

**RANCANG BANGUN SISTEM INFORMASI
PEMELIHARAAN MESIN PRODUKSI *FRONT FORK 2 W*
BERBASIS WEB MENGGUNAKAN PHP 5.6.3 DAN MYSQL
5.6.21 DI PT SHOWA INDONESIA MANUFACTURING**

TUGAS AKHIR

Diajukan untuk Memenuhi Sebagian Syarat Penyelesaian Jenjang Diploma Empat
(D-4) Program Studi Sistem Informasi pada Politeknik STMI Jakarta

OLEH

AGUSTINA SIREGAR

1312043



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2016**

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PEMELIHARAAN MESIN PADA PRODUKSI FRONT
FORK 2 W BERBASIS WEB MENGGUNAKAN PHP
5.6.3 DAN MYSQL 5.6.21 DI PT SHOWA
INDONESIA MANUFACTURING.

Disusun Oleh :
Nama : Agustina Siregar
Nim : 1312043
Program Studi : Sistem Informasi

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia pada hari Rabu tanggal 02 November 2016.

Jakarta, November 2016

Dosen Pembimbing

Ketua Penguji

Ulil Hamida ST, MT
NIP. 198103272005022001

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

Dosen Penguji

Dosen Penguji

Dr. Rizky Kramandita, S. Kom, MT
NIP. 197403022002121001

Dedy Trisanto, S. Kom, MMSI
NIP. 197805052005021002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PEMELIHARAAN MESIN PADA PRODUKSI FRONT
FORK 2 W BERBASIS WEB MENGGUNAKAN PHP
5.6.3 DAN MYSQL 5.6.21 DI PT SHOWA
INDONESIA MANUFACTURING.

Disusun Oleh :
Nama : Agustina Siregar
Nim : 1312043
Program Studi : Sistem Informasi
Tanggal Seminar : Jumat, 14 Oktober 2016
Tanggal Sidang : Rabu, 02 November 2016
Tanggal Lulus : Rabu, 02 November 2016

Jakarta, 14 November 2016

Menyetujui
Dosen Pembimbing

Ulil Hamida ST, MT
NIP. 198103272005022001

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PEMELIHARAAN MESIN PADA PRODUKSI FRONT
FORK 2 W BERBASIS WEB MENGGUNAKAN PHP
5.6.3 DAN MYSQL 5.6.21 DI PT SHOWA
INDONESIA MANUFACTURING.

Disusun Oleh :
Nama : Agustina Siregar
Nim : 1312043
Program Studi : Sistem Informasi
Tanggal Seminar : Jumat, 14 Oktober 2016
Tanggal Sidang : Rabu, 02 November 2016
Tanggal Lulus : Rabu, 02 November 2016

Jakarta, 14 November 2016

Menyetujui
Asisten Dosen Pembimbing

Noveriza Yuliasari, Ssi, MT
NIP. 197811212009012003

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Agustina Siregar

Nim : 1312043

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 Pemeliharaan Mesin Pada Produksi Front Fork 2 W Berbasis Web Menggunakan PHP 5.6.3 Dan MYSQL 5.6.21 Di PT Showa Indonesia Manufacturing”

- **Dibuat** dan diselesaikan sendiri, dengan menggunakan literatur hasil kuliah, survei lapangan, dosen pembimbing dan asisten dosen pembimbing, melalui tanya jawab maupun asistensi serta buku-buku acuan yang tertera dalam referensi pada karya Tugas Akhir ini.
- **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 karya Tugas Akhir ini.

Jika terbukti saya tidak memenuhi apa yang telah saya nyatakan di atas, maka saya bersedia menerima sanksi atas apa yang telah saya lakukan sesuai peraturan yang berlaku.

Jakarta, November 2016

Agustina Siregar

ABSTRAK

PT Showa Indonesia Manufacturing merupakan perusahaan yang bergerak di bidang industri otomotif yang berfokus dalam produksi *shock absorber* yaitu *front fork* (peredam kejut bagian depan) dan *rear cushion*. PT Showa Indonesia Manufacturing membutuhkan pengembangan sebuah sistem informasi permintaan pemeliharaan mesin mengenai pelaporan pemeliharaan mesin. Sistem informasi pemeliharaan mesin yang ada pada PT Showa Indonesia Manufacturing dalam pengolahan datanya masih menggunakan *Microsoft Excel* sehingga membutuhkan waktu yang lama dalam penginputan data permintaan karena dilakukan satu persatu. Selain itu, terjadi kesulitan dalam penyimpanan data memo order pekerjaan karena dokumen hanya disimpan dalam lemari arsip. Rancang bangun sistem informasi pemeliharaan mesin sangat diperlukan dalam perusahaan untuk melakukan proses pengolahan data pemeliharaan mesin. Sistem informasi yang diusulkan akan mempermudah dan mengintegrasikan proses pengolahan data sehingga membantu bagian di dalam sistem untuk saling bertukar informasi dan mengambil keputusan dengan cepat. Pengembangan sistem informasi pemeliharaan mesin menggunakan metode *evolutionary prototype*. Pemodelan sistem menggunakan *unified modeling language* (UML), *hierarchy plus input-process-output* (HIPO), dan *flowchart*. Perancangan dan pembuatan aplikasi sistem informasi pemeliharaan mesin ini menggunakan PHP 5.6.3 dan MySQL 5.6.21. Sistem informasi pemeliharaan mesin diharapkan dapat memudahkan perusahaan dalam proses pengolahan data terutama dalam pengolahan laporan *monitoring* pemeliharaan mesin dan memudahkan dalam menyimpan data pada bagian *maintenance*. Untuk penerapan sistem baru, disarankan untuk melakukan sosialisasi kepada bagian terkait dan pemeliharaan aplikasi agar dapat berjalan dengan baik.

Kata Kunci : sistem informasi, pemeliharaan mesin, pelaporan *monitoring*, pemeliharaan, *evolutionary prototype*, PHP 5.6.3, MySQL 5.6.21, UML.

KATA PENGANTAR

Puji syukur penulis panjatkan bagi Tuhan Yang Maha Esa, yang atas berkat dan kasih-Nya sehingga penulis mampu menyelesaikan Tugas Akhir dengan judul **“Rancang Bangun Sistem Informasi Pemeliharaan Mesin Pada Produksi *Front Fork 2 W* Berbasis Web Menggunakan PHP 5.6.3 Dan MYSQL 5.6.21 Di PT Showa Indonesia Manufacturing”**.

Penulisan Tugas Akhir ini bertujuan untuk memenuhi sebagian syarat penyelesaian program D-IV pada program studi Sistem Informasi pada Politeknik STMI Jakarta.

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

1. Bapak Mustofa, ST, MT. Selaku ketua Politeknik STMI Jakarta.
2. Bapak Drs. Jacob Saragih, MM. Selaku Ketua Jurusan Sistem Informasi Industri Otomotif.
3. Ibu Ulil Hamida, ST, MT. Selaku dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
4. Ibu Noveriza, S.Si, MT. Selaku asisten dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
5. Bapak Budiman Siregar dan Mama Longseri Silaban. Selaku orang tua penulis yang selalu mendoakan, memberikan motivasi dan pengorbanannya baik segi moril, materi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
6. Kakak dan adik, Febrina, Helen, Marlina, Risma, dan Daniel. Selaku saudara penulis yang selalu memberikan motivasi dan dukungan moril serta doa sehingga Tugas Akhir ini dapat diselesaikan dengan baik.

7. Bapak Sabda dan Bapak Agus selaku pembimbing di PT Showa Indonesia Manufacturing serta seluruh pekerja khususnya *Shock Absorber* Produksi *Front Fork* yang telah membimbing dan memberikan informasi yang berguna kepada penulis dalam penyusunan Tugas Akhir.
8. Seluruh dosen Sekolah Tinggi Manajemen Industri yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
9. Seluruh teman-teman jurusan Sistem Informasi angkatan 2012, khususnya SA02, yang telah memberikan semangat dan motivasi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
10. Serta semua pihak yang telah membantu penulis dan yang tidak dapat disebutkan satu per satu.

Semoga Tuhan Yang Maha Esa membalas kebaikan dan ketulusan semua pihak yang telah membantu dan penulis berharap semoga Tugas Akhir ini dapat memberikan manfaat dan informasi yang berguna bagi pembaca. Terima kasih.

Jakarta, Oktober 2016

Penulis,
Agustina Siregar

DAFTAR ISI

	Halaman
Halaman Judul	
Lembar Persetujuan Dosen Pembimbing	
Lembar Persetujuan Asisten Dosen Pembimbing	
Lembar Bimbingan Dengan Dosen Pembimbing	
Lembar Bimbingan Dengan Asisten Dosen Pembimbing	
Lembar Pernyataan Keaslian	
Abstrak	vii
Kata Pengantar	viii
Daftar Isi	x
Daftar Gambar	xiv
Daftar Tabel	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan	2
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat Tugas Akhir	3
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Konsep Dasar Sistem	6
2.1.1 Karakteristik Sistem	6
2.1.2 Klasifikasi Sistem	8
2.2 Konsep Dasar Informasi	9
2.2.1 Siklus Informasi	10
2.2.2 Nilai Informasi	11
2.3 Sistem Informasi	13

2.4	Konsep Dasar Pemeliharaan	14
2.4.1	Tujuan Pemeliharaan	14
2.4.2	Konsep <i>Breakdown</i> dan <i>Downtime</i>	15
2.4.3	Kegiatan-kegiatan <i>Maintenance</i>	17
2.5	Mesin.....	18
2.6	Pengertian <i>Shock Absorber Front Fork</i>	19
2.7	Laporan	20
2.7.1	Fungsi Laporan.....	20
2.7.2	Elemen Laporan	21
2.7.3	Tipe Laporan	22
2.8	Pengembangan Sistem	22
2.8.1	Metode-metode Pengembangan Sistem	23
2.8.2	<i>System Development Life Cycle (SLDC)</i>	23
2.9	Model Prototipe.....	24
2.9.1	Prototipe Evolusioner	26
2.9.2	<i>Requirement Prototype</i>	26
2.10	<i>Flowchart</i>	27
2.11	<i>Unified Modelling Language (UML)</i>	29
2.11.1	<i>Use Case Diagram</i>	30
2.11.2	<i>Activity Diagram</i>	32
2.11.3	<i>Sequence Diagram</i>	33
2.11.4	<i>Class Diagram</i>	34
2.11.5	<i>Deployment Diagram</i>	36
2.12	Kamus Data	36
2.13	HIPO (<i>Hierarchy plus Input-Process-Output</i>).....	38
2.14	PHP (<i>Hypertext Pre-Processor</i>).....	40
2.15	<i>Database</i>	41
2.15.1	MySQL	41
2.15.2	Tipe Data My SQL	42

BAB III	METODOLOGI PENELITIAN	44
3.1	Metodologi Penelitian	44
3.2	Identifikasi Masalah	44
3.3	Pengumpulan Data	45
3.4	Metode Pengembangan Sistem	46
3.5	Kerangka Penelitian	47
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA	51
4.1	Sekilas Tentang PT Showa Indonesia Manufacturing	51
4.2	Profil Perusahaan	52
4.3	Visi, Misi dan Tujuan Pendirian Perusahaan	53
4.4	Struktur Organisasi Perusahaan	53
4.5	Peran dan Tanggung Jawab	56
4.6	Sistem Kerja Karyawan.....	60
4.7	Fasilitas-Fasilitas Perusahaan	61
4.8	Mesin <i>Assembling</i> Produksi <i>Front Fork</i>	62
4.9	Analisi Dokumen	63
4.10	Analisi Sistem Pemeliharaan Mesin	65
4.10.1	Prosedur Pemeliharaan Mesin yang Sedang Berjalan	66
4.10.2	Sistem Pemodelan Pemeliharaan Mesin yang Berjalan	68
4.11	Evaluasi Sistem yang Sedang Berjalan	71
BAB V	ANALISIS DAN PEMBAHASAN	72
5.1	Analisis Kebutuhan Sistem	72
5.2	Prosedur Sistem Informasi Pemeliharaan Mesin Usulan	73
5.3	Analisis dan Perancangan Sistem Pemeliharaan Mesin Usulan	75
5.3.1	<i>Use Case Diagram</i>	75
5.3.2	<i>Activity Diagram</i>	82
5.3.3	<i>Sequence Diagram</i>	86

5.3.4	<i>Class Diagram</i>	92
5.3.5	Kamus Data	93
5.3.6	<i>Deployment Diagram</i>	96
5.4	Analisis Desain Program	98
5.4.1	HIPO (<i>Hierarchy plus Input-Process-Output</i>).....	98
5.4.2	<i>Flowchart</i> Program	99
5.4.3	Perancangan <i>Interface</i> Program	101
5.5	Implementasi Sistem <i>Software</i> dan <i>Hardware</i>	110
BAB VI	KESIMPULAN DAN SARAN	112
6.1	Kesimpulan	112
6.2	Saran	112
DAFTAR PUSTAKA	113

DAFTAR GAMBAR

		Halaman
Gambar II.1	Karakteristik Sistem	8
Gambar II.2	Siklus Informasi	10
Gambar II.3	Pengembangan Prototipe Evolusioner.....	26
Gambar II.4	Pengembangan <i>Prototype Requirement</i>	27
Gambar II.5	Klasifikasi Diagram UML	30
Gambar II.6	Contoh <i>Use Case Diagram</i>	31
Gambar II.7	Contoh <i>Activity Diagram Login</i>	32
Gambar II.8	Contoh <i>Sequence Diagram Login</i>	34
Gambar II.9	Contoh <i>Class Diagram</i>	36
Gambar II.10	<i>Visual Table Of Contents</i>	39
Gambar II.11	<i>Overview Diagram</i>	39
Gambar III.1	Kerangka Penelitian	50
Gambar IV.1	Struktur Organisasi PT Showa Indonesia Manufacturing.....	54
Gambar IV.2	Struktur Organisasi Produksi 2 W	55
Gambar IV.3	Struktur Organisasi Seksi <i>Assembling</i>	56
Gambar IV.4	Lembar Pertama Memo Order Pekerjaan.....	63
Gambar IV.5	Lembar Kedua Memo Order Pekerjaan	64
Gambar IV.6	Lembar Ketiga Memo Order Pekerjaan	64
Gambar IV.7	<i>Flowmap</i> yang Sedang Berjalan di Seksi <i>Assembling</i>	67
Gambar IV.8	<i>Use Case Diagram</i> Sistem Informasi Permintaan Mesin	70
Gambar V.1	<i>Flowmap</i> Sistem Informasi Pemeliharaan Mesin pada Produksi <i>Front Fork</i>	74
Gambar V.2	<i>Use Case Diagram</i> Sistem Informasi Pemeliharaan Mesin Usulan	78
Gambar V.3	<i>Activity Diagram Login</i>	82
Gambar V.4	<i>Activity Diagram</i> Mengelola Data Master	83
Gambar V.5	<i>Activity Diagram</i> Memasukan Data Order Pemeliharaan	84

Gambar V.6	<i>Activity Diagram View Order Pemeliharaan</i>	84
Gambar V.7	<i>Activity Diagram Menyetujui Data Order Pemeliharaan</i>	85
Gambar V.8	<i>Activity Diagram Laporan Monitoring Order Pemeliharaan</i> ..	85
Gambar V.9	<i>Activity Diagram Mencetak Laporan</i>	86
Gambar V.10	<i>Sequence Diagram Proses Login</i>	87
Gambar V.11	<i>Sequence Diagram Mengelola Data Master</i>	88
Gambar V.12	<i>Sequence Diagram Memasukan Data Order Pemeliharaan</i>	89
Gambar V.13	<i>Sequence Diagram Menyetujui Data Order Pemeliharaan</i>	89
Gambar V.14	<i>Sequence Diagram View Data Order Pemeliharaan</i>	90
Gambar V.15	<i>Sequence Diagram Laporan Monitoring Pemeliharaan</i>	90
Gambar V.16	<i>Sequence Diagram Mencetak Laporan Monitoring</i>	91
Gambar V.17	<i>Sequence Diagram Logout</i>	92
Gambar V.18	<i>Class Diagram Usulan</i>	93
Gambar V.19	<i>Deployment Diagram</i>	97
Gambar V.20	HIPO Sistem Informasi Pemeliharaan Mesin	99
Gambar V.21	Program <i>Logic Flowchart</i> Aplikasi Sistem Informasi Pemeliharaan Mesin	100
Gambar V.22	<i>Interface Form Login</i>	101
Gambar V.23	<i>Interface Form Menu Utama</i>	102
Gambar V.24	<i>Interface Form Data Master User</i>	102
Gambar V.25	<i>Interface Form Tambah Data Master User</i>	102
Gambar V.26	<i>Interface Form Data Lini</i>	103
Gambar V.27	<i>Interface Form Form Tambah Data Lini</i>	103
Gambar V.28	<i>Interface Form Data Karyawan</i>	104
Gambar V.29	<i>Interface Form Data Tambah Karyawan</i>	104
Gambar V.30	<i>Interface Form Data Mesin</i>	105
Gambar V.31	<i>Interface Form Tambah Data Mesin</i>	105
Gambar V.32	<i>Interface Form Data Shift</i>	106
Gambar V.33	<i>Interface Form Tambah Data Shift</i>	106
Gambar V.34	<i>Interface View Pembuatan Data Order Pemeliharaan</i>	106
Gambar V.35	<i>Interface Form Tambah Data Order Pemeliharaan</i>	107

Gambar V.36	<i>Interface Form Status Persetujuan</i>	107
Gambar V.37	<i>Interface Form Status Persetujuan Group Head</i>	108
Gambar V.38	<i>Interface Form Pilih Status Persetujuan</i>	108
Gambar V.39	<i>Interface Form Status Persetujuan Department Head</i>	108
Gambar V.40	<i>Interface Form Progres Order Pemeliharaan</i>	109
Gambar V.41	<i>Interface Form Pilih Npk Monitoring Pemeliharaan</i>	109
Gambar V.42	<i>Interface Form Laporan Monitoring Pemeliharaan</i>	109
Gambar V.43	<i>Interface Form Klasifikasi Order Pemeliharaan</i>	110
Gambar V.44	<i>Interface Form Laporan Klasifikasi Order Pemeliharaan</i>	110
Gambar V.45	<i>Interface Form Rekap Mesin Pemeliharaan</i>	110
Gambar V.46	<i>Interface Form Laporan Rekap Mesin Pemeliharaan</i>	111

DAFTAR TABEL

		Halaman
Tabel II.1	Simbol <i>Flow Direction</i>	28
Tabel II.2	Simbol Proses	28
Tabel II.3	Simbol <i>Input dan Output</i>	29
Tabel II.4	Simbol-Simbol <i>Use Case Diagram</i>	30
Tabel II.5	Simbol-Simbol <i>Activity Diagram</i>	32
Tabel II.6	Simbol-Simbol <i>Sequence Diagram</i>	33
Tabel II.7	Simbol-Simbol <i>Class Diagram</i>	35
Tabel II.8	Tipe <i>Multiplicity</i>	35
Tabel II.9	Simbol-Simbol Kamus Data	36
Tabel II.10	Tipe Data MySQL	43
Tabel IV.1	Mesin Seksi <i>Assembling</i>	62
Tabel IV.2	Laporan Monitoringg Pemeliharaan Mesin Produksi <i>Front Fork 2 W</i>	65
Tabel IV.3	Definisi Aktor Sistem Informasi Permintaan Pemeliharaan	68
Tabel IV.4	Definisi Use Case Sistem Informasi yang Berjalan	69
Tabel IV.5	Evaluasi Sistem yang Berjalan	71
Tabel V.1	Identifikasi Kebutuhan Sistem	72
Tabel V.2	Definisi Aktor <i>Use Case Diagram</i> Sistem Usulan	76
Tabel V.3	Definisi <i>Use Case Diagram</i> Sistem Usulan	76
Tabel V.4	Skenario <i>Use Case Login</i>	79
Tabel V.5	Skenario <i>Use Case</i> Mengelola Data <i>Master</i>	79
Tabel V.6	Skenario <i>Use Case</i> Memasukan Data Order Pemeliharaan	80
Tabel V.7	Skenario <i>Use Case</i> Menyetujui Data Order Pemeliharaan	80
Tabel V.8	Skenario <i>Use Case View</i> Order Pemeliharaan	81
Tabel V.9	Skenario <i>Use Case View</i> Laporan <i>Monitoring</i> Pemeliharaan ..	81
Tabel V.10	Skenario <i>Use Case</i> Mencetak Laporan <i>Monitoring</i> Pemeliharaan	82

Tabel V.11	Tabel <i>User</i>	94
Tabel V.12	Tabel Karyawan	94
Tabel V.13	Tabel Mesin	95
Tabel V.14	Tabel Lini	95
Tabel V.15	Tabel <i>Shift</i>	95
Tabel V.16	Tabel Keputusan	96
Tabel V.17	Tabel Order Pemeliharaan	96

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem informasi dan teknologi komputer berkembang sangat pesat sejalan dengan besarnya kebutuhan terhadap informasi. Perkembangan teknologi informasi tidak lepas dari pesatnya teknologi komputer yang dibutuhkan dalam suatu organisasi. Pemanfaatan sistem informasi dalam suatu organisasi dapat optimal apabila direncanakan dengan baik dalam suatu perencanaan strategis. Salah satu strategi dalam meningkatkan daya saing adalah melalui penerapan peran sistem informasi dalam perusahaan.

Penerapan sistem informasi dalam dunia bisnis banyak dimanfaatkan untuk mendukung kecepatan dan ketepatan proses bisnis. Saat ini kegiatan proses bisnis banyak yang dipermudah dan dipercepat oleh penggunaan sistem informasi. Sehingga perusahaan harus memiliki sistem yang bisa menghasilkan informasi yang tepat dan akurat agar dapat mempermudah dalam mengambil suatu keputusan. Penerapan sistem informasi dalam perusahaan sangat penting, hal tersebut dapat diterapkan antara lain pada bagian keuangan, administrasi, produksi untuk meningkatkan proses bisnis, dan pada bagian pemeliharaan mesin dalam perusahaan industri.

PT Showa Indonesia Manufacturing adalah sebuah perusahaan industri yang bergerak di bidang manufaktur otomotif. Perusahaan ini memproduksi *shock absorber* yang berlokasi di Cikarang. PT Showa Indonesia Manufacturing memiliki 6 gedung, yaitu gedung A, B, C, D, E, dan F. Pada setiap gedung menghasilkan *shock absorber* yang berbeda-beda baik roda 2 maupun roda 4.

Dalam memproduksi suatu produk, PT Showa Indonesia Manufacturing menggunakan tenaga sumber daya manusia (SDM) dan mesin yang berkualitas tinggi. Mesin dioperasikan oleh sumber daya manusia (SDM) atau dapat disebut juga sebagai *man power* untuk melakukan proses produksi.

Sistem pemeliharaan mesin yang digunakan oleh PT Showa Indonesia Manufacturing saat ini adalah sistem pengolahan data yang menggunakan memo order pekerjaan untuk pemeliharaan mesin. Pengumpulan data dilakukan secara manual yang dicatat pada kertas slip memo order pekerjaan. Hal yang dicatat adalah kondisi mesin yang membutuhkan pemeliharaan dan penyelesaian untuk setiap masalah terhadap mesin produksi *front fork* di Seksi *Assembling 2 W*. Pelaporan order pekerjaan pemeliharaan mesin dapat dilakukan oleh operator dan *section head* yang akan dilaporkan pada Departemen *Maintenance*, sehingga dalam penyampaian informasi antara departemen membutuhkan waktu yang lama. Hal ini dapat menyebabkan keterlambatan dalam penyampaian data dan menyebabkan produksi tidak maksimal.

Berdasarkan latar belakang di atas, maka tugas akhir ini berjudul “Rancang Bangun Sistem Informasi Pemeliharaan Mesin Produksi *Front Fork 2 W* Berbasis Web Menggunakan PHP 5.6.3 Dan MySQL 5.6.21 Di PT Showa Indonesia Manufacturing”.

1.2 Pokok Permasalahan

Adapun pokok permasalahan setelah mengamati sistem pemeliharaan mesin produksi yang berjalan pada PT Showa Indonesia Manufacturing adalah sebagai berikut:

1. Dokumen yang berjalan pada seksi *Assembling* produksi *front fork 2 W* masih menggunakan memo permintaan berupa kertas memo order pekerjaan, yang membutuhkan waktu yang cukup lama untuk melakukan pengisian atau penulisan sebuah memo order pemeliharaan, sehingga dapat terjadi resiko kehilangan dan kerusakan pada memo.
2. Seksi *Assembling* produksi *front fork 2 W* belum menggunakan basis data sehingga menyebabkan kesulitan dalam pengolahan data, serta pengelolaan laporan *monitoring* order pemeliharaan yang tersimpan dalam Ms. *Excel*.

1.3 Tujuan Penelitian

Adapun tujuan dari tugas akhir ini adalah sebagai berikut:

1. Membuat memo permintaan pemeliharaan mesin pada seksi *Assembling* produksi *front fork 2 W* di PT Showa Indonesia Manufacturing menjadi terkomputerisasi.
2. Merancang sistem informasi pemeliharaan mesin dan membuat basis data menggunakan MySQL sehingga memudahkan dalam penyimpanan data pemeliharaan mesin dan dalam pembuatan laporan *monitoring* order pemeliharaan.

1.4 Batasan Masalah

Dalam penyusunan tugas akhir agar lebih fokus dan terarah, maka perlu diadakan batasan-batasan sebagai berikut:

1. Tempat praktik adalah PT Showa Indonesia Manufacturing, Cikarang.
2. Penelitian dilakukan di Seksi *Assembling* PT Showa Indonesia Manufacturing, yaitu selama bulan September 2015.
3. Penelitian dan analisis hanya mengenai proses pemeliharaan mesin dan laporan *monitoring* pemeliharaan mesin.

1.5 Manfaat Tugas Akhir

Adapun manfaat yang dapat diperoleh dari penelitian tugas akhir ini adalah sebagai berikut:

1. Bagi Penulis
Sebagai alat untuk mempraktikkan teori-teori yang telah diperoleh selama perkuliahan, sehingga penulis dapat menambah pengetahuan secara praktis tentang masalah-masalah yang dihadapi oleh perusahaan.
2. Bagi Perusahaan
Hasil penelitian rancang bangun sistem informasi pemeliharaan mesin diharapkan dapat mempermudah dalam pengolahan data, membuat laporan

pemeliharaan dan membantu dalam penanganan setiap permasalahan mesin produksi.

3. Bagi Pihak Lain

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

1.6 Sistematika Penulisan

Adapun sistematika penulisan yang digunakan dalam tugas akhir ini terbagi dalam beberapa pokok bahasan yaitu:

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Bab ini membahas teori tentang pengertian sistem, informasi, sistem informasi, pemeliharaan, mesin, pengembangan sistem, *Diagram Alur (Flowchart)*, *Unified Modeling Language (UML)* sebagai alat bantu untuk membuat rancangan konseptual, pemrograman PHP, *database MYSQL*, XAMPP dan teori-teori lain yang berhubungan dengan perancangan sistem.

BAB III METODOLOGI PENELITIAN

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

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini menguraikan tentang hasil pengamatan dan data yang telah diperoleh, pengolahan data diagram alir sistem berjalan, dan *use case*

sistem berjalan berdasarkan penelitian selama melaksanakan penelitian di PT Showa Indonesia Manufacturing.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis kebutuhan sistem, pemodelan sistem dengan *Unified Modelling Language* (UML), kamus data, perancangan *flowchart* program dan perancangan antar muka.

BAB VI KESIMPULAN DAN SARAN

Pada bab ini dikemukakan kesimpulan-kesimpulan dari hasil penulisan tugas akhir dan saran-saran pada pihak perusahaan dalam berbagai hal yang berhubungan dengan sistem pemeliharaan mesin produksi *front fork* di Seksi *Assembling 2 W*.

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Pengertian sistem pada berbagai bidang berbeda-beda, tetapi meskipun istilah sistem yang digunakan bervariasi, semua sistem pada bidang-bidang tersebut mempunyai beberapa persyaratan umum, yaitu sistem harus mempunyai elemen, lingkungan, interaksi antar elemen, interaksi antara elemen dengan lingkungannya, dan yang terpenting adalah sistem harus mempunyai tujuan yang akan dicapai. Pengertian sistem menurut para ahli dapat didefinisikan sebagai berikut:

1. Sistem bisa berupa abstrak atau fisis. Sistem yang abstrak adalah susunan yang teratur dari gagasan-gagasan atau konsepsi yang saling bergantung, sedangkan sistem yang bersifat fisis adalah serangkaian unsur yang bekerjasama untuk mencapai suatu tujuan (Davis, 2012).
2. Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling bergantung satu sama lain, dan terpadu (Sutabri, 2012).
3. Sekumpulan komponen yang secara bersama-sama bekerja untuk mencapai suatu tujuan bersama (Hartono, 2013).

Berdasarkan beberapa pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa sistem adalah suatu kumpulan bagian-bagian baik manusia atau pun bukan manusia yang saling berinteraksi untuk mencapai suatu tujuan.

2.1.1 Karakteristik Sistem

Sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut (Mustakini, 2009):

1. **Komponen Sistem (*Component*)**
Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.
2. **Batasan Sistem (*Boundary*)**
Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.
3. **Lingkungan Luar Sistem (*Environment*)**
Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.
4. **Penghubung (*Interface*)**
Media yang menghubungkan sistem dengan subsistem yang lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.
5. **Masukan Sistem (*Input*)**
Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).
6. **Keluaran Sistem (*Output*)**
Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini dapat menjadi masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi *input* bagi subsistem lain.

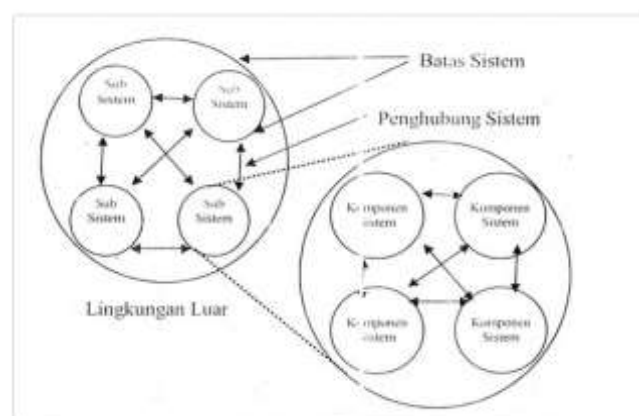
7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

8. Sasaran (*Objectives*) dan Tujuan (*Goal*)

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

Suatu sistem dikatakan berhasil apabila mengenai sasaran atau tujuannya. Jika suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya (lihat Gambar II.1).



Gambar II.1 Karakteristik Sistem
(Sumber: Mustakini, 2009)

4.1.4 MASUKAN SISTEM

Sistem dapat diklasifikasikan dari beberapa sudut pandang, di antaranya sebagai berikut (Yakub, 2012):

1. Sistem Abstrak (*Abstract System*)

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik.

2. Sistem fisik (*physical system*)

Sistem fisik adalah sistem yang ada secara fisik, sistem komputer, sistem akuntansi, sistem produksi, sistem sekolah, dan sistem transportasi merupakan contoh *physical system*.

3. Sistem tertentu (*deterministic system*)

Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi, interaksi antara bagian dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer sudah diprogramkan, merupakan contoh *deterministic system* karena program komputer dapat diprediksi dengan pasti.

4. Sistem tak tentu (*probabilistic system*)

Sistem tak tentu adalah suatu sistem yang kondisi masa depannya tidak dapat diprediksikan karena mengandung unsur probabilitas. Sistem arisan merupakan contoh *probabilistic system* karena sistem arisan tidak dapat diprediksikan dengan pasti.

5. Sistem tertutup (*close system*)

Sistem tertutup merupakan sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan. Sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan, misalnya reaksi kimia dalam tabung terisolasi. Sistem terbuka (*open system*) Sistem ini adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Sistem perdagangan merupakan contoh *open system*, karena dapat dipengaruhi oleh lingkungan.

2.2 Konsep Dasar Informasi

Pengertian informasi menurut para ahli dapat didefinisikan sebagai berikut:

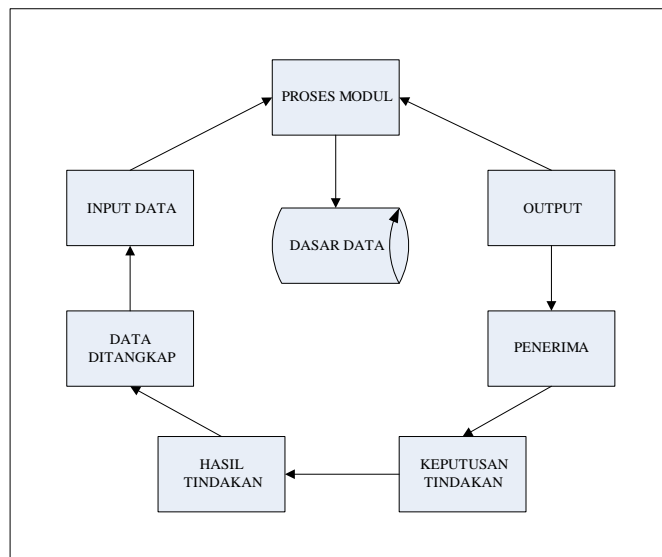
- a. Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan (Sutabri, 2012).
- b. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna bagi penerimanya (McLeod, 2012).

- c. Menurut Gordon B. Davis dalam bukunya Hartono (2013), Informasi adalah data yang telah diolah menjadi suatu bentuk yang berguna bagi penerimanya dan memiliki nilai bagi pengambilan keputusan saat ini atau di masa yang akan datang.

Berdasarkan beberapa pendapat yang dikemukakan dapat ditarik kesimpulan bahwa informasi merupakan data yang telah diolah, dibentuk, ataupun dimanipulasi sesuai dengan keperluan tertentu bagi penggunaannya.

2.2.1 Siklus Informasi

Data agar menjadi lebih berarti dan berguna dalam bentuk informasi maka perlu diolah menjadi suatu model tertentu. Data diolah melalui suatu model informasi. Si penerima akan menerima informasi tersebut untuk membuat suatu keputusan dan melakukan tindakan yang akan mengakibatkan munculnya sejumlah data lagi. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model, dan seterusnya sehingga membentuk suatu siklus. Siklus inilah yang disebut siklus informasi (*information cycle*) yang dapat dijelaskan pada Gambar II.2 (Sutabri, 2012).



Gambar II.2 Siklus Informasi
(Sumber: Sutabri, 2012)

2.2.2 Nilai Informasi

Nilai suatu informasi dapat ditentukan berdasarkan sifatnya. Menurut Sutarman (2012), berikut adalah nilai dari informasi:

1. Untuk memperoleh pemahaman dan manfaat.
2. Untuk mendapatkan pengalaman.
3. Pembelajaran yang terakumulasi sehingga dapat diaplikasikan dalam pemecahan masalah atau proses bisnis tertentu.
4. Untuk mengekstrak implikasi kritis dan merfleksikan pengalaman masa lampau yang menyediakan pengetahuan yang terorganisasi dengan nilai yang tinggi. Nilai ini bisa menghindari seorang manajer dari membuat kesalahan yang sama yang dilakukan oleh manajer lain sebelumnya.
5. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Sebagian besar informasi tidak dapat ditaksir keuntungannya dengan suatu nilai uang, tetapi dapat ditaksir nilai efektivitasnya.

Menurut Sutabri (2012), nilai dari informasi ditentukan dari manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai apabila manfaat yang diperoleh lebih berharga dibandingkan dengan biaya untuk mendapatkannya. Berikut ini adalah nilai informasi berdasarkan atas 10 (sepuluh) sifat, yaitu:

1. Mudah diperoleh
Sifat ini menunjukkan informasi dapat diperoleh dengan mudah dan cepat. Kecepatan memperoleh dapat diukur, misalnya 1 menit versus 24 jam. Akan tetapi beberapa nilainya bagi pemakai informasi sulit mengukurnya.
2. Luas dan lengkap
Sifat ini menunjukkan lengkapnya isi informasi. Hal ini tidak berarti hanya mengenai volumenya, tetapi juga mengenai keluaran informasinya. Sifat ini sangat kabur, karena itu sulit mengukurnya.
3. Ketelitian
Sifat ini menunjukkan minimnya kesalahan dalam informasi. Dalam hubungannya dengan volume data yang besar biasanya terjadi dua jenis kesalahan, yakni kesalahan pencatatan dan kesalahan perhitungan.
4. Kecocokan

Sifat ini menunjukkan seberapa baik keluaran informasi dalam hubungan dengan permintaan para pemakai. Isi informasi harus ada hubungannya dengan masalah yang sedang dihadapi, semua keluaran yang lainnya tidak berguna tetapi mahal mempersiapkannya.

5. Ketepatan waktu

Sifat ini menunjukkan tak ada keterlambatan jika ada yang sedang ingin mendapatkan informasi. Masukan, pengolahan dan pelaporan keluaran kepada para pemakai biasanya tepat waktu. Dalam beberapa hal, ketepatan waktu dapat diukur. Misalnya berapa banyak penjualan dapat ditambah dengan memberikan tanggapan segera kepada permintaan langganan mengenai tersedianya barang-barang inventaris.

6. Kejelasan

Sifat ini menunjukkan keluaran informasi yang bebas dari istilah-istilah yang tidak jelas.

7. Keluwesan

Sifat ini berhubungan dengan dapat disesuaikannya keluaran informasi tidak hanya beberapa keputusan, tetapi juga dengan beberapa pengambil keputusan. Sifat ini sulit diukur, tetapi dalam banyak hal dapat diberikan nilai yang dapat diukur.

8. Dapat dibuktikan

Sifat ini menunjukkan kemampuan beberapa pemakai informasi untuk menguji keluaran informasi dan sampai pada kesimpulan yang sama.

9. Tidak ada prasangka

Sifat ini berhubungan dengan tidak adanya keinginan untuk mengubah informasi guna mendapatkan kesimpulan yang telah dipertimbangkan sebelumnya.

10. Dapat diukur

Sifat ini menunjukkan hakikat informasi yang dihasilkan dari sistem informasi formal. Meskipun kabar angin, desas-desus, dugaan-dugaan, klenik, dan sebagainya sering dianggap informasi, hal-hal tersebut berada di luar lingkup pembicaraan.

2.3 Sistem Informasi

Pengertian sistem informasi menurut para ahli dapat didefinisikan sebagai berikut:

- i. Sistem informasi adalah sistem dapat didefinisikan dengan mengumpulkan, memproses, menyimpan, menganalisis, menyebarkan informasi untuk tujuan tertentu. Seperti sistem lainnya, sebuah sistem informasi terdiri atas *input* (data, instruksi) dan *output* (laporan, kalkulasi) (Sutarman, 2009).
- ii. Menurut Sutabri (2012), sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.
- iii. Sistem informasi adalah seperangkat komponen yang saling berhubungan, yang bekerja untuk mengumpulkan dan menyimpan data serta mengolahnya menjadi informasi yang digunakan (Hartono, 2013).

Berdasarkan beberapa pendapat yang dikemukakan dapat ditarik kesimpulan bahwa sistem informasi adalah sebuah sistem yang terdiri dari pengumpulan, pemasukan, pemrosesan data, penyimpanan, pengolahan, pengendalian dan pelaporan sehingga tercapai sebuah informasi yang mendukung pengambilan keputusan di dalam suatu organisasi untuk dapat mencapai sasaran dan tujuannya

2.4 Konsep Dasar Pemeliharaan

Pemeliharaan atau *maintenance* dapat diartikan sebagai kegiatan untuk memelihara, menjaga fasilitas peralatan pabrik dan mengadakan perbaikan atau penyesuaian atau penggantian yang diperlukan agar supaya terdapat suatu keadaan operasi produksi yang memuaskan sesuai dengan apa yang direncanakan, (Assauri, 2008). Pada umumnya sebuah produk yang dihasilkan oleh manusia, tidak ada yang tidak mungkin rusak, tetapi usia penggunaannya dapat

diperpanjang dengan melakukan perbaikan yang dikenal dengan pemeliharaan. Pemeliharaan dibedakan menjadi *preventive maintenance* dan *corrective maintenance*.

Preventive maintenance adalah kegiatan pemeliharaan dan perawatan yang dilakukan untuk mencegah timbulnya kerusakan-kerusakan yang tidak terduga dan menemukan kondisi atau keadaan yang dapat menyebabkan fasilitas produksi mengalami kerusakan pada waktu digunakan dalam proses produksi. *Preventive maintenance* disebut juga tindakan pencegahan atau *overhaul*, yaitu kegiatan pemeliharaan dan perawatan untuk mencegah kerusakan yang tak terduga dan menemukan kondisi atau keadaan yang menyebabkan fasilitas operasi lebih tepat.

Pemeliharaan *preventive* apabila direncanakan dengan baik dapat mencegah terjadinya kegagalan atau kerusakan, sebab apabila terjadi kerusakan peralatan operasi dapat berakibat kemacetan produksi secara total. *Preventive maintenance* sangat tepat dilakukan, karena kegunaannya sangat efektif dalam menghadapi fasilitas-fasilitas produksi yang termasuk dalam *critical unit*, yaitu peralatan atau fasilitas yang membahayakan kesehatan dan keselamatan kerja, mempengaruhi produk yang dihasilkan, dapat menyebabkan kemacetan seluruh proses produksi, dan apabila modal yang ditanam untuk fasilitas ini relatif lebih mahal.

Corrective maintenance, disebut juga *break down maintenance*, yaitu kegiatan pemeliharaan dan perawatan yang dilakukan setelah terjadi kerusakan, kegagalan, atau kelainan fasilitas produksi sehingga tidak dapat berfungsi dengan baik.

2.4.1 Tujuan Pemeliharaan

Menurut (Assauri, 2008), tujuan pemeliharaan mesin yang utama dapat didefinisikan sebagai berikut:

1. Kemampuan produksi dapat memenuhi kebutuhan sesuai dengan rencana produksi.
2. Menjaga kualitas pada tingkat yang tepat untuk memenuhi apa yang dibutuhkan oleh produk itu sendiri dan kegiatan produksi tidak terganggu.

3. Untuk membantu mengurangi pemakaian dan penyimpangan yang diluar batas dan menjaga modal yang diinvestasikan dalam perusahaan selama waktu yang ditentukan sesuai dengan kebijaksanaan perusahaan mengenai investasi tersebut.
4. Untuk mencapai tingkat biaya pemeliharaan serendah mungkin, dengan melaksanakan kegiatan *maintenance* secara efektif dan efisien keseluruhannya.
5. Menghindari kegiatan *maintenance* yang dapat membahayakan keselamatan para pekerja.
6. Mengadakan suatu kerja sama yang erat dengan fungsi-fungsi utama lainnya dari suatu perusahaan dalam rangka untuk mencapai tujuan utama perusahaan, yaitu tingkat keuntungan atau *return of investment* yang sebaik mungkin dan total biaya yang terendah.

2.4.2 Konsep *Breakdown* dan *Downtime*

Breakdown dapat didefinisikan sebagai berhentinya mesin pada saat produksi yang melibatkan *engineering* dalam perbaikan. Suatu mesin atau peralatan tidak dapat melakukan fungsinya lagi dengan baik, maka mesin atau peralatan tersebut dapat dikatakan mengalami kerusakan atau *breakdown*. *Downtime* didefinisikan sebagai waktu menganggur atau lama waktu dimana mesin tidak dapat lagi dijalankan untuk beroperasi sesuai dengan yang diharapkan (Assauri, 2008).

Downtime dapat didefinisikan sebagai waktu yang diperlukan selama peralatan atau mesin tidak dapat digunakan atau mesin mengalami kerusakan (gangguan), sehingga mesin atau peralatan tidak dapat menjalankan fungsinya sesuai dengan yang diharapkan dengan baik. *Breakdown* terjadi apabila suatu mesin atau peralatan mengalami kerusakan di mana kerusakan ini akan mempengaruhi kemampuan mesin secara keseluruhan dan menyebabkan terjadinya penurunan hasil proses dan juga akan mempengaruhi kualitas dari produk yang dihasilkan. Sedangkan *downtime* menunjukkan waktu yang

dibutuhkan mesin untuk mengembalikan kemampuan mesin untuk dapat menjalankan fungsi-fungsinya seperti semula.

Beberapa unsur di dalam konsep *downtime*:

1. *Maintenance delay*

Maintenance delay merupakan waktu yang diperlukan untuk menunggu ketersediaannya sumber daya *maintenance* yang akan melakukan proses perbaikan. Sumber daya *maintenance* dapat berupa teknisi, peralatan bantu, alat pengetesan, dan komponen pengganti.

2. *Supply delay*

Supply delay merupakan waktu yang dibutuhkan oleh *personal maintenance* untuk memperoleh komponen yang diperlukan dalam melakukan proses perbaikan. *Supply delay* terdiri dari *lead time* administrasi, *lead time* produksi, dan waktu transportasi komponen ke lokasi perbaikan.

3. *Access time*

Access time merupakan waktu yang dibutuhkan untuk memperoleh akses ke komponen yang mengalami kerusakan.

4. *Diagnosis time*

Diagnosis time merupakan waktu yang dibutuhkan untuk mengidentifikasi penyebab kerusakan yang terjadi serta mempersiapkan langkah-langkah yang diperlukan untuk memperbaiki kerusakan.

5. *Repair atau replacement time*

Repair atau replacement time merupakan waktu yang dibutuhkan untuk memperbaiki mesin agar mampu menjalankan fungsinya kembali dengan baik setelah mengetahui permasalahan dan mengakses ke komponen yang rusak.

6. *Verification and alignment*

Verification and alignment merupakan waktu yang digunakan untuk memastikan bahwa fungsi dari suatu mesin atau peralatan telah kembali seperti kondisi semula.

2.4.3 Kegiatan-Kegiatan *Maintenance*

Peranan *maintenance* adalah untuk menjaga agar perusahaan dapat bekerja secara optimal dengan menekan atau bahkan mengurangi kemacetan-kemacetan menjadi sekecil mungkin. Dalam setiap kegiatan yang akan dilakukan tidak terlepas dengan langkah-langkah untuk melakukan kegiatan tersebut. Kegiatan dari pada *maintenance* sebagai berikut (Assauri ,2008):

1. Inspeksi (*Inspection*)

Kegiatan inspeksi meliputi kegiatan pengecekan atau pemeriksaan secara berkala (*routine schedule check*) bangunan dan peralatan pabrik sesuai dengan rencana serta kegiatan pengecekan atau pemeriksaan terhadap peralatan yang mengalami kerusakan dan membuat laporan-laporan dari hasil pengecekan atau pemeriksaan tersebut.

2. Kegiatan Teknik (*Engineering*)

Kegiatan teknik meliputi kegiatan percobaan atas alat yang baru dibeli, dan kegiatan-kegiatan pengembangan peralatan atau komponen peralatan yang perlu diganti, serta melakukan penelitian-penelitian terhadap kemungkinan pengembangan tersebut. Dalam kegiatan teknik ini termasuk pula kegiatan penyelidikan sebab-sebab terjadinya kerusakan pada peralatan tertentu dengan cara-cara atau usaha-usaha untuk mengatasi/memperbaikinya yang sangat diperlukan dalam kegiatan produksi.

3. Kegiatan Produksi (*Production*)

Kegiatan produksi ini merupakan kegiatan pemeliharaan sebenarnya, yaitu mereparasi mesin-mesin dan peralatan. Kegiatan produksi ini dimaksudkan agar kegiatan pengolahan/pabrik dapat berjalan lancar sesuai dengan rencana, dan untuk ini diperlukan usaha-usaha perbaikan segera jika terdapat kerusakan pada peralatan.

4. Kegiatan Administrasi (*Clerical Work*)

Kegiatan administrasi ini merupakan kegiatan yang berhubungan dengan pencatatan mengenai biaya-biaya yang terjadi dalam melakukan pekerjaan pemeliharaan. Kegiatan pemeliharaan antara lain komponen (*spareparts*)

yang dibutuhkan, *progress report* tentang apa yang dikerjakan, waktu yang dilakukannya inspeksi dan perbaikan.

5. Pemeliharaan Bangunan (*House Keeping*)

Kegiatan pemeliharaan bangunan merupakan kegiatan untuk menjaga agar bangunan gedung tetap terpelihara dan terjamin kebersihannya.

2.5 Mesin

Mesin merupakan suatu fasilitas yang mutlak diperlukan perusahaan dalam berproduksi. Dengan menggunakan mesin, maka perusahaan dapat menekan tingkat kegagalan produknya, dapat meningkatkan standar kualitasnya, dapat mencapai ketepatan waktu dalam menyelesaikan produknya sesuai dengan permintaan pelanggan dan penggunaan sumber bahan baku akan lebih efisien karena dapat lebih terkontrol penggunaannya.

Mesin adalah suatu peralatan yang digerakan oleh suatu kekuatan atau tenaga yang dipergunakan untuk membantu manusia dalam mengerjakan produk atau bagian-bagian produk tertentu (Assauri, 2008). Mesin memiliki banyak variasi tetapi pada prinsipnya mesin-mesin ini dapat dibedakan atas dua macam yaitu (Assauri, 2008):

1. Mesin-mesin yang bersifat umum/serba guna (*general purpose machines*).
Mesin yang serba guna merupakan suatu mesin yang dibuat untuk mengerjakan pekerjaan-pekerjaan tertentu untuk berbagai jenis barang/produk atau bagian dari produk (*parts*). Ciri-ciri mesin yang serba guna, yaitu :
 - a. Mesin ini dibuat dengan bentuk standar dan selalu atas dasar pasar dan bukan atas dasar pesanan.
 - b. Mesin ini memproduksi dengan jumlah yang cukup besar sehingga harga dari mesin tersebut juga relatif lebih murah. Maka dari itu tidak jarang beberapa investor tertarik untuk berinvestasi di bidang ini karena cukup murah dan terjangkau.
 - c. Penggunaan mesin tersebut sangat fleksibel dan memiliki banyak variasi atau jenis.

- d. Diperlukan kegiatan pemeriksaan atau inspeksi atas apa yang dikerjakan pada mesin serba guna ini.
 - e. Biaya operasi produk tersebut lebih mahal.
 - f. Biaya pemeliharaan mesin serba guna ini lebih murah, karena bentuk mesin serba guna ini yang standar.
 - g. Mesin ini tidak mudah ketinggalan zaman.
2. Mesin yang bersifat khusus (*special purpose machine*)
- Mesin tersebut direncanakan dan dibuat untuk mengerjakan satu atau beberapa jenis kegiatan yang sama. Adapun beberapa ciri dari mesin yang bersifat khusus yaitu:
- a. Mesin tersebut dibuat atas dasar pesanan dan dalam jumlah yang sedikit atau kecil. Oleh karena itu juga, harga mesin-mesin tersebut biasanya relatif mahal, sehingga investasi dalam mesin ini menjadi mahal.
 - b. Mesin bersifat khusus ini biasanya otomatis, sehingga pekerjaannya lebih cepat dan biasanya dipergunakan dalam pabrik yang menghasilkan produknya dalam jumlah yang besar.
 - c. Biaya pemeliharaan dari mesin ini lebih mahal daripada mesin serba guna.
 - d. Biaya produksi per unit relatif lebih rendah.
 - e. Mesin ini mudah ketinggalan zaman.

Dengan adanya mesin yang dapat membantu manusia dalam melakukan proses produksi, manusia dapat memproduksi barang dalam jumlah yang sangat besar dengan waktu yang singkat. Bagi perusahaan perlu diperhatikan pemilihan mesin yang cocok untuk proses produksi, karena dengan pemilihan mesin secara bijak yang sesuai dengan kebutuhan dan kemampuan perusahaan maka proses produksi akan berjalan dengan baik sehingga produksi akan mencapai sasaran.

2.6 Pengertian *Shock Absorber Front Fork (Shock Breaker Depan)*

Shock absorber front fork atau *shock breaker* depan sepeda *motorshock absorber* adalah untuk meredam getaran atau kejutan dari roda bila melewati jalan yang tidak rata atau melewati lubang. Juga untuk membuat kendaraan lebih stabil saat melibas tikungan, melakukan rem.

Shock absorber adalah sebuah alat mekanik yang didesain untuk meredam hentakan yang disebabkan oleh energi kinetik. *Shock absorber* terdiri atas *piston rod* dan tabung. *Piston* memiliki fungsi penting dalam sistem kerja motor. Fungsi utama *piston* adalah menerima tenaga ledakan dari proses pembakaran dan meneruskan tenaga ke poros engkol menjadi tenaga putar dengan perantara batang piston. Atau mengubah tenaga panas menjadi tenaga mekanis. *Piston* bekerja untuk mengatur langkah kerja motor, baik motor dua langkah maupun motor empat langkah.

2.7 Laporan

Laporan adalah keluaran sistem informasi akuntansi dan berbentuk hasil cetak komputer dan tayangan pada layar monitor komputer (Mulyadi, 2008). Laporan adalah suatu bentuk penyampaian berita, keterangan, pemberitahuan ataupun pertanggungjawaban baik secara lisan maupun secara tertulis dari bawahan kepada atasan sesuai dengan hubungan wewenang dan tanggung jawab yang ada di antara mereka. Laporan berisi informasi yang didukung oleh data yang lengkap sesuai dengan fakta yang ditemukan, data disusun sedemikian rupa sehingga akurasi informasi yang kita berikan dapat dipercaya dan mudah dipahami (Soegito dalam Wardani 2008).

2.7.1 Fungsi Laporan

Laporan memiliki peran atau fungsi dalam suatu organisasi atau perusahaan, adapun fungsi laporan diantaranya sebagai berikut (Soedjadi, 2000):

1. Merupakan perwujudan dari tanggung jawab pelapor terhadap tugas yang dilimpahkan.

2. Sebagai alat untuk memperlancar kerjasama dan koordinasi maupun komunikasi yang saling mempengaruhi antara perseorangan dalam organisasi.
3. Sebagai alat untuk membuat anggaran, pelaksanaan, pengawasan, pengendalian, maupun pengambilan keputusan.
4. Sebagai alat untuk menukar informasi yang saling dibutuhkan oleh pekerja.

2.7.2 Elemen Laporan

Menurut Rama dan Jones (2008), elemen rancangan laporan terdiri dari:

1. *Report footer*, bagian diakhir laporan sering digunakan untuk menampilkan seluruh angka-angka ringkasan, seperti total semuanya dan untuk seluruh data dilaporan.
2. *Report header*, bagian laporan yang digunakan untuk mendapatkan informasi, seperti judul tanggal diawal laporan.
3. *Group footer*, bagian dari laporan yang mengikuti rincian kelompok.
4. *Group header*, bagian dari laporan yang mengikuti rincian kelompok dan biasanya digunakan untuk menentukan nama kelompok.
5. *Group detail*, laporan status perincian berkelompok.
6. *Group detail status report*, laporan yang menampilkan data ringkasan dan data *saldo* mengenai barang, jasa atau agen.
7. *Group event detail report*, laporan yang memiliki perincian kelompok dan yang menampilkan daftar kejadian selama satu periode biasanya disusun menurut barang, jasa, atau agen.

Menurut Narko (2004), elemen rancangan laporan terdiri dari:

1. *Header* laporan (*report header*) menunjukkan informasi yang diterapkan pada seluruh laporan (misalnya nama laporan perusahaan, tanggal laporan, dan nomor halaman).
2. *Header* halaman (*page header*) dapat digunakan untuk menentukan informasi yang tampak dibagian atas setiap halaman.

3. *Footer* halaman (*page footer*) tampak dibagian bawah setiap halaman dan biasanya mencakup nomor halaman
4. *Footer* laporan (*report footer*) terlihat satu kali dibagian akhir laporan biasanya digunakan untuk menyajikan informasi ringkasan seperti total semuanya
5. Perincian laporan (*report detail*) berisi informasi utama dilaporan, biasanya menyajikan data mengenai berbagai entitas (kejadian, agen, produk, dan jasa)

2.7.3 Tipe Laporan

Menurut Rama dan Jones (2008), laporan memiliki beberapa tipe yaitu:

1. *Simple event list*, laporan yang menyediakan daftar kejadian sederhana selama satu periode waktu yang disusun menurut tanggal kejadian atau nomor transaksi tanpa mengelompokkan atau sub total.
2. *Simple list*, satu daftar kejadian atau daftar acuan sederhana.
3. *Single entity report*, laporan yang hanya menyediakan perincian mengenai satu entitas seperti barang, jasa, agen atau kejadian.
4. *Status report*, laporan yang menyediakan data ringkasan mengenai barang, jasa atau agen.
5. *Summary report*, laporan yang meringkas data kejadian untuk sekelompok *record* terkait selama periode tertentu.
6. *Summary status report*, laporan yang mendaftarkan data acuan dan data ringkasan mengenai barang dan jasa atau agen.

1.1

1.2 2.8 Pengembangan Sistem

Pengembangan sistem (*System Development*) dapat berarti menyusun suatu sistem informasi yang benar-benar baru atau yang lebih sering terjadi, menyempurnakan sistem yang telah ada (Nugroho, 2004). Konsep siklus hidup sistem merupakan bagian dari langkah pengembangan. Siklus hidup sistem merupakan penerapan pendekatan sistem untuk tugas mengembangkan dan

menggunakan sistem berbasis komputer dilakukan dengan motivasi untuk memanfaatkan komputer sebagai alat bantu yang dikenal sebagai alat yang cepat, akurat, tidak cepat lelah, serta tidak mengenal kata bosan, untuk melaksanakan instruksi-instruksi pengguna.

1.9.1 2.8.1 Metode-metode Pengembangan Sistem

Keamanan sistem informasi yang berbasis komputer dapat dicapai salah satu diantaranya melalui penggunaan metode pengembangan sistem yang benar. Saat ini metode pengembangan sistem yang umum digunakan adalah metode pengembangan *System Development Life Cycle* (SDLC) atau sering juga disebut sebagai *Water Fall Method*, dan *Prototyping*.

2.8.2 System Development Life Cycle (SDLC)

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

Tahapan yang ada pada SDLC secara global adalah sebagai berikut:

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 dan mengembangkan kebutuhan *user*.

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*.

1.9.2 2.9 Model Prototipe

Model prototipe dapat digunakan untuk menyambungkan ketidakpahaman pelanggan mengenai hal teknis yang memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak (Rosa dan Shalahuddin, 2013).

Tahapan-tahapan pada model prototipe (*prototype model*) adalah sebagai berikut (Rosa dan Shalahuddin, 2013):

1. Mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat.
2. Membuat prototipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi.
3. Program prototipe selanjutnya dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*.

Kelemahan model prototipe adalah sebagai berikut (Rosa dan Shalahuddin, 2013):

1. *User* dapat sering mengubah-ubah atau menambah spesifikasi kebutuhan karena menganggap aplikasi sudah dengan cepat dikembangkan, karena adanya iterasi ini dapat menyebabkan pengembang banyak mengalah dengan *user* karena perubahan atau penambahan spesifikasi kebutuhan perangkat lunak.
2. Pengembang lebih sering mengambil kompromi dengan pelanggan untuk mendapatkan prototipe dengan waktu yang cepat sehingga pengembang lebih sering melakukan segala cara (tanpa idealis) guna menghasilkan prototipe untuk didemonstrasikan.

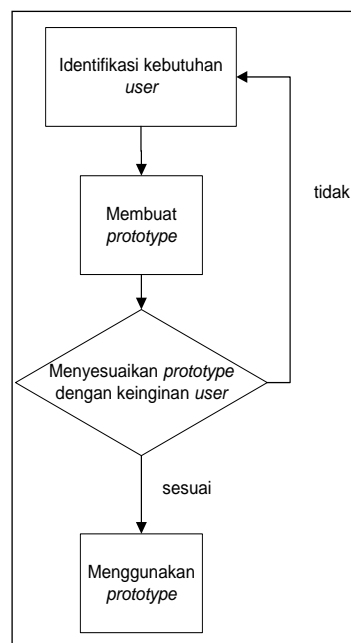
Prototipe adalah suatu versi sistem potensial yang disediakan bagi pengembang dan calon pengguna yang dapat memberikan gambaran bagaimana kira-kira sistem tersebut akan berfungsi bila telah disusun dalam bentuk yang lengkap (McLeod, 2011).

Proses dalam memproduksi suatu prototipe disebut *prototyping*. Tujuannya adalah menghasilkan prototipe secepat mungkin dan memperoleh

umpan balik dari pengguna yang akan memungkinkan prototipe untuk ditingkatkan sampai sistem dianggap sempurna. Adapun jenis *prototype* terdapat dua jenis prototipe yaitu prototipe evolusioner (*evolutionary prototype*) dan prototipe requirement (*requirement prototype*) (Mc Leod, 2011).

2.9.1 Prototipe Evolusioner

Prototipe evolusioner adalah prototipe yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi (McLeod, 2011). Ada empat langkah yang yang diambil dalam mengembangkan suatu prototipe evolusioner yaitu identifikasi kebutuhan pengguna, mengembangkan prototipe, menentukan prototipe dapat diterima atau tidak, dan penggunaan prototipe (lihat Gambar II.3).

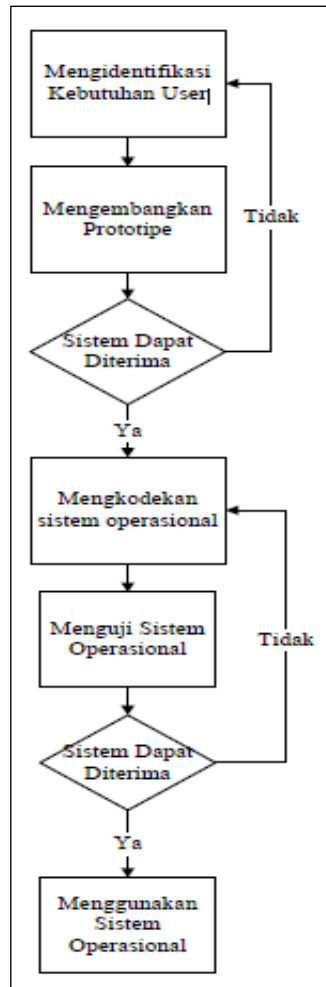


Gambar II.3 Pengembangan Prototipe Evolusioner
(Sumber: McLeod, 2011)

2.9.2 Requirement Prototype

Requirement prototype adalah suatu pengembangan untuk menentukan kebutuhan fungsional dari sistem baru pada saat para pengguna tidak mampu mengungkapkan dengan tepat apa yang mereka butuhkan. Saat kebutuhan telah

ditentukan prototipe requirement dapat mulai dikerjakan dan proyek siap untuk mengembangkan suatu sistem yang baru (lihat Gambar II.4).



Gambar II.4 Pengembangan Prototipe *Requirement*
(Sumber: McLeod, 2011)

2.10 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 (Sahid, 2010). *Flowchart* menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoprasian. (Febriani, 2015).

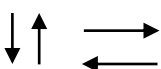


Simbol-simbol *flowchart* yang digunakan merupakan simbol-simbol *flowchart* standar yang dikeluarkan oleh ANSI dan ISO. Berikut simbol-simbol

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

1. *Flow Direction Symbols*

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

Tabel II.1 Simbol *Flow Direction*






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

Sumber: Febriani (2015)

2. Simbol Proses

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

Tabel II.2 Simbol Proses




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

Sumber: Febriani (2015)

3. Simbol *Input* dan *Output*

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

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

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

Sumber: Febriani (2015)

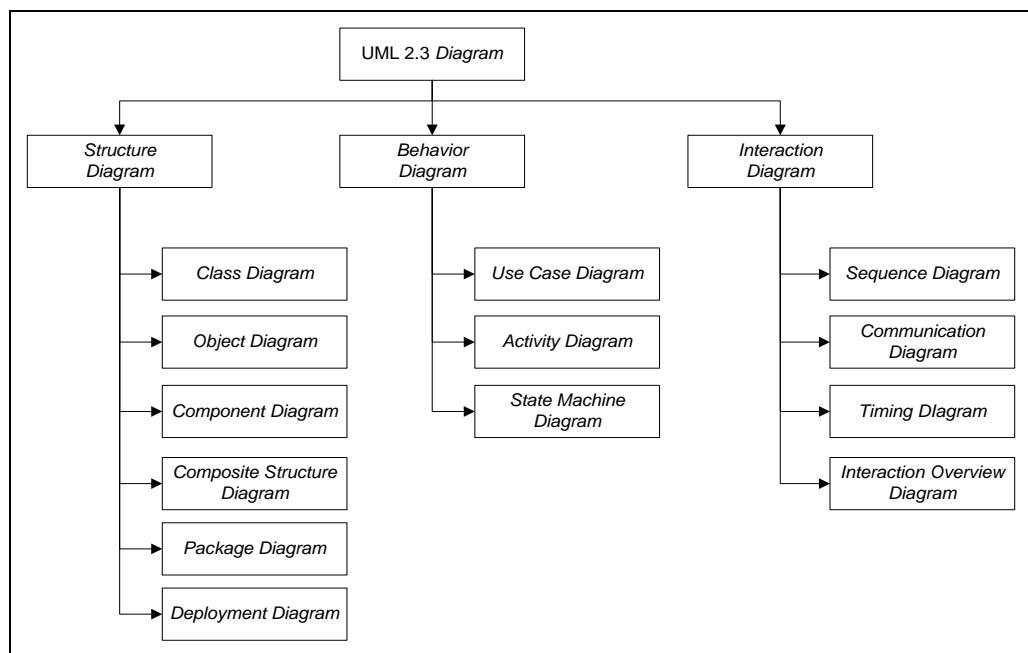
2.11 *Unified Modelling Language (UML)*

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. *UML (Unified Modeling Language)* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan (*modeling*) digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami (Nugroho, 2010).

UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Penggunaan UML dalam industri terus meningkat. UML merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri piranti lunak dan pengembangan sistem (Rama dan Jones, 2008). UML (*Unifed Modeling Language*) adalah sebuah standarisasi bahasa pemodelan untuk

pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek (Rosa dan Shalahuddin, 2013).

Pada UML terdiri dari 13 macam *diagram* yang dikelompokkan dalam 3 kategori. Pembagian kategori tersebut dapat dilihat pada Gambar II.5 berikut ini:

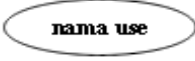




Gambar II.5 Klasifikasi *Diagram* UML
(Sumber: Rosa dan Shalahuddin, 2013)

2.11.1 Use Case Diagram

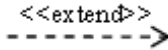

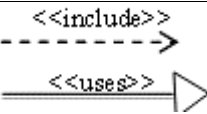
Use case mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (lihat Tabel II.4).

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

No.	Nama	Simbol	Keterangan
1.	<i>Use Case</i>		Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2.	Aktor		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi dibuat itu sendiri.

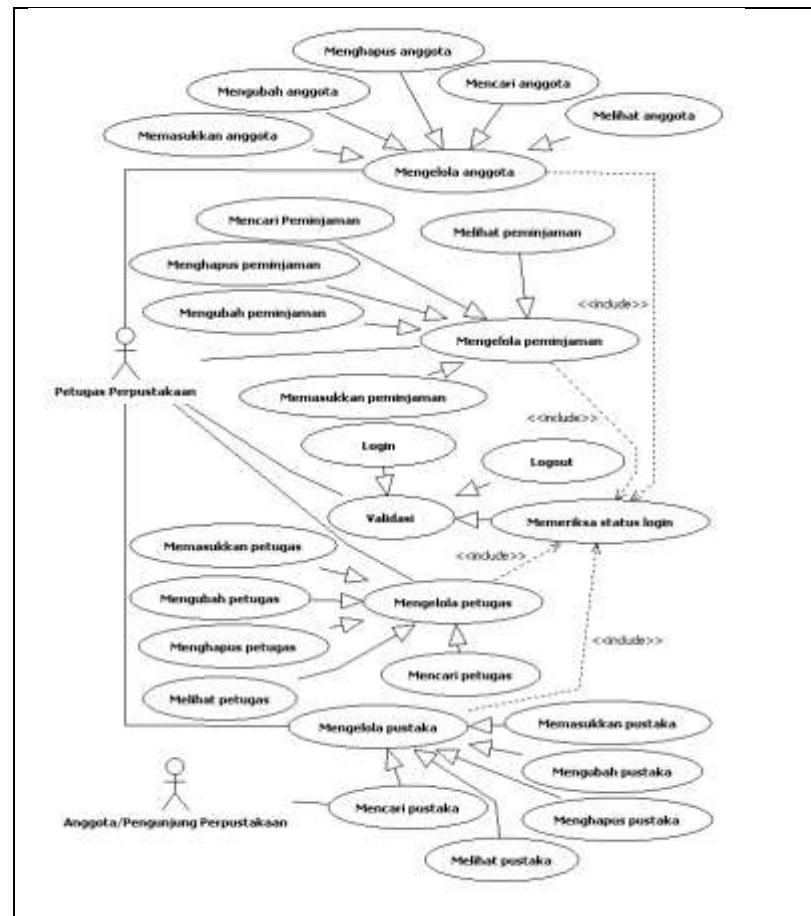
3.	Asosiasi/ <i>association</i>		Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
----	------------------------------	---	--

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

No.	Nama	Simbol	Keterangan
4.	Ekstensi/ <i>extend</i>		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5.	Generalisasi/ <i>generalization</i>		Hubungan generalisasi dan spesialisasi antara dua buah <i>use case</i> dimana fungsi yang satu fungsi yang lebih umum dari lainnya.
6.	Menggunakan/ <i>include/uses</i>		Fungsinya atau sebagai syarat dijalankan <i>use case</i> .

(Sumber: Rosa dan Shalahuddin, 2014)

Contoh gambaran *Use Case Diagram* dari *Use Case Diagram* Sistem Informasi Manajemen Perpustakaan dapat dilihat pada Gambar II.6.








Gambar II.6 Contoh *Use Case Diagram* Sistem Informasi Manajemen Perpustakaan
(Sumber: Rosa dan Shalahuddin, 2013)

2.11.2 Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2011) lihat Tabel II.5.

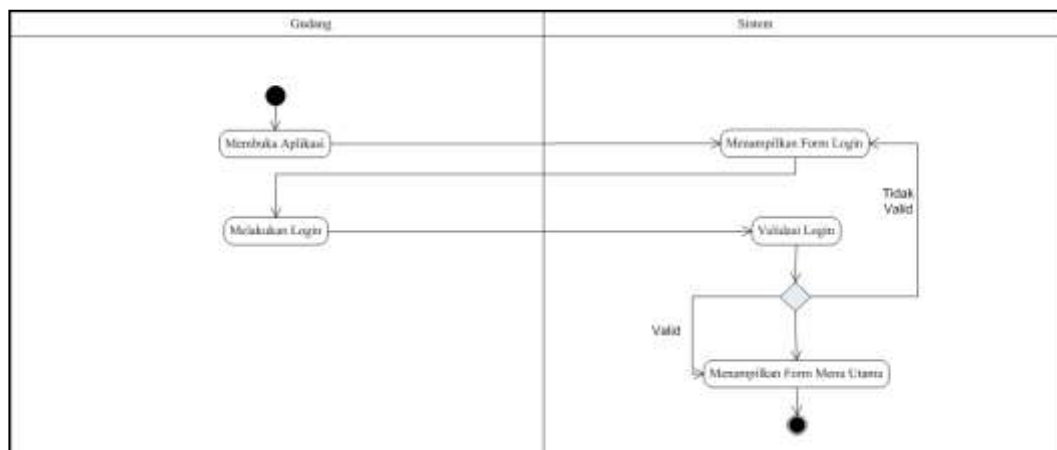
Tabel II.5 Simbol-simbol *Activity Diagram*

No	Nama	Simbol	Keterangan
1.	Status awal	●	Status awal aktivitas sistem, sebuah <i>diagram</i> aktivitas memiliki sebuah status awal.
2.	Aktivitas	○	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.

3.	Status akhir		Status akhir aktivitas sistem, sebuah <i>diagram</i> aktivitas memiliki sebuah status akhir.
4.	<i>Decision</i>		Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya.
5.	<i>Directional Association</i>		Menghubungkan antar proses yang saling berhubungan dan membentur suatu alur proses.
6.	<i>Fork Node</i>		Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.
7.	<i>Join Node</i>		Beberapa aliran yang pada tahap tertentu berubah menjadi satu aliran

(Sumber: Rosa dan Shalahuddin, 2011)

Contoh gambaran dari *Activity Diagram Login* lihat pada Gambar II.7.



Gambar II.7 Contoh *Activity Diagram Login*

(Sumber: Rosa dan Shalahuddin, 2013)

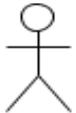

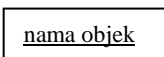

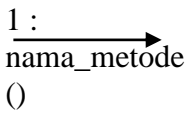
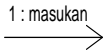
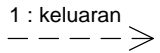
2.11.3 *Sequence Diagram*

Sequence diagram menggambarkan interaksi antar objek di dalam maupun di luar sistem berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu.

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima

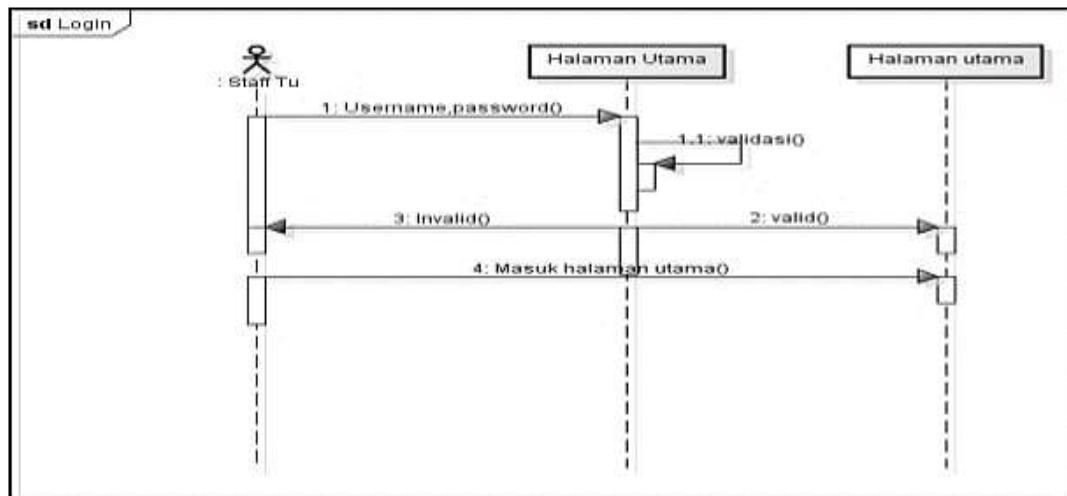
antar objek. Banyaknya *sequence diagram* yang harus digambar adalah minimal sebanyak pendefinisian *use case* (Rosa dan Shalahuddin, 2013) lihat Tabel II.6.

Tabel II.6 Simbol-simbol *Sequence Diagram*

No	Nama	Simbol	Keterangan
1.	<i>Actor</i>		Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi dibuat.
2.	<i>LifeLine</i>		Objek <i>entity</i> , antarmuka yang saling berinteraksi.
3.	Objek		Menyatakan objek yang berinteraksi pesan.
4.	Waktu aktif		Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
5.	Pesan tipe <i>call</i>		Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
6.	Pesan tipe <i>send</i>		Menyatakan suatu objek mengirimkan data ke objek lain.
7.	Pesan tipe <i>return</i>		Menyatakan suatu objek telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu.

(Sumber: Rosa dan Shalahuddin, 2013)

Contoh gambaran *Sequence Diagram* dari proses *Login* dapat dilihat pada Gambar II.8.



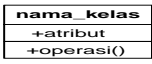

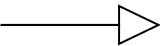
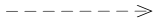

Gambar II.8 Contoh *Sequence Diagram Login*
(Sumber: Rosa dan Shalahuddin, 2013)

2.11.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Berikut adalah susunan struktur kelas (Rosa dan Shalahuddin, 2011), lihat Tabel II.7:

1. Kelas main
Kelas main adalah kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
2. Kelas yang menangani tampilan sistem
Kelas yang menangani tampilan sistem merupakan kelas yang mendefinisikan dan mengatur tampilan ke pemakai.
3. Kelas yang diambil dari pendefinisian *use case*
kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*.
4. Kelas yang diambil dari pendefinisian data
Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Tabel II.7 Simbol-Simbol *Class Diagram*

No	Nama	Simbol	Keterangan
1.	Kelas		Kelas pada struktur sistem.
2.	Asosiasi		Hubungan dimana salah satu kelas mengirimkan pesan kepada kelas lain.
3.	<i>Directional Association</i>		Menggambarkan bahwa pesan terjadi dari hanya salah satu kelas.
4	<i>Generalization</i>		Relasi antara dua kelas dengan makna umum khusus.
5.	Kebergantungan		Relasi antar kelas dengan makna kebergantungan antar kelas.
6.	Agregasi/aggregation		Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>).

(Sumber: Rosa dan Shalahuddin, 2011)

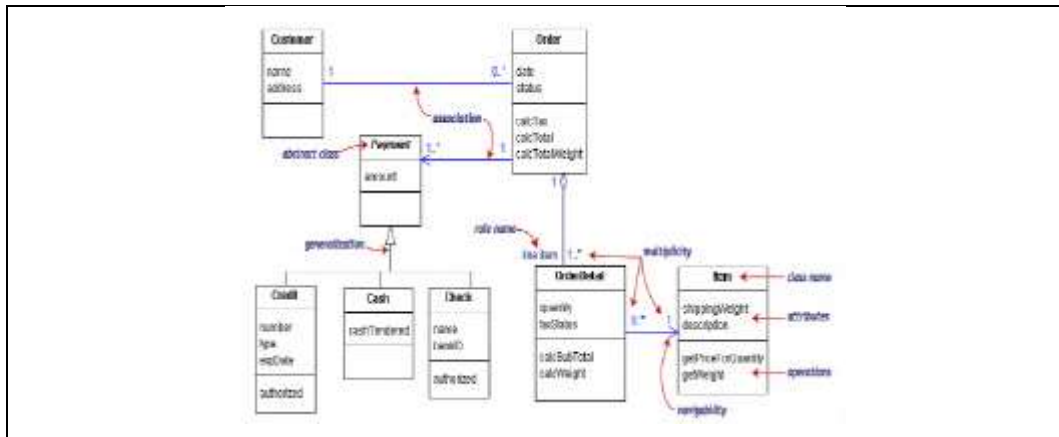
Pada relasi terdapat suatu penanda yang disebut *multiplicity*. *Multiplicity* ini akan mengindikasikan berapa banyak obyek dari suatu kelas terelasi ke obyek lain. Notasi UML untuk *multiplicity* ini adalah sebagai berikut (Wahono dan Dharwarwiyanti, 2003):

Tabel II.8 Tipe *Multiplicity*

No.	<i>Multiplicity</i>	<i>Option</i>	<i>Cardinality</i>
1.	0..0	0	Nol
2.	0..1		Antara nol sampai satu
3.	1..1	1	Satu
4.	0..*	*	Antara nol sampai banyak
5.	1..*		Antara satu sampai banyak
6.	5..5	5	Tepat lima
7.	m..n		Sedikitnya m tetapi tidak boleh lebih dari n

(Sumber: Wahono dan Dharwarwiyanti, 2003)

Contoh gambaran *Class Diagram* beserta notasi dapat dilihat pada Gambar II.9:





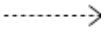
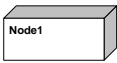
Gambar II.9 Contoh *Class Diagram* Studi Kasus Sistem Informasi Perpustakaan
(Sumber: Wahono dan Dharwarwiyanti, 2003)

2.11.5 Deployment Diagram

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* juga dapat digunakan untuk memodelkan hal-hal berikut (lihat Tabel II.8):

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

Tabel II.8 Simbol-Simbol *Deployment Diagram*

No	Nama	Simbol	Keterangan
1.	<i>Package</i>		Merupakan sebuah kumpulan dari satu atau lebih komponen.
2.	<i>Link</i>		Relasi antar objek.
3.	<i>Dependency</i>		Hubungan elemen mandiri akan mempengaruhi elemen yang bergantung padanya.
4.	<i>Node</i>		Perangkat keras dan perangkat lunak.

(Sumber: Rosa dan Shalahuddin, 2013)

2.12 Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga user dan analis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Kamus data sangat membantu analis sistem dalam mendefinisikan data yang mengalir di dalam sistem, sehingga pendefinisian data itu dapat dilakukan dengan lengkap dan terstruktur. Pengertian kamus data menurut para ahli sebagai berikut:

1. Pembentukan kamus data dilaksanakan dalam tahap analisis dan perancangan suatu sistem (Yakub, 2012).
2. Kumpulan elemen-elemen atau simbolsymbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap field atau file di dalam sistem (Kristanto, 2008).
3. Rosa dan M. Shalahuddin (2014), kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).

Kamus data biasanya berisi:

- a. Nama-nama dari data.
- b. Digunakan pada merupakan proses-proses yang terkait data.
- c. Deskripsi merupakan deskripsi data.
- d. Informasi tambahan seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut (lihat Tabel II.9):

Tabel II.9 Simbol-Symbol Kamus Data

No.	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik...atau...
4.	{ } ⁿ	N kali diulang/bernilai banyak
5.	()	Data optional
6.	*...*	Batas komentar

(Sumber: Rosa dan Shalahuddin, 2014)

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

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

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

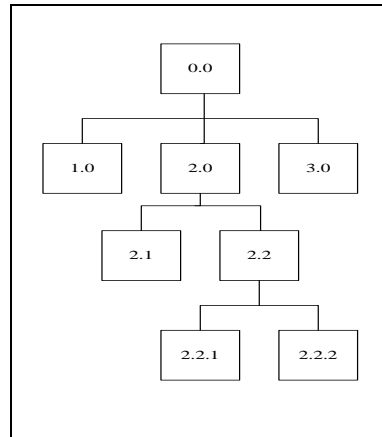
1. Untuk menyediakan suatu struktur guna memahami fungsi-fungsi dari sistem.
2. Untuk lebih menekankan fungsi-fungsi yang harus diselesaikan oleh program, bukannya menunjukkan statemen-statemen program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan yang jelas dari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

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

1. *Visual Table Of Contents* (VTOC)

Visual table of contents menggambarkan hubungan fungsi-fungsi di system secara berjenjang, *VTOC* menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur. Pada diagram ini nama dan nomor dari program HIPO, struktur paket diagram dan hubungan fungsi

diidentifikasi dalam bentuk hirarki. Keterangan masing-masing fungsi diberikan pada bagian penjelasan yang diikutsertakan dalam diagram ini. Berikut adalah *table of contents*.



Gambar II.7 *Visual*

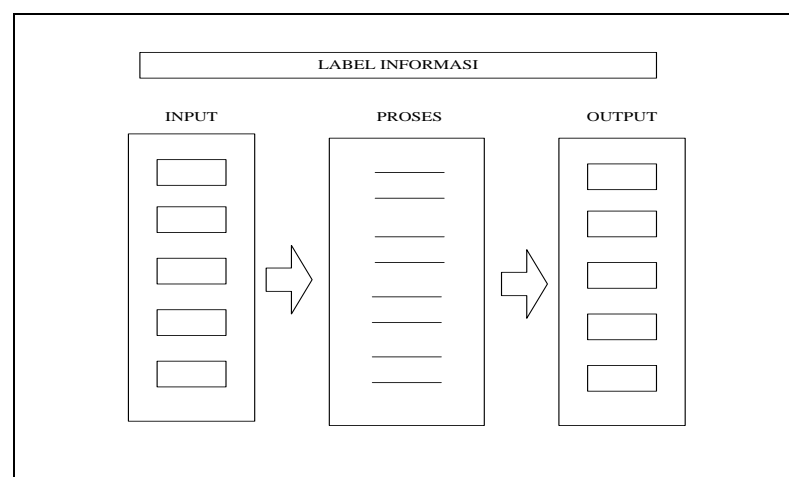
Gambar II.10 *Visual Table Of Contents*

Sumber: Jogiyanto (2005)

2. *Overview Diagram*

Overview Diagram menunjukkan secara garis besar hubungan dari *input*, proses dan *output*. Bagian *input* menunjukkan item-item data yang akan digunakan oleh bagian proses. Bagian proses berisi sejumlah langkah-langkah yang menggambarkan kerja dari fungsi. Bagian *output* berisi dengan item-item data yang dihasilkan atau dimodifikasi oleh langkah-langkah proses.

Berikut adalah Gambar II.8 *Overview diagram*.



Gambar II.11 *Overview Diagram*

(Sumber: Jogiyanto, 2005)

3. *Detail Diagram*

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

2.14 **PHP (*Hypertext Pre-Processor*)**

PHP adalah pemrograman berbasis web yang sudah sangat dikenal, bahasa pemrograman PHP termaksud bahasa pemrograman berbasis web yang bersifat *cross platform* atau dapat dijalankan dalam berbagai macam sistem operasi (Wahana, 2013). Menurut Winarno dan Ali Zaki (2014), PHP adalah sebuah bahasa pemrograman *web* berbasis *server* (*server-side*) yang mampu membuat kode PHP dari kode *web* dengan ekstensi *.php*, sehingga menghasilkan tampilan *website* yang dinamis di sisi *client* (*browser*). Sebuah halaman PHP adalah sebuah halaman HTML yang memiliki *server-side scripts* yang ditempatkan dalam *server* dan diproses oleh *web server* sebelum dikirim ke *browser* pemakai (Welling dan Thompson, 2001).

Menurut Welling dan Thomson (2001), beberapa keunggulan PHP adalah:

1. *High Performance*

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

2. *Database Integration*

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

3. *Built-in-Libraries*

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

4. Harga yang murah

PHP adalah perangkat lunak gratis.

5. Mudah dalam pembelajaran dan penggunaan

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

6. *Portability*

PHP dapat digunakan di banyak sistem operasi yang berbeda.

7. Ketersediaan *Source Code*

Kode PHP dapat langsung diakses dan dimodifikasi secara bebas.

2.15 Penyajian Data

Purwanto (2008) mengartikan penyajian data adalah kegiatan menyusun data mentah yang berserakan menjadi lebih teratur sehingga mudah dibaca, dipahami dan dianalisis. Penyajian data itu dapat dilakukan dengan dua cara yaitu membuat tabel atau daftar dan grafik atau diagram. Berdasarkan cara penyajiannya, tabel dapat dibagi menjadi beberapa jenis yaitu sebagai berikut:

1. Penyajian Data Tabel

Tabel adalah penyusunan data untuk membaca dan menganalisis data.

- Tabel baris dan kolom

Tabel ini memuat keterangan mengenai baris dan kolom.

- Tabel distribusi frekuensi

Tabel distribusi frekuensi tunggal adalah tabel yang digunakan untuk menyusun distribusi data dalam frekuensi dengan distribusi yang bersifat tunggal.

2. Penyajian Data Grafik

Purwanto (2008), penyajian data dalam bentuk grafik adalah menggambarkan data secara visual dalam sebuah gambar. Data yang disajikan merupakan data nominal, maka penyajian data menggunakan grafik berupa batang, lambang, garis atau lingkaran. Sedang bila data bersifat kontinu maka penyajian data biasanya menggunakan histogram, polygon dan kurva.

a. Grafik batang

Grafik batang digunakan untuk sejumlah nilai yang akan di bandingkan relative sedikit. Grafik ini dibuat dengan menggunakan batang sebagai gambaran kelompok data secara *vertical* dan *horizontal*.

b. Grafik garis

Grafik garis sering disebut juga peta garis (*line chart*) atau kurva (*curve*), merupakan bentuk penyajian yang paling banyak dipakai dalam berbagai laporan perusahaan maupun penelitian ilmiah.

c. Grafik lingkaran

Grafik lingkaran ini menarik, namun memiliki sisi kelemahan dalam hal tujuan untuk perbandingan antara sektor-sektor yang terdapat dalam lingkarannya.

2.16 Database

Database adalah Struktur penyimpanan data. Untuk menambah, mengakses dan memproses data yang disimpan dalam sebuah *database* komputer, diperlukan sistem manajemen *database* seperti *MYSQL Server*. Pengertian *database* menurut para ahli:

1. *Database* adalah sekumpulan tabel-tabel yang berisi data dan merupakan kumpulan dari *field* atau kolom (Anhar, 2010).
2. *Database* adalah sebagai kumpulan data dari penempatan tenaga kerja yang saling terkait dan mempengaruhi sesuai dengan tingkat kepentingannya sehingga data tersebut terintegrasi dan *independence* (Martono, 2009).

2.16.1 MySQL

My Structured Query Language (MySQL) adalah *Relational Database Management System* (RDBMS) yang sangat cepat dan kuat. Sebuah basis data memungkinkan *user* untuk melakukan penyimpanan yang efisien, pencarian dan pengurutan data. MySQL merupakan *database server* yang yang dapat digunakan banyak *user* dan banyak pekerjaan (Welling dan Thomson, 2003).

Menurut Anhar (2010) MySQL adalah salah satu databases management system (DBMS) dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL, dan lainnya. MySQL berfungsi untuk mengolah database menggunakan bahasa SQL. MySQL bersifat open source sehingga kita bisa menggunakannya secara gratis.

Beberapa keunggulan dari MySQL yaitu:

1. Cepat, handal dan Mudah dalam penggunaannya MySQL lebih cepat tiga sampai empat kali dari pada database server komersial.
2. Didukung oleh berbagai bahasa. Database *server* MySQL dapat memberikan pesan eror dalam berbagai bahasa.
3. Mampu membuat tabel berukuran sangat besar Ukuran maksimal dari setiap tabel yang dapat dibuat dengan MySQL adalah 4 GB sampai dengan ukuran *file* yang dapat ditangani oleh sistem operasi yang dipakai.
4. Lebih Murah MySQL bersifat *open source* dan didistribusikan dengan gratis tanpa biaya untuk UNIX platform, OS/2 dan Windows platform

2.16.2 Tipe Data MySQL

Tipe data MySQL masing-masing memiliki kegunaan dan keterangan untuk digunakan dalam merancang tabel. Beberapa tipe data yang terdapat pada MySQL (lihat Tabel II.10):

Tabel II.10 Tipe Data MySQL

Tipe Data	Ukuran	Keterangan
CHAR	M	Menampung maksimal M karakter (kombinasi huruf, angka, dan simbol-simbol). Jumlah memori yang dibutuhkan selalu M byte. M terbesar adalah 255.
VARCHAR	M	Karakter yang disimpan maksimal M karakter. Jumlah memori yang dibutuhkan tergantung jumlah karakter. M bisa mencapai 65535.
DATE	8 byte	Menyatakan tanggal.
TIME	8 byte	Menyatakan waktu (jam:menit:detik).
TINYINT	1 byte	Bilangan antara -128 sampai dengan +127.
SMALLINT	2 byte	Bilangan antara -32768 sampai dengan +32768.
INT	4 byte	Bilangan antara -2147683648 sampai dengan +2147683647.
FLOAT	4 byte	Bilangan pecahan.
DOUBLE	8 byte	Bilangan pecahan dengan presisi tinggi.

BOOL	1 byte	Untuk menampung nilai <i>true</i> (benar) dan <i>false</i> (salah). Identik dengan TINYINT.
ENUM	-	Menyatakan suatu tipe yang nilainya tertentu (disebutkan dalam pendefinisian).
TEXT	-	Menyimpan teks yang ukurannya sangat panjang.
BLOB	-	Untuk menyimpan data biner (misalnya gambar atau suara).

(Sumber: Kadir, 2008)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian merupakan ilmu mengenai jenjang-jenjang yang harus dilalui dalam suatu proses penelitian. Metodologi penelitian juga dikenal sebagai metode ilmiah dalam mencari, mengembangkan dan menguji suatu kebenaran pengetahuan. Metodologi penelitian juga membuat penelitian lebih terarah. Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, dan memilih langkah-langkah sistematis (Hasan, 2002).

Metodologi berasal dari bahasa Yunani "*metodos*", kata ini terdiri dari dua suku kata yaitu "*metha*" yang berarti melalui atau melewati dan "*hodos*" yang berarti jalan atau cara. Metodologi digunakan untuk memperoleh kebenaran, tergantung dari realitas yang sedang dikaji, dalam metodologi penelitian sudah ditentukan garis besar urutan-urutan tahapan penelitian yang akan dikerjakan oleh peneliti. Dengan demikian peneliti tidak akan menyimpang dari prosedur ilmiah yang telah ditetapkan.

3.2 Identifikasi Masalah

Identifikasi masalah adalah salah satu proses penelitian yang boleh dikatakan paling penting di antara proses lain. Masalah penelitian akan menentukan kualitas dari penelitian, bahkan juga menentukan apakah sebuah kegiatan bisa disebut penelitian atau tidak. Masalah penelitian secara umum bisa ditemukan lewat studi literatur atau lewat pengamatan lapangan, dilakukan identifikasi terhadap suatu masalah dengan cara, antara lain:

- a. Mengurai berbagai pertanyaan tentang tema tertentu, yaitu kebutuhan informasi. Hal ini dimaksudkan untuk memperoleh gambaran secara utuh tentang suatu masalah.
- b. Menampilkan indikasi terjadinya masalah di seksi *Assembling* yang bisa dikaitkan dengan kebutuhan informasi. Hal ini dimaksudkan untuk membantu menemukan gejala, variabel dan indikator yang akan dibahas dari suatu masalah.

- c. Menginventaris berbagai masalah yang terjadi di seksi *Assembling*, khususnya yang terjadi pada pemeliharaan mesin. Hal ini dimaksudkan untuk membantu menemukan masalah-masalah penting sebagai fokus masalah dalam penelitian yang akan dilakukan.
- d. Merumuskan masalah dalam suatu kalimat yang dapat mewakili dan mendeskripsikan penelitian yang akan diteliti. Perumusan masalah dilakukan dengan menggabungkan antara fokus masalah penelitian dengan teori-teori sebelumnya yang berkaitan. Hasil akhir dari perumusan masalah tersebut diwujudkan dalam judul penelitian.

3.3 Pengumpulan Data

Pengumpulan data merupakan sebuah sarana yang menampung dan mengumpulkan data-data dari berbagai sumber dan bentuk yang akan digunakan dalam keperluan menganalisis sistem untuk perancangan sistem usulan. Sumber data atau informasi penelitian ini berdasarkan kepada jenis data yang diperlukan.

Data yang diperoleh dari responden secara langsung yang dikumpulkan melalui survei lapangan dengan menggunakan teknik pengumpulan data. Teknik pengumpulan data dalam rangka pengumpulan informasi mengenai objek penelitian ini, yaitu:

1. Studi Literatur

Tahap ini merupakan tahap awal pengumpulan data dari sumber-sumber seperti buku yang berhubungan analisis dan desain sistem informasi, buku pemrograman menggunakan PHP, dan juga mencari data tambahan yang diperlukan melalui internet.

2. Observasi

Tahap ini merupakan tahap melakukan pengamatan langsung terhadap pengguna sistem dan pengamatan langsung terhadap sistem yang sedang berjalan. Tahap ini melakukan observasi pada departemen produksi *front fork 2 W* di Seksi *Assembling* yang belum memiliki sistem permintaan pemeliharaan mesin dan laporan monitoring yang terintegrasi dengan baik.

3. Wawancara

Pada tahap ini dilakukan wawancara dengan para pengguna sistem tersebut untuk memperoleh keterangan mengenai alur proses *input* data permintaan pemeliharaan mesin mengenai kebutuhan sistem yang akan dikembangkan selanjutnya. Keterangan yang diperoleh bertujuan agar sistem selanjutnya dapat

lebih mudah, cepat, dan akurat dalam melakukan penyampaian data dan penanganan dalam pemeliharaan mesin produksi *front fork 2 W*. Wawancara dilakukan, yaitu untuk memperoleh keterangan tujuan penelitian dengan cara tanya jawab sambil bertatap muka antara pewawancara dengan informan atau orang yang benar-benar mengetahui dan memahami obyek penelitian, dengan atau tanpa menggunakan pedoman wawancara. Wawancara dilakukan terhadap karyawan dan operator yang melaksanakan proses produksi *front fork 2 W* untuk mengetahui lebih mendalam kebutuhan informasi dalam pemeliharaan mesin di seksi *Assembling*.

3.4 Metode Pengembangan Sistem

Metodologi pengembangan sistem merupakan kerangka formal dalam mengimplementasikan konsep SDLC untuk mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik) (Rosa dan Shalahuddin, 2014). Pada penelitian ini untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat pengembangan sistem. Dalam pengembangan sistem ini digunakan metode prototipe evolusioner.

Metode prototipe merupakan cara yang potensial disediakan oleh pengembang kepada calon pengguna dengan tujuan memperoleh umpan balik dari pengguna untuk ditingkatkan sampai sistem dianggap sempurna. prototipe evolusioner adalah prototipe yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi (McLeod, 2011).

Tahapan-tahapan dalam prototipe evolusioner adalah sebagai berikut:

1. Identifikasi kebutuhan pengguna
Pengembang melakukan diskusi dengan Seksi *Assembling* produksi *front fork 2 W* dan *Maintenance* bagian pemeliharaan mesin tentang kebutuhan sistem yang digunakan.
2. Mengembangkan prototipe
Pengembang membuat prototipe dari sistem yang telah dijelaskan oleh pihak terkait.
3. Menyesuaikan dan evaluasi prototipe dengan keinginan *user*

Pengembang menanyakan Seksi *Assembling* produksi *front fork 2 W* dan *Maintenance* serta tentang prototipe yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.

4. Menggunakan prototipe

Sistem mulai dikembangkan dengan prototipe yang sudah dibuat.

3.5 Kerangka Penelitian

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

1. Studi Pendahuluan

Studi pendahuluan merupakan langkah awal yang dilakukan untuk memulai penelitian. Dengan melakukan studi pustaka, observasi, dan wawancara. Tujuan melakukan studi pendahuluan adalah untuk mendapatkan pengetahuan umum mengenai sistem apa yang sedang diteliti.

2. Menentukan Rumusan Masalah

Tahap ini bertujuan untuk mencari jalan keluar dari masalah yang ada pada sistem. Rumusan masalah dari permasalahan, yaitu pemeliharaan mesin produksi produksi *front fork 2 W* di seksi *Assembling* yang digunakan menggunakan memo berupa kertas.

3. Menentukan Tujuan Penelitian

Tujuan penelitian dimaksudkan agar dapat mencapai tujuan yang hendak dicapai dari penelitian dengan merancang sistem informasi pemeliharaan mesin produksi *front fork 2 W* di seksi *Assembling*.

4. Menentukan Batasan Masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya. Batasan Masalah yaitu melakukan penelitian dan analisis hanya mengenai proses pemeliharaan mesin di seksi *Assembling* pada PT Showa Indonesia Manufacturing.

5. Mengidentifikasi Kebutuhan Pengguna

Pada tahap ini bertujuan untuk mengidentifikasi pengguna dengan pengumpulan dan pengolahan data.

6. Mengembangkan Metode Prototipe Evolusioner
 - a. Mengidentifikasi kebutuhan pelanggan

Menganalisis kebutuhan *user* terhadap program yang akan dibuat. Dengan melakukan metode pengumpulan data dan pengolahan data.
 - b. Pengembangan prototipe

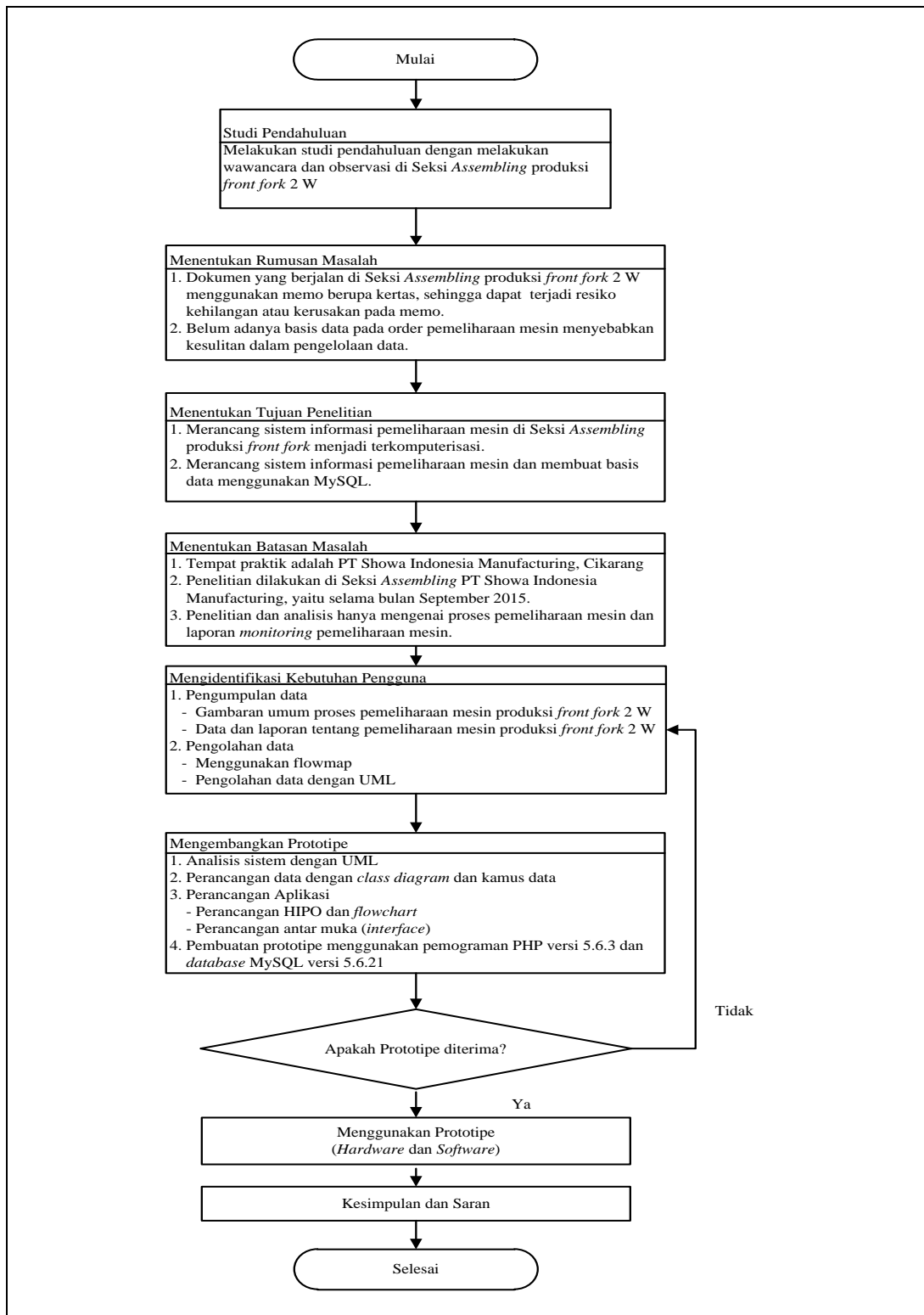
Mengembangkan sebuah prototipe sesuai dengan analisis yang dilakukan. Pada tahap ini akan dirancang aplikasi tentang sistem penggajian. Perancangan sistem nantinya akan memuat tentang:
 - i. Pemodelan sistem menggunakan *Unified Modeling Language* (UML).
 - ii. Pemodelan *database* dengan *class diagram* dan kamus data.
 - iii. Perancangan aplikasi
 - Perancangan HIPO dan *flowchart*
 - Perancangan antar muka (*interface*)
 - iv. Pembuatan prototipe menggunakan pemograman PHP versi 5.6.3 dan database MySQL versi 5.6.21.
 - v. PHP Dan MySQL 5.6.21
 - c. Menyesuaikan dan evaluasi prototipe dengan keinginan *user*

Melakukan evaluasi prototipe dengan menanyakan seksi *Assembling* untuk menyesuaikan dengan kebutuhan sistem.
 - d. Menggunakan prototipe

Memutuskan untuk menggunakan prototipe yang telah divalidasi menggunakan metode prototipe evolusioner. Apabila prototipe tidak diterima maka akan kembali ke tahap identifikasi.
7. Kesimpulan dan Saran

Kesimpulan digunakan untuk membandingkan hasil penelitian atau pengembangan sistem dengan sistem sebelumnya. Sedangkan saran berisi masukan untuk pengembangan sistem selanjutnya.

Berikut adalah *flowchart* kerangka penelitian untuk menyelesaikan masalah dalam Tugas Akhir ini:



Gambar III.1 Kerangka Penelitian
Sumber: Pengumpulan dan Pengolahan Data (2016)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Tentang PT Showa Indonesia Manufacturing

PT Showa Indonesia Manufacturing merupakan perusahaan kerja sama antara Showa *Corporation* Jepang (55%) dengan PT Astra Honda Motor (45%). PT Showa Indonesia Manufacturing didirikan di Jakarta pada tanggal 8 Maret 1978 tepatnya di Jalan Raya Gaya Motor Sunter II Jakarta Utara dengan Akte Notaris Kartini Mulyadi SH. Pada saat itu PT Showa Indonesia Manufacturing hanya berukuran dengan luas tanah 18,142 m².

PT Showa Indonesia Manufacturing adalah perusahaan yang bergerak di bidang otomotif untuk kendaraan roda 2 maupun roda 4, sebagian besar hasil produksinya adalah *shock absorber* atau sering dikenal dengan nama *shock braker* disebut juga peredam kejut. PT Showa Indonesia Manufacturing hanya memproduksi *shock absorber* untuk kendaraan roda 2 untuk motor jenis Honda tepatnya pada tahun 1979. Pada tahun 1980, perusahaan tersebut mencoba untuk memproduksi *shock absorber* roda 4 untuk mobil merek Honda dan Suzuki.

Pada tanggal 23 Mei 1996 PT Showa Indonesia Manufacturing melakukan perpindahan produksinya secara total ke Kawasan Industri Cikarang-Jawa Barat, Jl Jababeka VI kav. 28-32 Cikarang Bekasi 17530, dengan luas tanah 104.640 m². PT Showa Indonesia Manufacturing memiliki 6 gedung, yaitu gedung A, B, C, D, E dan F.

PT Showa Indonesia Manufacturing mulai melakukan ekspor ke Jepang pada tahun 1988 dan tahun 1989, mengekspor komponen *shock absorber* ke Malaysia dan Thailand. Mulai tahun 1990 baru memproduksi Mitsubishi, dengan semakin bertambahnya produksi maka menuntut para manajemen untuk mengeksperikan pabrik. Produk-produk dengan kualitas yang menjamin menjadikan PT Showa Indonesia Manufacturing kini mempunyai pelanggan tetap bahkan semakin berkembang dan bertambah baik.

4.2 Profil Perusahaan

Nama Perusahaan	:	PT Showa Indonesia Manufacturing
Status Perusahaan	:	Perseroan Terbatas
Status Investasi	:	PMA (Penanaman Modal Asing)
Tanggal Pendirian	:	8 Maret 1978
Produksi Komersial	:	1 Juni 1979
Aktivitas	:	Manufaktur Komponen Otomotif Terutama <i>shock absorber</i> untuk 2 Roda & 4 Roda.
Status Perusahaan	:	Perseroan Terbatas
Status Investasi	:	PMA (Penanaman Modal Asing)
Kepemilikan	:	Showa Corporation Japan (55 %) Pt . Astra Honda Motor (45 %)
Alamat	:	Gedung Graha Kirana (<i>representative office</i>) Jl. Yos Sudarso Kav. 88 Jakarta Utara Cikarang Plant (pabrik dan kantor) Jl.Jababeka VI kav. 28-36 Cikarang <i>Industrial Real Estate</i> – Bekasi telp : (021)8934855-56 fax : (021) 8934875
Luas Pabrik	:	50,568.00 m2 (<i>Land 1</i>) 31,040.00 m2 (<i>Land 2</i>) 11,520.00 m2 (<i>Land 3</i>) 10,000.00 m2 (<i>Land 4</i>) 40,800 m2 (<i>Factory</i>)
Jumlah Pegawai	:	Total : 2.469 orang
Jam Kerja	:	Pabrik <i>Shift</i> : 07.30 - 16.15 WIB
Website	:	http://www.showa.co.id

4.3 Visi, Misi dan Tujuan Pendirian Perusahaan

PT. Showa Indonesia Manufacturing merupakan perusahaan yang bergerak di bidang otomotif yang mempunyai visi, misi, dan tujuan perusahaan sebagai berikut:

Visi

Menyediakan komponen otomotif kelas dunia, terutama *shock absorber* dan *steering stem* untuk pasar global OEM (*Original Requirement Manufacture*) dan REM (*Replacement Equipment Manufacture*) dengan kualitas terbaik dan aman untuk pihak- pihak yang berkepentingan.

Misi

1. Menjadikan pemimpin Industri *Shock Absorber* OEM (*Original Requirement Manufacture*) di Kawasan ASEAN dan sebagai pemasok global *Shock Absorber* dan *Steering Steam*.
2. Menciptakan kepuasan pelanggan dengan kualitas tertinggi, harga bersaing, sumber daya manusia yang kompeten, komitmen sabagai perusahaan yang ramah lingkungan, aman, nyaman efesien dan produktif.

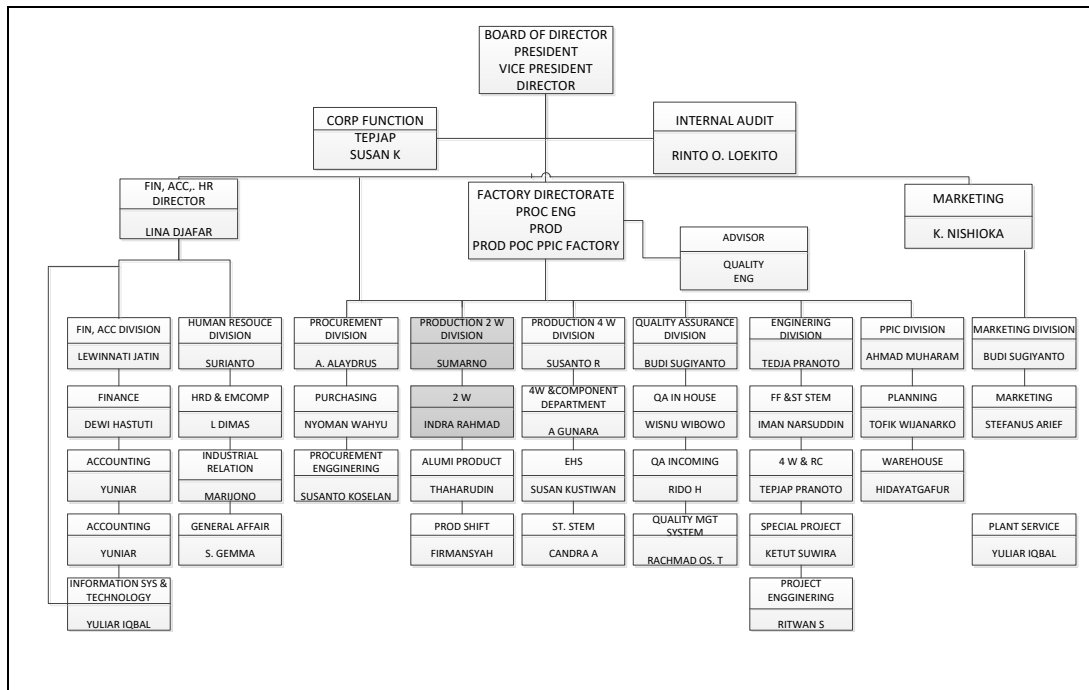
Tujuan

1. Mengurangi keluarnya devisa lebih banyak dari pembelian impor penuh (*buildup*) menjadi CKD (*Complete Knot Down*).
2. Impor (*Assembling*).
3. Menciptakan lapangan kerja baru.
4. Memperkuat industri komponen kendaraan bermotor.
5. Melakukan inovatif dan pengembangan produk.

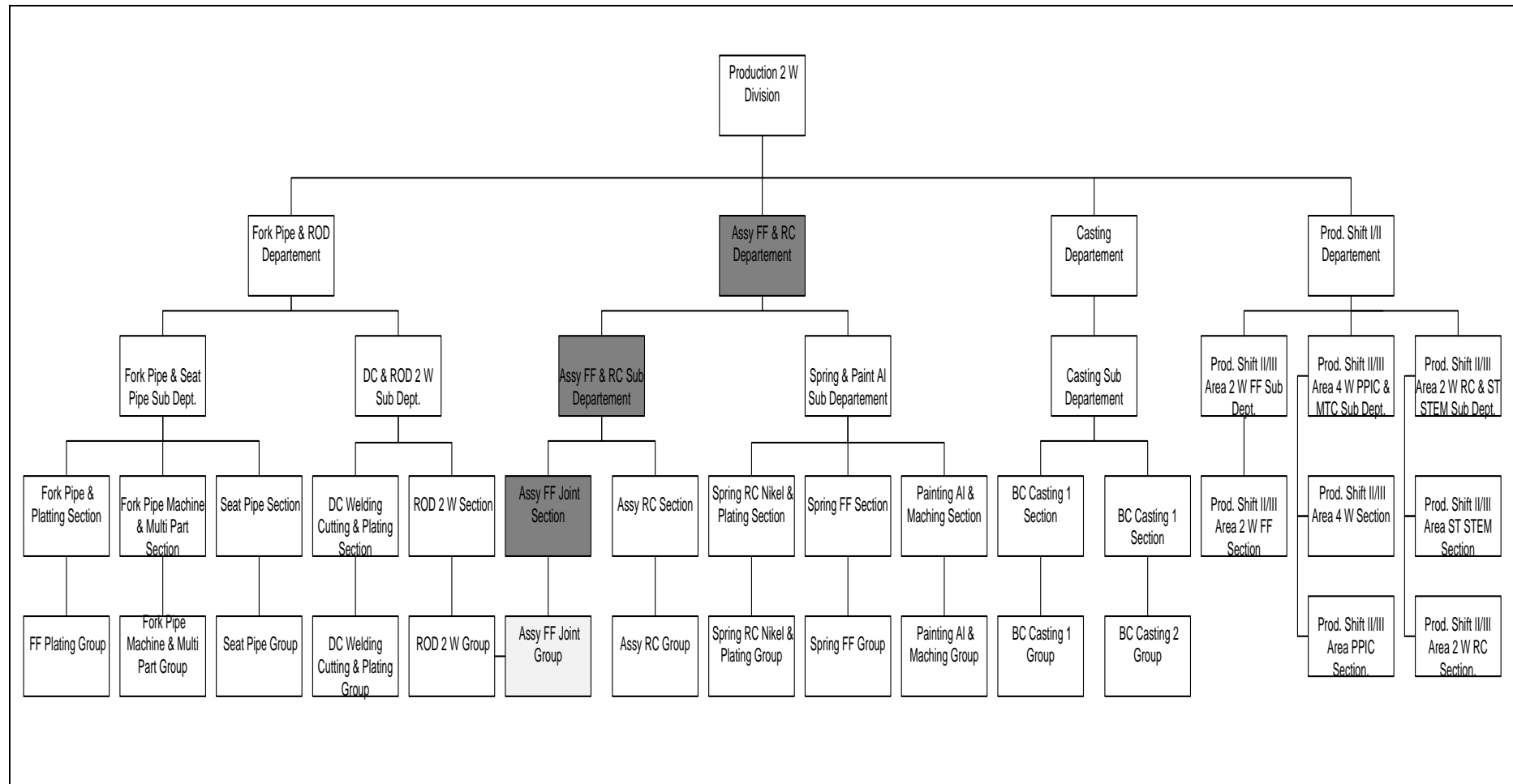
4.4 Struktur Organisasi Perusahaan

Setiap perusahaan memerlukan suatu struktur organisasi yang baik. Struktur organisasi dapat diartikan sebagai susunan dan hubungan antar bagian serta posisi dalam suatu perusahaan. Suatu struktur organisasi koordinasi dan komunikasi dalam rangka mencapai satu tujuan yang sama, yaitu menjalankan fungsi perusahaan untuk mencapai tujuan perusahaan.

Berikut adalah struktur organisasi PT Showa Indonesia Manufacturing, struktur organisasi *production 2 wheeler* dan struktur organisasi di seksi *assembling front fork* PT Showa Indonesia Manufacturing (lihat Gambar IV.1, Gambar IV.2 dan Gambar IV.3):

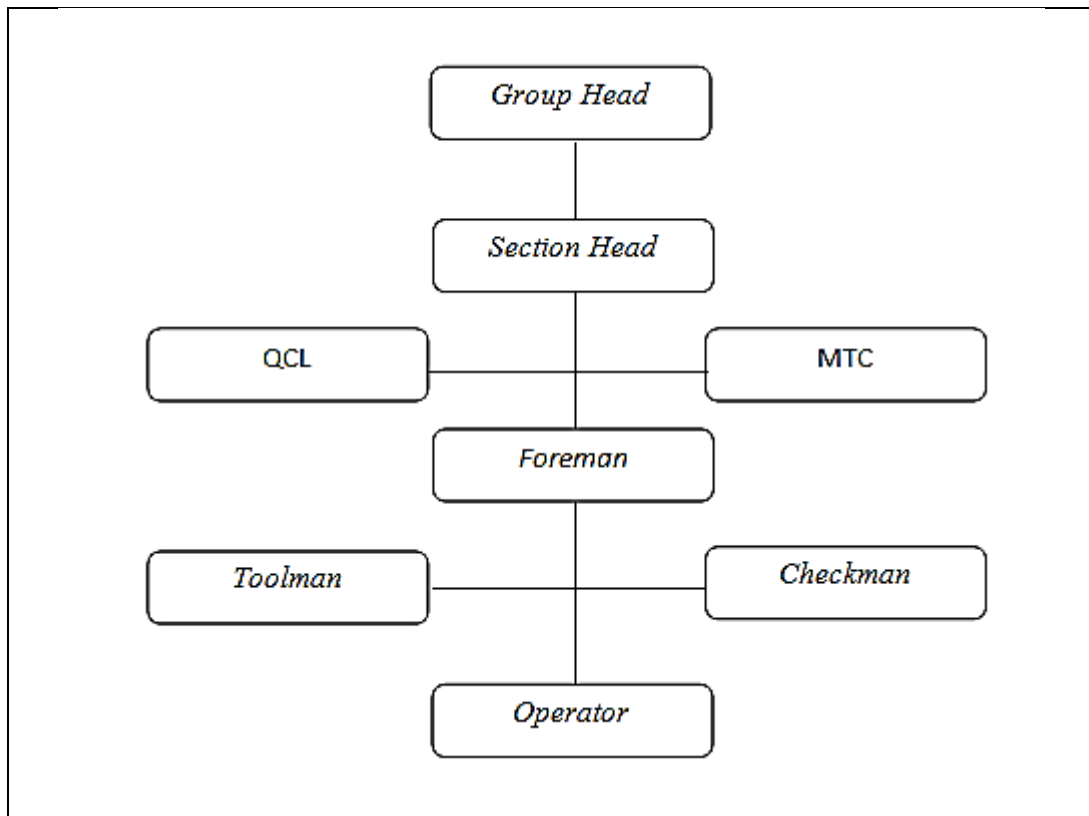


Gambar IV.1 Struktur Organisasi PT Showa Indonesia Manufacturing
(Sumber: PT Showa Indonesia Manufacturing, 2015)



Gambar IV.2 Struktur Organisasi *Production 2 Wheeler* PT Showa Indonesia Manufacturing

(Sumber: PT Showa Indonesia Manufacturing, 2015)



Gambar IV.3 Struktur Organisasi di Seksi *Assembling Front Fork*
(Sumber: PT Showa Indonesia Manufacturing, 2015)

4.3 Tujuan dan Tanggung Jawab

Tugas dan wewenang tiap jabatan di Seksi *Assembling* (PT Showa Indonesia Manufacturing, 2015).

1. *Group Head*:

Tugas pokok:

- Membagi tugas-tugas harian antar *operator* secara seimbang dan tepat guna.
- Membantu dan mengawasi *operator* untuk dapat melakukan pekerjaannya.
- Mengatur penggunaan alat-alat/mesin secara tepat dan efisien serta menjaga sarana-sarana proses untuk selalu siap pakai.
- Mengontrol pemakaian *tools/indirect material* sesuai standar.
- Mengisi absensi *operator* dan mengajukan daftar *overtime* bila dibutuhkan.

- Merekap masalah bila terjadi *trouble* mesin dan membuat informasi produksi untuk dilaporkan ke seksi-seksi yang terkait.
- Kontrol *man power*, mesin, peralatan dan lingkungan yang berkenaan dengan *quality*, 5K, dan *safety*.

Wewenang:

- Memberikan contoh yang baik dan mengarahkan operator bekerja sesuai dengan peraturan perusahaan.
- Melaporkan segera setiap penyimpangan yang terjadi.
- Memeriksa, memberikan persetujuan dan menandatangani *checksheet quality* dan perbaikan pada *order* pekerjaan.

2. *Section Head*:

Tugas pokok:

- Mengkoordinir pembagian/pengaturan tugas-tugas bawahnya secara terarah untuk menyelesaikan tugas tepat waktu.
- Upaya aktif dalam menyelesaikan problem kualitas harian 5 langkah penyelesaian masalah dan kualitas *improvement meeting*.
- Menyusun *budget* pemakaian bahan/*material secure* cermat untuk efisiensi penggunaan energi dan bahan.
- Membuat laporan bulanan seksi sebagai tanggung jawab hasil produktivitas dan kualitas di seksi, maupun pengendalian sistem manajemen lingkungan.
- Upaya aktif perbaikan mutu sebagai promotor QCC diseksinya.

Wewenang:

- Memeriksa dan memaraf informasi produk yang keluar dari seksi.
- Memberikan surat peringatan kepada bawahannya yang melanggar etika kerja.
- Melaksanakan penilaian karyawan bawahan dengan benar dan terarah guna kesempatan pengembangan Sumber Daya Manusia.

3. QCL:

Tugas pokok:

- Mengaudit kondisi kualitas proses di *line* seksi agar tidak terjadi penyimpangan.
- Memberikan keputusan akan kualitas yang sedang dikerjakan seksi.
- Menjaga kondisi alat-alat ukur yang akan digunakan di seksi agar layak pakai.

Wewenang:

- Memeriksa dan memparaf label produksi setelah melakukan audit kualitas sesuai standar.
- Audit disemua *line* proses tiap mesin sesuai dengan seksi yang ditugaskan.

4. MTC:

Tugas pokok:

- Mengusulkan kepala seksi untuk pembuatan order pekerjaan.

Wewenang:

- Mengelola secara administrasi untuk *part-part reject*.
- Mengadakan *part-part* pengganti yang diperlukan untuk di *repair*.
- Mengendalikan *part-part* yang ada di lini produksi, melengkapinya bila terjadi kekurangan.
- Membantu *repairman* menyelesaikan unit motor *repair* menjadi *finish good*.
- Melaksanakan tugas-tugas dari *foreman* yang dilimpahkan kepadanya.

5. *Foreman*:

Tugas pokok:

- Menjaga ketertiban bawahan pada waktu kerja.
- Membuat laporan harian produksi.
- Melaksanakan program lingkungan yang menjadi sasaran perusahaan.
- Mengatur jalannya lini produksi sesuai dengan *planning* dan target yang ditentukan dan standar kualitas yang ditentukan.

Wewenang:

- Memberikan pengarahan atau nasehat kepada bawahan yang melakukan pelanggaran.
- Memberikan sanksi kepada bawahan yang melakukan pelanggaran.
- Mengusulkan sistem rotasi.
- Memberi masukan/pertimbangan pada kepala seksi dalam membuat penilaian kerja.

6. *Toolman*

Tugas pokok:

- Pendataan *tools* dan *equipment*.
- Pendataan *setting tools* dan *diez*.
- Cek dan data kondisi *setting* parameter mesin.
- Melakukan *lot control* produksi.
- Menggantikan operator.
- Bertanggungjawab terhadap masalah di seksi.

Wewenang:

- Mengoperasikan mesin & memutuskan hasil pemeriksaan mesin.
- Mengusulkan penghentian produksi kepada *foreman* berkaitan dengan masalah *tools*.

7. *Checkman*

Tugas pokok:

- Melakukan *scan* turun produksi.

Wewenang:

- Melakukan kegiatan produksi sesuai *planning* harian.
- Mencatat hasil produksi per periode.
- Melaksanakan perintah atasan dalam rangka memperlancar jalannya produksi.

8. Operator

Tugas pokok:

- Hadir tepat pada waktu yang telah ditentukan.

- Melakukan persiapan pada mesin yang akan dioperasikan sebelum proses (\pm 5 menit).
- Melakukan proses sesuai dengan *planning* dan instruksi kerja.
- Pencapaian hasil sesuai dengan target.
- Melakukan perawatan sarana produksi dan mesin agar tidak terjadi *line* proses *stop*.
- Mengisi laporan hasil kerja.
- Melaporkan masalah-masalah yang terkait pada mesin di pos kerja kepada atasan terkait.
- Melakukan pembersihan pada mesin dan lingkungan 5 menit sebelum pulang.

Wewenang:

- Melaksanakan tugas sesuai dengan tata tertib perusahaan yang berlaku.

4.6 Sistem Kerja Karyawan

PT Showa Indonesia Manufacturing memiliki karyawan sebanyak 2.469 orang, dengan perbandingan 2.402 orang karyawan laki-laki dan 62 orang karyawan orang perempuan.

PT Showa Indonesia Manufacturing membagi waktu kerja menjadi 3 *shift*, diantaranya sebagai berikut:

1. *Non Shift*

Karyawan bekerja mulai 07:30 – 16:15 WIB dari hari Senin sampai Kamis, sedangkan hari Jumat karyawan bekerja mulai pukul 07:30 – 16:30 WIB.

2. *Shift*

Karyawan pada bagian *shift* di bagi menjadi 3 *shift*, yaitu dengan waktu kerja 8 jam dengan hari kerja dari Senin sampai Jumat, termasuk 1 jam untuk makan dan istirahat.

Dimana sebagian waktunya adalah sebagai berikut:

Shift pagi : Karyawan bekerja mulai pukul 07:30-16:15 WIB

Shift sore : Karyawan bekerja mulai pukul 16:30-24:00 WIB

Shift malam : Karyawan bekerja mulai pukul 24:00-07:30 WIB

4.7 Fasilitas-Fasilitas Perusahaan

PT Showa Indonesia Manufacturing menyediakan fasilitas-fasilitas untuk menunjang kegiatan karyawan, perusahaan mempunyai tempat ibadah, fasilitas olahraga dan seni yang sangat memadai dan fasilitas kesehatan. Perusahaan memiliki klinik umum dan gigi dengan dokter yang selalu siap memberikan konsultasi kesehatan. Pelatihan dan konseling merupakan fasilitas perusahaan bagi karyawan sehingga masalah-masalah yang membebani karyawan bisa dapat dengan cepat diselesaikan. Kesempatan untuk mengikuti *training* baik di dalam maupun di luar negeri, terbuka luas bagi setiap karyawan yang ingin meningkatkan kemampuan dan keahlian. PT Showa Indonesia Manufacturing menyediakan psikolog atau konselor dari luar perusahaan apabila ada karyawan yang ingin berkonsultasi secara pribadi untuk mengatasi masalah-masalah yang dihadapi di tempat kerja maupun di lingkungan keluarga.

PT Showa Indonesia Manufacturing menyediakan fasilitas-fasilitas bagi semua karyawan. Setiap karyawan mendapatkan fasilitas-fasilitas, di antaranya:

- Jaminan sosial tenaga kerja,
- Bantuan perawatan kesehatan,
- Penyediaan makanan,
- Tunjangan kecelakaan kerja,
- Sumbangan pernikahan dan duka,
- *Fogging*,
- Penyuluhan kesehatan,
- Posyandu PKK,
- Donasi,
- Bea siswa,
- Donor,
- *Rescue team*,
- *Village development*,
- Santunan kematian, serta
- Seragam kerja yang dilengkapi dengan Alat Pelindung Diri (APD).

4.8 Mesin Assembling Produksi Front Fork

Setiap perusahaan umumnya memiliki produk yang berbeda-beda untuk memenuhi kebutuhan konsumennya. Demikian pula dengan PT Showa Indonesia Manufacturing. Perusahaan ini bergerak di bidang industri komponen otomotif yang memfokuskan produksinya dalam pembuatan *shock absorber* untuk kendaraan roda 2 maupun roda 4.

Dalam proses untuk perakitan *shock absorber front fork*, PT Showa Indonesia Manufacturing menggunakan mesin proses berteknologi tinggi. Adapun mesin yang digunakan untuk memproses produksi *front fork* adalah sebagai berikut:

Tabel IV.1 Mesin Seksi Assembling

No.	Nama Mesin	Proses	Keterangan
1.	Mesin <i>Rolling</i>	<i>Rolling F/P</i>	Proses <i>rolling fork pipe</i> bagian bawah.
2.	Mesin <i>Cleaning</i>	<i>Cleaning Buttom Case</i>	Proses pembersihan <i>buttom case</i> bagian dalam. Pada proses ini <i>buttom case</i> disemprotkan air dan disikat.
3.	Mesin <i>Center</i>	<i>Center Fastener</i>	Proses untuk membuat lingkaran dalam ditengah.
4.	Mesin <i>Oil Seal Press</i>	<i>Oil Seal Press</i>	Proses pemasangan kawat yang berguna untuk menahan <i>seal</i> . <i>seal</i> berguna untuk menahan oli agar oli tidak keluar.
5.	Mesin <i>Dust Seal Press</i>	<i>Dust Seal Press</i>	Proses pemasangan <i>dust</i> yang berguna untuk melindungi <i>seal</i> dari kotoran-kotoran.
6.	Mesin <i>Leak Test</i>	<i>Leak Test</i>	Proses pengecekan kebocoran
7.	Mesin <i>Oil Filling and Joint Spring</i>	<i>Oil Filling and Joint Spring</i>	Proses pengisian oli dan dimasukan <i>spring</i> .
8.	Mesin <i>Upper Spring Seat</i>	<i>Upper Spring Seat</i>	Pemasangan <i>upper spring seat</i> .
9.	Mesin <i>Dumpping Manual</i>	<i>Dumpping Manual</i>	Proses pengecekan <i>front fork</i> apakah sudah berfungsi dengan benar atau tidak.
10.	Mesin <i>Final Inspection</i>	<i>Final Inspection</i>	Pengecekan terakhir

(Sumber: PT Showa Indonesia Manufacturing, 2015)

4.9 Analisis Dokumen

Analisis dokumen berikut ini akan menjelaskan dokumen apa saja yang digunakan dalam proses pemeliharaan mesin produksi *front fork 2 W* di Seksi *Assembling* di PT Showa Indonesia Manufacturing. Dokumen-dokumen tersebut adalah sebagai berikut:

1. Memo Order Pekerjaan

Deskripsi:

Memo yang dibutuhkan untuk menginformasikan permasalahan apa yang ada pada mesin produksi *front fork* di Seksi *Assembling*. Memo dibuat oleh Operator Lini Produksi dan diserahkan kepada *section head* kemudian meminta persetujuan oleh *Group Head* dan *Departement Head* untuk diberikan kepada *Maintenance* dan melakukan perbaikan atau pemeliharaan yang dibutuhkan. *Maintenance* akan melakukan pengecekan dan pekerjaan yang telah di order oleh Seksi *Assembling*. Terdapat tiga slip memo yang pada order pekerjaan (lihat Gambar IV.4, Gambar IV.5 dan Gambar IV.6).

PT. SHOWA INDONESIA MFG.
GENERAL AFFAIR DEPT.

DEPT/DIV		ORDER PEKERJAAN		NOMOR	
SEKSI				TANGGAL	
JENIS PEKERJAAN		URAIAN PEKERJAAN			
<input checked="" type="checkbox"/> 1	<input type="checkbox"/> 1 PEMBUATAN <input type="checkbox"/> 2 MODIFIKASI	<input type="checkbox"/> 3 SERVICE <input type="checkbox"/> 4 LAIN-LAIN			
PRIORITAS PEKERJAAN					
<input checked="" type="checkbox"/> 2	<input type="checkbox"/> 1 EMERGENCY <input checked="" type="checkbox"/> 2 SCHEDULE				
KONDISI YANG ADA		KONDISI YANG DIHARAPKAN			
MATERIAL YANG DIBUTUHKAN				TGL SELESAI	
NO	NAMA BARANG	JUMLAH	KETERANGAN		
DITERIMA MTC. BUILD	MENYETUJUI	DITERIMA ADMINISTRASI	MENGETAHUI	PEMOHON ORDER	PEMBAUT
Jam	DEPT/SECT. HEAD	Jam	DIV/DEPT. HEAD	SECTION HEAD	NAMA JELAS
Tgl	Tgl	Tgl	Tgl	Tgl	Tgl

Gambar IV.4 Lembar Pertama Memo Order Pekerjaan
(Sumber: PT Showa Indonesia Manufacturing, 2015)

PT. SHOWA INDONESIA MFG. GENERAL AFFAIR DEPT.					
DEPT/DIV			ORDER PEKERJAAN		NOMOR
SEKSI					TANGGAL
JENIS PEKERJAAN			URAIAN PEKERJAAN		
<input type="checkbox"/> 1 PEMBUATAN <input type="checkbox"/> 2 MODIFIKASI			<input type="checkbox"/> 3 SERVICE <input type="checkbox"/> 4 LAIN-LAIN		
PRIORITAS PEKERJAAN					
<input type="checkbox"/> 1 EMERGENCY <input type="checkbox"/> 2 SCHEDULE					
KONDISI YANG ADA			KONDISI YANG DIHARAPKAN		
MATERIAL YANG DIBUTUHKAN			TGL SELESAI		
NO	NAMA BARANG	JUMLAH	KETERANGAN		
DITERIMA MTC. BUILD	MENYETUJUI	DITERIMA ADMINISTRASI	PEMOHON ORDER		
			MENGETAHUI	MENYETUJUI	PEMBUAT
Jam	DEPT/SECT. HEAD	Jam	DIV/DEPT. HEAD	SECTION HEAD	NAMA JELAS
Tgl	Tgl	Tgl	Tgl	Tgl	Tgl

Gambar IV.5 Lembar Kedua Memo Order Pekerjaan
(Sumber: PT Showa Indonesia Manufacturing, 2015)

PT. SHOWA INDONESIA MFG. GENERAL AFFAIR DEPT.					
DEPT/DIV			ORDER PEKERJAAN		NOMOR
SEKSI					TANGGAL
JENIS PEKERJAAN			URAIAN PEKERJAAN		
<input type="checkbox"/> 1 PEMBUATAN <input type="checkbox"/> 2 MODIFIKASI			<input type="checkbox"/> 3 SERVICE <input type="checkbox"/> 4 LAIN-LAIN		
PRIORITAS PEKERJAAN					
<input type="checkbox"/> 1 EMERGENCY <input type="checkbox"/> 2 SCHEDULE					
KONDISI YANG ADA			KONDISI YANG DIHARAPKAN		
MATERIAL YANG DIBUTUHKAN			TGL SELESAI		
NO	NAMA BARANG	JUMLAH	KETERANGAN		
DITERIMA MTC. BUILD	MENYETUJUI	DITERIMA ADMINISTRASI	PEMOHON ORDER		
			MENGETAHUI	MENYETUJUI	PEMBUAT
Jam	DEPT/SECT. HEAD	Jam	DIV/DEPT. HEAD	SECTION HEAD	NAMA JELAS
Tgl	Tgl	Tgl	Tgl	Tgl	Tgl

Gambar IV.6 Lembar Ketiga Memo Order Pekerjaan
(Sumber: PT Showa Indonesia Manufacturing, 2015)

2. Laporan *Monitoring* Pemeliharaan Mesin Produksi *Front Fork 2 W*

Deskripsi:

Merupakan laporan harian hasil dari pemeliharaan mesin produksi *front fork 2 W* yang berisikan permasalahan mesin yang terjadi di Seksi *Assembling*, dapat dilihat pada Tabel IV.4.

Tabel IV.2 Laporan *Monitoring* Pemeliharaan Mesin Produksi *Front Fork*

LAPORAN MONITORING ORDER PEKERJAAN									
Shift	I								
Periode	September, 2015								
Dokumen Status	Diterima								
Tgl dan Order	Shift	Nomor OP	Pemeliharaan OP	Lokasi Masalah	No. Mesin	Nama Mesin	Sifat	Klasifikasi	Uraian Masalah
23-09-15	1	50	HARDIYANTNA	sear pipe c	483	NCLATHE	schedule	perbaikan	perbaikan base cylinder
23-09-15	1	54	PARYADI	sear pipe c	483	NCLATHE	schedule	perbaikan	perbaikan roller bawah
23-09-15	1	53	MULTAKIN	SS bracket line-e	1019	MACHINE CENTER	schedule	perbaikan	perbaikan support an clamp
23-09-15	1	52	HIDAYAT	piston rod	1018	MACHINE CENTER	schedule	Reparasi	tidak bisa auto
23-09-15	1	51	HIDAYAT	spring FF	1693	NCLATHE	schedule	Reparasi	proses welding pemotretan tidak bekerja
22-09-15	1	50	ZAINAL	DC welding	1109	AIR BLOW MACHINE	emergency	Down time	axis motor over heat
22-09-15	1	49	PARYADI	spring FF	607	AUTO DRILL	schedule	Service	stopper mc sering macet
22-09-15	1	48	ZAINAL	SS bracket line-e	1014	PUNCH	emergency	Down time	mc 1109 error alarm
22-09-15	1	47	AU PRASTIONO	piston rod	1431	AUTO DRILL	emergency	Reparasi	tidak bisa auto

(Sumber: PT Showa Indonesia Manufacturing, 2015)

4.10 Analisis Sistem Pemeliharaan Mesin

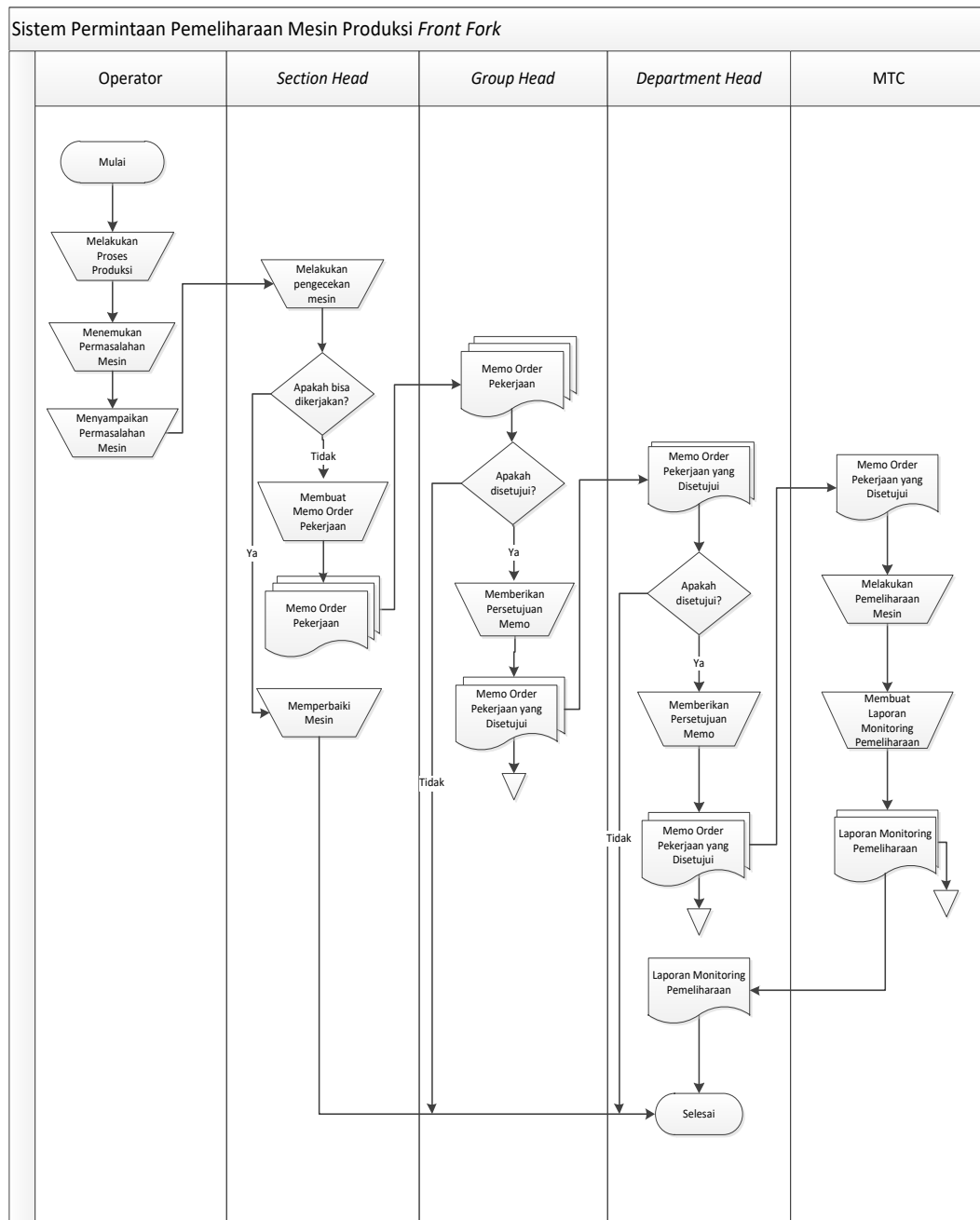
Flowmap menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem, menunjukkan arus dari pekerjaan secara keseluruhan dari sistem termasuk dokumen-dokumennya yang terkait dan digunakan untuk alat bantu komunikasi dan untuk dokumentasi. *Flowmap* ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem. Analisis sistem yang sedang berjalan digambarkan dengan *flowmap* yang bertujuan untuk memudahkan pemahaman terhadap informasi tersebut dan menyederhanakan rangkaian proses atau prosedur dengan menggambarannya secara ringkas, jelas, dan logis. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem.

4.10.1 Prosedur Pemeliharaan Mesin yang Sedang Berjalan

Prosedur proses pemeliharaan mesin yang sedang berjalan adalah sebagai berikut:

1. Operator melakukan proses produksi dan menemukan permasalahan mesin.
2. *Section head* melakukan pengecekan permasalahan mesin yang diminta oleh Operator.
 - Jika permintaan permasalahan dapat diperbaiki, maka akan langsung diperbaiki dan pekerjaan selesai.
 - Apabila permintaan pekerjaan tidak dapat dikerjakan, maka *Section Head* akan membuat memo order pekerjaan, lalu diberikan kepada *Group Head*.
3. *Group head* mendapatkan memo dan memberikan persetujuan dan menyimpan slip kedua untuk pertinggal.
 - Apabila permintaan pekerjaan tidak dapat dipenuhi dan permintaan pekerjaan tidak dilakukan.
 - Permintaan pekerjaan yang sudah disetujui maka akan diteruskan kepada *Department Head*.
4. Setelah mendapatkan persetujuan dari *Group Head* maka permintaan pekerjaan diteruskan sebagai sepengetahuan untuk meminta persetujuan dari *Department Head*.
5. *Department Head* kemudian memberikan persetujuan atas permintaan pekerjaan dan menyimpan slip ketiga sebagai pertinggal.
 - Apabila permintaan pekerjaan tidak dapat dipenuhi dan permintaan pekerjaan tidak dilakukan.
 - Permintaan pekerjaan yang sudah disetujui maka akan diteruskan kepada *Maintenance*.
6. Permintaan pekerjaan yang sudah mendapat persetujuan kemudian diterima oleh *Maintenance* dan melakukan permintaan pemeliharaan.
7. *Maintenance* membuat laporan monitoring permintaan pemeliharaan.

Berikut adalah *flowmap* proses permintaan pekerjaan pemeliharaan mesin *front fork 2 W* di Seksi *Assembling* PT Showa Indonesia Manufacturing, (lihat Gambar IV.7):



Gambar IV.7 *Flowmap* yang Sedang Berjalan di Seksi *Assembling* PT Showa Indonesia Manufacturing
(Sumber: Hasil Pengolahan Data, 2015)

4.10.2 Sistem Pemodelan Pemeliharaan Mesin yang Berjalan

Analisis terhadap sistem yang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan kelemahan sistem serta solusi atas masalah tersebut sehingga dapat dijadikan rencang bangun sistem yang baru. Kegiatan analisis sitem yang berjalan dengan menggunakan analisis sistem yang berorientasi pada objek-objek yang sangat diperlukan oleh sistem yang akan dirancang. Dengan maksud untuk menitikberatkan kepada fungsionalitas sistem yang berjalan dengan tidak terlalu menitikberatkan pada alur proses dari sistem.

Hasil analisis ini direpresentasikan dengan *UML* melalui diagram *Use Case*. Pertimbangan dari diagram ini dapat mewakili secara keseluruhan sistem yang berjalan yang dapat dimengerti oleh *user*. Berikut detail dari hasil analisis adalah sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi permintaan pemeliharaan mesin yang sedang berjalan pada di Seksi *Assembling* di PT Showa Indonesia Manufacturing dapat dilihat pada Tabel IV.3 berikut:

Tabel IV.3 Definisi Aktor Sistem Informasi Permintaan Pemeliharaan

No.	Aktor	Definisi
1.	Operator	Operator melakukan proses produksi.
2.	<i>Section Head</i>	<i>Section Head</i> menerima informasi tentang permintaan pemeliharaan dari operator dan menuliskan pada memo.
3.	<i>Group Head</i>	Group Head menerima memo pemeliharaan dan memberikan persetujuan.
4.	<i>Department Head</i>	<i>Department Head</i> menerima memo permintaan pemeliharaan dan memberikan persetujuan pada memo tersebut.
5.	MTC	MTC melakukan permintaan pemeliharaan dan membuat laporan monitoring pemeliharaan mesin yang telah selesai dikerjakan.

(Sumber : Hasil Pengolahan Data, 2015)

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* sistem informasi permintaan pemeliharaan mesin yang sedang berjalan pada seksi

assembling di PT Showa Indonesia Manufacturing dapat dilihat pada Tabel IV.4 berikut:

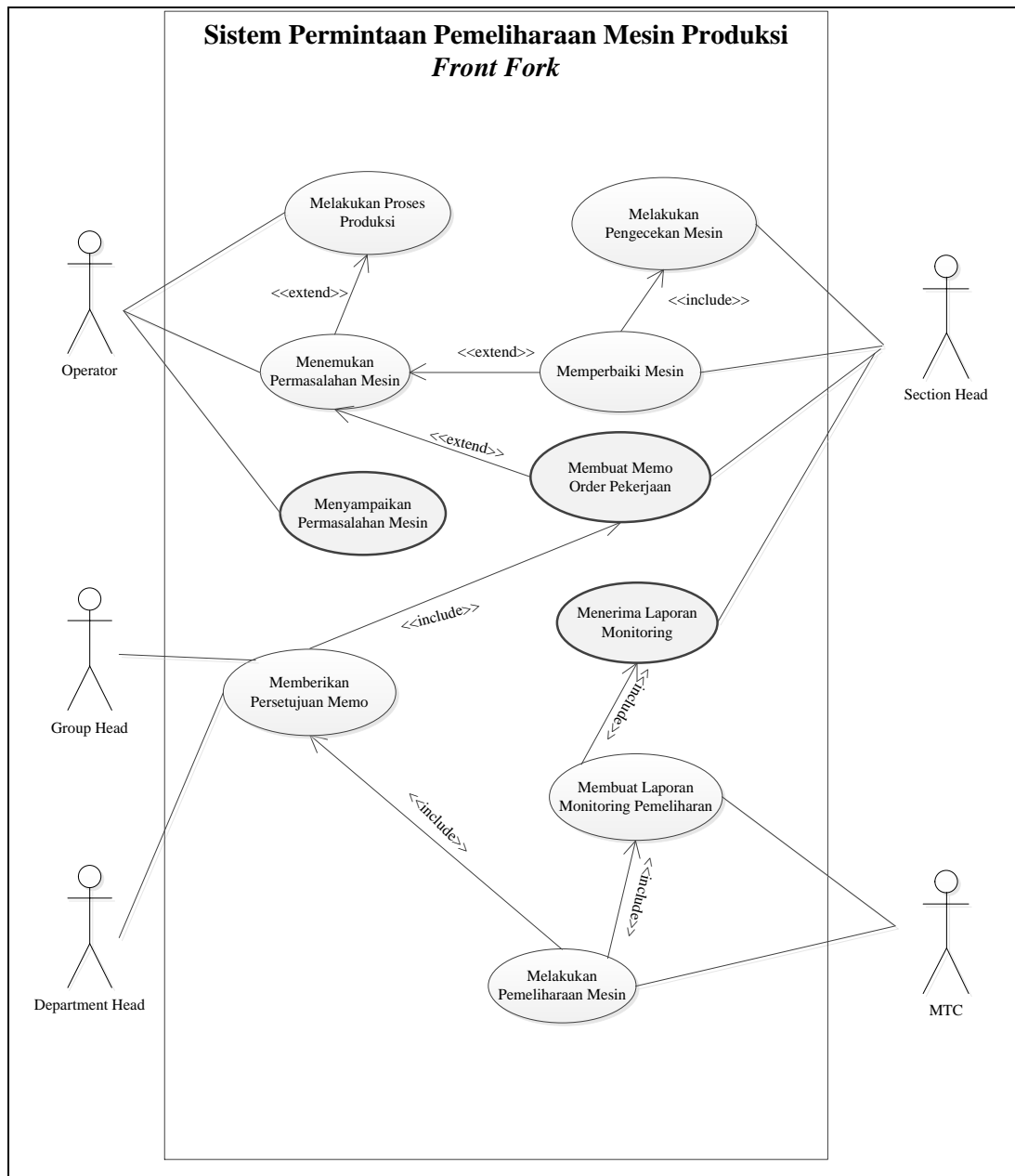
Tabel IV.4 Definisi *Use Case* Sistem Informasi Sistem Informasi yang Berjalan

No.	<i>Use Case</i>	Definisi
1.	Melakukan proses produksi.	Proses melakukan proses produksi.
2.	Menemukan permasalahan mesin.	Proses menemukan permasalahan mesin.
3.	Melakukan pengecekan mesin	Proses pengecekan mesin untuk menemukan permasalahan mesin.
4.	Membuat memo order pekerjaan.	Proses membuat memo order pekerjaan untuk menginformasikan permasalahan mesin.
5.	Memperbaiki mesin	Proses perbaikan dilakukan <i>Section Head</i> .
6.	Memberikan persetujuan memo.	Proses memberikan persetujuan untuk melakukan permintaan pemeliharaan mesin.
7.	Melakukan pemeliharaan mesin.	Melakukan pemeliharaan mesin sesuai dengan permasalahan mesin yang diminta.
8.	Membuat laporan <i>monitoring</i> pemeliharaan	Proses pembuatan laporan <i>monitoring</i> permintaan pemeliharaan mesin.
9.	Menerima laporan <i>monitoring</i>	Proses menerima laporan <i>monitoring</i> pemeliharaan

(Sumber : Hasil Pengolahan Data, 2015)

3. *Use Case Diagram*

Use case diagram sistem informasi permintaan pemeliharaan yang sedang berjalan pada di seksi *assembling* di PT Showa Indonesia Manufacturing., lihat Gambar IV.8 di bawah ini:



Gambar IV.8 Use Case Diagram Sistem Informasi Permintaan Pemeliharaan Mesin

(Sumber: Hasil Pengolahan Data, 2015)

Penjelasan *use case diagram* sistem informasi permintaan pemeliharaan mesin yang sedang berjalan di seksi *assembling* di PT Showa Indonesia Manufacturing adalah sebagai berikut:

4.11 Evaluasi Sistem yang Sedang Berjalan

Evaluasi sistem berfungsi sebagai proses identifikasi dan merupakan pengumpulan hasil dari proses pengolahan data yang telah dilakukan. Berbagai permasalahan diklasifikasi dan dicarikan upaya atau rencana pemecahan dari permasalahan tersebut.

Berikut ini dijelaskan berbagai permasalahan yang telah didapat dari hasil pengolahan data beserta pemecahan permasalahannya (lihat Tabel IV.5):

Tabel IV.5 Evaluasi Sistem yang Berjalan

No.	Masalah	Pemecahan Masalah
1.	Memerlukan kertas memo order pekerjaan dan waktu yang lama dalam penyampaian informasi.	Proses pengolahan data akan dilakukan dengan aplikasi berbasis <i>web</i> .
2.	Media penyimpanan dan sistem pencatatan masih menggunakan cara manual maka proses pencarian data pemeliharaan mesin akan memerlukan waktu yang lama.	Diperlukan <i>database</i> khusus mengenai data produksi yang tersedia beserta informasi-informasi yang mendukung data produksi tersebut.
3.	Tidak adanya informasi pengolahan data pemeliharaan mesin produksi <i>front fork</i> di seksi <i>assembling</i> .	Dibuatkan aplikasi pemeliharaan mesin berbasis <i>web</i> agar lebih mudah dalam mengolah informasi produksi.

(Sumber : Hasil Pengolahan Data, 2015)

Berdasarkan pengolahan data dan analisis permasalahan di atas, maka akan dirancang suatu aplikasi produksi berbasis *web* yang bertujuan untuk pengolahan data hasil pemeliharaan mesin produksi *front fork* di seksi *Assembling*. Perancangan aplikasi tersebut akan diimplementasikan menggunakan bahasa pemrograman PHP versi 5.6.3 dan MySQL versi 5.6.21 sebagai basis data. Aplikasi yang dibuat ini dapat mempercepat pengolahan dan pencarian data, sehingga setiap proses pemeliharaan mesin produksi *front fork* 2 W di Seksi *Assembling* dapat tercatat secara tepat waktu dan informasi pemeliharaan yang diharapkan dapat diberikan seakurat mungkin.

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mendapatkan informasi kebutuhan sistem yang diharapkan dapat diusulkan perbaikan terhadap sistem yang diamati. Berdasarkan hasil pengolahan data dan analisis sistem, dibutuhkan perancangan aplikasi sistem informasi pemeliharaan mesin pada produksi *front fork 2 W* sebagai sarana untuk memberikan informasi mengenai pelaporan pemeliharaan mesin dari kegiatan produksi Seksi *Assembling*. Berikut adalah daftar kebutuhan sistem informasi pemeliharaan mesin (lihat Tabel V.1):

Tabel V.1 Identifikasi Kebutuhan Sistem

Kebutuhan Sistem	
<i>Project Name</i>	Sistem Informasi Pemeliharaan Mesin pada Produksi <i>Front Fork</i> .
<i>Business Need</i>	<ol style="list-style-type: none"> 1. Terwujudnya sebuah sistem informasi pemeliharaan mesin pada produksi <i>front fork</i> yang dapat memudahkan <i>Section Head</i> menyampaikan permasalahan mesin dari Operator ke <i>Maintenace</i>. 2. Terwujudnya sebuah sistem yang mempercepat proses pengolahan laporan <i>monitoring</i> pemeliharaan. 3. Mempermudah pendistribusian pemeliharaan mesin yang telah disetujui.
<i>Business Requirement</i>	<ol style="list-style-type: none"> 1. Memberikan sistem pengolahan data yang cepat dan akurat. 2. Memberikan sistem informasi pemeliharaan mesin yang membantu dalam pembuatan laporan pemeliharaan <i>monitoring</i>.
<i>Business Value</i>	<ol style="list-style-type: none"> 1. Membuat memo permintaan pemeliharaan mesin pada seksi <i>Assembling</i> produksi <i>front fork 2 W</i> di PT Showa Indonesia Manufacturing menjadi

	terkomputerisasi.
--	-------------------

Tabel V.1 Identifikasi Kebutuhan Sistem (lanjutan)

Kebutuhan Sistem	
	2. Merancang sistem informasi pemeliharaan mesin dan membuat basis data menggunakan MySQL sehingga memudahkan dalam penyimpanan data pemeliharaan mesin dan mencegah terjadinya kehilangan data.
<i>Special Issues or Constrains</i>	1. <i>Input</i> : Data Master <i>Proses</i> : Pengolahan data transaksi <i>Output</i> : Laporan <i>monitoiring</i>

(Sumber: Hasil Analisis Data, 2016)

5.2 Prosedur Sistem Informasi Pemeliharaan Mesin Usulan

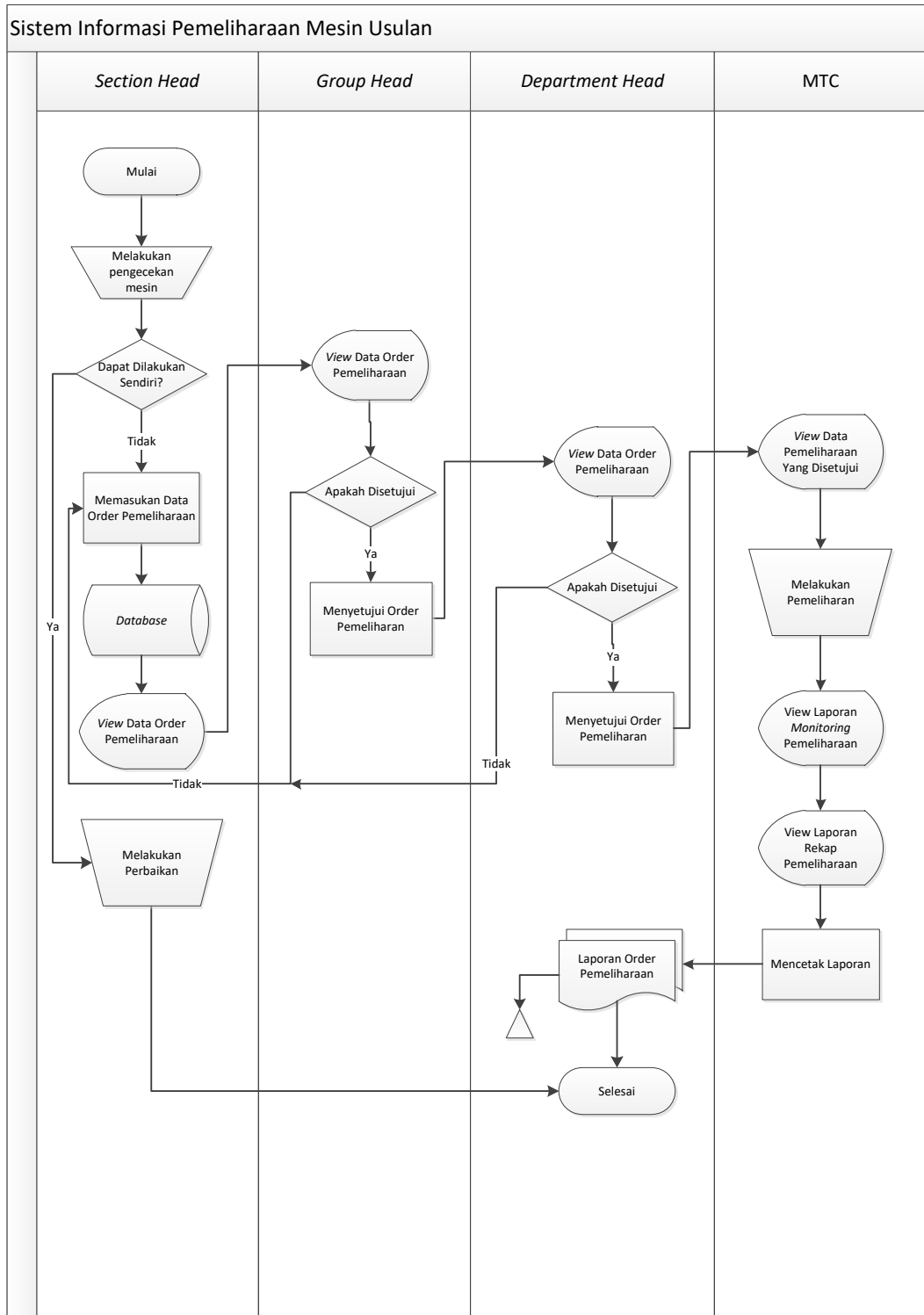
Prosedur sistem informasi pemeliharaan mesin produksi *front fork 2 W* Seksi *Assembling* yang diusulkan melalui tahapan sebagai berikut:

1. *Section Head* melakukan pengecekan mesin
 - Jika *Section Head* dapat melakukan perbaikan maka pekerjaan selesai.
 - Apabila *Section Head* tidak dapat melakukan perbaikan maka *Section Head* akan memasukkan data permasalahan mesin produksi pada order pemeliharaan.
2. *Group Head* memberikan persetujuan terhadap order pemeliharaan yang telah menginformasikan permasalahan mesin yang terjadi.
3. *Department Head* memberikan persetujuan pada order pemeliharaan yang telah disetujui sebelumnya oleh *Section Head*.
4. *Maintenance* menerima order pemeliharaan kemudian melakukan permintaan pemeliharaan mesin.
5. *Maintenance memonitoring* pemeliharaan sebagai laporan pemeliharaan.
6. *Maintenance* mencetak laporan harian pemeliharaan dan memberikan kepada *Departement Head*.

7. *Maintenance* melakukan evaluasi hasil pemeliharaan mesin sesuai laporan *harian* pemeliharaan untuk memberikan penanganan terhadap permasalahan-permasalahan mesin dalam pemeliharaan mesin produksi selanjutnya.

Flowmap Sistem Informasi Pemeliharaan Mesin pada Produksi *Front Fork*

2 W Usulan sebagai berikut (Lihat Gambar V.1):



Gambar V.1 *Flowmap* Sistem Informasi Pemeliharaan Mesin pada Produksi *Front Fork*

(Sumber: Hasil Analisis Data, 2016)

5.3 Analisis dan Perancangan Sistem Pemeliharaan Mesin Usulan

Permasalahan dalam pemeliharaan mesin yang ada pada Seksi *Assembling* diatasi dengan mengajukan usulan sistem baru dengan menerapkan penggunaan aplikasi komputer untuk pengolahan data yang akan mendukung beberapa proses bisnis yang sedang berjalan pada perusahaan. Dengan menggunakan aplikasi ini membuat sistem menjadi terintegrasi sehingga dapat menyajikan informasi secara cepat dan dapat meningkatkan fungsionalitas proses pemeliharaan mesin produksi *front front 2 W* yang ada pada Seksi *Assembling* di PT Showa Indonesia Manufacturing.

Perancangan aplikasi yang diusulkan menggunakan model pengembangan perangkat lunak prototipe evolusioner. Tahap ini adalah tahap kedua pada metodologi prototipe evolusioner yaitu membuat prototipe untuk model sistem setelah sebelumnya melakukan tahap mengidentifikasi kebutuhan. Analisis proses sistem informasi pemeliharaan mesin menggunakan *tools* pemodelan sistem UML (*unified modeling language*) yang meliputi pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*, *deployment diagram* dan juga pembuatan kamus data. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun.

5.3.1 Use Case Diagram

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Hasil analisis direpresentasikan dengan UML melalui diagram *Use Case*. Pertimbangan dari diagram ini dapat mewakili secara keseluruhan sistem yang berjalan yang dapat dimengerti oleh *user*. Berikut detail dari hasil analisis adalah sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi pemeliharaan mesin pada di Seksi *Assembling* di PT Showa Indonesia Manufacturing dapat dilihat pada Tabel IV.2 berikut:

Tabel IV.2 Definisi Aktor *Use Case Diagram* Sistem Usulan

No.	Aktor	Definisi
1.	<i>Section Head</i>	<i>Section Head</i> adalah karyawan yang memiliki hak akses untuk <i>input form</i> order pemeliharaan untuk memberikan informasi permasalahan mesin dan dapat <i>view</i> data order pemeliharaan mesin.
2.	<i>Group Head</i>	<i>Group Head</i> bertanggung jawab terhadap proses penentuan disetujui/ditolak pada order pemeliharaan yang diajukan. Dalam hal ini hak akses yang diberikan adalah <i>view</i> order pemeliharaan dan <i>input</i> persetujuan atau penolakan order pemeliharaan.
3.	<i>Department Head</i>	<i>Department Head</i> bertanggung jawab terhadap proses pengelolaan data master pada Seksi <i>Assembling</i> produksi <i>front fork</i> . Dalam hal ini hak akses yang diberikan yaitu melakukan pengelolaan data master, seperti data <i>shift</i> kerja, data mesin, data jabatan, dan data karyawan. <i>Department Head</i> juga mempunyai hak akses untuk <i>view</i> order pemeliharaan dan bertanggung jawab penentuan disetujui/ditolak pada order pemeliharaan.
4.	MTC	MTC bertanggung jawab terhadap proses melakukan pemeliharaan dan mengelola laporan <i>monitoring</i> pemeliharaan. Dalam hal ini hak akses yang diberikan adalah <i>view</i> order pemeliharaan yang telah ditolak/disetujui, <i>input</i> laporan order pemeliharaan apabila sudah dikerjakan atau masih dalam pengerjaan, <i>view</i> laporan <i>monitoring</i> pemeliharaan.

(Sumber: Hasil Analisis Data, 2016)

2. Definisi Use Case

Pendefinisian *use case* pada sistem informasi pemeliharaan mesin usulan dapat dilihat pada Tabel V.3:

Tabel V.3 Definisi *Use Case Diagram* Sistem Usulan

No.	Use Case	Deskripsi
1.	<i>Login</i>	Proses untuk melakukan <i>login</i> pada aplikasi
2.	Mengelola data <i>master</i>	<i>Use case</i> ini menggambarkan proses pengelolaan data <i>master</i> yang terdiri dari data <i>user</i> , karyawan, <i>shift</i> , lini dan mesin yaitu menambah, mengubah, dan mencari data.

Tabel V.3 Definisi Aktor *Use Case Diagram* Sistem Usulan (lanjutan)

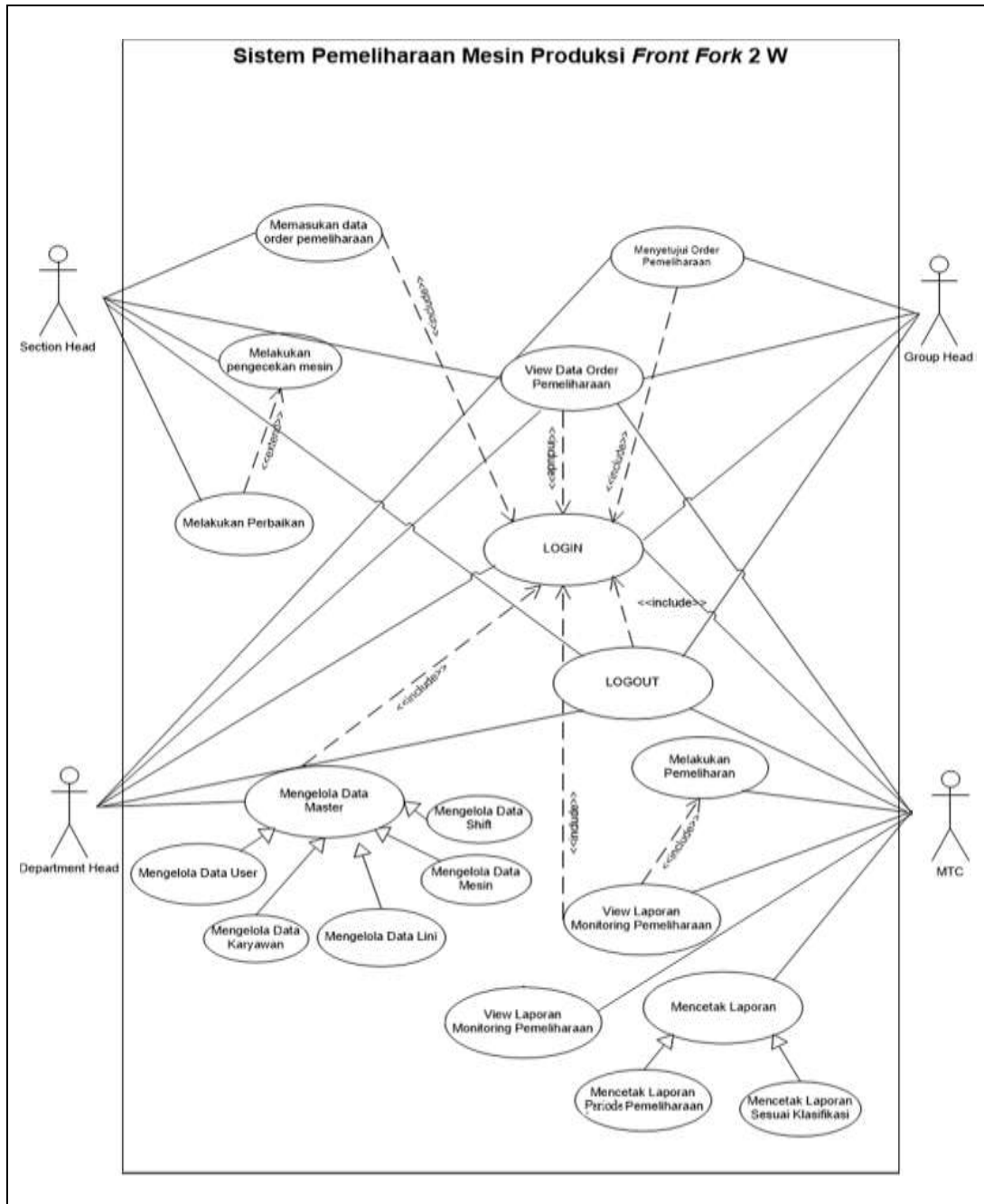
No.	Use Case	Deskripsi
3.	Melakukan pengecekan mesin	Proses apabila <i>Section Head</i> melakukan pengecekan baik permintaan atau pemeriksaan berkala.
4.	Melakukan perbaikan mesin	Proses melakukan perbaikan apabila dapat dikerjakan oleh <i>Section Head</i> .
5.	Memasukan data order pemeliharaan	Proses dimana <i>Section Head</i> memasukan data order pemeliharaan dan penanganan yang dibutuhkan.
6.	Menyetujui order pemeliharaan	Proses memberikan <i>input</i> terhadap disetujui/ditolak order pemeliharaan yang diajukan oleh <i>Section Head</i> .
7.	<i>View</i> order pemeliharaan	Proses menampilkan order pemeliharaan mesin yang disetujui/ditolak.
8.	Melakukan pemeliharaan	Proses melakukan pemeliharaan sesuai dengan order pemeliharaan yang diminta.
9.	<i>View</i> Laporan <i>Monitoring</i> pemeliharaan	Proses <i>view</i> laporan setelah <i>memonitoring</i> pemeliharaan baik dalam tahap pengerjaan atau telah diselesaikan.
10.	<i>View</i> laporan rekap order pemeliharaan	Proses melihat laporan rekap order pemeliharaan.
11.	Mencetak laporan <i>monitoring</i> pemeliharaan	Proses mencetak laporan <i>monitoring</i> pemeliharaan.

(Sumber: Hasil Analisis Data, 2016)

3. *Use Case Diagram*

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Rancangan *use case*

diagram sistem informasi pemeliharaan mesin pada produksi *front fork* yang diusulkan dapat dilihat pada Gambar V.2.



Gambar V.2 Use Case Diagram Sistem Informasi Pemeliharaan Mesin Usulan
(Sumber: Hasil Analisis Data, 2016)

4. Skenario Use Case

Skenario *use case* masing-masing pada sistem informasi pemeliharaan mesin usulan dapat dilihat pada poin berikut:

a. Use Case Login

Berikut adalah skenario *use case login* yang terdapat pada Tabel V.4:

Tabel V.4 Skenario Use Case Login

Nama Use Case	Login
<i>Primary Actor</i>	<i>Section Head, Group Head, Department Head, MTC</i>
<i>Use Case Description</i>	<i>Use case menggambarkan login sesuai dengan hak akses dari user.</i>
<i>Relationship</i>	<i>Association: Section Head, Group Head, Department Head, dan MTC.</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> melakukan <i>login</i> kemudian, masukkan npk dan <i>password</i> pada <i>login form</i>. 2. Sistem mengecek ke basis data, apakah npk dan <i>password</i> benar. 3. Jika npk dan <i>password</i> benar, maka muncul tampilan menu utama. 4. Jika npk dan <i>password</i> salah (tidak <i>valid</i>), maka muncul pesan “Npk dan <i>Password</i> Salah”.

(Sumber: Hasil Analisis Data, 2016)

b. Use Case Mengelola Data Master

Berikut skenario *use case* mengelola data *master* yang terdapat pada tabel V.5:

Tabel V.5 Use Case Description Mengelola Data Master

Nama Use Case	Mengelola Data Master
<i>Primary Actor</i>	<i>Departement Head</i>
<i>Use Case Description</i>	<i>Use case menggambarkan proses pengelolaan data master yang terdiri dari pengguna, lini, karyawan, mesin dan shift yaitu menambah, mengubah, mencari, dan menghapus data.</i>
<i>Relationship</i>	<p>Generalisasi: Data pengguna, lini, karyawan, mesin, dan <i>shift</i>.</p> <p><i>Include: - Login</i></p>

<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu <i>file data master</i>. 3. <i>User</i> memilih sub menu yang terdiri dari data pengguna, lini, karyawan, mesin dan <i>shift</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 4. <i>User</i> melakukan proses cari, tambah, ubah, dan hapus data di basis data masing masing pada data <i>master</i>.

(Sumber: Hasil Analisis Data, 2016)

c. *Use Case* Memasukan Data Order Pemeliharaan

Berikut skenario *use case* memasukan data order pemeliharaan (Lihat Tabel V.6):

Tabel V.6 Skenario *Use Case* Memasukkan Data Order Pemeliharaan

Nama <i>Use Case</i>	Memasukkan Data Pemeliharaan
<i>Primary Actor</i>	<i>Section Head</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses memasukkan data pemeliharaan yang didapat dari kebutuhan untuk pemeliharaan
<i>Relationship</i>	<i>Association: Section Head</i> <i>Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih <i>file</i> order pemeliharaan, pilih sub menu pembuatan order pemeliharaan. 3. Sistem menampilkan <i>form</i> pembuatan order pemeliharaan yang sudah ada sebelumnya. 4. <i>User</i> mengisi <i>form</i> pembuatan order pemeliharaan, kemudian memilih simpan untuk menyimpan data order pemeliharaan.

(Sumber: Hasil Analisis Data, 2016)

d. *Use Case* Menyetujui Data Order Pemeliharaan

Berikut adalah skenario *use case* menyetujui data order pemeliharaan yang terdapat pada Tabel V.7:

Tabel V.7 Skenario *Use Case* Menyetujui Data Order Pemeliharaan

Nama <i>Use Case</i>	Menyetujui Data Order Pemeliharaan
<i>Primary Actor</i>	<i>Group Head, Departement Head</i>

<i>Use Case Description</i>	<i>Use case ini menggambarkan proses disetujui/ditolak data order pemeliharaan.</i>
<i>Relationship</i>	<i>Association: Group Head, Departement Head Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu <i>file</i> persetujuan <i>user</i>. 3. Sistem menampilkan daftar order pemeliharaan baru. 4. <i>User</i> memeriksa kesesuaian data, kemudian memilih status persetujuan <i>user</i> “Disetujui, “Tidak Disetujui”.

(Sumber: Hasil Analisis Data, 2016)

e. *Use Case View* Order Pemeliharaan

Berikut adalah skenario *use case view* order pemeliharaan yang terdapat pada Tabel V.8:

Tabel V.8 Skenario *Use Case View* Order Pemeliharaan

Nama Use Case	View Order Pemeliharaan
<i>Primary Actor</i>	<i>Section Head, Group Head, Department Head, MTC</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan proses melihar order pemeliharaan baik yang sudah diberi persetujuan atau tidak.</i>
<i>Relationship</i>	<i>Association: Group Head, Department Head, MTC Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu <i>file</i> persetujuan <i>user</i>. 3. <i>User</i> memilih <i>progress</i> order pemeliharaan.

(Sumber: Hasil Analisis Data, 2016)

f. *Use Case View* Laporan *Monitoring* Pemeliharaan

Berikut skenario *use case view* laporan *monitoring* pemeliharaan yang terdapat pada Tabel V.9:

Tabel V.9 Skenario *Use Case View* Laporan *Monitoring* Pemeliharaan

Nama Use Case	View Laporan Monitoring Pemeliharaan
<i>Primary Actor</i>	MTC

<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses <i>view</i> laporan <i>monitoring</i> pemeliharaan MTC.
<i>Relationship</i>	<i>Association</i> : MTC <i>Include</i> : - <i>Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. MTC <i>Login</i> dan masuk ke tampilan menu utama. 2. MTC memilih menu laporan. 3. Memilih sub menu laporan <i>monitoring</i> harian atau laporan sesuai dengan klasifikasi, dan laporan rekap order pemeliharaan mesin. 4. Sistem menampilkan laporan order pemeliharaan sesuai yang ditentukan.

(Sumber: Hasil Analisis Data, 2016)

g. *Use Case* Mencetak Laporan

Berikut adalah skenario *use case* mencetak laporan *monitoring* pemeliharaan, laporan sesuai klasifikasi dan laporan rekap order pemeliharaan mesin yang terdapat pada Tabel V.10:

Tabel V.10 Skenario *Use Case* Mencetak Laporan Order Pemeliharaan

Nama <i>Use Case</i>	Mencetak Laporan Order Pemeliharaan
<i>Primary Actor</i>	MTC
<i>Use Case Description</i>	<i>Use case</i> menggambarkan proses mencetak laporan harian pemeliharaan atau laporan klasifikasi pemeliharaan.
<i>Relationship</i>	<i>Association</i> : MTC <i>Include</i> : - <i>Login</i> memasukkan data pemeliharaan mesin, menerima laporan <i>monitoring</i> pemeliharaan. <i>Generalization</i> : mencetak laporan <i>monitoring</i> harian dan atau mencetak laporan sesuai klasifikasi pemeliharaan.
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu laporan. 3. <i>User</i> memilih sub menu laporan harian pemeliharaan dan atau laporan sesuai klasifikasi pemeliharaan.

4. <i>User</i> menekan tombol cetak.

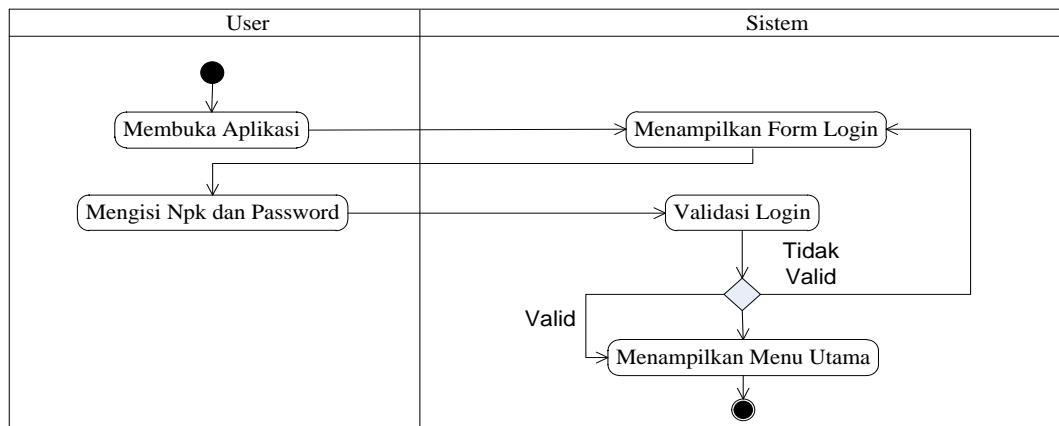
(Sumber: Hasil Analisis Data, 2016)

5.3.2 *Activity Diagram*

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar lebih memahami sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem informasi pemeliharaan mesin.

1. *Activity Diagram Login*

Activity diagram login menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu *Section Head*, *Group Head*, *Department Head* dan MTC untuk dapat masuk ke dalam sistem informasi pemeliharaan mesin (Lihat Gambar V.3):

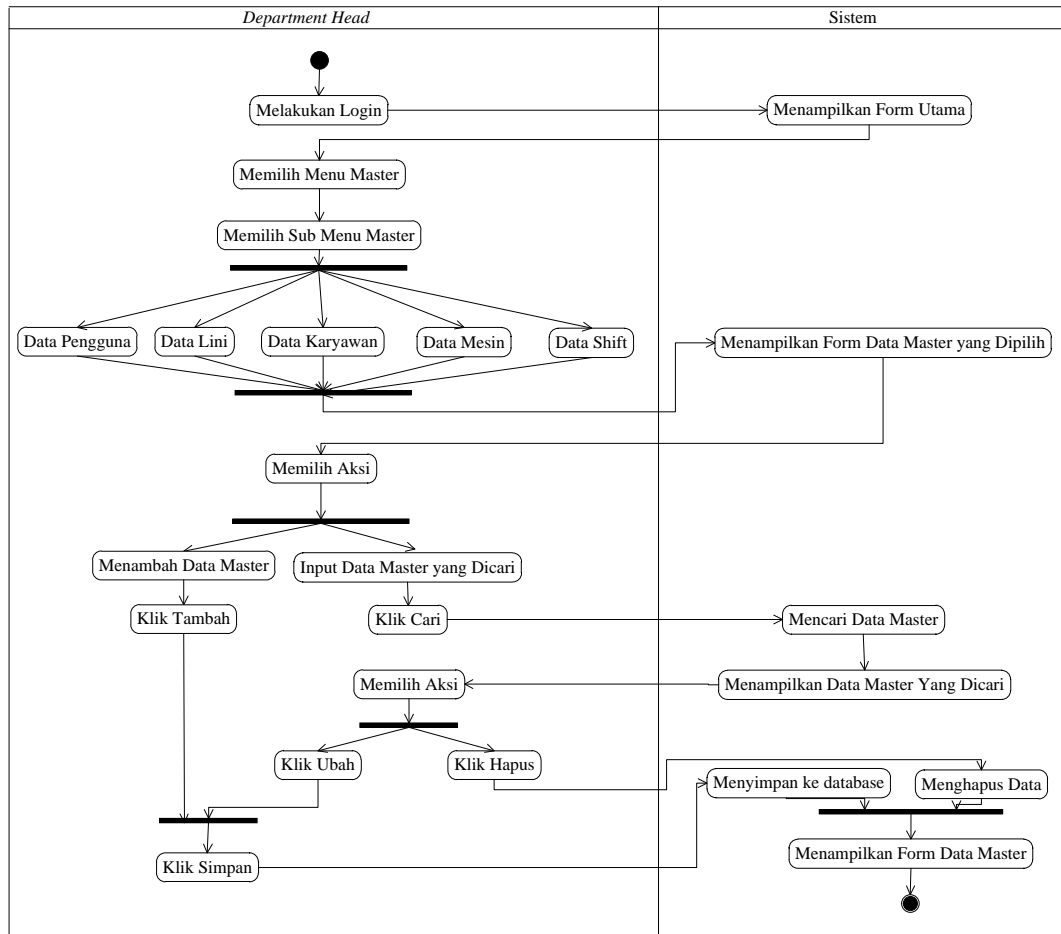


Gambar V.3 *Activity Diagram Login*

(Sumber: Hasil Analisis Data, 2016)

2. *Activity Diagram Mengelola Data Master*

Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam mengelola data *master* (Lihat Gambar V.4):

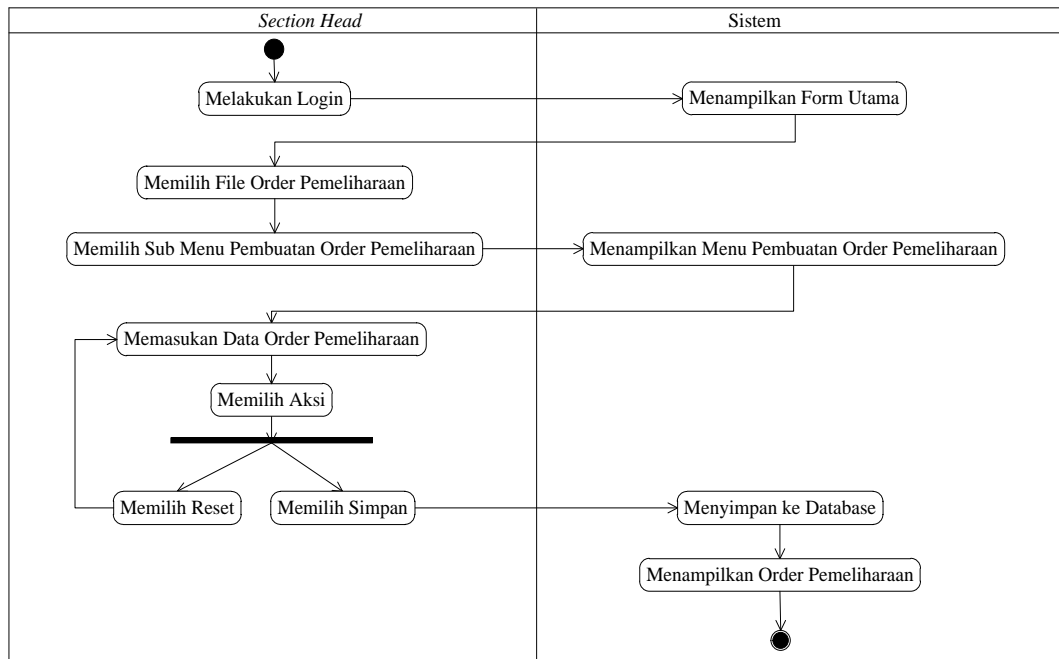


Gambar V.4 Activity Diagram Mengelola Data Master

(Sumber: Hasil Analisis Data, 2016)

3. Activity Diagram Memasukan Data Order Pemeliharaan

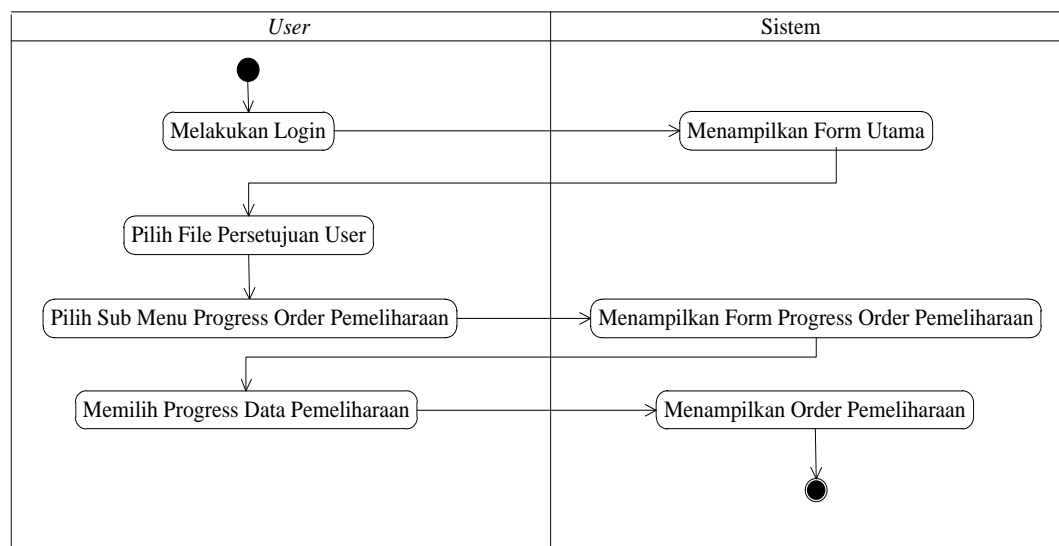
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam memasukan data order pemeliharaan (Lihat Gambar V.5):



Gambar V.5 Activity Diagram Memasukkan Data Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

4. Activity Diagram View Order Pemeliharaan

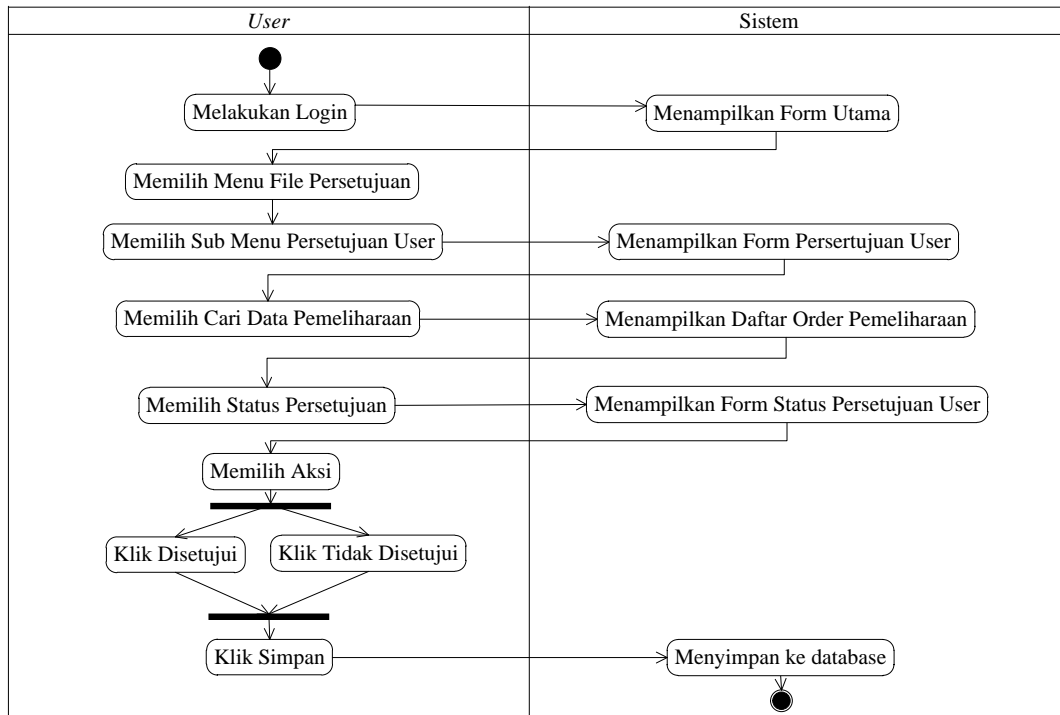
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam view order pemeliharaan (Lihat Gambar V.6):



Gambar V.6 Activity Diagram View Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

5. *Activity Diagram* Menyetujui Data Order Pemeliharaan

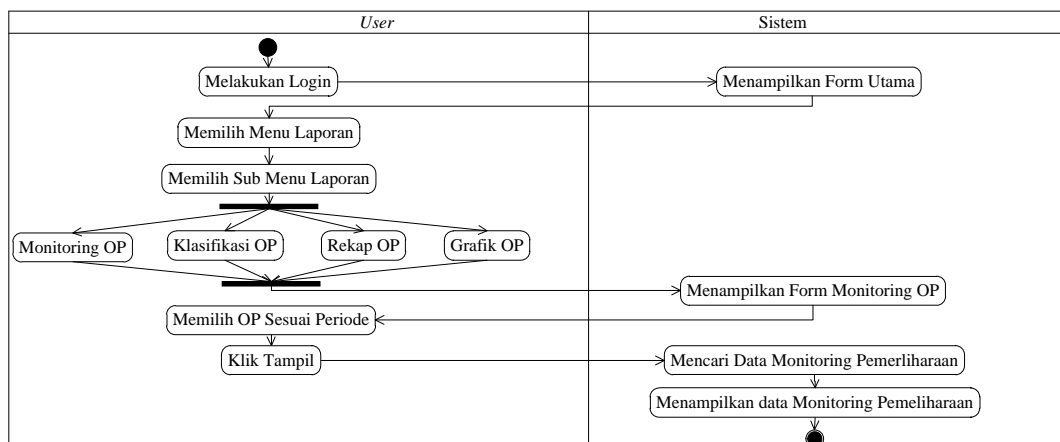
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam menyetujui data order pemeliharaan (Lihat Gambar V.7):



Gambar V.7 *Activity Diagram* Menyetujui Data Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

6. *Activity Diagram* Laporan Order Pemeliharaan

Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam mengelola laporan order pemeliharaan (Lihat Gambar V.8):

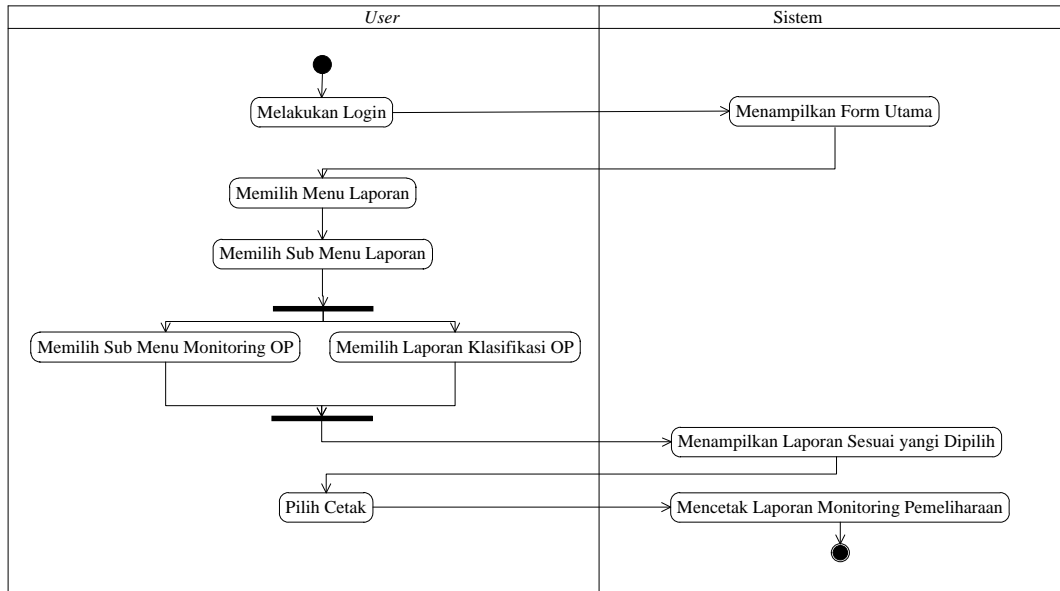


Gambar V.8 *Activity Diagram* Laporan Monitoring Order Pemeliharaan

(Sumber: Hasil Analisis Data, 2016)

7. Activity Diagram Mencetak Laporan Order Pemeliharaan

Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam view laporan monitoring pemeliharaan (Lihat Gambar V.9):



Gambar V.9 Activity Diagram Mencetak Laporan Monitoring Pemeliharaan

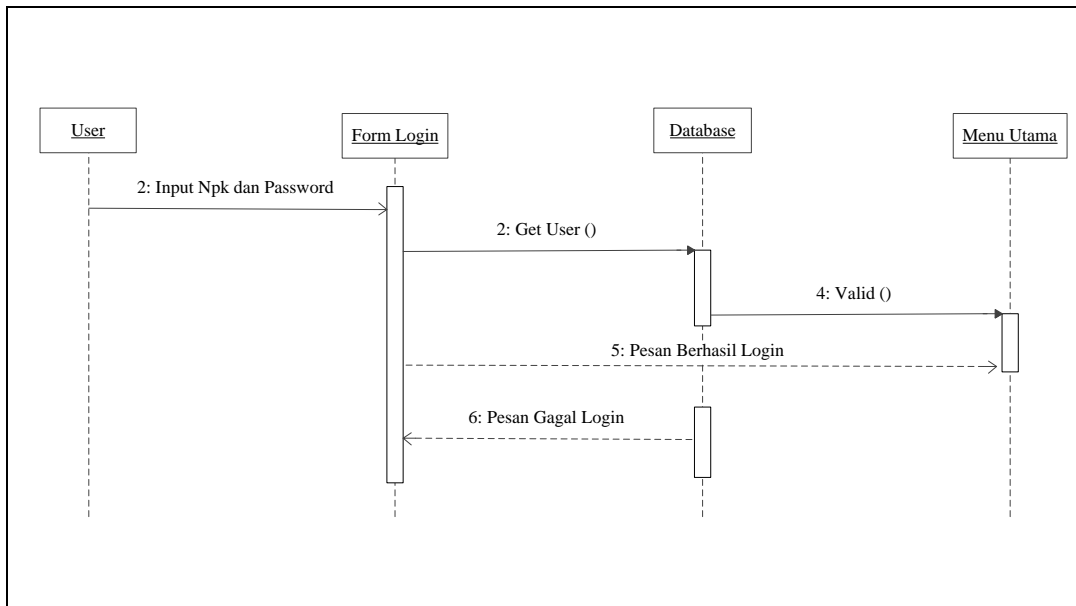
Sumber: Hasil Analisis Data (2016)

5.3.3 Sequence Diagram

Sequence diagram menggambarkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu, di mana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada *diagram*. Hubungan yang ada pada gambar di bawah ini adalah proses yang dilakukan oleh sistem ketika melakukan proses yang sesuai dengan suatu objek *use case diagram*, berikut adalah *sequence diagram* pada sistem informasi pemeliharaan mesin:

1. Sequence Diagram Proses Login

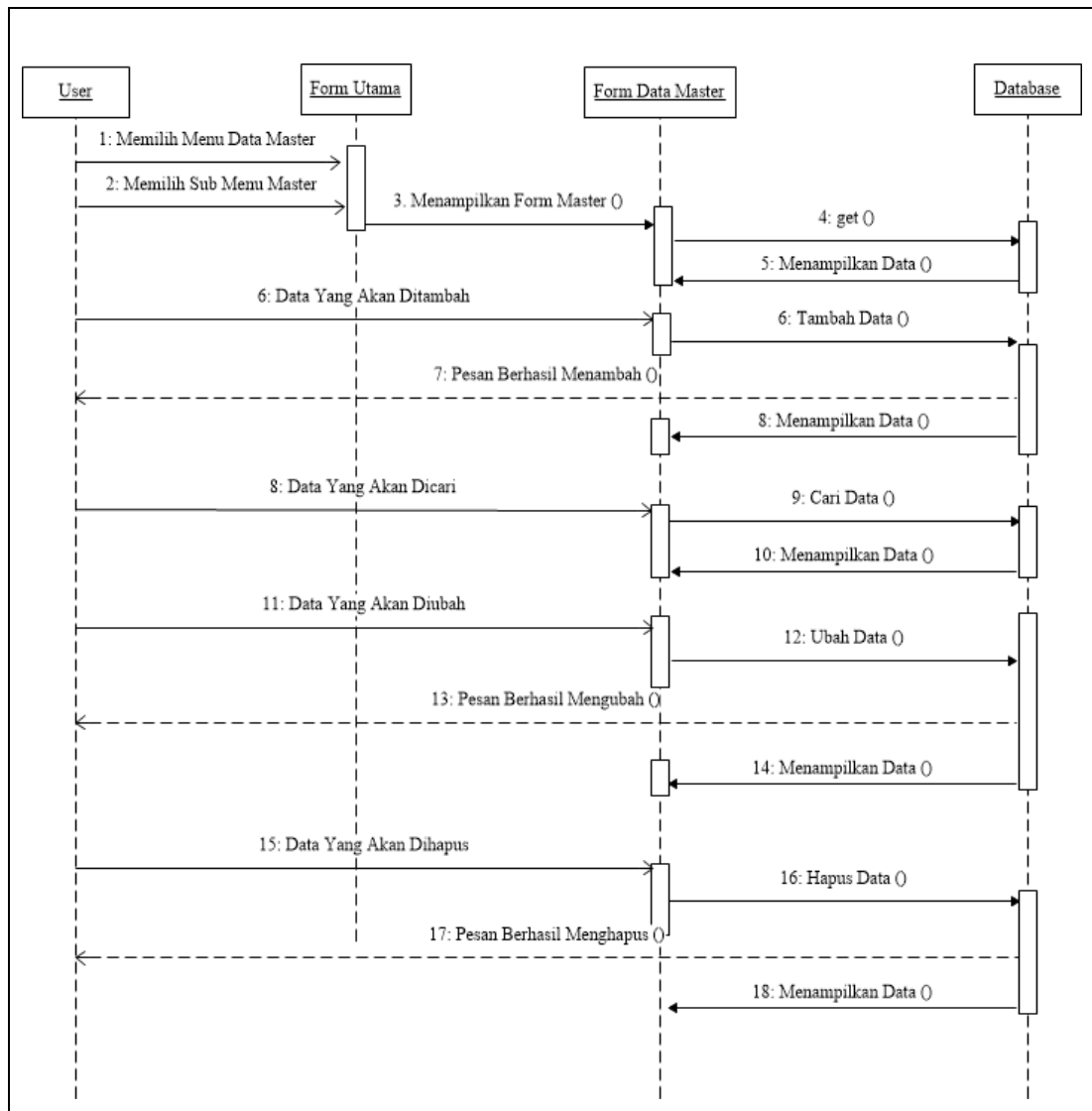
Sequence diagram login menjelaskan proses ini dilakukan oleh *user* sebelum masuk sistem. Proses ini dilakukan oleh *user* sebelum masuk ke sistem. Adapun *sequence diagram* dari *use case login* (Lihat Gambar V.10):



Gambar V.10 *Sequence Diagram* Proses Login
 Sumber: Hasil Analisis Data (2016)

2. *Sequence Diagram* Mengelola Data Master

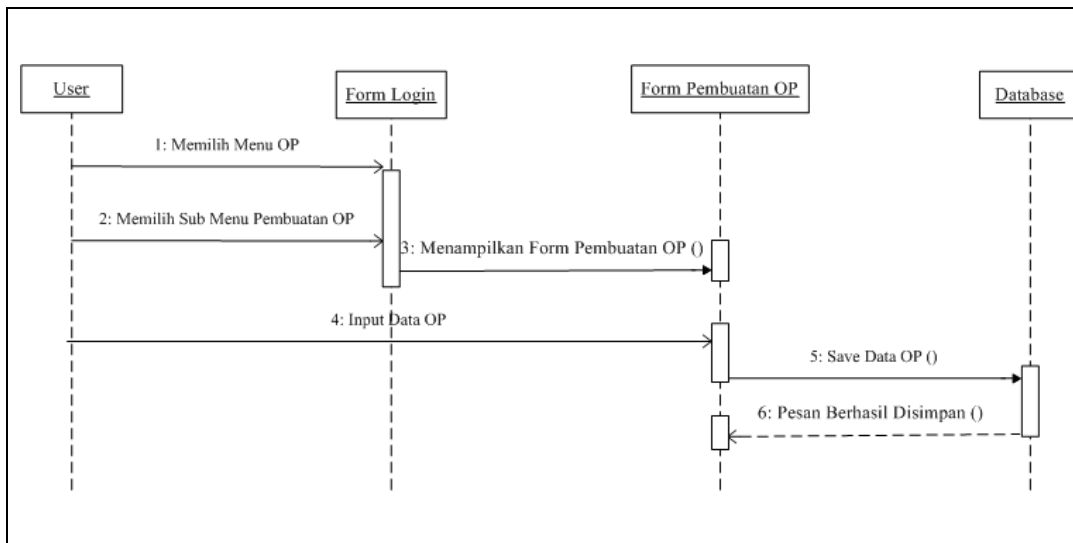
Sequence diagram menjelaskan sebuah *sequence diagram* dalam proses menambah, menghapus dan mengubah data *master*. Adapun *sequence diagram* dari *use case* mengelola data *master* meliputi data pengguna, lini, karyawan, mesin dan *shift* yang dapat dilihat pada Gambar V.11 sebagai berikut:



Gambar V.11 *Sequence Diagram* Mengelola Data Master
(Sumber: Hasil Analisis Data, 2016)

3. *Sequence Diagram* Memasukan Data Order Pemeliharaan

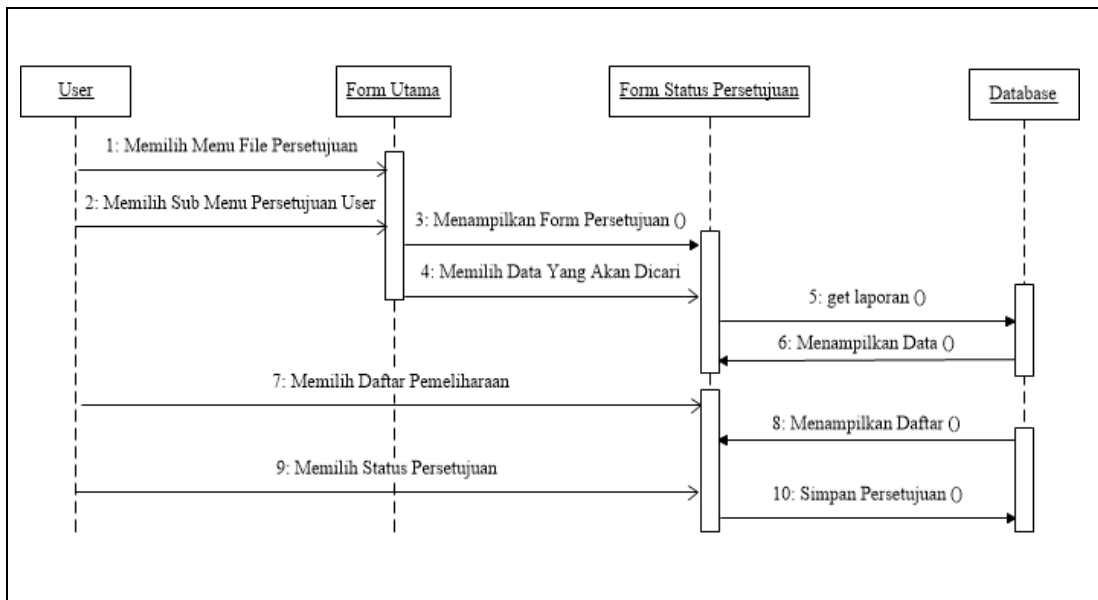
Sequence diagram memasukan data order pemeliharaan menjelaskan proses menambah dan menyimpan data order pemeliharaan mesin yang dilakukan oleh *Section Head*. Adapun *sequence diagram* dari *use case* memasukan data order pemeliharaan dapat dilihat pada Gambar V.12 sebagai berikut:



Gambar V.12 *Sequence Diagram* Memasukkan Data Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

4. *Sequence Diagram* Menyetujui Order Pemeliharaan

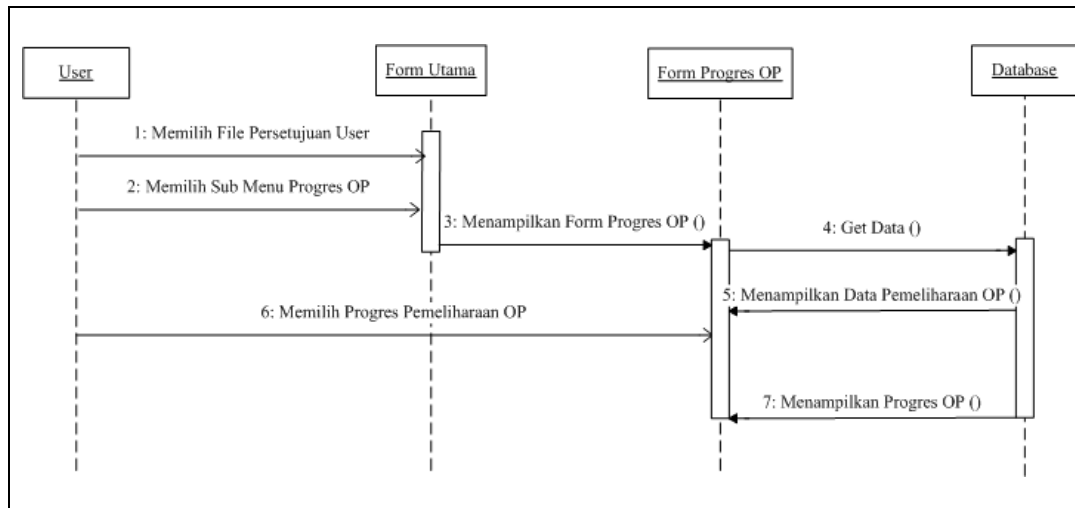
Sequence diagram menyetujui order pemeliharaan menjelaskan sebuah *sequence diagram* dalam proses menyetujui dan tidak disetujui order pemeliharaan. Adapun *sequence diagram* dari *use case* menyetujui order pemeliharaan dapat dilihat pada Gambar V.13 sebagai berikut:



Gambar V.13 *Sequence Diagram* Menyetujui Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

5. *Sequence Diagram View Order Pemeliharaan*

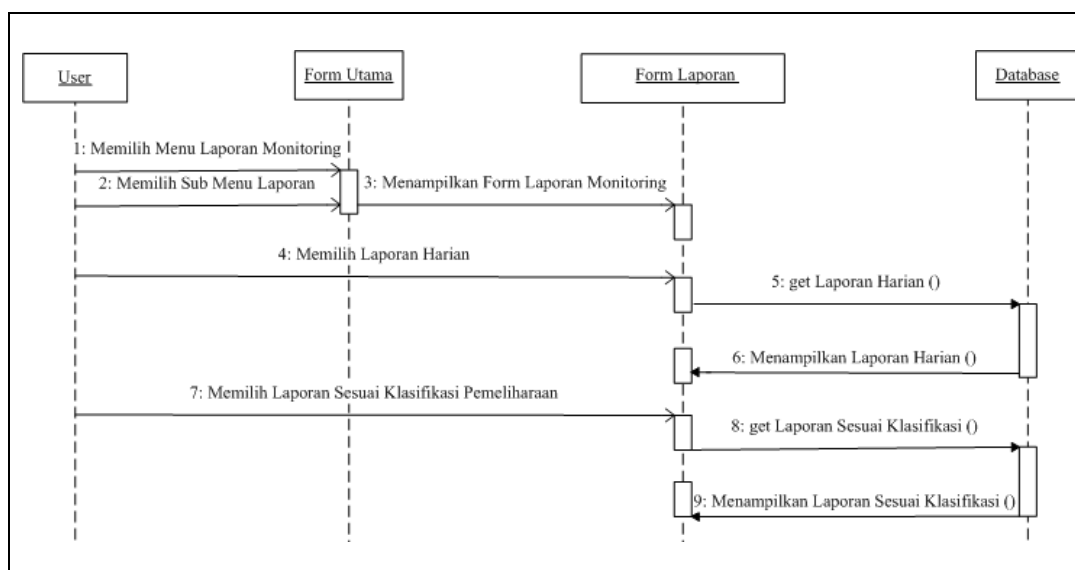
Sequence diagram view order pemeliharaan menjelaskan sebuah *sequence diagram* dalam proses menampilkan pemeliharaan yang telah setuju atau ditolak. Adapun *sequence diagram* dari *use case view order pemeliharaan* dapat dilihat pada Gambar V.14 sebagai berikut:



Gambar V.14 *Sequence Diagram View Order Pemeliharaan*
(Sumber: Hasil Analisis Data, 2016)

6. *Sequence Diagram View Laporan Monitoring Order Pemeliharaan*

Sequence diagram menjelaskan dalam proses melihat data *monitoring* dalam order pemeliharaan (Lihat Gambar V.15):

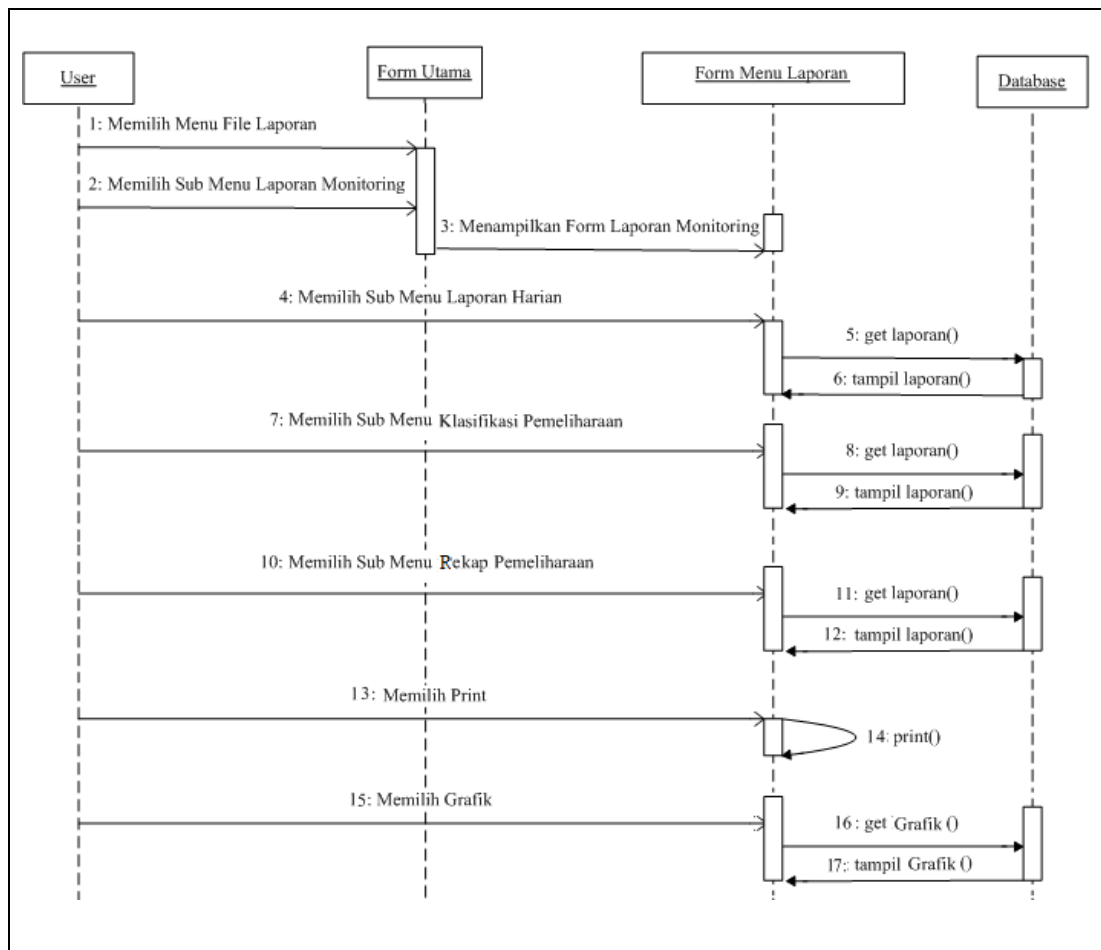


Gambar V.15 *Sequence Diagram Laporan Monitoring Pemeliharaan*

(Sumber: Hasil Analisis Data, 2016)

7. *Sequence Diagram* Mencetak Laporan *Monitoring* Pemeliharaan

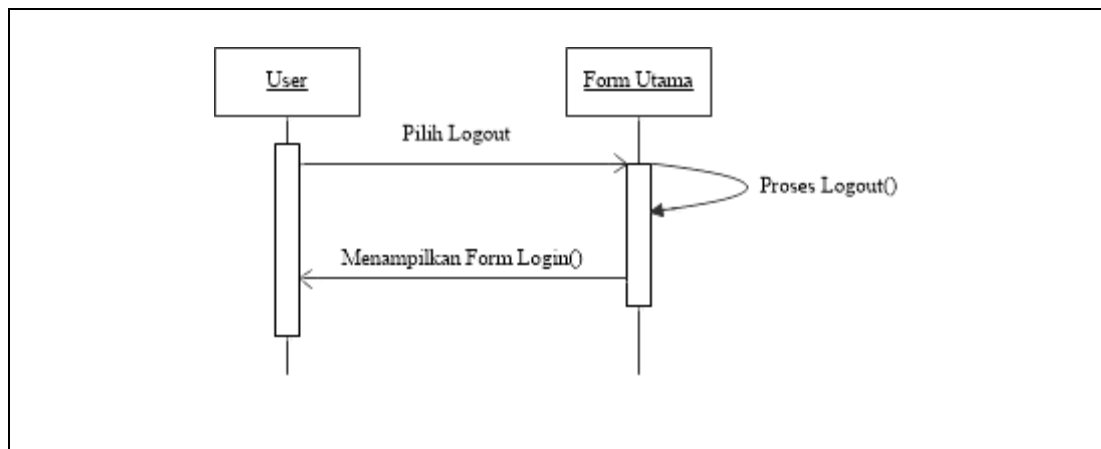
Sequence diagram mencetak laporan *monitoring* pemeliharaan menjelaskan sebuah *sequence diagram* dalam proses mencetak laporan *monitoring* pemeliharaan berdasarkan laporan harian dan laporan sesuai klasifikasi pemeliharaan. Adapun *sequence diagram* dari *use case* laporan *monitoring* pemeliharaan dapat dilihat pada Gambar V.16 sebagai berikut:



Gambar V.16 *Sequence Diagram* Mencetak Laporan *Monitoring* Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

8. *Sequence Diagram* Logout

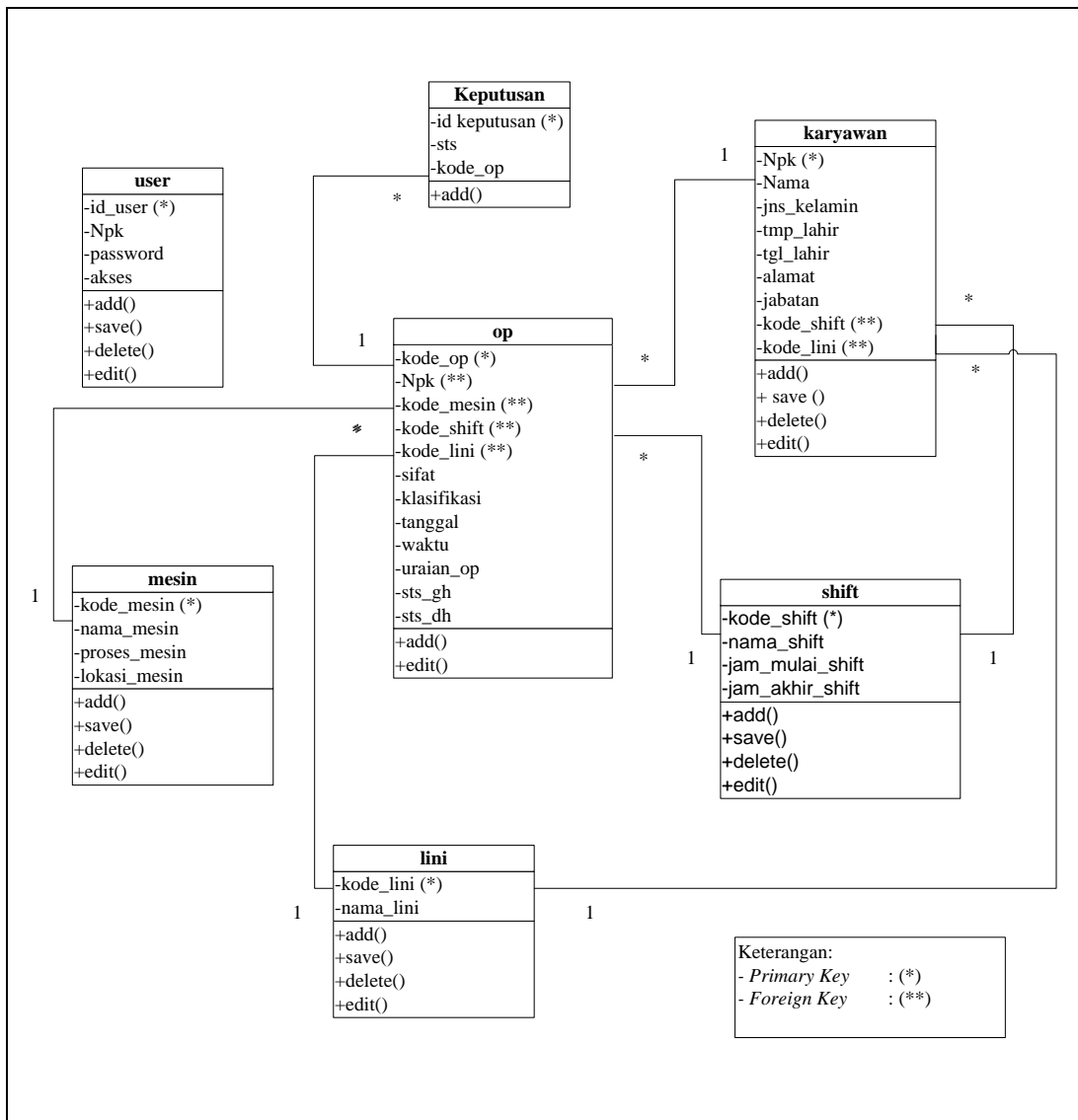
Sequence diagram login menjelaskan sebuah *sequence diagram* dalam proses *logout*. Proses ini dilakukan oleh *user* untuk keluar ke sistem. Adapun *sequence diagram* dari *use case* *logout* dapat dilihat pada gambar V.17 di bawah ini.



Gambar V.17 *Sequence Diagram Logout*
(Sumber: Hasil Analisis Data, 2016)

5.3.4 *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan *detail* tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi order pemeliharaan mesin yang diusulkan dapat dilihat pada Gambar V.18 sebagai berikut:



Gambar V.18 Class Diagram Usulan
 (Sumber: Hasil Analisis Data, 2016)

5.3.5 Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analisi sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan dan basis data. Berikut adalah kamus data sistem informasi order pemeliharaan mesin:

1. Spesifikasi Tabel *User*

Nama Tabel : *user*

Fungsi : Untuk menyimpan data *user*

Tipe : File data *master*

Tabel V.11 Tabel *User*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id <i>User</i>	id_user	<i>Integer</i>	3	<i>Primary Key</i>
2.	Npk	Npk	<i>Char</i>	7	<i>Foreign Key</i>
3.	<i>Password</i>	password	<i>Char</i>	7	
4.	Akses	Akses	<i>Char</i>	3	

(Sumber: Hasil Analisis Data, 2016)

2. Spesifikasi Tabel Karyawan

Nama Tabel : karyawan

Fungsi : Untuk menyimpan data karyawan

Tipe : File data *master*

Tabel V.12 Tabel Karyawan

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Npk	Npk	<i>Char</i>	7	<i>Primary Key</i>
2.	Nama Karyawan	Nama	<i>Varchar</i>	25	
3.	Jenis Kelamin	jns_kelamin	<i>Varchar</i>	10	
4.	Tempat Lahir	tmp_lahir	<i>Varchar</i>	20	
5.	Tanggal Lahir	tgl_lahir	<i>Date</i>	-	
6.	Alamat	alamat	<i>Text</i>	-	
7.	Jabatan	jabatan	<i>Varchar</i>	20	
8.	Kode Shift	kode_shift	<i>Char</i>	2	<i>Foreign Key</i>
9.	Kode Lini	kode_lini	<i>Char</i>	3	<i>Foreign Key</i>

(Sumber: Hasil Analisis Data, 2016)

3. Spesifikasi Tabel Mesin

Nama Tabel : mesin

Fungsi : Untuk menyimpan data Mesin

Tipe : File data *master*

Tabel V.13 Tabel Mesin

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Kode Mesin	kode_mesin	<i>Char</i>	7	<i>Primary Key</i>
2.	Nama Mesin	nama_mesin	<i>Varchar</i>	20	
3.	Proses Mesin	proses_mesin	<i>Varchar</i>	25	
4.	Lokasi Mesin	lokasi_mesin	<i>Text</i>	20	

(Sumber: Hasil Analisis Data, 2016)

4. Spesifikasi Tabel Lini

Nama Tabel : lini

Fungsi : Untuk menyimpan data tipe lini

Tipe : File data *master*

Tabel V.16 Tabel Lini

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Kode Lini	kode_lini	<i>Char</i>	3	<i>Primary key</i>
2.	Nama Lini	nama_lini	<i>Varchar</i>	25	

(Sumber: Hasil Analisis Data, 2016)

5. Spesifikasi Tabel *Shift*

Nama Tabel : *shift*

Fungsi : Untuk menyimpan data *shift*

Tipe : File data *master*

Tabel V.14 Tabel *Shift*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Kode <i>Shift</i>	kode_ <i>shift</i>	<i>Char</i>	2	<i>Primary</i>

					<i>Key</i>
2.	Nama <i>Shift</i>	<i>nama_shift</i>	<i>Varchar</i>	10	
2.	Jam Mulai <i>Shift</i>	<i>jam_mulai_shift</i>	<i>Time</i>	-	
3.	Jam Akhir <i>Shift</i>	<i>Jam_akhir_shift</i>	<i>Time</i>	-	

(Sumber: Hasil Analisis Data, 2016)

6. Spesifikasi Tabel Keputusan

Nama Tabel : keputusan

Fungsi : Untuk menyimpan data Keputusan

Tipe : File Persetujuan

Tabel V.15 Tabel Keputusan

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	Id Status	Idstatus	<i>Char</i>	3	<i>Primary Key</i>
2.	Status	Sts	<i>Varchar</i>	15	
3.	Kode op	kode_op	<i>Char</i>	3	<i>Foreign Key</i>

(Sumber: Hasil Analisis Data, 2016)

7. Spesifikasi Tabel Order Pemeliharaan

Nama Tabel : op

Fungsi : Untuk menyimpan data order pemeliharaan

Tipe : File Order Pemeliharaan

Tabel V.16 Tabel Order Pemeliharaan

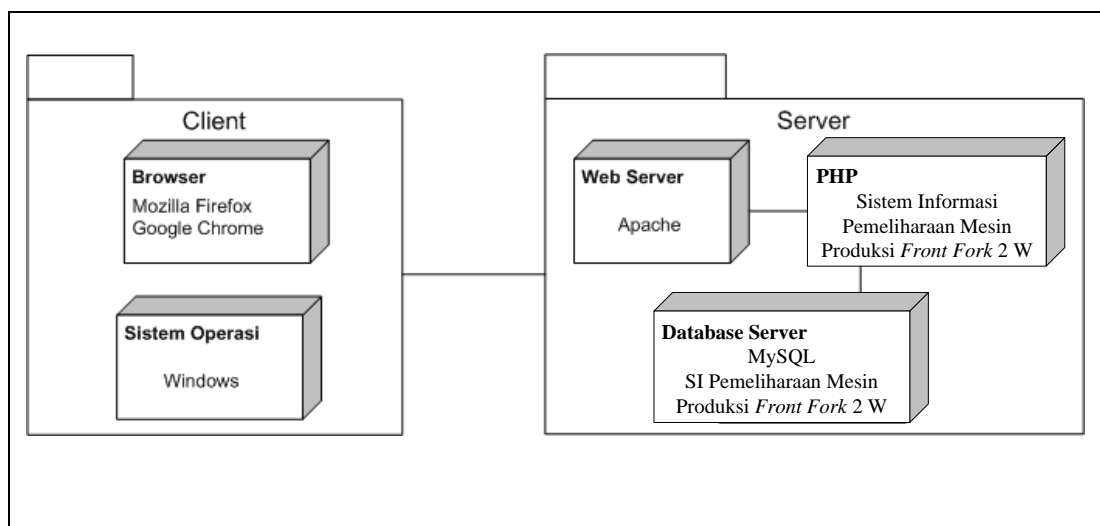
No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Kode OP	kode_op	<i>Char</i>	3	<i>Primary Key</i>
2.	Npk	Npk	<i>Char</i>	7	<i>Foreign Key</i>
3.	Kode Mesin	kode_mesin	<i>Integer</i>	7	<i>Foreign Key</i>
4.	Kode <i>Shift</i>	kode_shift	<i>Char</i>	2	<i>Foreign Key</i>

4.	Sifat	Sifat	<i>Varchar</i>	15	
5.	Kode Lini	kode_lini	<i>Char</i>	3	
6.	Klasifikasi	klasifikasi	<i>Varchar</i>	15	
7.	Tanggal	Tanggal	<i>Date</i>	-	
8.	Waktu	Waktu	<i>Time</i>		
9.	Uraian OP	uraian_op	<i>Text</i>	-	
10.	Status <i>Group</i> <i>Head</i>	sts_gh	<i>Varchar</i>	15	
11.	Status <i>Dept.</i> <i>Head</i>	sts_dh	<i>Varchar</i>	15	

(Sumber: Hasil Analisis Data, 2016)

5.3.6 Deployment Diagram

Deployment diagram digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur yang menggambarkan arsitektur fisik dari system, seperti *web browser* dan sistem yang menjadi pendukung. *Deployment diagram* sistem informasi pemeliharaan mesin yang diusulkan dapat dilihat pada Gambar V.19.



Gambar V.19 *Deployment Diagram*

(Sumber: Hasil Analisis Data, 2016)

Berikut adalah penjelasan Gambar V.19 *deployment diagram* sistem informasi order pemeliharaan mesin:

1. PHP

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

2. *Database*

Database adalah program komputer yang menyediakan layanan data lainnya ke komputer atau program komputer, seperti yang ditetapkan oleh model *client-server*. Istilah ini juga merujuk kepada sebuah komputer yang didedikasikan untuk menjalankan program *server database*.

3. *Web Server*

Suatu program sekaligus mesin yang menjalankan program komputer yang mengerti protokol HTTP dan dapat menanggapi permintaan-permintaan dari *web browser*.

4. *Web Browser*

Web browser memiliki arti sebagai penjelajah *web*, seperti *Mozilla Firefox* atau *Google Chrome*. Fungsi *web browser* itu sendiri adalah untuk menampilkan dan melakukan interaksi dengan dokumen-dokumen yang disediakan oleh *web server*.

5. Sistem Operasi

Sistem operasi adalah perangkat lunak komputer atau *software* yang bertugas untuk melakukan kontrol dan manajemen perangkat keras dan juga operasi-operasi dasar sistem. Sistem operasi yang digunakan adalah *windows*.

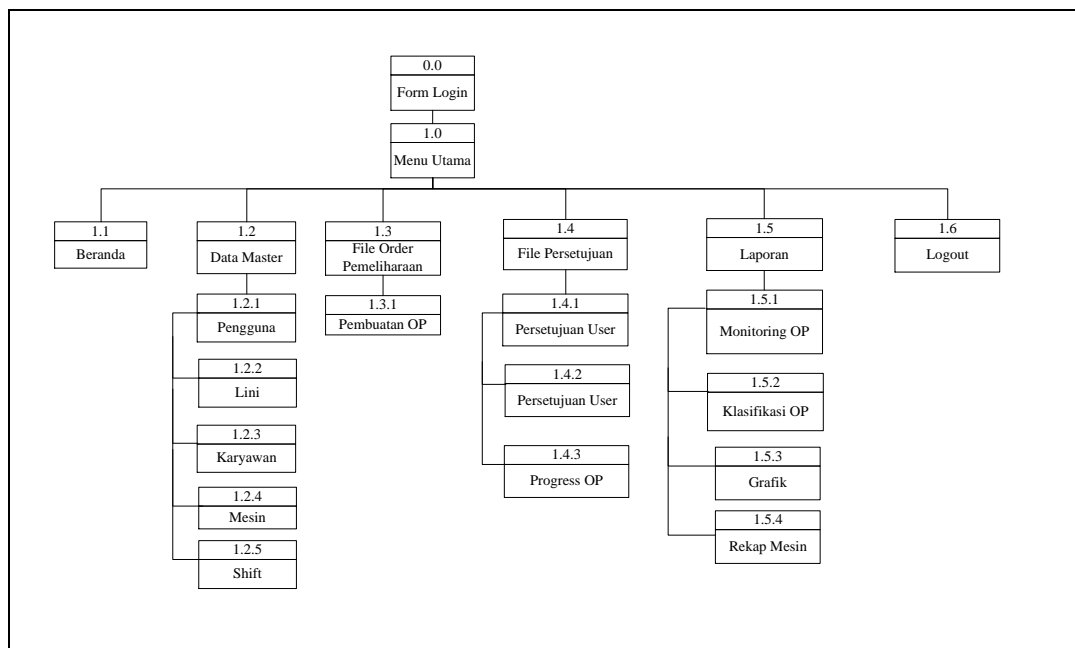
5.4 Analisis Desain Program

Tahap ini merupakan tahap kedua dalam metodologi *prototype evolutioner*, yaitu tahap membuat sebuah prototipe dari program/aplikasi. Dimulai dengan analisis desain usulan meliputi pembuatan struktur menu program, *flowchart* program, dan *interface* program sampai dengan program dapat

dijalankan. Pada tahap ini dilakukan upaya untuk memperbaiki sistem ataupun membangun dan menghasilkan sistem yang baru dengan memanfaatkan teknologi terbaru dan fasilitas yang tersedia untuk mengurangi dan mengatasi berbagai permasalahan yang telah terjadi pada sistem.

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

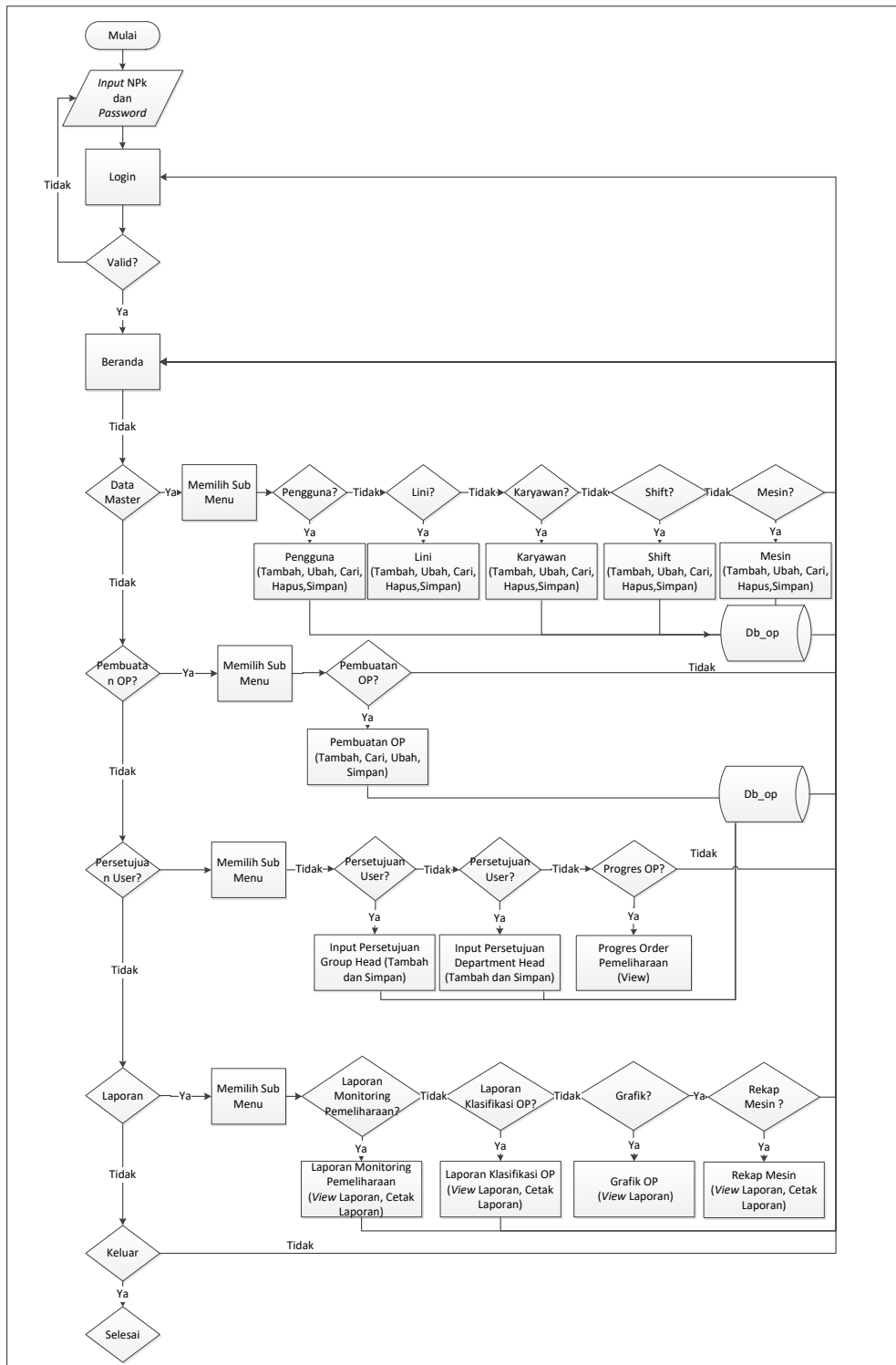
Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hierarkis. Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya. Perancangan HIPO aplikasi usulan sistem informasi pemeliharaan mesin dapat dilihat pada Gambar V.20 sebagai berikut:



Gambar V.20 HIPO Sistem Informasi Pemeliharaan Mesin
(Sumber: Hasil Analisis Data, 2016)

5.4.2 Flowchart Program

Flowchart yang digunakan untuk mendokumentasikan aplikasi sistem informasi pemeliharaan mesin ini menggunakan bagan alir logika program (*program logic flowchart*). Bagan alir ini digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. (Lihat Gambar V.21):



Gambar V.21 Program Logic Flowchart Aplikasi Sistem Informasi Pemeliharaan Mesin
(Sumber: Hasil Analisis Data, 2016)

5.4.3 Perancangan *Interface* Program

Rancangan *interface* dari program pemeliharaan mesin ini adalah sebagai berikut:

1. Tampilan *Form Login*

Form login adalah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan npk pengguna dan sandi yang benar. Rancangan *form login* dapat dilihat pada Gambar V.22.



The image shows a login form with a light gray background and a thin black border. At the top left, the text "PLEASE LOGIN" is displayed in bold black font. Below this, there are two stacked rectangular input fields. The top field is labeled "Npk" and the bottom field is labeled "Password". Below these fields is a single rectangular button with the text "Log In" centered on it. At the bottom of the form, the text "Order Pemeliharaan © 2016" is centered.

Gambar V.22 *Interface Form Login*
(Sumber: Hasil Analisis Data, 2016)

2. Tampilan *Form* Menu Utama

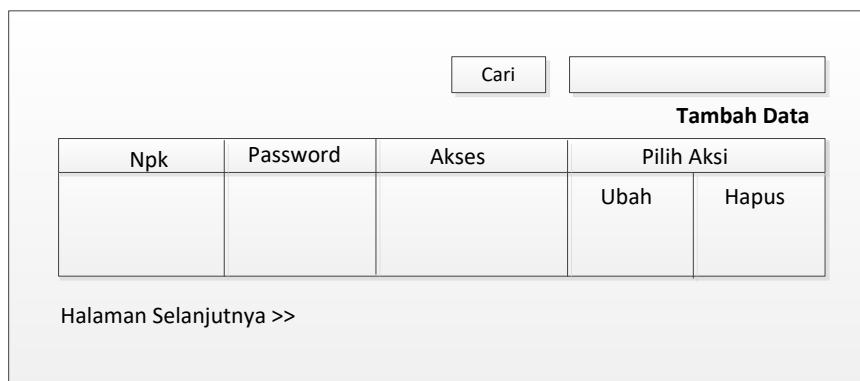
Form menu utama adalah tampilan awal pada aplikasi yang berisi enam menu dan sub menu pada menu utama. Berikut adalah *interface* menu utama pada order pemeliharaan mesin. Rancangan menu utama dapat dilihat pada Gambar V.23:



Gambar V.23 *Interface Form Menu Utama*
(Sumber: Hasil Analisis Data, 2016)

3. Tampilan *Form Data Master User*

Form data master user adalah *form* yang digunakan untuk mengelola data *user*. Rancangan *interface* dari *form data user* dapat dilihat pada Gambar V.24 dan Gambar V.25 berikut:



Gambar V.24 *Interface Form Data Master User*
(Sumber: Hasil Analisis Data, 2016)



Gambar V.25 *Interface Form Tambah Data Master User*

(Sumber: Hasil Analisis Data, 2016)

4. *Form* Data Master Lini

Form data master lini adalah form yang digunakan untuk mengelola data lini. Rancangan *interface* dari *form* data lini dan tampilan tambah data dapat dilihat pada Gambar V.26 dan Gambar V.27 berikut:

		Cari	<input type="text"/>
Tambah Data			
Kode Lini	Nama Lini	Pilih Aksi	
		Ubah	Hapus

Halaman Selanjutnya >>

Gambar V.26 *Interface Form* Data Lini

(Sumber: Hasil Analisis Data, 2016)

Kode Lini

Batal Simpan

Gambar V.27 *Interface Form* Tambah Data Lini

(Sumber: Hasil Analisis Data, 2016)

5. *Form* Data Master Karyawan

Form data master karyawan adalah form yang digunakan untuk mengelola data karyawan. Rancangan *interface* dari *form* data karyawan dapat dilihat pada Gambar V.28 dan Gambar V.29 berikut:

Cari

Tambah Data

Npk	Nama	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Alamat	Jabatan	Shift	Lini Produksi	Pilih Aksi	
									Ubah	Hapus

Halaman Selanjutnya >>

Gambar V.28 Interface Form Data Karyawan
(Sumber: Hasil Analisis Data, 2016)

Form Data Karyawan

Npk

Nama

Jenis Kelamin

Tempat Lahir

Tanggal Lahir

Alamat

Jabatan

Shift

Lini Produksi

Gambar V.29 Interface Form Tambah Karyawan Data Master
(Sumber: Hasil Analisis Data, 2016)

6. Tampilan *Form* Data Mesin

Form data mesin adalah *form* yang digunakan untuk melakukan aktivitas menyimpan, mencari, mengubah dan menghapus data mesin. Rancangan *interface* dari *form* data mesin dan tambah data mesin dapat dilihat pada Gambar V.30 dan Gambar V.31 berikut:

Cari

Tambah Data

Kode Mesin	Nama Mesin	Proses Mesin	Lokasi Mesin	Pilih Aksi	
				Ubah	Hapus

Halaman Selanjutnya >>

Gambar V.30 *Interface Form Data Mesin*
(Sumber: Hasil Analisis Data, 2016)

Form Data Mesin

Kode Mesin

Nama Mesin

Proses Mesin

Lokasi Mesin

Gambar V.31 *Interface Form Tambah Data Mesin*
(Sumber: Hasil Analisis Data, 2016)

7. Tampilan *Form Data Shift*

Form data shift adalah *form* yang digunakan untuk melakukan aktivitas menyimpan, mencari, mengubah dan menghapus data *shift*. Rancangan *interface* dari *form data shift* dan tambah data dapat dilihat pada Gambar V.32 dan Gambar V.33 berikut:

Cari

Tambah Data

Kode Shift	Nama Shift	Jam Mulai Shift	Jam Akhir Shift	Pilih Aksi	
				Ubah	Hapus

Halaman Selanjutnya >>

Gambar V.32 Interface Form Data Shift
(Sumber: Hasil Analisis Data, 2016)

Form Data Shift

Kode Shift

Nama Shift

Jam Mulai Shift

Jam Akhir Shift

Gambar V.33 Interface Form Tambah Data Shift
(Sumber: Hasil Analisis Data, 2016)

8. Tampilan *Form Data Order Pemeliharaan*

Form data order pemeliharaan adalah *form* yang digunakan untuk melakukan permintaan order pemeliharaan. Rancangan *interface* dari data order pemeliharaan dapat dilihat pada Gambar V.34 dan Gambar V.35 berikut:

Tambah Data Cari

Kode OP	Npk	Kode Mesin	Shift	Sifat	Klasifikasi	Tanggal	Waktu	Uraian	Pilih Aksi
									Direvisi

Halaman Selanjutnya >>

Gambar V.34 Interface View Pembuatan Data Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

Form Pembuatan OP	
Kode Op	<input type="text"/>
Nama Karyawan	<input type="text"/>
Mesin	<input type="text"/>
Shift	<input type="text"/>
Sifat	<input type="text" value="v"/>
Lini	<input type="text" value="v"/>
Klasifikasi	<input type="text" value="v"/>
Tanggal	<input type="text" value="v"/>
Waktu	<input type="text" value="v"/>
Uraian	<input type="text" value="v"/>
<input type="button" value="Simpan"/>	

Gambar V.35 *Interface Form* Tambah Data Order Pemeliharaan

(Sumber: Hasil Analisis Data, 2016)

9. Tampilan *Form Persetujuan User*

Form yang digunakan oleh *Group head* dan *Department Head* untuk memberikan keputusan kepada setiap order pemeliharaan untuk disetujui atau tidak disetujui. Rancangan *interface* dari pemberian persetujuan dapat dilihat pada Gambar V.36 dan Gambar V.37 dan Gambar V.38 dan Gambar V.39 berikut:

Form Status Persetujuan	
Kode Shift	<input type="text" value="v"/>
<input type="button" value="Tampil"/>	

Gambar V.36 *Interface Form* Status Persetujuan

(Sumber: Hasil Analisis Data, 2016)

Cari

Kode OP	Npk	Kode Mesin	Shift	Sifat	Klasifikasi	Tanggal	Waktu	Uraian	Status Group Head	Pilih Status
										Pilih Status

Halaman Selanjutnya >>

Gambar V.37 *Interface Form Status Persetujuan Group Head*

(Sumber: Hasil Analisis Data, 2016)

Form Pilih Status Persetujuan

Status Persetujuan Kode OP -PILIH-

DISETUJUI
TIDAK DISETUJUI

Gambar V.38 *Interface Form Pilih Status Persetujuan User*

(Sumber: Hasil Analisis Data, 2016)

Cari

Kode OP	Npk	Kode Mesin	Shift	Sifat	Klasifikasi	Tanggal	Waktu	Uraian	Status Group Head	Status Dept. Head	Pilih Status
											Pilih Status

Halaman Selanjutnya >>

Gambar V.39 *Interface Form Status Persetujuan Department Head*

(Sumber: Hasil Analisis Data, 2016)

10. Tampilan *Form* Progres OP

Form laporan adalah *form* yang digunakan untuk melihat progres dari permintaan order yang diajukan. Rancangan *interface* dari *form* progress order pemeliharaan dapat dilihat pada Gambar V.40 berikut:

Cari

Kode OP	Npk	Kode Mesin	Shift	Sifat	Klasifikasi	Tanggal	Waktu	Uraian	Status Group Head	Status Dept. Head

Halaman Selanjutnya >>

Gambar V.40 *Interface Form Progres Order Pemeliharaan*
(Sumber: Hasil Analisis Data, 2016)

11. Tampilan *Form Laporan Monitoring OP*

Form laporan adalah *form* yang digunakan untuk mencari, melihat dan mencetak laporan baik laporan *monitoring* order pemeliharaan dan laporan sesuai klasifikasi permintaan. Laporan *monitoring* berdasarkan periode dan klasifikasi pada permintaan order pemeliharaan mesin Lihat Gambar V.41, Gambar V.42 dan Gambar V.43 berikut:

Laporan Monitoring Order Pemeliharaan

Tanggal: ___/___/___ s/d ___/___/___ Cari

Kode OP	Npk	Kode Mesin	Shift	Sifat	Klasifikasi	Tanggal	Waktu	Uraian	Status Group Head	Status Dept. Head

Halaman Selanjutnya >> Cetak

Gambar V.41 *Interface Form Laporan Monitoring Pemeliharaan*
(Sumber: Hasil Analisis Data, 2016)

PILIH KIASIFIKASI

KLASIFIKASI **V**

Tampil Semua
Pembuatan
Service
Repair
Downtime

Gambar V.42 *Interface Form Klasifikasi Order Pemeliharaan*

(Sumber: Hasil Analisis Data, 2016)

Laporan Klasifikasi Order Pemeliharaan

Pilih Klasifikasi

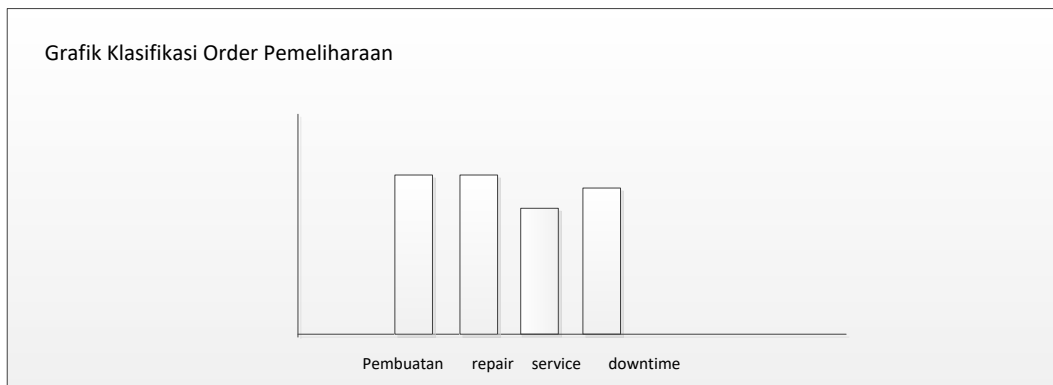
Kode OP	Npk	Kode Mesin	Shift	Sifat	Klasifikasi	Tanggal	Waktu	Uraian	Status Group Head	Status Dept. Head

Halaman Selanjutnya >> Cetak

Gambar V.43 *Interface Form* Laporan Klasifikasi Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

12. Tampilan *Form* Grafik OP

Form grafik adalah *form* yang digunakan untuk memperlihatkan klasifikasi order pemeliharaan. Grafik yang digunakan adalah jenis grafik batang menjelaskan klasifikasi order pemeliharaan dengan jumlah setiap order pemeliharaan yang diminta (lihat Gambar V.44).



Gambar V.44 *Interface Form* Laporan Klasifikasi Order Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

Gambar V.45 *Interface Form* Rekap Mesin Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

Kode OP	Npk	Kode Mesin	Shift	Sifat	Klasifikasi	Tanggal	Waktu	Uraian	Status Group Head	Status Dept. Head

Gambar V.46 *Interface Form* Laporan Rekap Mesin Pemeliharaan
(Sumber: Hasil Analisis Data, 2016)

5.5 Implementasi Sistem *Software* dan *Hardware*

Tahap ini adalah tahap pengkodean program menggunakan bahasa pemrograman PHP dengan *Macromedia Dreamweaver 8* dan *Notepad++* sebagai aplikasi editor. Setiap *interface* berisikan kode program agar program dapat dijalankan sesuai dengan fungsinya. Untuk mendukung kebutuhan implementasi sistem diperlukan suatu spesifikasi perangkat lunak (*software*) maupun perangkat keras (*hardware*). Adapun spesifikasinya sebagai berikut:

1. Analisis Kebutuhan *Software*
 - a. Sistem Operasi : *Microsoft Windows 7*
 - b. *Database Server* : *MySQL versi 5.5.26*
 - c. Bahasa Pemrograman : *PHP 5.4.7*
 - d. *Web Browser* : *Mozilla Firefox, Google Chrome*

2. Analisis Kebutuhan *Hardware*

- a. *Processor* : Minimal *Processor Pentium IV*
- b. *RAM* : Minimal *RAM 512 MB*
- c. *Harddisk* : Minimal *Harddisk 64 GB*
- d. *Mouse, Keyboard, Monitor* sebagai peralatan antar muka.
Printer sebagai media media pencetakan dokumen.

BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

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

1. Dengan adanya sistem informasi pemeliharaan mesin *front fork* maka proses pengolahan data pada order pemeliharaan mesin menjadi lebih cepat. Sebelumnya sistem permintaan order pemeliharaan ditemui kelemahan-kelemahan dimana permintaan melalui memo, dengan adanya aplikasi ini proses order pemeliharaan mesin pada bagian produksi berjalan dapat terkomputerisasi.
2. Aplikasi Sistem informasi pemeliharaan mesin *front fork* berbasis web ini dapat mengelola laporan *monitoring* order pemeliharaan mesin dan tersimpan dengan aman dan mudah di dalam basis data, sehingga mudah mencari dan mengetahui penanganan dalam permintaan pada pemeliharaan mesin selanjutnya.

6.2. Saran

Saran yang diberikan untuk pengembangan sistem informasi pemeliharaan mesin produksi *front-fork 2 W* selanjutnya adalah sebagai berikut:

1. Penerapan dalam sistem baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diharapkan adanya pengembangan lebih lanjut dari sistem informasi dalam penggunaan sistem ini kepada bagian yang terkait.
2. Aplikasi yang digunakan sebaiknya dilakukan pemeliharaan secara berkala, sehingga aplikasi dapat berjalan dengan baik.

DAFTAR PUSTAKA

- Andri, Kristanto. 2008. *Perancangan Sistem Informasi dan Aplikasinya*. Yogyakarta : Gava Media.
- Anhar. 2010. *Panduan Menguasai PHP & MySQL Secara Otodidak*. Jakarta: Agromedia Pustaka.
- Assauri, Sofjan. 2004, *Manajemen Operasi dan Produksi*. Edisi Empat, Jakarta: Lembaga Penerbit Fakultas Ekonomi UI.
- Assauri, Sofjan. 2008. *Manajemen Produksi dan Operasi*. Edisi Revisi. Jakarta: Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia.
- Dharwiyanti. S. dan Wahono. R. S. 2003, *Pengantar Unified Modelling Language (UML)*.
- Davis, Gordon B. 2002. *Sistem Informasi Manajemen*. Cetakan Kedua Belas, Jakarta: PT Pustaka Binawan Pressindo.
- Febriani, 2015. Flowchart, <http://febriani.staff.gunadarma.ac.id/Downloads/files/5616/Flowchart.pdf>. (Tanggal Akses: 20 Juni 2016)
- Hartono, Bambang. 2013. *Sistem Informasi Manajemen Berbasis Komputer*. Jakarta : Rineka Cipta.
- Hasan, M Iqbal. 2002. Pokok-pokok Materi Statistika 1 (Statistik Deskriptif). Edisi Kedua. Jakarta: PT Bumi Aksara.
- Jogiyanto. 2005. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Andi
- Kadir, A. 2009. *Membuat Aplikasi Web dengan PHP dan Database MySQL*. Yogyakarta: Andi.

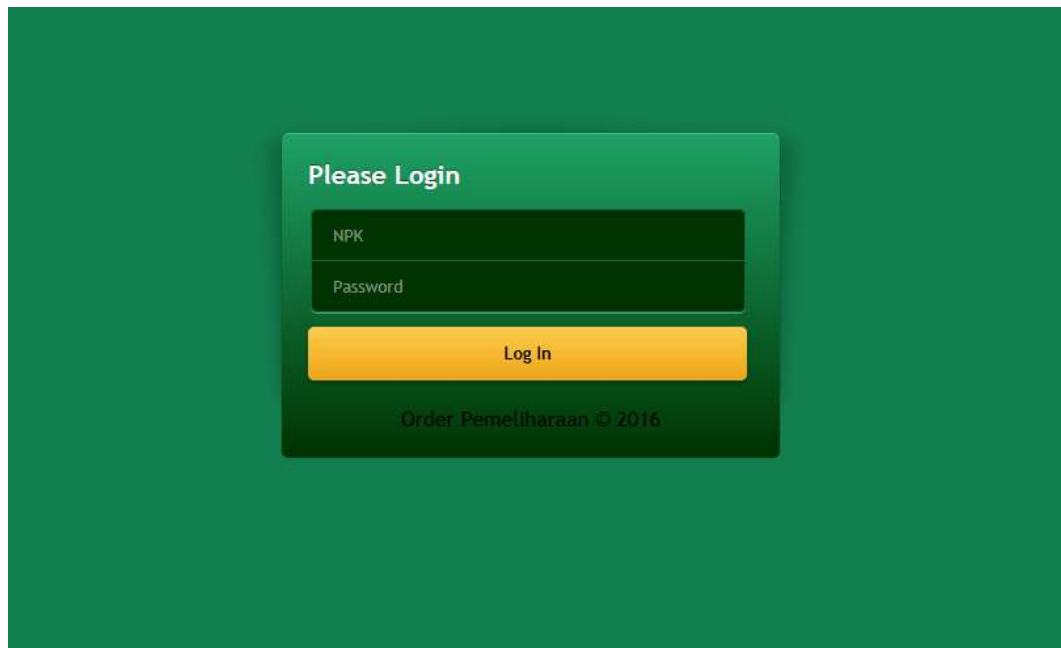
- Martono, D. F. Murad and P. 2009. *Desain dan Implementasi Data Warehouse Sebagai Pengukuran Kinerja*. vol. 2 No 3. CCIT.
- McLeod, Raymond. S, George. 2011. *Sistem Informasi Manajemen*. Jakarta: PT Indeks.
- Mulyadi. 2008. *Sistem Akuntansi*, Edisi ketiga, Cetakan Keempat. Jakarta: Salemba Empat.
- Mustakini, Jogiyanto Hartono. 2009. *Sistem Informasi Teknologi*. Yogyakarta: Andi Offset.
- Narko. 2004. *Sistem Akuntansi*. Yogyakarta: Yayasan Pustaka Nusatama.
- Nugroho, Adi. 2004. *Konsep Pengembangan Sistem Basis Data*. Bandung : Informatika.
- Nugroho, Adi. 2010. *Rekayasa Perangkat Lunak Berbasis Objek dengan Metode USDP*. Yogyakarta: Andi.
- PT Showa Indonesia Manufacturing, Pemasaran SIM. <http://www.showa.co.id/customer.php/customer-sim/>, 2012. (Tanggal Akses; 25 September 2015).
- PT Showa Indonesia Manufacturing, Profil SIM, http://www.showa.co.id/s_contents.php/profil-showa/, 2012. (Tanggal Akses; 25 November 2015).
- PT Showa Indonesia Manufacturing, Visi-MisiSIM,<http://www.showa.co.id/index.php/visi-misi/>, 2012. (Tanggal Akses; 25 September 2015).
- Purwanto. 2008. *Metodologi Penelitian Kuantitatif untuk Psikologi dan Pendidikan*. Yogyakarta : Pustaka Pelajar.
- Rama, Dasaratha V. dan Jones, Frederick L. 2008. *Sistem Informasi Akuntansi*, Buku Satu, Alih Bahasa M. Slamet Wibowo. Jakarta: Salemba Empat.

- Rosa, A.S., M. Shalahuddin. 2014. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Edisi Kedua. Bandung: Informatika.
- Soedjadi. 2000. *Kiat Pendidikan Matematika di Indonesia*. Jakarta: Direktorat Jendral Pendidikan Tinggi Departemen Pendidikan Nasional.
- Sutabri, Tata. 2012. *Konsep Dasar Informasi*. Yogyakarta: Andi.
- Sutarman. 2009. *Pengantar Teknologi Informasi*. Edisi Pertama. Jakarta: PT Bumi Aksara.
- Wahana Komputer. 2013, *Step By Step Menjadi Programmer Android*. Yogyakarta: Andi.
- Wardani, dkk. 2008. *Pengantar Pendidikan Luar Biasa*. Jakarta: Universitas Terbuka.
- Welling, L, dan Thomson, L., 2001, *PHP and MySQL Web Development*. Indianapolis United State of America: Sams Publishing.
- Winarno, Edi dan Zaki, Ali. 2012. *Programming dengan Android SDK untuk Advanced*. Jakarta : Elexmedia Komputindo.
- Yakub. 2012. *Pengantar Sistem Informasi*, Yogyakarta: Graha Ilmu.

LAMPIRAN A

TAMPILAN PROGRAM

1. Login



2. Menu Utama



LAMPIRAN B

1. Log_in.php

```
<?php
    // Mengakses database mysql
include"connect.php";
$NPK=$_REQUEST['Npk'];
$Password=$_REQUEST['password'];

$ambildata = "select user.Npk, user.password, user.akses, karyawan>Nama,
karyawan.jabatan from user LEFT JOIN karyawan ON user.Npk = karyawan.Npk
where user.Npk = ".$NPK." and user.password = ".$Password."";
$hasil = mysql_query ($ambildata);
$hasilcari = mysql_fetch_array($hasil);
if ($hasilcari){
    session_start();
    $_SESSION["id"]=$NPK.$Password;
    $_SESSION["akses"]=$hasilcari['akses'];
    $_SESSION["Npk"]=$hasilcari['Npk'];
    $_SESSION["Nama"]=$hasilcari['Nama'];
    $_SESSION["jabatan"]=$hasilcari['jabatan'];
    header("location:tamp_utama.php");
}else {
    echo"<script>alert('Login Gagal !');</script>";
    echo"<script>>window.location='login.php';</script>";
}
```

tamp_utama.php

```
<?php
include "session.php";
include "connect.php";

echo"<html>
<head>
<meta http-equiv=\"Content-Type\" content=\"text/html; charset=iso-8859-1\">
<script src=\"script/fungsi.js\" type=\"text/javascript\"></script>
<link rel=\"stylesheet\" href=\"style/style.css\" type=\"text/css\"/>
</head>
<body>
<div id=navigation>
    <ul id=nav>";
?>

<?php
if($_SESSION['akses'] == 4){
    /*sec*/
echo "
        <li><a href=halamanutama.php
target=main_frame>BERANDA</a></li>
        <li><a href=#>FILE ORDER PEMELIHARAAN</a>
        <ul>
            <li><a href=showop.php
target=main_frame>Pembuatan OP</a></li>
        </ul>
        </li>
        <li><a href=#>FILE PERSETUJUAN</a>
```

```

        <ul>
            <li><a href=showprogres.php
target=main_frame>Progress OP</a></li>
        </ul>
    </li>
    <li><a href=logout.php>LOGOUT</a></li>";
}
?>

<?php
if($_SESSION['akses'] == 0){
echo "
        <li><a href=halamanutama.php
target=main_frame>BERANDA</a></li>
        <li><a href=#>FILE DATA MASTER</a>
            <ul>
                <li><a href=showpengguna.php
target=main_frame>Pengguna</a></li>
                <li><a href=showlini.php
target=main_frame>Lini</a></li>
                <li><a href=showkaryawan.php
target=main_frame>Karyawan</a></li>
                <li><a href=showmesin.php
target=main_frame>Mesin</a></li>
                <li><a href=showshift.php
target=main_frame>Shift</a></li>
            </ul>
        </li>
        <li><a href=#>FILE ORDER PEMELIHARAAN</a>
            <ul>

```

```

        <li><a href=showop.php
target=main_frame>Pembuatan OP</a></li>
    </ul>
</li>
<li><a href=#>FILE PERSETUJUAN</a>
    <ul>
        <li><a href=masterpersetujuan.php
target=main_frame>Persetujuan User</a></li>
        <li><a href=masterdp.php
target=main_frame>Persetujuan User</a></li>
        <li><a href=showprogres.php
target=main_frame>Progress OP</a></li>
    </ul>
</li>
<li><a href=#>LAPORAN</a>
    <ul>
        <li><a href=viewlaporan.php target=main_frame>Monitoring
OP</a></li>
        <li><a href=masterklasifikasi.php
target=main_frame>Klasifikasi OP</a></li>
        <li><a href=grafik.php target=main_frame>Grafik
Klasifikasi</a></li>
    </ul>
</li>
<li><a href=logout.php>LOGOUT</a></li>";
}
?>

<?php
if($_SESSION['akses'] == 3){ /*mtc*/
echo "

```

```

        <li><a href=halamanutama.php
target=main_frame>BERANDA</a></li>
        <li><a href=#>LAPORAN</a>
            <ul>
                <li><a href=viewlaporan.php target=main_frame>Monitoring
OP</a></li>
                <li><a href=masterklasifikasi.php
target=main_frame>Klasifikasi OP</a></li>
                <li><a href=grafik.php target=main_frame>Grafik
Klasifikasi</a></li>
            </ul>
        </li>
        <li><a href=logout.php>LOGOUT</a></li>";
    }
?>

<?php
if($_SESSION['akses'] == 2 ){                                /*gh*/
echo "
        <li><a href=halamanutama.php
target=main_frame>BERANDA</a></li>
        <li><a href=#>FILE PERSETUJUAN</a>
            <ul>
                <li><a href=masterpersetujuan.php
target=main_frame>Persetujuan User</a></li>
                <li><a href=showprogres.php
target=main_frame>Progress OP</a></li>
            </ul>
        </li>
        <li><a href=#></a>

```

```
        <ul>
        </ul>
    </li>
    <li><a href=logout.php>LOGOUT</a></li>";
}
?>
```

```
<?php
if($_SESSION['akses'] == 1){          /*dh*/
echo "
        <li><a href=halamanutama.php
target=main_frame>BERANDA</a></li>
        <li><a href=#>FILE DATA MASTER</a>
        <ul>
            <li><a href=showpengguna.php
target=main_frame>Pengguna</a></li>
            <li><a href=showlini.php
target=main_frame>Lini</a></li>
            <li><a href=showkaryawan.php
target=main_frame>Karyawan</a></li>
            <li><a href=showmesin.php
target=main_frame>Mesin</a></li>
            <li><a href=showshift.php
target=main_frame>Shift</a></li>
        </ul>
        </li>
        <li><a href=#>FILE PERSETUJUAN</a>
        <ul>
            <li><a href=masterdp.php
target=main_frame>Persetujuan User</a></li>
```

```

                <li><a href=showprogres.php
target=main_frame>Progress OP</a></li>
            </ul>
        </li>
        <li><a href=logout.php>LOGOUT</a></li>";
    }
?>

<?php
echo "
<td width=77% valign=top height=100%><iframe name=main_frame
src=halamanutama.php height=85% width=100% frameborder=0</iframe></td>
<table border=0 cellpadding=2 cellspacing=2 width=816 align=center
class=footer>
    <tr>
        <td colspan=2><div align=center>Order Pemeliharaan © 2016</div>
    </tr></table>
</body>
</html>";
?>

```

mastermesin.php

```

<?php
include "session.php";
include "connect.php";
error_reporting (E_ALL ^ (E_NOTICE | E_WARNING));
$act=$_REQUEST['act'];
if(!isset($_REQUEST['kode_mesin']))
{
    $kode_mesin="0";
}
else{

```

```

        $kode_mesin=$_REQUEST['kode_mesin'];
    }

        $query = "select * from mesin where kode_mesin =
".$kode_mesin;
        $rslt = mysql_query($query);
        $tb_row = mysql_fetch_row($rslt);
        $nama_mesin=$tb_row[1];
        $proses_mesin=$tb_row[2];
        $lokasi_mesin=$tb_row[3];
        if($kode_mesin==0)
    {
        $kode_mesin="";
    }

echo "<html><title>Form Input Mesin</title>
<script src=\"script/fungsi.js\" type=\"text/javascript\"></script>
<link rel=\"stylesheet\" href=\"style/style.css\" type=\"text/css\"/>
<p>&nbsp;</p>
<p>&nbsp;</p>
<head></head>
<body>
<form name=\"form_input\" action=\"fungsimesin.php\" onSubmit=\"return
form_input_val()\">
<input type=hidden name= act value=\".$act.\">
<table cellpadding=2 cellspacing=0 width=500 class=hovetable align=center>
<br></br>
    <tr class=start1>
        <td colspan=2 height=26>Form Data Mesin</td>
    </tr>
    <tr class=row>
        <td width=30%>Kode Mesin</td>

```

```

        <td><input type=text name=\"kode_mesin\"
value=\"$kode_mesin\"></td>
    </tr>
    <tr class=row>
        <td>Nama Mesin</td>
        <td><input type=text name=\"nama_mesin\"
value=\"$nama_mesin\"></td>
    </tr>
    <tr class=row>
        <td>Proses Mesin</td>
        <td><input type=text name=\"proses_mesin\"
value=\"$proses_mesin\"></td>
    </tr>
    <tr class=row>
        <td>Lokasi Mesin</td>
        <td><input type=text name=\"lokasi_mesin\"
value=\"$lokasi_mesin\"></td>
    </tr>
    <tr class=row>
        <td colspan=2 align=right>
            <input name=simpan type=submit
onClick=\"simpaninputmesin()\" value=\"Simpan\"></td>
            <input name=batal type=reset value=\"Batal\"></td></td>
    </tr>
</table>
</form>
</body>
</html>";
?>

```

fungsimesin.php

```

<?php
//koneksi database
include "connect.php";
$act=$_REQUEST['act']; //1=input, 2=edit, 3=delete
$kode_mesin=$_REQUEST['kode_mesin'];
$nama_mesin=$_REQUEST['nama_mesin'];
$proses_mesin=$_REQUEST['proses_mesin'];
$lokasi_mesin=$_REQUEST['lokasi_mesin'];
//input data
if($act==1){
    $query="insert into mesin
values(\"$kode_mesin\", \"$nama_mesin\", \"$proses_mesin\", \"$lokasi_mesin\");
    mysql_query($query);
//edit data
}else if ($act==2){
    $query="update mesin set
kode_mesin=\"$kode_mesin\", nama_mesin=\"$nama_mesin\", proses_mesin=\"$p
roses_mesin\", lokasi_mesin=\"$lokasi_mesin\" where
kode_mesin=\"$kode_mesin\"";
    mysql_query($query);
//delete data
}else if($act==3){
    $query="delete from mesin where kode_mesin=\"$kode_mesin\"";
    mysql_query($query); }
header('location:showmesin.php?kode_mesin=');
?>

```

Source Code Koneksi

```

<?php
$koneksi = mysql_connect('localhost','root','');

```

```
mysql_select_db('db_ip',$koneksi);
```

```
?>
```

Source Code Login

```
<?php
```

```
session_start();
```

```
if(isset($_SESSION['nrp'])){
```

```
    header('location:index.php');
```

```
}
```

```
?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

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

```
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
    <meta name="description" content="">
```

```
    <meta name="author" content="">
```

```
    <title>Aplikasi Minimarket | Login</title>
```

```
    <!-- Bootstrap Core CSS -->
```

```
    <link href="assets/bower_components/bootstrap/dist/css/bootstrap.min.css"
rel="stylesheet">
```

```
    <!-- Custom CSS -->
```

```
    <link href="assets/dist/css/sb-admin-2.css" rel="stylesheet">
```

```
    <!-- Custom Fonts -->
```

```
    <link href="assets/bower_components/font-awesome/css/font-awesome.min.css"
rel="stylesheet" type="text/css">
```

```

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

</head>

<body>

    <div class="container">
        <div class="row">
            <div class="col-md-4 col-md-offset-4">
                <div class="login-panel panel panel-success">
                    <div class="panel-heading">
                        <h3 class="panel-title">Please Sign In</h3>
                    </div>
                    <div class="panel-body">

                        <?php if(isset($_GET['logout'])): ?>
                            <div class="alert alert-success"
role="alert">
                                <?php echo 'Selamat Anda Berhasil Logout'; ?>
                            </div>
                                <?php elseif(isset($_GET['failed'])): ?>
                                    <div class="alert alert-danger" role="alert">
                                        <?php echo 'NRP dan Password anda salah'; ?>
                                    </div>
                                <?php endif ?>
                            <form role="form" action="auth/auth.php" method="POST">
                                <fieldset>

```

```
        <div class="form-group">
            <input class="form-control" placeholder="NRP" name="nrp"
type="text" autofocus>
        </div>
        <div class="form-group">
            <input class="form-control" placeholder="Password"
name="password" type="password" value="">
        </div>

        <!-- Change this to a button or input when using this as a form -->
        <input type="submit" class="btn btn-success btn btn-block"
value="Login" name="submit">
        </fieldset>
    </form>
</div>
</div>
</div>
</div>
</div>
</div>

<!-- jQuery -->
<script src="assets/bower_components/jquery/dist/jquery.min.js"></script>

<!-- Bootstrap Core JavaScript -->
<script src="assets/bower_components/bootstrap/dist/js/bootstrap.min.js"></script>

<!-- Custom Theme JavaScript -->
<script src="assets/dist/js/sb-admin-2.js"></script>

</body>

</html>
```

Source Code Menu Utama

```
<div class="navbar-default sidebar" role="navigation">
  <div class="sidebar-nav navbar-collapse">
    <ul class="nav" id="side-menu">
      <li class="sidebar-search">
        <div class="input-group custom-search-form">
          <input type="text" class="form-control" placeholder="Search...">
          <span class="input-group-btn">
            <button class="btn btn-default" type="button">
              <i class="fa fa-search"></i>
            </button>
          </span>
        </div>
      <!-- /input-group -->
    </li>
    <li>
      <?php
        if ($_SESSION['akses']==4){
          ?>
          <li>
            <a href="index.php"><i class="fa fa-dashboard fa-fw"></i> Menu
            Utama</a>
          </li>
          <li>
            <a href="#"><i class="fa fa-sitemap fa-fw"></i> Idea Proposal<span
            class="fa arrow"></span></a>
            <ul class="nav nav-second-level">
              <li>
                <a href="ip.php">Idea Proposal </a>
              </li>
              <li>
                <a href="view_point_user.php">Point Idea Proposal</a>
              </li>
            </ul>
          </li>
        }
      </?php>
    </li>
  </ul>
</div>
</div>
```

```

        </ul>
        <!-- /.nav-second-level -->
    </li>
    <?php }
    if($_SESSION['akses']=='2'){
    ?>
    <li>
        <a href="index.php"><i class="fa fa-dashboard fa-fw"></i> Menu
Utama</a>
    </li>
    <li>
        <a href="#"><i class="fa fa-sitemap fa-fw"></i> Idea Proposal<span
class="fa arrow"></span></a>
        <ul class="nav nav-second-level">
            <li>
                <a href="view_ip.php">Idea Proposal </a>
            </li>
            <li>
                <a href="view_point.php">Point Idea Proposal</a>
            </li>
        </ul>
    </li>
    </ul>
    <!-- /.nav-second-level -->
    </li>

    <?php }

    if($_SESSION['akses']=='3'){
    ?>
    <li>

```

```
<a href="index.php"><i class="fa fa-dashboard fa-fw"></i> Menu
Utama</a>
```

```
</li>
```

```
<li>
```

```
<a href="#"><i class="fa fa-sitemap fa-fw"></i> Data Master<span
class="fa arrow"></span></a>
```

```
<ul class="nav nav-second-level">
```

```
<li>
```

```
<a href="jabatan.php">Menu Jabatan </a>
```

```
</li>
```

```
<li>
```

```
<a href="lini_produksi.php">Menu Lini Produksi </a>
```

```
</li>
```

```
<li>
```

```
<a href="shift_kerja.php">Menu Shift Kerja </a>
```

```
</li>
```

```
<li>
```

```
<a href="karyawan.php">Menu Data Karyawan </a>
```

```
</li>
```

```
</ul>
```

```
<li>
```

```
<a href="user.php"><i class="fa fa-sitemap fa-fw"></i> Menu Data
User</a>
```

```
</li>
```

```
<li>
```

```
<a href="#"><i class="fa fa-sitemap fa-fw"></i> Idea Proposal<span
class="fa arrow"></span></a>
```

```
<ul class="nav nav-second-level">
```

```
<li>
```

```
<a href="view_ip_qcl.php">Idea Proposal </a>
```

```
</li>
                                <li>
                                <a href="view_point_qcl.php">Point Idea Proposal</a>
                                </li>
                                <li>
                                <a href="#">Laporan Idea Proposal <span class="fa
                                arrow"></span></a>
                                <ul class="nav nav-third-level">
                                <li>
                                <a href="view_laporan.php">Laporan IP diterima</a>
                                </li>
                                <li>
                                <a href="view_laporan_bulanan.php">Laporan IP
                                Bulanan</a>
                                </li>
                                </ul>
                                <?php } ?>
                                </div>
                                <!-- /.sidebar-collapse -->
                                </div>
                                <!-- /.navbar-static-side -->
                                </nav>
```