence Diagram Login*

(Sumber: Hasil Pengolahan Data, 2017)

2. *Sequence Diagram* Data *User*

Proses *sequence* dimulai ketika *Part Control* memilih menu *user*, kemudian melengkapi *field* yang disediakan pada halaman *user*. Setelah data dianggap valid, simpan, edit dan hapus di-*execute* oleh sistem, lalu manipulasi data simpan, ubah dan hapus pada tabel *user* di *database* terjadi.

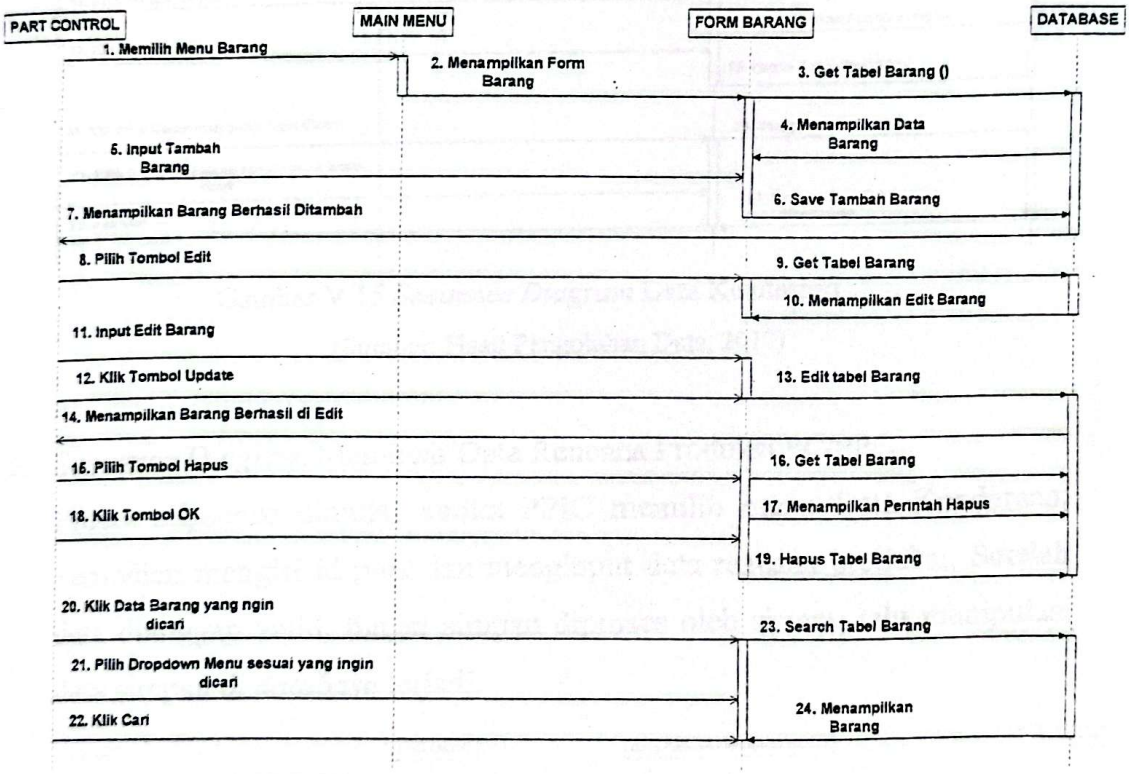


Gambar V.13 *Sequence Diagram Data User*

(Sumber: Hasil Pengolahan Data, 2017)

3. *Sequence Diagram* Data Barang

Proses *sequence* dimulai ketika *Part Control* memilih menu *part*, kemudian melengkapi *field* yang disediakan pada halaman data barang. Setelah data dianggap valid, simpan, ubah dan hapus di-*execute* oleh sistem, lalu manipulasi data simpan, ubah dan hapus pada tabel barang di *database* terjadi.

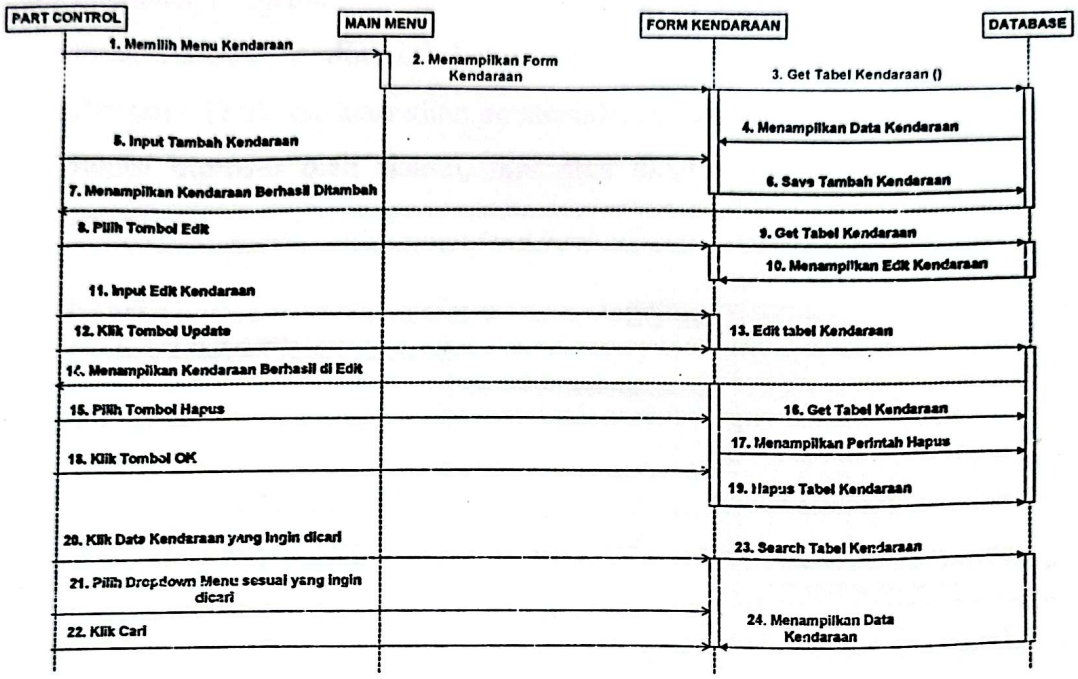


Gambar V.14 *Sequence Diagram* Data Barang

(Sumber: Hasil Pengolahan Data, 2017)

4. *Sequence Diagram* Data Kendaraan

Proses *sequence* dimulai ketika *Part Control* memilih menu kendaraan, kemudian melengkapi *field* yang disediakan pada halaman data kendaraan. Setelah data dianggap valid, simpan, ubah dan hapus di-*execute* oleh sistem, lalu manipulasi data simpan, ubah dan hapus pada tabel kendaraan di *database* terjadi.

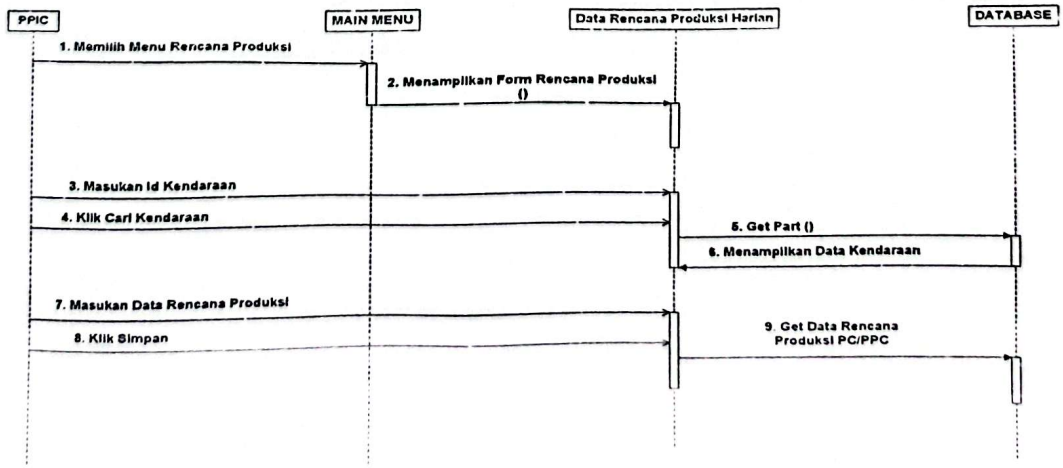


Gambar V.15 *Sequence Diagram* Data Kendaraan

(Sumber: Hasil Pengolahan Data, 2017)

5. *Sequence Diagram* Membuat Data Rencana Produksi PC/PPC

Proses *sequence* dimulai ketika PPIC memilih menu Data Kendaraan, kemudian mengisi id *part* dan menginput data rencana produksi. Setelah data dianggap valid, fungsi simpan diproses oleh sistem, lalu manipulasi data simpan di *database* terjadi.

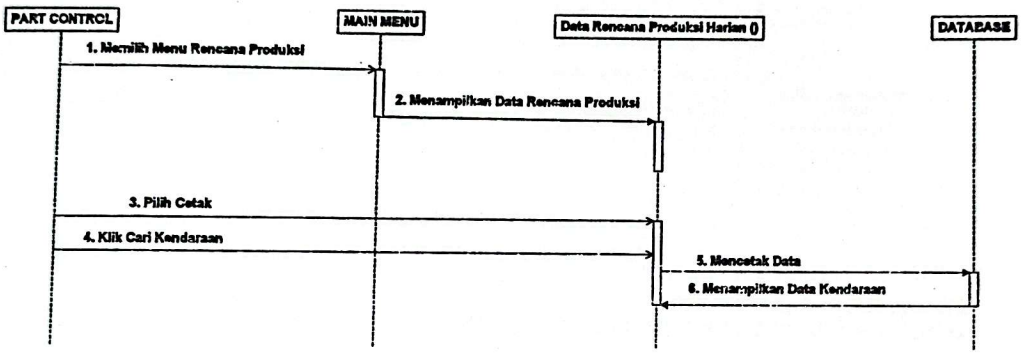


Gambar V.16 *Sequence Diagram* Membuat Data Rencana Produksi PC/PPC

(Sumber: Hasil Pengolahan Data, 2017)

6. *Sequence Diagram* Mencetak Data Rencana Produksi PC/PPC

   Proses *sequence* dimulai ketika *Part Control* membuka menu Data Rencana Produksi kemudian mencetaknya, setelah data dianggap valid, fungsi diproses oleh sistem, lalu data dicetak dalam bentuk dokumen kertas.

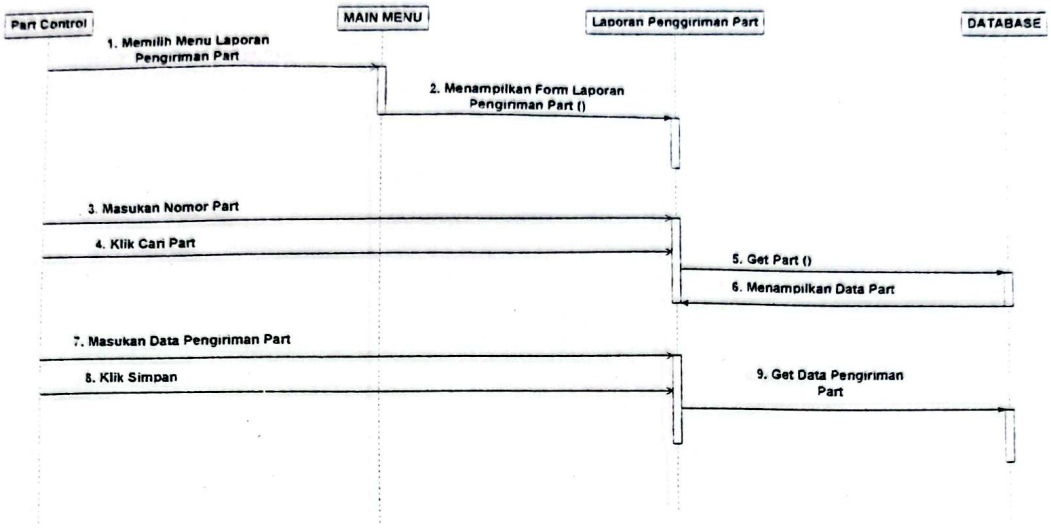


Gambar V.17 *Sequence Diagram* Mencetak Data Rencana Produksi PC/PPC

(Sumber: Hasil Pengolahan Data, 2017)

7. *Sequence Diagram* Membuat Laporan Pengiriman *Part* Rawan Hilang/Rusak

   Proses *sequence* dimulai ketika *Part Control* memilih menu Data *Part*, kemudian mengisi id *part* dan menginput data pengiriman *part*. Setelah data dianggap valid, fungsi simpan diproses oleh sistem, lalu manipulasi data simpan di *database* terjadi.

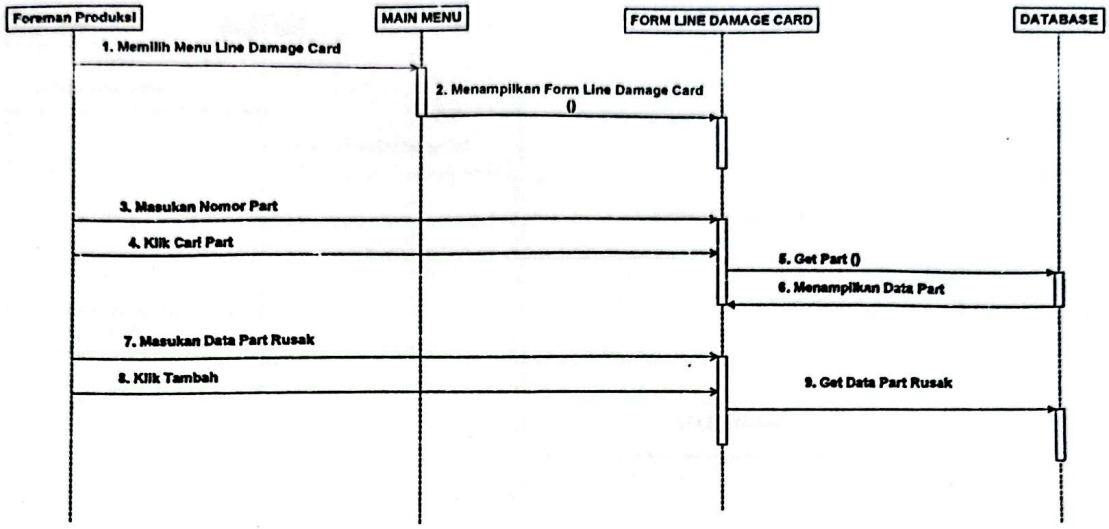


Gambar V.18 *Sequence Diagram* Membuat Laporan Pengiriman *Part* Rawan Hilang/Rusak

(Sumber: Hasil Pengolahan Data, 2017)

8. *Sequence Diagram* Membuat *Line Damage Card*

   Proses *sequence* dimulai ketika *Foreman* Produksi memilih menu *Line Damage Card*, kemudian memilih nomor *part* dan menginput data *part* yang rusak. Setelah data dianggap valid, simpan, ubah dan hapus di-*execute* oleh sistem, lalu manipulasi data simpan, ubah dan hapus pada tabel barang di *database* terjadi.

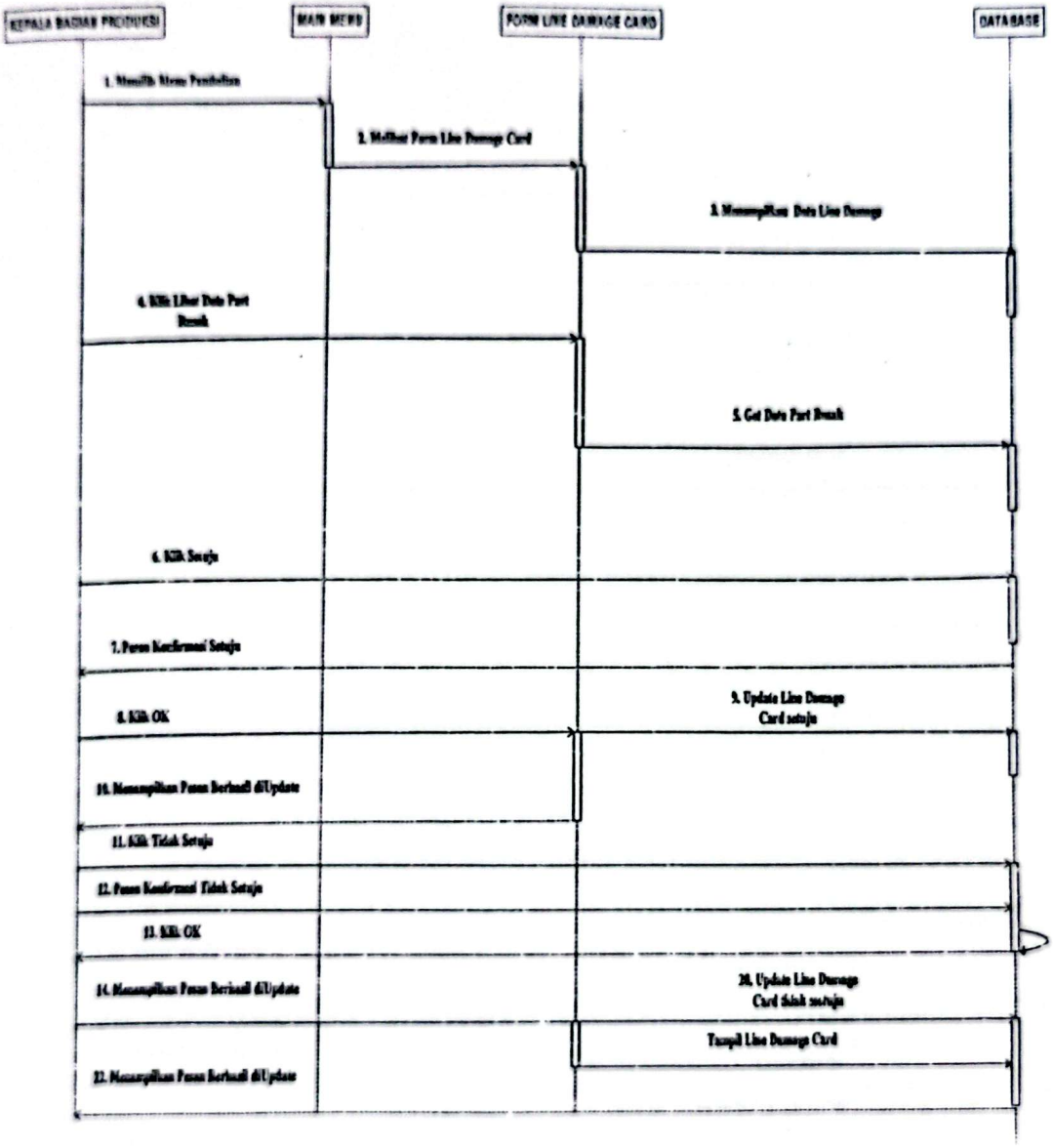


Gambar V.19 *Sequence Diagram* Membuat *Line Damage Card*

(Sumber: Hasil Pengolahan Data, 2017)

9. *Sequence Diagram* Menyetujui *Line Damage Card* oleh Kepala Bagian Produksi

   Proses *sequence* dimulai ketika Kepala Bagian Produksi memilih menu *Line Damage Card*, kemudian mengecek data yang ada pada *Line Damage Card.* Setelah data dianggap valid, fungsi diproses oleh sistem, lalu manipulasi data pada *database* terjadi.

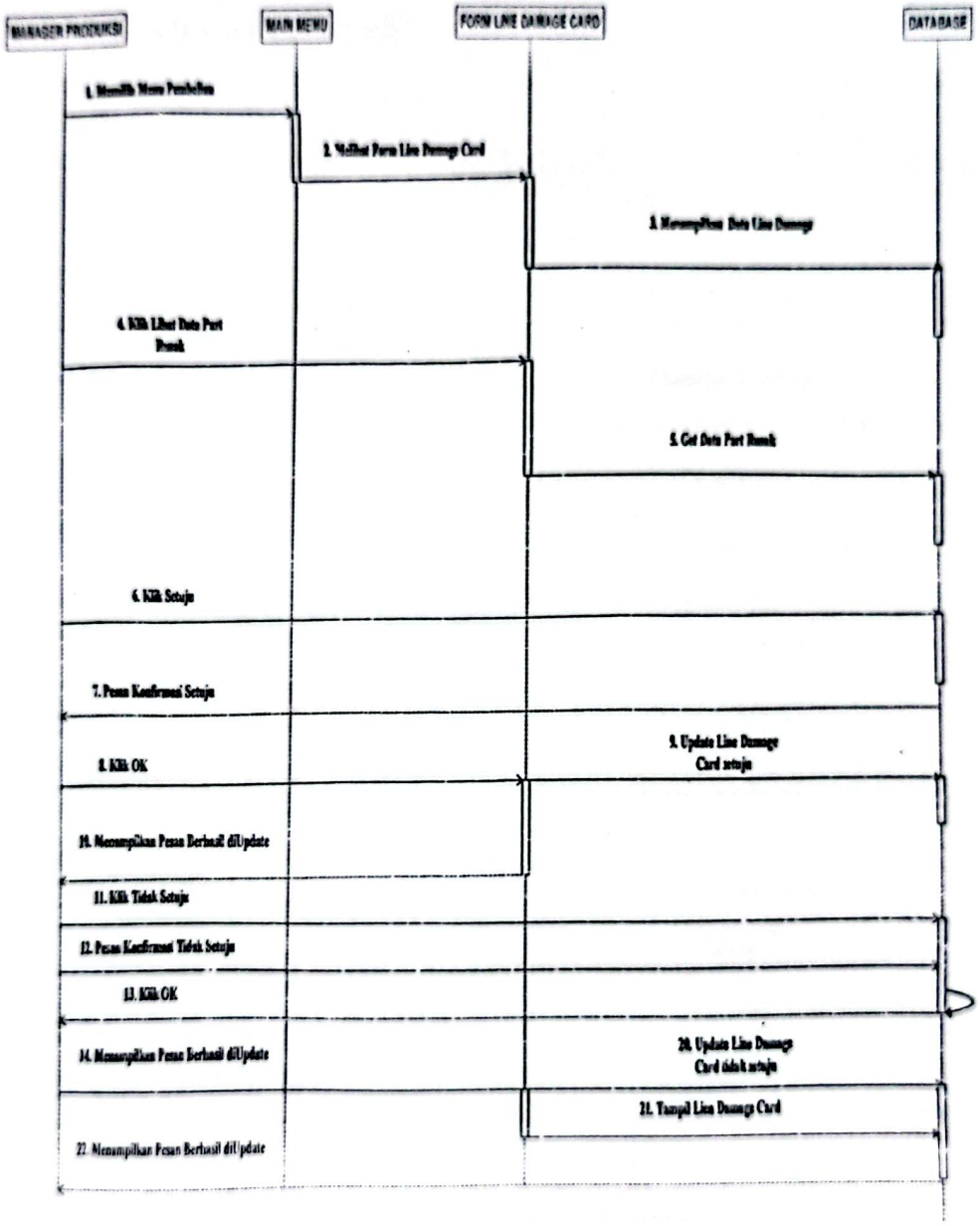


Gambar V.20 *Sequence Diagram* Menyetujui *Line Damage Card* oleh Kepala Bagian Produksi

(Sumber: Hasil Pengolahan Data, 2017)

10. *Sequence Diagram* Menyetujui *Line Damage Card* oleh Manager Produksi

Proses *sequence* dimulai ketika Manager Produksi memilih menu *Line Damage Card*, kemudian mengecek data *part* rusak *Line Damage Card*. Setelah data dianggap valid, setuju, fungsi diproses oleh sistem, lalu manipulasi data pada *database* terjadi.



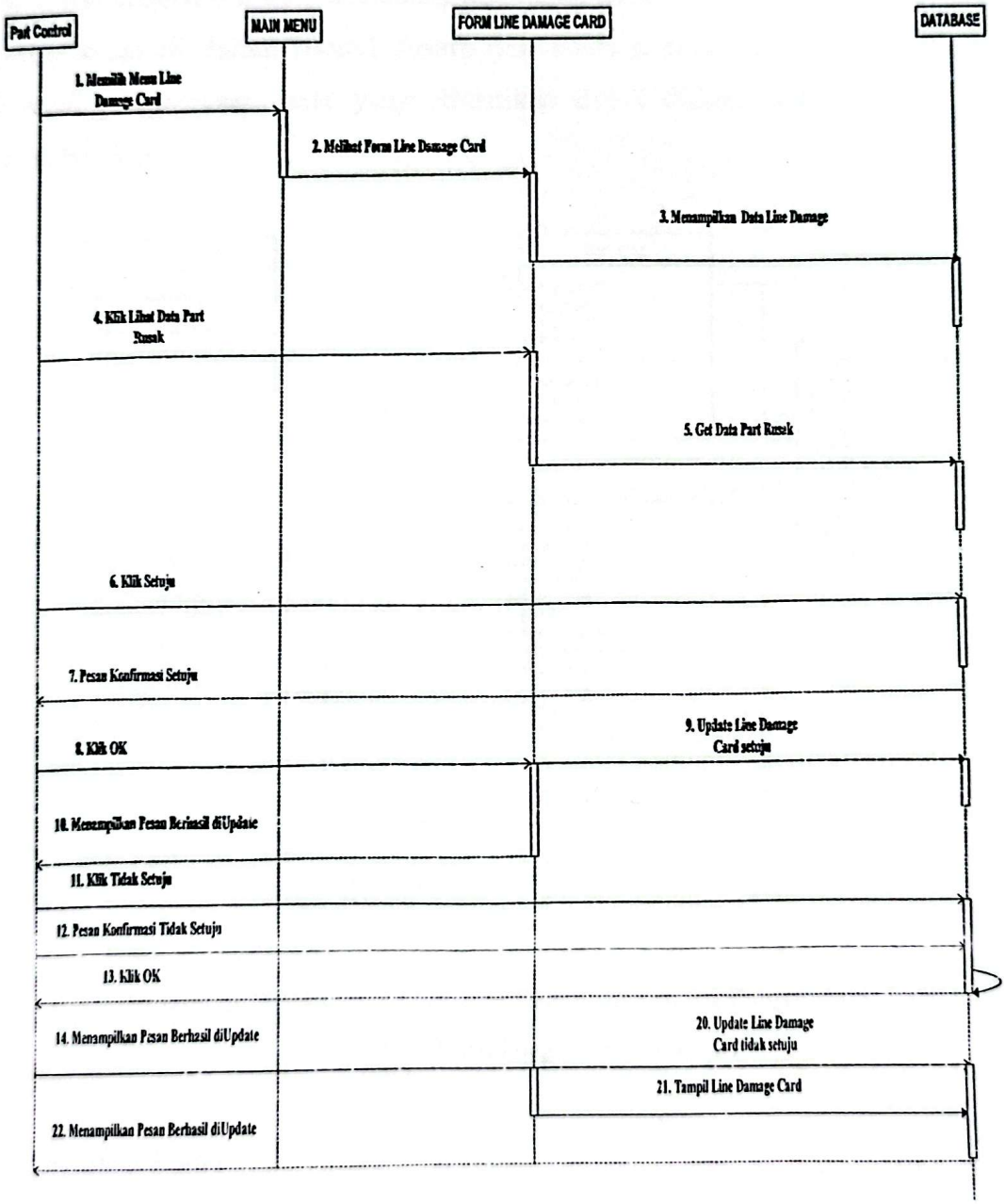
Gambar V.21 *Sequence Diagram* Menyetujui *Line Damage Card* oleh Manager Produksi

(Sumber: Hasil Pengolahan Data, 2017)

11. *Sequence Diagram* Persetujuan *Line Damage Card* oleh *Part Control*

Proses *sequence* dimulai ketika *Part Control* memilih menu *Line Damage Card*, kemudian mengecek tanda persetujuan dari Kepala Bagian Produksi dan Manager Produksi kemudian menyetujui *Line Damage Card*. Setelah

data dianggap valid, setuju, fungsi diproses oleh sistem, lalu manipulasi data pada *database* terjadi.

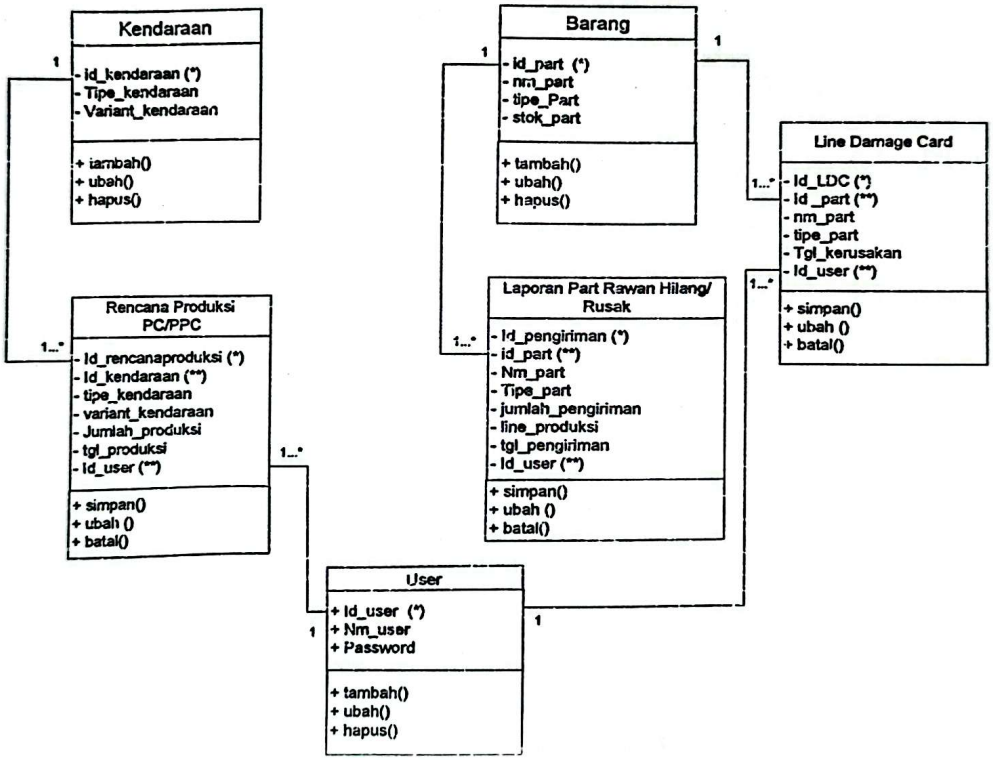


Gambar V.22 *Sequence Diagram* Menyetujui *Line Damage Card* oleh *Part Control*

(Sumber: Hasil Pengolahan Data, 2017)

5.3 *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem informasi pengawasan *part*. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan *detail* tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi pengawasan *part* yang diusulkan dapat dilihat pada Gambar V.23 sebagai berikut:



Gambar V.23 *Class Diagram* Sistem Informasi Pengawasan *Part*

(Sumber: Hasil Pengolahan Data, 2017)

5.3.1 Kamus Data

Kamus data membantu dalam pendefinisian data agar pendefinisian data tersebut dapat dilakukan dengan lengkap dan terstruktur. Pembentukan kamus data dilaksanakan dalam tahap analisis dan perancangan suatu sistem. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum

sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar penulisan). Berikut ini adalah kamus data yang terdapat dalam aplikasi sistem informasi pengawasan *part*:

1. Spesifikasi Tabel *User*

   Nama Tabel : *user*

   Fungsi : Untuk menyimpan data *user*

   Tipe : File data *master*

Tabel V.15 Tabel *User*

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
Id_*user*	*Integer*	5	*Primary key*	Nomor Id pengguna
Nm_*user*	*Varchar*	30		Nama pengguna
Username	*Varchar*	30		*User Name* pengguna
Password	*Varchar*	50		*Password* Pengguna

(Sumber: Hasil Analisis Data, 2017)

2. Spesifikasi Tabel Barang

   Nama Tabel : Barang

   Fungsi : Untuk menyimpan data barang

   Tipe : File data Master

Tabel V.16 Tabel Barang

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
Id_part	*Char*	10	*Primary key*	Kode part
nm_part	*Varchar*	30		Nama Part
tipe_part	*Varchar*	10		Tipe Part
stok_part	*Integer*	10		Stok Part

(Sumber: Hasil Analisis Data, 2017)

3. Spesifikasi Tabel Kendaraan

   Nama Tabel : Kendaraan

   Fungsi : Untuk menyimpan data kendaraan

Tabel V.17 Tabel Data Kendaraan

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
Id_kendaraan	*Varchar*	7	*Primary key*	Nomor Id kendaraan
tipe_kendaraan	*Varchar*	30		Tipe Kendaraan
variant_kendaraan	*Varchar*	30		Variant Kendaraan

(Sumber: Hasil Analisis Data, 2017)

4. Spesifikasi Tabel Rencana Produksi PC/PPC

Nama Tabel : MembuatRencana Produksi PC/PPC

Fungsi : Membuat data rencana produksi harian

Tipe : File data Rencana Produksi PC/PPC

Tabel V.18 Tabel Membuat Rencana Produksi PC/PPC

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
Id_rencanaproduksi	*Integer*	10	*Primary Key*	Nomor Id Rencana Produksi
Id_kendaraan	*Integer*	5	*Foreign Key*	Nomor Id kendaraan
tipe_kendaraan	*Varchar*	10		Tipe Kendaraan
variant_kendaraan	*Varchar*	10		Variant Kendaraan
jumlah_poduksi	*Varchar*	5		Jumlah Produksi Kendaraan
tgl_produksi	*Datetime*			Tanggal Produksi Kendaraan
Id_user	*Integer*	5	*Foreign Key*	Nomor id pengguna

(Sumber: Hasil Analisis Data, 2017)

5. Spesifikasi Tabel Laporan Pengiriman *Part* Rawan Hilang/Rusak

Nama Tabel : Laporan Pengiriman *Part* Rawan Hilang/Rusak

Fungsi : Membuat data laporan pengiriman part

Tipe : File Laporan Pengiriman *Part* Rawan Hilang/Rusak

Tabel V.19 Tabel Membuat Laporan Pengiriman *Part* Rawan Hilang/Rusak

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
Id_pengiriman	*Varchar*	10	*Primary Key*	Nomor pengiriman
Id_part	*Char*	5	*Foreign key*	Nomor Id Part
nm_part	*Varchar*	10		Nama Part
tipe_part	*Varchar*	10		Tipe Part
jumlah_pengiriman	*Varchar*	10		Jumlah Pengiriman
line_produksi	*Varchar*	20		
tgl_pengiriman	*Datetime*			Tanggal Pengiriman
Id_user	*Integer*	5		Nomor Id pengguna

(Sumber: Hasil Analisis Data, 2017)

6. Spesifikasi Tabel *Line Damage Card*

   Nama Tabel : *Line Damage Card*

   Fungsi : Untuk membuat data*part* rusak

   Tipe : *Form Line Damage Card*

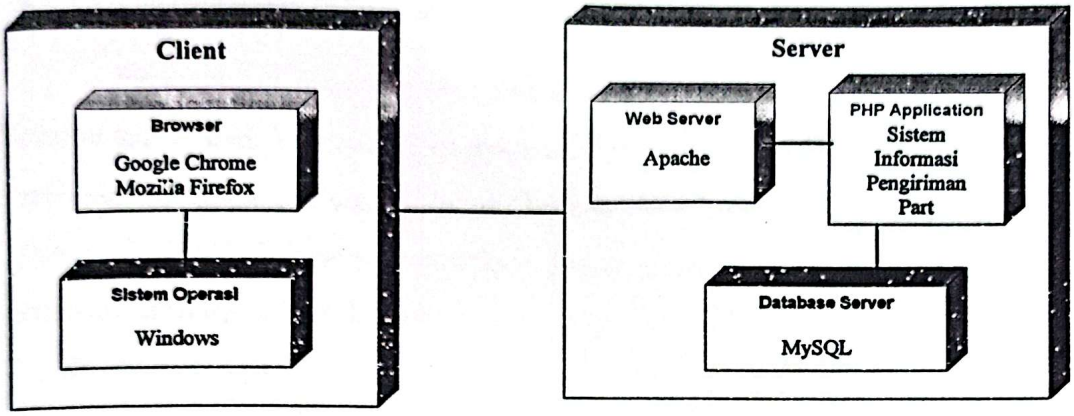
Tabel V.20 Tabel Membuat *Line Damage Card*

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
Id_LDC	*Varchar*	15	*Primary Key*	Nomor id Line Damage Card
Id_Part	*Char*	10	*Foreign key*	Kode Part
nm_part	*Varchar*	30		Nama Part
tipe_part	*Varchar*	10		Tipe Part
tgl_kerusakan	*Datetime*			Tanggal Part Rusak
Jumlah Damage	*Integer*	5		Jumlah Part Rusak
Id_User	*Enum*			Nomor Id pengguna

(Sumber: Hasil Analisis Data, 2017)

5.4 *Deployment Diagram*

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Seperti yang dapat dilihat pada gambar V.24 sebagai berikut:

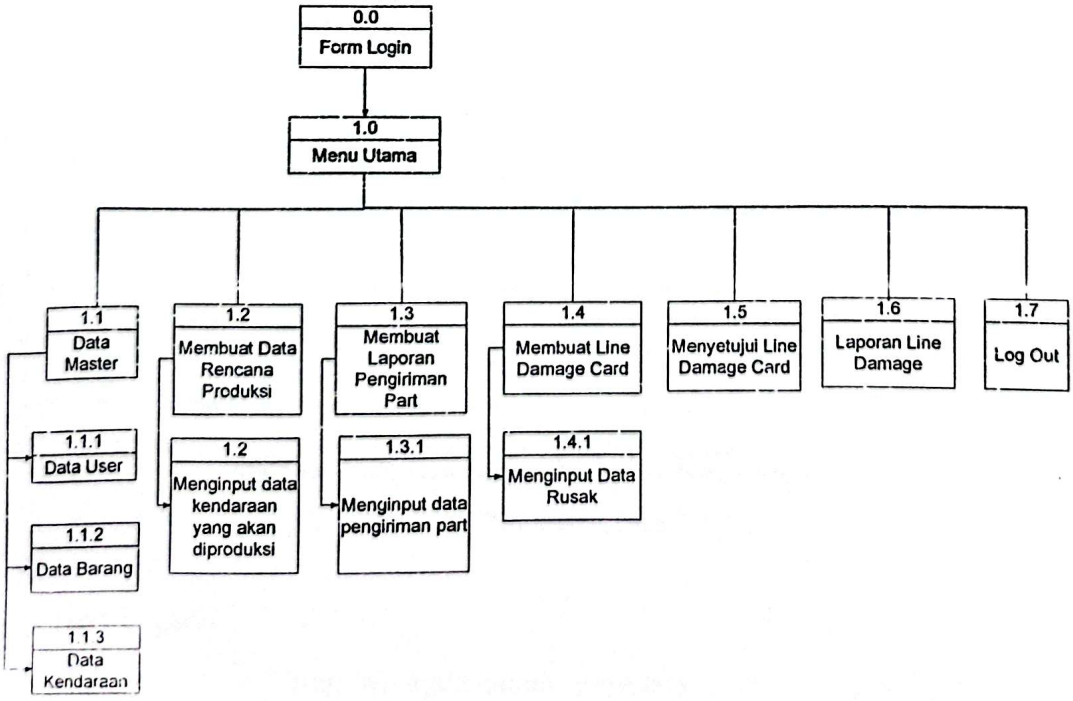


Gambar V.24 *Deployment Diagram* yang Diusulkan

(Sumber: Hasil Pengolahan Data, 2017)

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

Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hierarkis. Berikut adalah struktur menu program Sistem Informasi Pengawasan *Part* pada PT Krama Yudha Ratu Motor.



Gambar V.25 HIPO Sistem Informasi Pengawasan *Part*

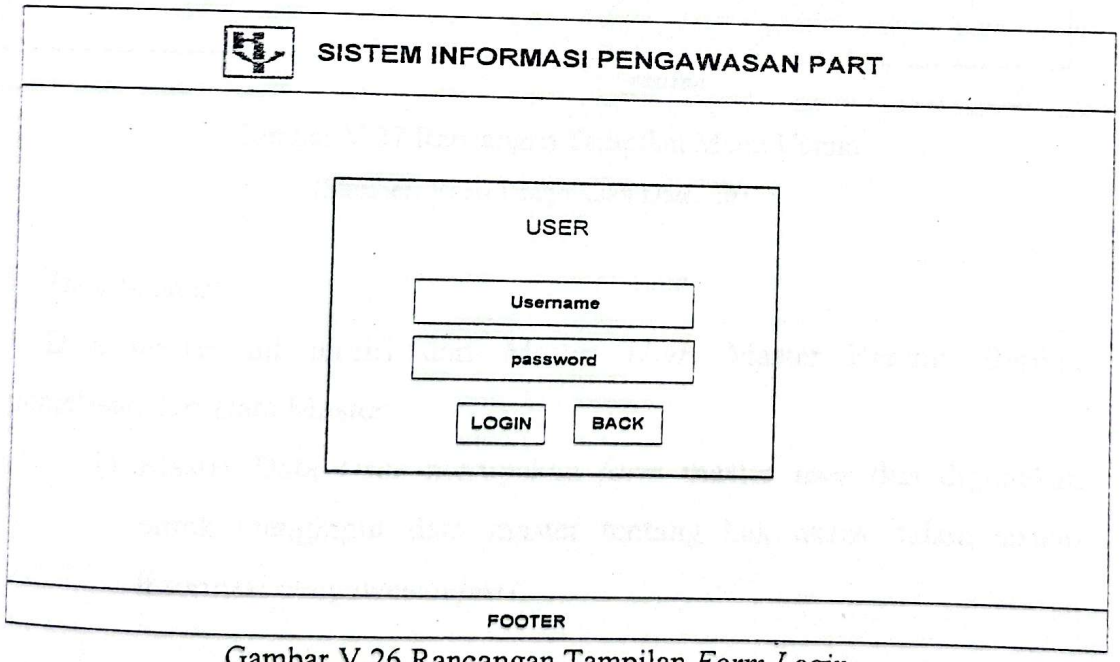
(Sumber: Hasil Pengolahan Data, 2017)

5.4.2 Perancangan *Interface* Sistem Informasi Pengawasan *Part*

Perancangan *interface* merupakan tahapan untuk membuat tampilan atau *design* dari Sistem Informasi Pengawasan *Part* . Rancangan tampilan yang dibuat meliputi beberapa rancangan *input* (berupa *form input*) dan rancangan *output* (berupa laporan). Rancangan *interface* dalam Sistem informasi pengawasan *part* tersebut dapat dilihat berikut ini:

1. *Form Login*

   Form Login digunakan untuk membedakan hak akses pengguna (*user*). Melalui *Form Login* ini pengguna yang boleh masuk sistem adalah pengguna yang mengetahui *username* dan *password* atau pengguna yang memiliki wewenang untuk menggunakan sistem yaitu sistem informasi pengawasan *part.*



Gambar V.26 Rancangan Tampilan *Form Login*

(Sumber: Hasil Pengolahan Data, 2017)

Keterangan:

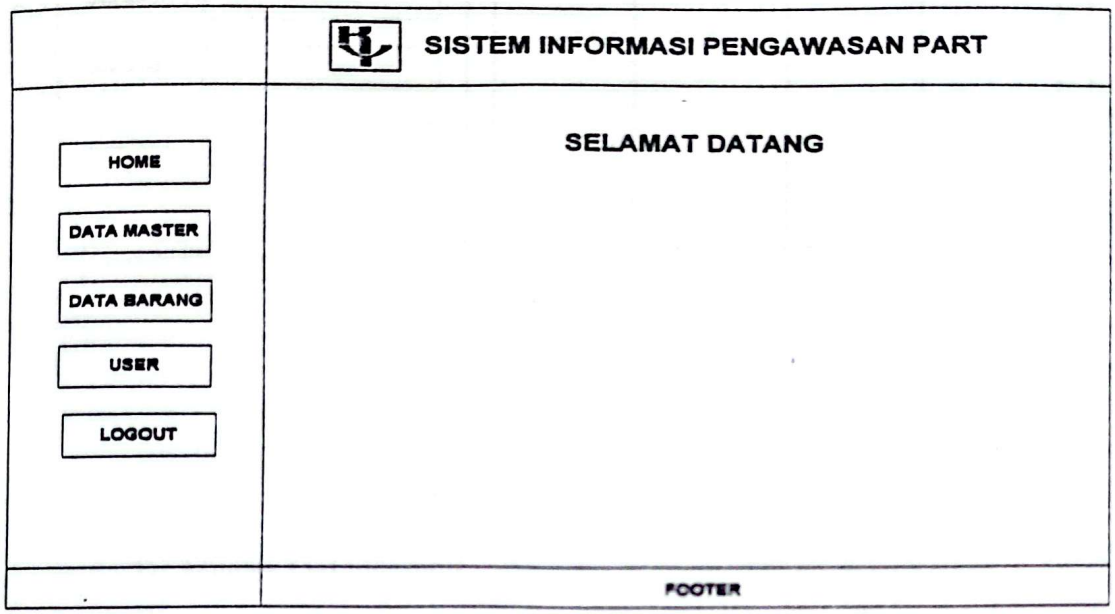
Username : Untuk Menginputkan *username.*

Password : Untuk menginputkan *password.*

Button Login : Digunakan untuk proses masuk ke dalam menu utama.

2. Menu Utama

Menu utama ini dirancang untuk memudahkan *user* dalam mengakses aplikasi dengan pilihan-pilihan menu yang sudah disediakan sesuai dengan kebutuhan dan hak akses karena tidak semua menu dapat digunakan.



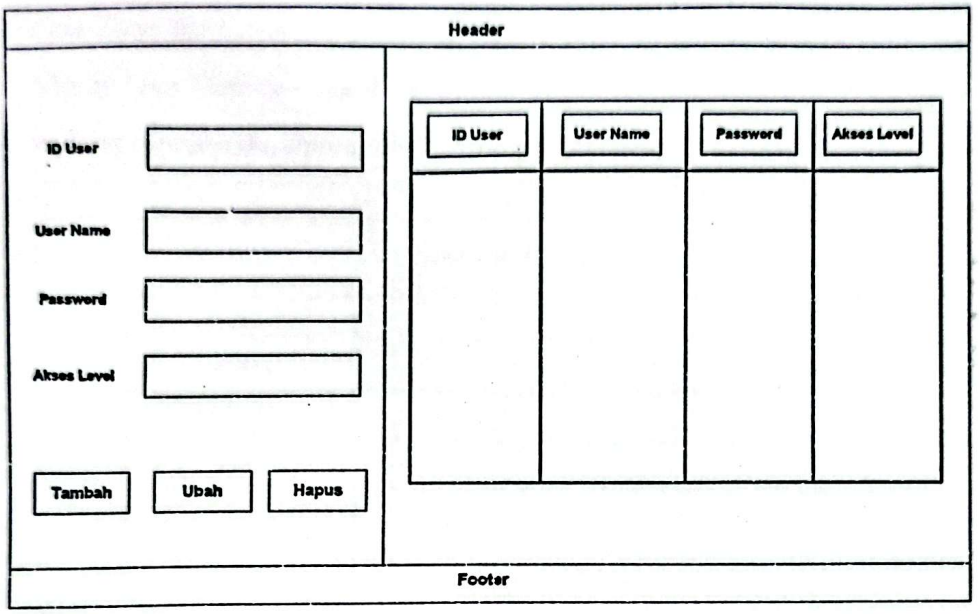
Gambar V.27 Rancangan Tampilan Menu Utama

(Sumber: Hasil Pengolahan Data, 2017)

3. Data Master

Data master ini terdiri dari Master *User*, Master Barang. Berikut penjelasan dari Data Master:

1) Master Data *User* merupakan *form* master *user* dan digunakan untuk menginput data master tentang hak akses dalam sistem informasi pengawasan *part*.

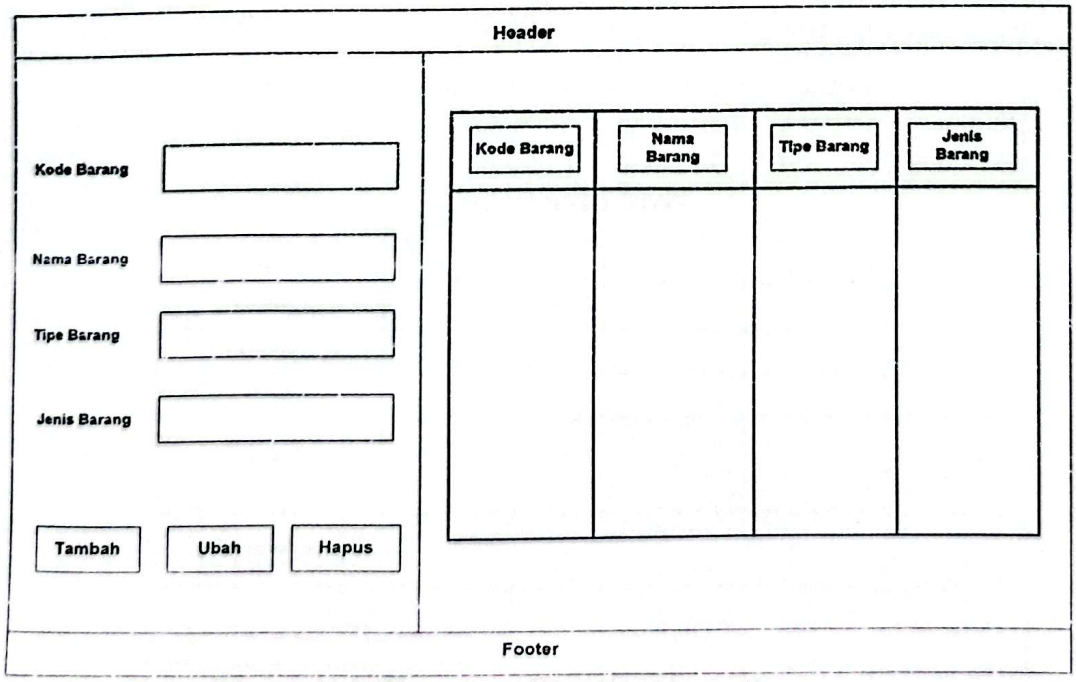


Gambar V.28 Rancangan Tampilan Master *User*

(Sumber: Hasil Pengolahan Data, 2017)

2) Master Data Barang

Form master barang digunakan untuk menginput data master tentang barang yang rusak saat di produksi.

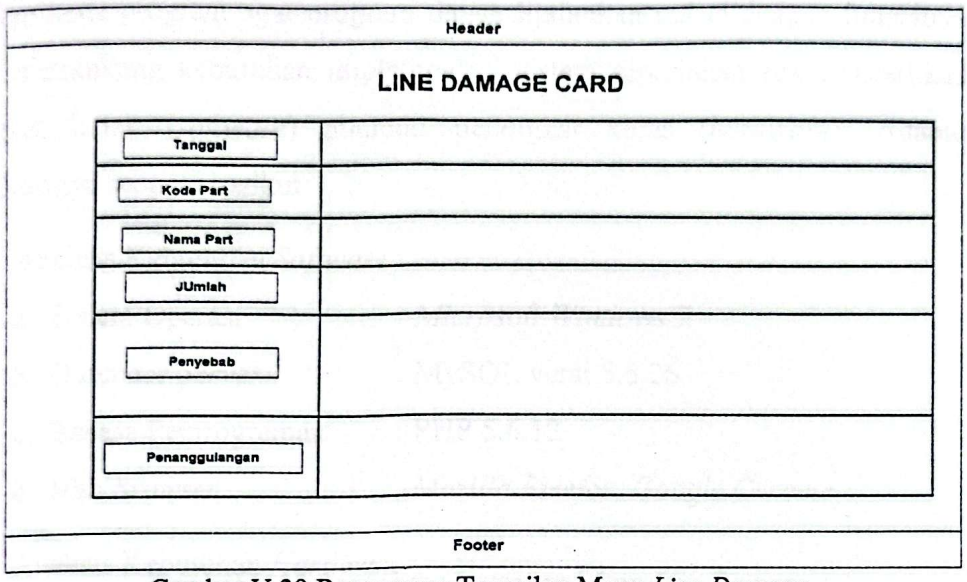


Gambar V.29 Rancangan Tampilan Master Barang

(Sumber: Hasil Pengolahan Data, 2017)

4. *Line Damage Card*

Menu *Line Damage Card* berfungsi untuk mencatat data *part* rusak saat sedang diproduksi, dibuat oleh *Foreman* Produksi.

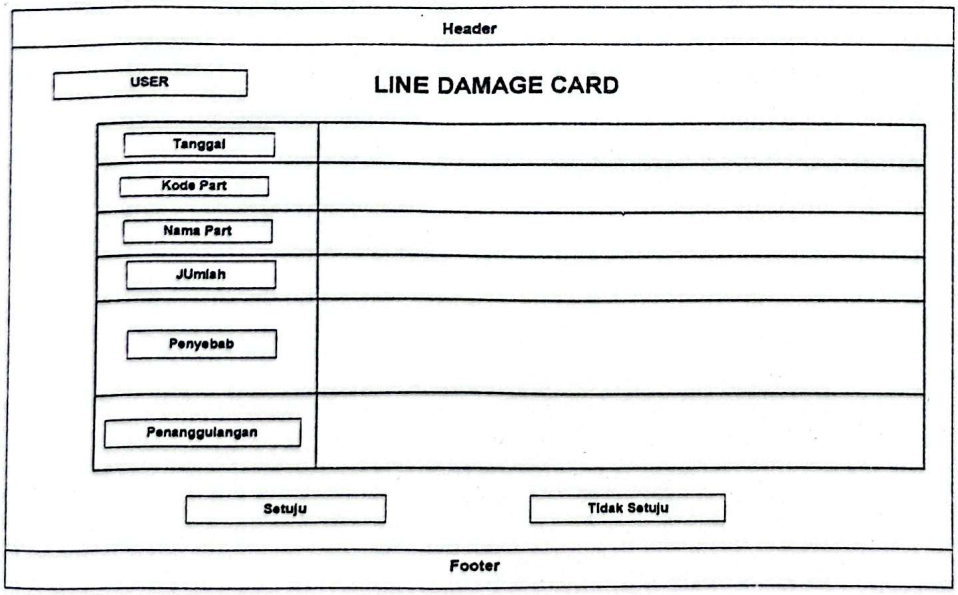


Gambar V.30 Rancangan Tampilan Menu *Line Damage*

(Sumber: Hasil Pengolahan Data, 2017)

5. Menyetujui *Line Damage Card*

Menu ini berfungsi untuk meminta persetujuan mengenai *part* rusak dari Kepala Bagian Produksi, *Manager* Produksi, dan *Part Control.*



Gambar V.31 Rancangan Tampilan Menyetujui *Line Damage Card*

(Sumber: Hasil Pengolahan Data, 2017)

BAB VI
KESIMPULAN DAN SARAN

6.1 Kesimpulan

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

1. Aplikasi sistem informasi pengawasan part ini memudahkan *Part Control* dalam membuat data pengiriman *part* serta data *part* rusak yang sebelumnya dilakukan secara manual sehingga kurang efisien.
2. Aplikasi sistem informasi pengawasan *part* ini dapat mempercepat proses permintaan dan persetujuan penggantian *part* karena persetujuan dari Kepala Bagian Produksi dan juga *Manager* Produksi dilakukan secara terkomputerisasi sehingga lebih efektif serta mempermudah dalam pencarian data – data *part* yang rusak.

6.2 Saran

Saran yang diberikan untuk pengembangan sistem informasi pengawasan *part* ini selanjutnya adalah sebagai berikut:

1. Untuk penerapan sistem terkomputerisasi sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi penggunaan sistem ini kepada bagian yang terkait.
2. Perlunya pemeliharaan terhadap sistem yang telah dibuat agar tetap berjalan dengan baik.
3. Menambahkan fitur-fitur yang lebih mendukung terhadap sistem yang baru seperti fitur pembuatan *Line Damage Card* dan fitur lainya yang berhubungan dengan proses pengawasan *part*.

DAFTAR PUSTAKA

Akakom, STMIK. 2009. *Modul 1 Mengenal Unified Modeling Language.* (http://ns.akakom.ac.id/.../Microsoft%20Word%20-%20MODUL%20I.pdf, diakses tanggal 2 September 2015)

https://malikazisahmad.wordpress.com/2012/01/13/pengertian-pengawasan/

Kadir, Abdul. 2003. *Pengenalan Sistem Informasi.* Yogyakarta: Andi Offset.

Munawar. 2005. *Pemodelan dengan UML.* Yogyakarta: Graha Ilmu.

Rosa, A.S., & Shalahuddin, M. 2013. *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek).* Cetakan Pertama. Bandung: Modula.

Sutabri, Tata. 2012. *Konsep Sistem Informasi.* Yogyakarta: Andi Offset.

Telkomuniversity. 2015. *Class Diagram*

(http://ftass.telkomuniversity.ac.id, diakses tanggal 3April 2015).

Unair, DSI. 2012. *Tipe-Tipe Data di MySQL.*

(http://nimoz-fisip11.web.unair.ac.id/artikel_detail-59844-kuliah-Tipe%20%20tipe%20data%20di%20My%20SQL.html, diakses tanggal 7 Desember 2015).

Wibowo, Pengertian Bahan Baku, www.wibowopajak.com, 2012. (Tanggal akses: 25 Februari 2015)

Widigdo, Anon K. 2003. *Dasar Pemrograman PHP dan MySQL.* (http://ikc.dinus.ac.id/umum/anon-phpmysql.php, diakses tanggal 7 Desember 2015).