

**RANCANG BANGUN SISTEM INFORMASI PENGENDALIAN
KUALITAS *ASSEMBLY ENGINE 2W* BERBASIS WEB
MENGUNAKAN PHP 5.5 DAN MYSQL 5.6
PT SUZUKI INDOMOBIL MOTOR**

TUGAS AKHIR

Diajukan Untuk Memenuhi Sebagian Syarat Penyelesaian Program Diploma
Empat (D-IV) Sistem Informasi pada Sekolah Tinggi Manajemen Industri

OLEH

JOHN YANSIN SIHOMBING

(1311020)



**SEKOLAH TINGGI MANAJEMEN INDUSTRI
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2016**

**SEKOLAH TINGGI MANAJEMEN INDUSTRI
KEMENTERIAN PERINDUSTRIAN R.I.**

LEMBAR PENGESAHAN

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem
Informasi Sekolah Tinggi Manajemen Industri Kementerian Perindustrian R.I
Pada Hari Selasa 19 April 2016

Menyetujui

Dosen Pembimbing

Dosen Penguji

Dr. Ridzky Kramanandita, S.Kom, MT

NIP. 197403022002121001

Dedy Trisanto, S.Kom, MMSI

NIP. 197805052005021002

Dosen Penguji

Dosen Penguji

Drs. Jacob Saragih, MM

NIP. 195404281986031002

Ulil Hamida, ST, MT

NIP. 198103272005022001

**SEKOLAH TINGGI MANAJEMEN INDUSTRI
KEMENTERIAN PERINDUSTRIAN R.I.**

TANDA PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : **RANCANG BANGUN SISTEM INFORMASI
PENGENDALIAN KUALITAS *ASSEMBLY*
ENGINE 2W MENGGUNAKAN PHP 5.5 DAN
MYSQL 5.6 PT SUZUKI INDOMOBIL MOTOR**

Disusun Oleh :

Nama : John Yansin Sihombing

Nim : 1311020

Program Studi : Sistem Informasi

Tanggal Seminar : 07 Maret 2016

Tanggal Sidang : 19 April 2016

Tanggal Lulus : 19 April 2016

Menyetujui

Jakarta, Mei 2016

Dosen Pembimbing

Dr. Ridzky Kramanandita, S.Kom, MT

NIP. 197403022002121001

**SEKOLAH TINGGI MANAJEMEN INDUSTRI
KEMENTERIAN PERINDUSTRIAN R.I.**

TANDA PERSETUJUAN ASISTEN DOSEN PEMBIMBING

Judul Tugas Akhir : **RANCANG BANGUN SISTEM INFORMASI
PENGENDALIAN KUALITAS *ASSEMBLY*
ENGINE 2W MENGGUNAKAN PHP 5.5 DAN
MYSQL 5.6 PT SUZUKI INDOMOBIL MOTOR**

Disusun Oleh :

Nama : John Yansin Sihombing

Nim : 1311020

Program Studi : Sistem Informasi

Tanggal Seminar : 07 Maret 2016

Tanggal Sidang : 19 April 2016

Tanggal Lulus : 19 April 2016

Menyetujui

Jakarta, Mei 2016

Assisten Dosen Pembimbing

Noveriza Yuliasari, SSi, MT

NIP. 197811212009012003

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : John Yansin Sihombing

NIM : 1311020

Judul TA : Rancang Bangun Sistem Informasi Pengendalian Kualitas
Assembly Engine 2W Berbasis Web Menggunakan PHP 5.5 dan MySQL
5.6 PT Suzuki Indomobil Motor

Pembimbing : Dr. Ridzky Kramanandita, S.Kom, MT

Asisten Pembimbing : Noveriza Yuliasari, SSi, MT

Tanggal	BAB	Keterangan	Paraf
03/09/2015	BAB I	Pengajuan BAB I	
08/09/2015	BAB I dan II	Perbaikan BAB I, Pengajuan BAB II	
10/09/2015	BAB I,II dan III	Perbaikan BAB I, II, Pengajuan BAB III	
23/09/2015	BAB III dan IV	Perbaikan BAB II, pengajuan BAB IV	
07/10/2015	BAB IV dan V	Perbaikan BAB IV, pengajuan BAB V	
20/10/2015	BAB V	Perbaikan BAB V Activity Diagram	
23/10/2015	BAB V	Perbaikan BAB V Sequence Diagram	
02/11/2015	BAB V	Perbaikan BAB V Class Diagram	
04/11/2015	BAB V, VI	Perbaikan BAB V Tampilan Program, Pengajuan BAB VI	
27/11/2015	BAB VI	Revisi Bab VI	
02/12/2015	BAB I-VI	Revisi Bab VI	
04/12/2015	-		
	Abstrak dan	Perbaikan Abstrak	
	Daftar	Perbaikan Daftar Pustaka	
	Pustaka	Demo Program	

Mengetahui,
Ka Prodi
Sistem Informasi

Asisten Pembimbing

Dedy Trisanto, S.Kom, MMSI
NIP : 197805052005021002

Noveriza Yuliasari, SSi, MT
NIP : 197811212009012003

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : John Yansin Sihombing

NIM : 1311020

Judul TA : Rancang Bangun Sistem Informasi Pengendalian Kualitas
Assembly Engine 2W Berbasis Web Menggunakan PHP 5.5 dan MySQL
5.6 PT Suzuki Indomobil Motor

Pembimbing : Dr. Ridzky Kramanandita, S.Kom, MT

Asisten Pembimbing : Noveriza Yuliasari, SSI, MT

Tanggal	BAB	Keterangan	Paraf
27/08/2015	Proposal	Acc Judul	
03/09/2015	BAB I	Perbaiki BAB I	
10/09/2015	BAB I,II, dan III	Perbaiki BAB I	
17/09/2015	BAB I,II, dan III	Perbaiki BAB II	
23/09/2015	BAB IV	Perbaiki BAB IV	
07/10/2015	BAB IV	Perbaiki BAB IV	
13/10/2015	BAB IV	Selesaikan BAB V	
22/10/2015	BAB V	Perbaiki BAB V	
05/11/2015	BAB I, II, III, IV, V VI	Perbaiki BAB II, V	
11/11/2015	BAB I, II, III, IV, V VI	Perbaiki BAB V	
20/11/2015	BAB I, II, III, IV, V VI	Perbaiki BAB V	
24/11/2015	BAB I, II, III, IV, V VI dan program	Demo Program	

Mengetahui,
Ka Prodi
Sistem Informasi

Dosen Pembimbing

Dedy Trisanto, S.Kom, MMSI
NIP : 197805052005021002

Dr. Ridzky Kramanandita, S.Kom, MT
NIP : 197403022002121001

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : John Yansin Sihombing

Nim : 1311020

Berstatus sebagai mahasiswa Program Studi Sistem Informasi di Sekolah Tinggi Manajemen Industri Kementerian Perindustrian RI. Dengan ini menyatakan bahwa hasil karya Tugas Akhir yang saya buat dengan judul:

“Rancang Bangun Sistem Informasi Pengendalian Kualitas *Assembly Engine* 2W Berbasis Web Menggunakan PHP 5.5 dan MYSQL 5.6 PT Suzuki Indomobil Motor”.

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

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

Jakarta, 10 Desember 2015

John Yansin Sihombing

ABSTRAK

Kualitas merupakan salah satu faktor yang mempengaruhi konsumen untuk menentukan produk dan jasa yang digunakan. Konsumen biasanya memilih produk dan jasa yang berkualitas yaitu yang sesuai dengan keinginan dan kebutuhan. PT Suzuki Indomobil Motor merupakan perusahaan industri yang bergerak di bidang otomotif. Salah satu departemen yang menangani proses produksi adalah Departemen Produksi, sedangkan Seksi *Assembly Engine 2W* yang berfungsi merakit setiap komponen *engine* sepeda motor. Sistem informasi pengendalian kualitas yang berjalan masih dilakukan secara manual. Proses pengolahan data kerusakan *engine* masih ditulis dalam formulir kerusakan *engine*. Hal ini menyebabkan pengolahan data kerusakan *engine* membutuhkan waktu yang lama. Proses pembuatan laporan juga masih menggunakan *Microsoft Excel*. Hal ini menyebabkan manajer sering kesulitan dalam membaca hasil laporan kerusakan *engine*. Selain itu, belum adanya program aplikasi yang terintegrasi dengan database sebagai penyimpanan data mengakibatkan terjadinya kehilangan data kerusakan *engine*. Untuk mengatasi permasalahan tersebut maka perlu dibuat sebuah sistem informasi pengendalian kualitas yang dapat memberikan informasi dengan lebih efisien dan aktual. Metodologi pengembangan sistem informasi yang digunakan pada penelitian ini adalah *evolutionary prototype*. Analisis dan perancangan sistem informasi pengendalian kualitas menggunakan analisis perancangan berorientasi objek dan menggunakan *tools* pemodelan *Unified Modelling Language* (UML). Sistem informasi pengendalian kualitas ini dibangun dengan menggunakan PHP 5.5 sebagai bahasa pemrograman dan MySQL 5.6 sebagai perangkat lunak manajemen basis datanya. Sistem informasi pengendalian kualitas ini memberikan informasi dengan lebih efisien dan aktual karena pencatatan data kerusakan sudah terkomputerisasi sehingga laporan kerusakan dapat langsung diolah secara otomatis.

Kata kunci: Sistem Informasi, *Assembly Engine 2W*, Pengendalian Kualitas, *Evolutionary Prototype*, UML, PHP 5.5, MySQL 5.6.

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas limpahan nikmat, rahmat serta karunia yang senantiasa diberikan sehingga penulis dapat menyelesaikan Laporan Tugas Akhir dengan judul “RANCANG BANGUN SISTEM INFORMASI PENGENDALIAN KUALITAS *ASSEMBLY ENGINE* 2W BERBASIS WEB MENGGUNAKAN PHP 5.5 DAN MYSQL 5.6 PT. SUZUKI INDOMOBIL MOTOR”. Laporan Tugas Akhir ini diajukan untuk memenuhi salah satu syarat penyelesaian Program Studi D-IV Sistem Informasi pada Sekolah Tinggi Manajemen Industri.

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

1. Kedua Orang tua dan keluarga tercinta yang telah memberikan doa, dukungan, pengorbanan, semangat dan kasih sayang hingga saat ini.
2. Dr. Mustofa, ST, MT, selaku ketua Sekolah Tinggi Manajemen Industri.
3. Bapak Drs. Jacob Saragih, MM, selaku Ketua Program Studi Sistem Informasi.
4. Bapak Dr. Ridzky Kramanandita, S.Kom, MT selaku dosen pembimbing yang memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
5. Ibu Noveriza Yuliasari, SSi, MT, selaku asisten dosen Pembimbing yang telah membantu penyelesaian Laporan Tugas Akhir.
6. Bapak Agus, selaku Divisi HRD yang telah mengorganisir jalannya Kerja Lapangan di PT Suzuki Indomobil Motor.
7. Bapak Yasfer, selaku pembimbing di PT Suzuki Indomobil Motor yang telah membimbing dalam melaksanakan Kerja Lapangan.

8. Seluruh dosen Sekolah Tinggi Manajemen Industri yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
9. Kepada Harry Christianto yang sudah membantu dalam penyelesaian Tugas Akhir ini.
10. Rekan-rekan mahasiswa Sekolah Tinggi Manajemen Industri Program Studi Sistem Informasi Angkatan 2011 atas kebersamaan selama ini.

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

Jakarta, Desember 2015

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING	
LEMBAR PENGESAHAN	
LEMBAR BIMBINGAN TUGAS AKHIR	
LEMBAR BIMBINGAN TUGAS AKHIR	
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	4
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Pengertian Rancang Bangun	6
2.2 Sistem Informasi	6
2.2.1 Pengertian Sistem	7
2.2.2 Pengertian Informasi	7
2.2.3 Pengertian Sistem Informasi	8
2.2.4 Komponen Sistem Informasi	9

2.3	Rancang Bangun Sistem Informasi	10
2.4	Pengendalian.....	11
2.4.1	Proses Pengendalian	11
2.4.2	Cara Pengendalian	12
2.5	Kualitas	12
2.5.1	Dimensi Kualitas	13
2.5.2	Pengendalian Kualitas	13
2.5.3	Tujuan Pengendalian Kualitas.....	15
2.6	Instrumen Dasar Peningkatan Kualitas.....	15
2.6.1	Lembar Pengumpulan Data (<i>Check Sheet</i>).....	15
2.6.2	<i>Pareto Chart</i>	16
2.6.3	<i>Histogram</i>	17
2.7	Pengembangan Sistem.....	18
2.7.1	<i>Prototyping</i>	19
2.7.2	<i>Evolutionary Prototype</i>	20
2.8	<i>Flowchart</i>	22
2.9	<i>Unified Modeling Language (UML)</i>	23
2.9.1	Tujuan UML.....	24
2.9.2	<i>Diagram UML</i>	24
2.10	<i>Multiplicity</i>	31
2.11	Kamus Data	32
2.12	<i>Hierarchy plus Input-Proses-Output (HIPO)</i>	32
2.13	PHP	34
2.14	MySQL	35
2.15	XAMPP.....	36
BAB III	METODOLOGI PENELITIAN	38
3.1	Metodologi Penelitian.....	38
3.2	Jenis dan Sumber Data.....	38
3.3	Metode Pengumpulan Data.....	39
3.4	Metode Pengembangan Sistem.....	40
3.5	Kerangka Penelitian.....	40

BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA	44
4.1	Sekilas Perusahaan.....	44
4.1.1	Visi dan Misi Perusahaan	44
4.1.2	Program Perusahaan	45
4.1.3	Struktur Organisasi Perusahaan.....	46
4.1.4	Jaringan Pemasaran	48
4.2	Proses <i>Assembly Engine</i>	50
4.3	Sistem Informasi yang Berjalan.....	51
4.3.1	Arus Dokumen.....	52
4.3.2	Arus Proses	55
4.3.3	Diagram <i>Use Case</i> Sistem Informasi yang Berjalan ...	56
BAB V	ANALISIS DAN PEMBAHASAN	63
5.1	Analisis Sistem Informasi Berjalan	63
5.2	Analisis Kebutuhan Sistem.....	63
5.3	Analisis Kebutuhan Rinci Sistem	64
5.4	Perancangan <i>Flowmap</i> Sistem Usulan.....	66
5.5	Pemodelan Sistem Usulan	68
5.5.1	<i>Use Case Diagram</i>	69
5.5.2	<i>Activity Diagram</i>	78
5.5.3	<i>Sequence Diagram</i>	87
5.6	Pemodelan Data	102
5.6.1	<i>Class Diagram</i>	102
5.6.2	Kamus Data	103
5.6.3	<i>Deployment Diagram</i>	108
5.7	Perancangan Aplikasi	109
5.7.1	HIPO (<i>Hierarchy plus Input-Process-Output</i>)	109
5.7.2	<i>Flowchart</i> Aplikasi Usulan.....	111
5.7.3	Perancangan <i>Interface</i> Aplikasi Usulan	112
5.8	Implementasi Sistem.....	131
BAB VI	KESIMPULAN DAN SARAN	132

	Halaman
6.1 Kesimpulan	132
6.2 Saran	132
DAFTAR PUSTAKA	133
LAMPIRAN.....	L-1

DAFTAR GAMBAR

	Halaman
Gambar II.1 Transformasi Data Menjadi Informasi	8
Gambar II.2 Komponen Sistem Informasi	9
Gambar II.3 Siklus P-D-A-C	14
Gambar II.4 Contoh <i>Check Sheet</i>	16
Gambar II.5 Contoh <i>Pareto Chart</i>	17
Gambar II.6 Contoh <i>Histogram</i>	17
Gambar II.7 Langkah Pembuatan <i>Prototype Evolutioner</i>	21
Gambar II.8 <i>Visual Table Of Contents</i>	33
Gambar III.1 Kerangka Penelitian	43
Gambar IV.1 Struktur Organisasi PT Suzuki Indomobil Motor	46
Gambar IV.2 Struktur Organisasi PT Suzuki Indomobil Motor Cakung.....	47
Gambar IV.3 Pemasaran Sepeda Motor PT Suzuki Indomobil Motor di Indonesia	49
Gambar IV.4 Pemasaran Sepeda Motor PT Suzuki Indomobil Motor di Dunia	50
Gambar IV.5 Hasil <i>Assy Counter Line</i> PT Suzuki Indomobil Motor	50
Gambar IV.6 Hasil <i>Assy Main Line 1</i> PT Suzuki Indomobil Motor	51
Gambar IV.7 Hasil <i>Assy Main Line 2</i> PT Suzuki Indomobil Motor	51
Gambar IV.8 <i>Form NG</i>	52
Gambar IV.9 <i>Form Laporan Kerusakan</i>	53
Gambar IV.10 Laporan Kerusakan <i>Before Process</i>	53
Gambar IV.11 Laporan Kerusakan <i>In Process</i>	54
Gambar IV.12 Laporan Evaluasi.....	54
Gambar IV.13 <i>Flowmap</i> Sistem yang Berjalan	56
Gambar IV.14 <i>Use Case Diagram</i> Sistem Informasi Pengendalian Kualitas Berjalan	57
Gambar V.1 <i>Flowmap</i> Sistem Usulan	68

Gambar V.2	<i>Use Case Diagram Usulan</i>	69
Gambar V.3	<i>Activity Diagram Proses Login</i>	78
Gambar V.4	<i>Activity Diagram Proses Mengelola Data User</i>	79
Gambar V.5	<i>Activity Diagram Proses Mengelola Data Master</i>	80
Gambar V.6	<i>Activity Diagram Proses Input Laporan Kerusakan</i>	81
Gambar V.7	<i>Activity Diagram Proses Melihat Laporan Kerusakan</i>	82
Gambar V.8	<i>Activity Diagram Proses Mengolah Laporan Kerusakan</i>	83
Gambar V.9	<i>Activity Diagram Proses Mencetak Laporan Kerusakan</i>	84
Gambar V.10	<i>Activity Diagram Proses Input Hasil Evaluasi</i>	85
Gambar V.11	<i>Activity Diagram Proses Melihat Laporan Evaluasi</i>	86
Gambar V.12	<i>Activity Diagram Proses Mencetak Laporan Evaluasi</i>	87
Gambar V.13	<i>Sequence Diagram Proses Login</i>	88
Gambar V.14	<i>Sequence Diagram Mengelola User</i>	89
Gambar V.15	<i>Sequence Diagram Mengelola Data Master Komponen</i>	90
Gambar V.16	<i>Sequence Diagram Mengelola Data Master Tipe Engine</i>	91
Gambar V.17	<i>Sequence Diagram Mengelola Data Master Jenis Kerusakan</i>	92
Gambar V.18	<i>Sequence Diagram Mengelola Data Master Lini Produksi</i>	93
Gambar V.19	<i>Sequence Diagram Data Master Asal Masalah Komponen</i>	94
Gambar V.20	<i>Sequence Diagram Mengelola Data Master Petugas</i>	95
Gambar V.21	<i>Sequence Diagram Input Kerusakan Before Process</i>	96
Gambar V.22	<i>Sequence Diagram Input Kerusakan In Process Engine Rusak</i>	97
Gambar V.23	<i>Sequence Diagram Input Kerusakan In Process Engine Diperbaiki</i>	98
Gambar V.24	<i>Sequence Diagram Melihat dan Mencetak Laporan Before Process</i>	99
Gambar V.25	<i>Sequence Diagram Melihat dan Mencetak Laporan Kerusakan In Process</i>	100
Gambar V.26	<i>Sequence Diagram Input Hasil Evaluasi</i>	101
Gambar V.27	<i>Sequence Diagram Melihat dan Mencetak Laporan Evaluasi</i>	102
Gambar V.28	<i>Class Diagram Sistem Informasi Pengendalian Kualitas</i>	103
Gambar V.29	<i>Deployment Diagram Usulan</i>	108

Gambar V.30	HIPO Sistem Informasi Usulan.....	110
Gambar V.31	<i>Flowchart</i> Sistem Informasi Usulan	111
Gambar V.32	Rancangan <i>Form Login</i>	112
Gambar V.33	Tampilan Menu Utama Kepala Seksi	113
Gambar V.34	Tampilan Menu Utama Teknisi	114
Gambar V.35	Tampilan Menu Utama <i>Hintan</i> dan <i>Manager Assembly</i>	115
Gambar V.36	Rancangan Tampilan Mengelola Data <i>User</i>	115
Gambar V.37	Rancangan Tampilan <i>Form</i> Tambah <i>User</i>	116
Gambar V.38	Rancangan Tampilan <i>Form</i> Data Komponen	117
Gambar V.39	Rancangan Tampilan <i>Form</i> Tambah Komponen.....	117
Gambar V.40	Rancangan Tampilan <i>Form</i> data Jenis Kerusakan	118
Gambar V.41	Rancangan Tampilan <i>Form</i> Tambah Jenis Kerusakan.....	119
Gambar V.42	Rancangan Tampilan <i>Form</i> Data Tipe <i>Engine</i>	119
Gambar V.43	Rancangan Tampilan <i>Form</i> Tambah Tipe <i>Engine</i>	120
Gambar V.44	Rancangan Tampilan <i>Form</i> Data Lini Produksi	121
Gambar V.45	Rancangan Tampilan <i>Form</i> Tambah Lini Produksi.....	121
Gambar V.46	Rancangan Tampilan <i>Form</i> Data Asal Masalah Komponen.....	122
Gambar V.47	Rancangan Tampilan <i>Form</i> Tambah Asal Masalah Komponen .	122
Gambar V.48	Rancangan Tampilan <i>Form</i> Data Petugas.....	123
Gambar V.49	Rancangan Tampilan <i>Form</i> Petugas	124
Gambar V.50	Rancangan Tampilan <i>Form</i> Data Kerusakan <i>Before Process</i>	124
Gambar V.51	Rancangan Tampilan <i>Form</i> Tambah Kerusakan <i>Before Process</i>	125
Gambar V.52	Rancangan Tampilan <i>Form</i> Data Kerusakan <i>In Process</i>	126
Gambar V.53	Rancangan Tampilan <i>Form Input</i> Laporan <i>In Process</i>	126
Gambar V.54	Rancangan Tampilan Data Kerusakan <i>In Process</i> Diperbaiki.....	127
Gambar V.55	Rancangan Tampilan <i>Form</i> Data Evaluasi	127
Gambar V.56	Rancangan Tampilan <i>Form</i> Tambah Hasil Evaluasi	128
Gambar V.57	Lihat Laporan Kerusakan <i>Before Process</i> oleh <i>Hintan</i>	129
Gambar V.58	Lihat Laporan <i>Before Process</i> oleh <i>Manager</i> dan Kepala Seksi .	129
Gambar V.59	Rancangan Tampilan Melihat Laporan Kerusakan <i>In Process</i>	130
Gambar V.60	Rancangan Tampilan Melihat Laporan Evaluasi	131

DAFTAR TABEL

		Halaman
Tabel II.1	Simbol-Simbol <i>Flowchart</i>	22
Tabel II.2	Simbol-simbol <i>Use Case Diagram</i>	25
Tabel II.3	Simbol-simbol <i>Activity Diagram</i>	26
Tabel II.4	Simbol-simbol <i>Sequence Diagram</i>	27
Tabel II.5	Simbol-simbol <i>Class Diagram</i>	28
Tabel II.6	Simbol-simbol <i>Component Diagram</i>	29
Tabel II.7	Simbol-simbol <i>Deployment Diagram</i>	30
Tabel II.8	<i>Multiplicity</i>	31
Tabel II.9	Contoh Kamus Data Untuk Tabel Pemasok	32
Tabel II.10	Tipe Data pada MySQL	36
Tabel IV.1	Definisi Aktor <i>Usecase Diagram</i>	58
Tabel IV.2	Definisi <i>Use Case Diagram</i> Sistem Berjalan.....	58
Tabel IV.3	Skenario <i>Use Case</i> Melakukan Analisis Kerusakan	59
Tabel IV.4	Skenario <i>Use Case</i> Melakukan Perbaikan	60
Tabel IV.5	Skenario <i>Use Case</i> Mengisi <i>Form</i> NG.....	60
Tabel IV.6	Skenario <i>Use Case</i> Menempelkan <i>Form</i> NG pada <i>Engine</i>	60
Tabel IV.7	Skenario <i>Use Case</i> Mengisi <i>Form</i> Laporan Kerusakan	60
Tabel IV.8	Skenario <i>Use Case</i> Memonitor <i>Form</i> Laporan Kerusakan.....	61
Tabel IV.9	Skenario <i>Use Case</i> Melakukan Evaluasi Proses Kerja	61
Tabel IV.10	Skenario <i>Use Case</i> Membuat Laporan Evaluasi	61
Tabel IV.11	Skenario <i>Use Case</i> Membuat Laporan Kerusakan.....	61
Tabel IV.12	Skenario <i>Use Case</i> Memonitor Laporan Kerusakan.....	62
Tabel IV.13	Skenario <i>Use Case</i> Melakukan Evaluasi Kinerja	62
Tabel V.1	Kebutuhan Sistem Informasi Usulan	63
Tabel V.2	Kebutuhan Rinci Sistem.....	64
Tabel V.3	Definisi Aktor <i>Use Case Diagram</i> Sistem Informasi Usulan ..	70
Tabel V.4	Definisi <i>Use Case Diagram</i> Sistem Informasi Usulan	70

Tabel V.5	Skenario <i>Use Case Login</i>	72
Tabel V.6	Skenario <i>Use Case</i> Mengelola Data <i>User</i>	72
Tabel V.7	Skenario <i>Use Case</i> Mengelola Data <i>Master</i>	72
Tabel V.8	Skenario <i>Use Case Input</i> Kerusakan	73
Tabel V.9	Skenario <i>Use Case</i> Melihat Laporan Kerusakan	74
Tabel V.10	Skenario <i>Use Case</i> Mengolah Laporan Kerusakan	75
Tabel V.11	Skenario <i>Use Case</i> Mencetak Laporan Kerusakan	75
Tabel V.12	Skenario <i>Use Case Input</i> Hasil Evaluasi	76
Tabel V.13	Skenario <i>Use Case</i> Melihat Laporan Evaluasi	77
Tabel V.14	Skenario <i>Use Case</i> Mencetak Laporan Evaluasi	77
Tabel V.15	Tabel <i>user</i>	104
Tabel V.16	Tabel <i>mst_komponen</i>	104
Tabel V.17	Tabel <i>mst_tipeengine</i>	104
Tabel V.18	Tabel <i>mst_jeniskerusakan</i>	105
Tabel V.19	Tabel <i>mst_asal_masalah_komponen</i>	105
Tabel V.20	Tabel <i>mst_petugas</i>	105
Tabel V.21	Tabel <i>mst_liniproduksi</i>	106
Tabel V.22	Tabel <i>laporan_beforeprocess</i>	106
Tabel V.23	Tabel <i>laporan_inprocess</i>	107
Tabel V.24	Tabel <i>evaluasi</i>	107

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang cepat belakangan ini telah memasuki hampir semua bidang kehidupan. Perkembangan ditandai dengan banyaknya pengguna komputer, baik untuk kepentingan perusahaan atau bisnis. Segala kepentingan memerlukan teknologi informasi yang cepat dan akurat, salah satunya dengan membangun sistem informasi.

Sistem informasi dibangun untuk mempermudah penyajian informasi kepada pemakainya. Sistem informasi banyak dimanfaatkan oleh perusahaan-perusahaan industri, baik manufaktur maupun jasa. Sistem informasi digunakan sebagai alat untuk pendukung pengambilan keputusan. Salah satu contoh proses pengambilan keputusan yang langsung terhubung dengan konsumen adalah pengambilan keputusan terhadap kualitas produk yang dihasilkan.

Kualitas merupakan salah satu faktor yang mempengaruhi konsumen untuk menentukan produk dan jasa yang digunakan. Konsumen biasanya memilih produk dan jasa yang berkualitas yaitu yang sesuai dengan keinginan dan kebutuhan. Hal ini mengharuskan setiap perusahaan untuk mengetahui keinginan dan kebutuhan konsumennya. Komitmen terhadap kualitas produk dan jasa merupakan upaya yang harus dilakukan produsen untuk memenuhi keinginan konsumen.

PT Suzuki Indomobil Motor yang terletak di daerah Cakung merupakan perusahaan industri yang bergerak di bidang otomotif. Produk yang dihasilkan adalah *engine* yang digunakan pada kendaraan roda dua (sepeda motor) dan roda empat (mobil). PT Suzuki Indomobil Motor terdiri dari beberapa departemen yang saling berkaitan dalam melakukan kegiatan produksi. Salah satu departemen yang menangani proses produksi adalah Departemen Produksi, sedangkan Seksi *Assembly Engine 2W* yang berfungsi merakit setiap komponen *engine* kendaraan roda dua (sepeda motor).

PT Suzuki Indomobil Motor sudah menerapkan sistem informasi dalam menjalankan kegiatannya. Tetapi, masih terdapat beberapa departemen yang belum menerapkan sistem informasi dalam pengolahan datanya. Salah satu departemen yang belum menggunakan sistem informasi di PT Suzuki Indomobil Motor adalah Departemen Produksi Seksi *Assembly Engine 2W* yang masih menggunakan *Microsoft Excel*. Setiap data kerusakan *engine* ditulis dalam formulir kerusakan *engine* dan diolah menggunakan *Microsoft Excel*. Kepala Seksi *Assembly Engine 2W* harus mengolah data kerusakan *engine* yang jumlahnya sampai ratusan *record* setiap bulannya, sehingga pengolahan data dan laporan menjadi lambat. Data kerusakan *engine* dibuat menjadi laporan total kerusakan *assembly engine 2W* dalam periode bulanan yang disimpan pada *file server* perusahaan. Pembuatan laporan pada Departemen Produksi Seksi *Assembly Engine 2W* menggunakan program aplikasi *Microsoft Excel* membutuhkan waktu yang lama.

Demi meningkatkan efektivitas dalam mendukung kegiatan pengendalian kualitas pada Departemen Produksi Seksi *Assembly Engine 2W*, diperlukan pengembangan suatu sistem informasi yang dapat menangani pengolahan data kerusakan *engine*. Oleh karena itu, dibuatlah Tugas Akhir ini dengan judul “Rancang Bangun Sistem Informasi Pengendalian Kualitas *Assembly Engine 2W* Berbasis Web Menggunakan PHP 5.5 dan MySQL 5.6 PT Suzuki Indomobil Motor”.

1.2 Pokok Permasalahan

Permasalahan yang terjadi pada Departemen Produksi Seksi *Assembly Engine 2W* adalah sebagai berikut:

1. Proses pengolahan data kerusakan *engine* membutuhkan waktu yang lama karena setiap data yang ada masih ditulis dalam formulir kerusakan *engine*.
2. Proses pembuatan laporan kerusakan *assembly engine 2W* yang dilakukan oleh Departemen Produksi Seksi *Assembly Engine 2W* masih menggunakan *Microsoft Excel* yang memiliki kekurangan dalam penyajian laporan yang

informatif. Hal ini menyebabkan Manajer Departemen Produksi *Assembly Engine 2W* sering kesulitan dalam membaca hasil laporan kerusakan *engine*.

3. Belum adanya program aplikasi yang terintegrasi dengan *database* sebagai media penyimpanan data yang mengakibatkan terjadinya kehilangan data kerusakan *engine* karena semua data hanya tercatat dalam bentuk formulir kerusakan *engine* serta tersimpan dalam *Microsoft Excel*.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Membuat proses pengisian formulir kerusakan *engine* menjadi terkomputerisasi agar sistem pengolahan data kerusakan *engine* menjadi lebih cepat dan mudah.
2. Membuat laporan kerusakan *assembly engine 2W* menjadi lebih informatif agar lebih mudah dipahami dengan tampilan tabel yang dapat membantu *manager assembly engine 2W* dalam membaca setiap laporan yang dihasilkan.
3. Membuat sistem informasi pengendalian kualitas *assembly engine 2W* berbasis web menggunakan *database* sehingga memudahkan dalam penyimpanan data kerusakan *engine* dan mencegah terjadinya kehilangan data.

1.4 Batasan Masalah

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

1. Penelitian dilakukan di Departemen Produksi Seksi *Assembly Engine 2W*.
2. Analisis dan penelitian hanya mengenai sistem informasi pengendalian kualitas *assembly engine 2W* dalam bentuk laporan kerusakan di PT Suzuki Indomobil Motor.
3. Penelitian tidak membahas mengenai proses produksi *Assembly Engine 2W* yang dilakukan di PT Suzuki Indomobil Motor.

1.5 Manfaat Tugas Akhir

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

1. Bagi mahasiswa

Memberikan wawasan dan pengalaman kepada mahasiswa dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi untuk memberikan informasi yang akurat.

2. Bagi perusahaan

Hasil penelitian Rancang Bangun Sistem Pengendalian Kualitas *Assembly Engine 2W* diharapkan dapat mempermudah dalam pengolahan data, membuat laporan kerusakan *engine* dan membantu dalam pengambilan keputusan oleh *manager assembly engine 2W*.

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang telah dicantumkan. Adapun sistematika penulisan Tugas Akhir ini diuraikan dalam enam bab, 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, kualitas, pengembangan sistem, Diagram Alir (*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 yaitu metode *evolutionary prototype*.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini akan menguraikan tentang hasil pengamatan yang telah dilakukan, seperti tentang sekilas perusahaan dan Departemen Produksi Seksi *Assembly Engine 2W* sebagai objek penelitian yang mencakup pengolahan data kerusakan *assembly engine 2W*, dokumen yang terlibat, laporan yang dihasilkan oleh sistem dan analisis sistem yang berjalan.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi tentang analisis dan perancangan sistem yang diusulkan. Analisis dan perancangan sistem yang diusulkan menggunakan OOAD (*Object Oriented Analysis and Design*), dengan *tools* pemodelan UML (*Unified Modeling Language*) yaitu pemodelan sistem menggunakan *use case diagram*, *activity diagram* dan *sequence diagram*. Untuk pemodelan data menggunakan *Class Diagram*, Kamus Data dan *Deployment diagram*. Selanjutnya dibahas tentang perancangan aplikasi meliputi, HIPO (*Hierarchy plus Input-Process-Output*), *flowchart* dan perancangan tampilan. Tahapan terakhir yaitu melakukan implementasi sistem.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran dalam penerapan sistem informasi pengendalian kualitas *assembly engine 2W* untuk perusahaan dan pengembangan sistem yang ingin dilakukan selanjutnya.

BAB II

LANDASAN TEORI

2.1 Pengertian Rancang Bangun

Menurut Tim Penyusun KBBI (2008) “Rancang bangun adalah kata kerja untuk desain bangunan”. Apabila bangunan dianalogikan sebagai suatu sistem, jadi rancang bangun adalah desain sistem. Desain sistem dapat diartikan sebagai berikut ini (Jogiyanto, 2005):

1. Tahap setelah analisis dari siklus pengembangan sistem;
2. pendefinisian dari kebutuhan-kebutuhan fungsional;
3. persiapan untuk rancang bangun implementasi;
4. menggambarkan bagaimana suatu sistem dibentuk;
5. yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi;
6. termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem.

Berdasarkan penganalogian dan penjelasan di atas dapat disimpulkan bahwa rancang bangun adalah persiapan perancangan berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen terpisah juga konfigurasi dari komponen perangkat lunak dan perangkat keras dari suatu sistem.

2.2 Sistem Informasi

Sebelum mengetahui arti dari sistem informasi, ada baiknya mengetahui terlebih dahulu pengertian sistem dan pengertian informasi sehingga dapat memahami pengertian sistem informasi dengan baik.

2.2.1 Pengertian Sistem

Sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Sistem telah didefinisikan oleh para ahli dalam cara yang berbeda. Perbedaan tersebut terjadi karena perbedaan cara pandang dan lingkup sistem yang ditinjau. Sistem yang diperoleh dari beberapa sumber referensi, mendefinisikan bahwa sistem:

1. Menurut Indrajit (2001) “Mengemukakan bahwa sistem mengandung arti kumpulan-kumpulan dari komponen-komponen yang dimiliki unsur keterkaitan antara satu dengan lainnya”.
2. Menurut Jogianto (2005) “Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi”.
3. Menurut Murdick (1991) “Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data pada waktu rujukan tertentu untuk menghasilkan informasi”.
4. Menurut Davis (1991) “Sistem secara fisik adalah kumpulan dari elemen-elemen yang beroperasi bersama-sama untuk menyelesaikan suatu sasaran”.
5. Menurut Lani (1995) “Sistem adalah himpunan dari bagian-bagian yang saling berhubungan yang secara bersama mencapai tujuan”.

Secara umum, sistem dapat didefinisikan sebagai sekumpulan hal atau kegiatan atau elemen atau subsistem yang saling bekerja sama atau yang dihubungkan dengan cara-cara tertentu sehingga membentuk suatu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan (Sutanta, 2003).

2.2.2 Pengertian Informasi

Data dapat didefinisikan sebagai bahan keterangan tentang kejadian-kejadian nyata atau fakta-fakta yang dirumuskan dalam sekelompok lambang

tertentu yang tidak acak yang menunjukkan jumlah, tindakan atau hal. Data dapat berupa catatan-catatan dalam kertas, buku, atau tersimpan sebagai file dalam basis data. Berikut ini pengertian informasi yang diperoleh dari beberapa sumber referensi, mendefinisikan bahwa informasi:

1. Menurut Sutanta (2003) “Informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang”. Untuk memperoleh informasi, diperlukan adanya data yang akan diolah dan unit pengolah. Transformasi data menjadi informasi diperlihatkan pada gambar II.1.
2. Menurut Jogiyanto (2005) “Informasi adalah hasil pengolahan data yang dapat digunakan sebagai dasar pengambilan keputusan, Secara umum informasi dapat didefinisikan sebagai data yang diolah menjadi bentuk yang lebih berarti bagi yang menerimanya”.



Gambar II.1 Transformasi Data Menjadi Informasi
(Sumber: Sutanta, 2003)

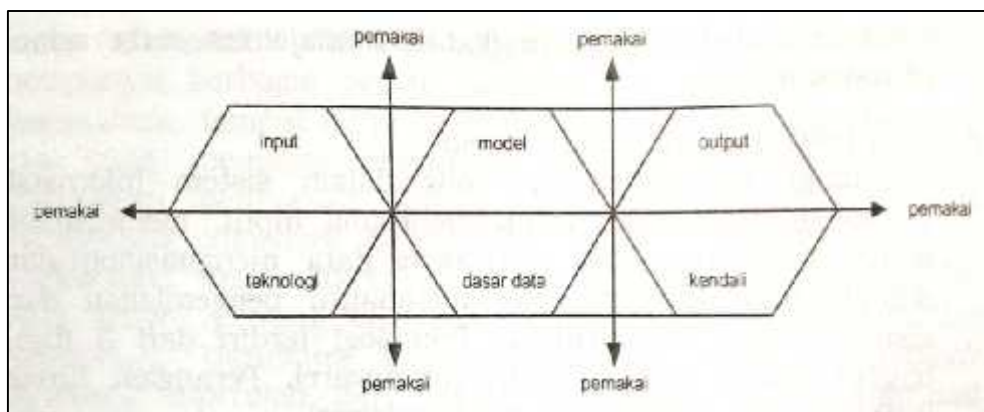
2.2.3 Pengertian Sistem Informasi

Menurut Sutanta (2003) “Sistem informasi dapat didefinisikan sebagai kumpulan subsistem yang saling berhubungan, berkumpul, bekerja bersama-sama dan membentuk satu kesatuan, saling berinteraksi dan bekerjasama antara bagian satu dengan yang lainnya dengan cara-cara tertentu untuk melakukan fungsi pengolahan data, menerima masukan (*input*) berupa data-data, kemudian mengolahnya (*processing*), dan menghasilkan keluaran (*output*) berupa informasi sebagai dasar bagi pengambilan keputusan yang berguna dan mempunyai nilai nyata yang dapat dirasakan akibatnya baik pada saat itu juga maupun di masa

mendatang, mendukung kegiatan operasional, manajerial, dan strategis organisasi, dengan memanfaatkan sumber daya yang ada dan tersedia bagi fungsi tersebut guna mencapai tujuan”.

2.2.4 Komponen Sistem Informasi

Menurut Sutabri (2004) “Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data, dan blok kendali. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran”. Komponen tersebut saling berhubungan dan saling terkait seperti Gambar II.2.



Gambar II.2 Komponen Sistem Informasi
(Sumber: Sutabri, 2004)

Berikut ini adalah penjelasan dari masing-masing komponen yang terlibat di dalam, yaitu (Sutabri, 2004):

1. Blok masukan (*input block*)

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

2. Blok model (*model block*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (*output block*)

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

4. Blok teknologi (*technology block*)

Teknologi merupakan *tool box* dalam sistem informasi, Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama yaitu teknisi (*brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*).

5. Blok basis data (*database block*)

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

6. Blok kendali (*control block*)

Merupakan sebuah komponen yang bertugas mendefinisikan bagaimana kontrol terhadap sistem dilakukan sehingga sistem dapat berjalan dengan baik. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dicegah dan apabila terlanjur terjadi maka kesalahan-kesalahan dapat dengan cepat diatasi.

2.3 Rancang Bangun Sistem Informasi

Berdasarkan penjelasan sebelumnya mengenai rancang bangun dan sistem informasi, maka dapat dibuat suatu kesimpulan bahwa rancang bangun sistem

informasi merupakan persiapan perancangan pengolahan data berupa gambaran atau sketsa, rencana dan pengaturan.

2.4 Pengendalian

Pengendalian berasal dari kata “kendali” yang berarti kekang. Menurut Tim Penyusun KBBI (2008) “pengendalian adalah proses, cara, perbuatan mengendalikan, atau pengawasan atas kemajuan (tugas) dengan membandingkan hasil dan sasaran secara teratur serta menyesuaikan usaha (kegiatan) dengan hasil pengawasan”.

Pengendalian menurut Glenn A. Welsch, Hilton, dan Gordon yang diterjemahkan oleh Purwatiningsih dan Maudy Warouw (2003) adalah “suatu proses untuk menjamin terciptanya kinerja yang efisien yang memungkinkan tercapainya tujuan perusahaan”. Menurut Gasperz (2005) “pengendalian dapat diartikan sebagai kegiatan yang dilakukan untuk memantau aktivitas dan memastikan kinerja sebenarnya yang dilakukan telah sesuai dengan yang direncanakan”.

Berdasarkan pengertian di atas dapat ditarik kesimpulan bahwa pengendalian adalah usaha untuk membandingkan prestasi kerja dengan rencana, pemeriksaan, evaluasi dan pemantauan atas pekerjaan yang telah dijalankan sesuai dengan rencana atau tujuan yang telah ditetapkan sebelumnya, dilakukan secara terus menerus agar dapat berfungsi sebaik mungkin untuk mencapai tujuan perusahaan.

2.4.1 Proses Pengendalian

Tahapan untuk mengawasi kegiatan periodik dalam suatu usaha dari tiap pusat pertanggungjawaban adalah sebagai berikut (Gasperz, 2005):

1. Menyiapkan laporan kinerja yang aktual.
2. Menganalisis jenis dan tindakan lain untuk mengetahui penyebab terjadinya ketidaksamaan atau penyimpangan.
3. Mengembangkan penyebab lain dari tindakan untuk memperbaiki kesalahan dan belajar dari keberhasilan.

4. Membuat pilihan, tindakan koreksi, perbaikan dari serangkaian alternatif dan menerapkannya.
5. Menindaklanjuti penilaian atas efektivitas perbaikan, memprosesnya untuk melakukan perencanaan ulang.

Berdasarkan uraian di atas, maka proses pengendalian memerlukan suatu standar yang telah ditetapkan sebelumnya, sebagai dasar atau patokan untuk menilai hasil yang ingin dicapai dalam pelaksanaan operasional perusahaan.

2.4.2 Cara Pengendalian

Setiap tugas yang dilaksanakan oleh seorang pemimpin, harus mampu mencapai suatu sasaran yang telah ditetapkan. Untuk itu para pemimpin harus dapat mengatur teknik atau cara-cara untuk melaksanakan pengendalian. Teknik atau cara pengendalian menurut Hasibuan (1996) sebagai berikut:

1. Pengendalian langsung, adalah pengendalian yang dilakukan sendiri secara langsung oleh seorang pemimpin, untuk memeriksa apakah hasil-hasil yang dicapai sesuai dengan yang dikehendakinya.
2. Pengendalian tidak langsung, adalah pengendalian jarak jauh artinya dengan melalui laporan yang diberikan oleh bawahan. Laporan ini dapat berupa lisan, tulisan, tentang pelaksanaan pekerjaan dan hasil-hasil yang telah dicapai.
3. Pengendalian berdasarkan kekecualian, adalah pengendalian yang dikhususkan untuk kesalahan-kesalahan yang luar biasa dari hasil standar yang diharapkan.

2.5 Kualitas

Kualitas memiliki makna yang berbeda bagi setiap orang, tergantung dari mana kita memandang. Pengertian kualitas menurut beberapa pakar (Yamit, 2013):

1. Kualitas adalah apapun yang menjadi kebutuhan dan keinginan konsumen (Deming, 1982).

2. Kualitas adalah sebagai nihil cacat, kesempurnaan, dan kesesuaian terhadap persyaratan (Crosby, 1979).
3. Mutu adalah kesesuaian terhadap spesifikasi (Juran, 1962).

Perusahaan jasa dan pelayanan lebih menekankan pada kualitas proses, karena konsumen biasanya terlibat langsung dalam proses tersebut. Sedangkan perusahaan yang menghasilkan produk lebih menekankan kepada hasil, karena konsumen tak terlibat langsung di dalam prosesnya. Untuk itu diperlukan sistem manajemen kualitas yang dapat memberikan jaminan kepada pihak konsumen bahwa produk tersebut dihasilkan oleh proses yang berkualitas (Yamit, 2013).

2.5.1 Dimensi Kualitas

Berdasarkan perspektif kualitas, terdapat delapan dimensi yang dapat digunakan sebagai dasar perencanaan strategis terutama bagi perusahaan atau manufaktur yang menghasilkan barang. Kedelapan dimensi itu adalah sebagai berikut (Yamit, 2013):

1. *Performance* (kinerja), yaitu karakteristik pokok dari produk inti.
2. *Features*, yaitu karakteristik pelengkap atau tambahan.
3. *Reliability* (kehandalan), yaitu kemungkinan tingkat kegagalan pemakaian.
4. *Conformance* (kesesuaian), yaitu sejauh mana karakteristik desain dan operasi memenuhi standar-standar yang telah ditetapkan sebelumnya.
5. *Durability* (daya tahan), yaitu berapa lama produk dapat terus digunakan.
6. *Serviceability*, yaitu meliputi kecepatan, kompetensi, kenyamanan, kemudahan dalam pemeliharaan, dan penanganan keluhan yang memuaskan.
7. Estetika, yaitu menyangkut corak, rasa, dan daya tarik produk.
8. *Perception*, yaitu menyangkut citra dan reputasi produk serta tanggung jawab perusahaan terhadapnya.

2.5.2 Pengendalian Kualitas

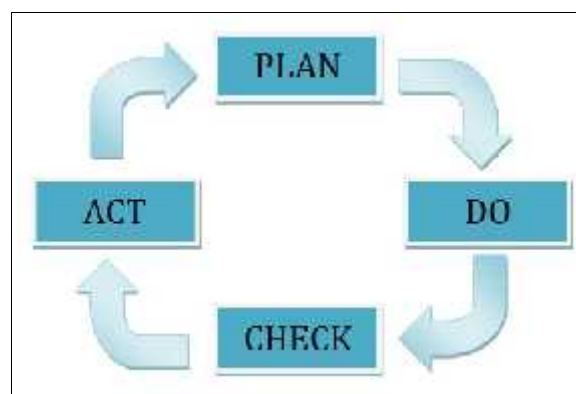
Pengendalian kualitas adalah aktifitas pengendalian proses untuk mengukur ciri-ciri kualitas produk, membandingkan dengan spesifikasi atau

persyaratan, dan mengambil tindakan penyehatan yang sesuai apabila ada perbedaan antara penampilan yang sebenarnya dan yang standar. Pengendalian Kualitas dapat membantu perusahaan dalam mengetahui kelayakan kualitas produk berdasarkan batas-batas kontrol yang telah ditentukan (Purnomo, 2004).

Pengendalian kualitas merupakan suatu sistem verifikasi dan penjagaan/perawatan dari suatu tingkat/derajat kualitas produk atau proses yang dikehendaki dengan perencanaan yang seksama, pemakaian peralatan yang sesuai, inspeksi yang terus menerus serta tindakan korektif bilamana diperlukan. Jadi pengendalian kualitas tidak hanya kegiatan inspeksi ataupun menentukan apakah produk itu baik (*accept*) atau jelek (*reject*) (Ginting, 2007).

Pengendalian adalah keseluruhan fungsi atau kegiatan yang harus dilakukan untuk menjamin tercapainya sasaran perusahaan dalam hal kualitas produk dan jasa pelayanan yang diproduksi. Pengendalian kualitas pelayanan pada dasarnya adalah pengendalian kualitas kerja dan proses kegiatan untuk menciptakan kepuasan pelanggan yang dilakukan setiap orang dalam bagian organisasi (Yamit, 2013).

Pengertian pengendalian kualitas diatas mengacu pada siklus pengendalian (*control cycle*) dengan memutar siklus “*Plan-Do-Check-Action*” (P-D-C-A). Pendekatan siklus P-D-C-A dapat dijadikan model untuk menganalisis proses dan kualitas. Hubungan peningkatan kualitas dengan peningkatan perbaikan berdasarkan siklus P-D-A-C diperlihatkan dalam gambar II.3.



Gambar II.3 Siklus P-D-A-C
(Sumber: Yamit, 2013)

2.5.3 Tujuan Pengendalian Kualitas

Pengendalian kualitas memiliki tujuan tertentu. Adapun tujuan dari pengendalian kualitas menurut Assauri (2004) adalah:

1. Agar barang hasil produksi dapat mencapai standar kualitas yang telah ditetapkan.
2. Mengusahakan agar biaya inspeksi dapat menjadi sekecil mungkin.
3. Mengusahakan agar biaya desain dari produk dan proses dengan menggunakan kualitas produksi tertentu dapat menjadi serendah mungkin.
4. Mengusahakan agar biaya produksi dapat menjadi serendah mungkin.

Tujuan utama pengendalian kualitas adalah untuk mendapatkan jaminan bahwa kualitas produk atau jasa yang dihasilkan sesuai dengan standar kualitas yang telah ditetapkan dengan mengeluarkan biaya yang ekonomis atau serendah mungkin.

2.6 Instrumen Dasar Peningkatan Kualitas

Sebagai konsep pengembangan berkelanjutan yang melibatkan tenaga kerja, diperlukan instrumen yang dapat membantu mengatasi masalah secara sistematis. Instrumen pertama dalam peningkatan kualitas adalah berhubungan dengan *Total Quality Management* (TQM) yang difokuskan pada aspek penyelesaian masalah tentang *issue* operasional yang terjadi setiap hari. Instrumen dasar peningkatan kualitas ini banyak digunakan diseluruh dunia oleh perusahaan-perusahaan yang melakukan upaya *total quality*. Teori ini disebut sebagai *The Basic Quality Improvement Tools*, yaitu (Yamit, 2013):

1. Pengumpulan data (*Check Sheet*)
2. *Pareto Chart*
3. *Histogram*

2.6.1 Lembar Pengumpulan Data (*Check Sheet*)

Lembar pengumpulan data (*check sheet*) diperlihatkan pada gambar II.4 merupakan alat bantu untuk memudahkan pengumpulan data. Aplikasi

penggunaan *check sheet* ini dalam pengendalian kualitas biasanya untuk keperluan, antara lain (Yamit, 2013):

1. Membantu memahami situasi sebenarnya yang terjadi pada suatu proses (membedakan antara opini dan fakta).
2. Menganalisis permasalahan (seberapa sering suatu masalah terjadi).
3. Mengendalikan proses yang sedang berjalan.

Check Sheet Data Permasalahan Karyawan				
Problem	Januari	Februari	Maret	Total
A	/	/	-	2
B	/	/	-	2
C	/	//	//	5
D	-	///	//	5
Total	3	7	4	14

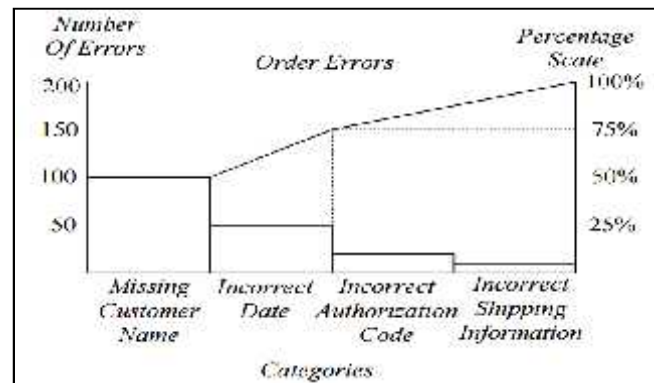
Gambar II.4 Contoh *Check Sheet*
(Sumber: Purnomo, 2004)

2.6.2 *Pareto Chart*

Pareto chart merupakan metode yang digunakan untuk menentukan masalah mana yang harus dikerjakan lebih dahulu. *Pareto chart* mendasarkan keputusannya pada data kuantitatif (Yamit, 2013). Contoh *pareto chart* ditampilkan pada gambar II.5.

Beberapa kegunaan dari *pareto chart* ini diantaranya adalah sebagai berikut:

1. Menunjukkan persoalan utama yang ada pada suatu proses atau rangkaian proses.
2. Menyatakan perbandingan masing-masing persoalan terhadap keseluruhan.
3. Skala prioritas dari setiap permasalahan yang sedang dibahas
4. Sebagai alat untuk melakukan evaluasi, terhadap tingkat keberhasilan dari suatu proses perbaikan.

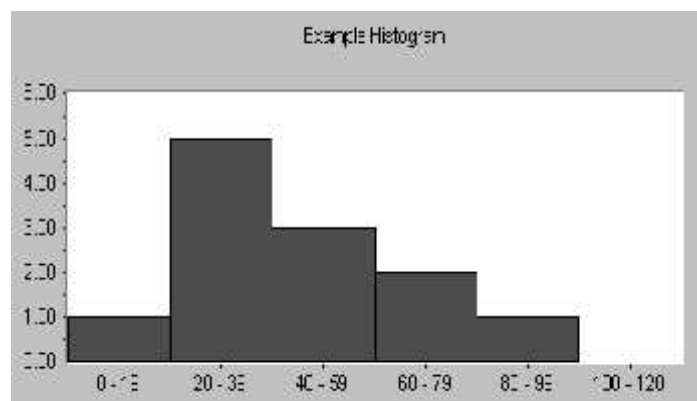


Gambar II.5 Contoh *Pareto Chart*
(Sumber: Yamit, 2013)

2.6.3 *Histogram*

Histogram yang ditampilkan pada gambar II.6 adalah salah satu alat bantu dalam memecahkan masalah yang berupa grafik khusus yang menggambarkan penyebaran data sebagai hasil dari satu macam pengukuran dari suatu proses, yang dapat digunakan untuk (Yamit, 2013):

1. Membuktikan atau menyelidiki apakah suatu proses benar-benar terjadi. Dimana histogram akan berfungsi sebagai indikator masalah dan dengan penyelidikan lebih lanjut dapat dibuktikan sumber atau sebab masalah tersebut.
2. Menyampaikan informasi mengenai variasi dalam suatu proses. Mengambil keputusan dengan memusatkan perhatian pada upaya perbaikan.



Gambar II.6 Contoh *Histogram*
(Sumber: Purnomo, 2004)

2.7 Pengembangan Sistem

Proses perancangan atau pengembangan sistem informasi, mulai dari konsep sampai dengan implementasinya disebut dengan istilah *System Development Life Cycle* (SDLC) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Rosa dan Shalahuddin, 2014).

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

1. Inisiasi (*initiation*)
Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.
2. Pengembangan konsep sistem (*system concept development*)
Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem analisis manfaat biaya, manajemen rencana dan pembelajaran kemudahan sistem.
3. Perencanaan (*planning*)
Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.
4. Analisis kebutuhan (*requirements analysis*)
Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.
5. Desain (*design*)
Mentransformasikan kebutuhan *detail* menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
6. Pengembangan (*development*)
Mengkonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan, membuat basis data dan mempersiapkan prosedur kasus pengujian,

mempersiapkan berkas atau *file* pengujian, pengkodean, pengkompilasian, memperbaiki dan membersihkan program serta peninjauan pengujian.

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

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

8. Implementasi (*implementation*)

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

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

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

10. Disposisi (*disposition*)

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

2.7.1 Prototyping

Menurut McLeod (2011) “Dalam pengembangan sistem, prototipe adalah satu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, bagaimana sistem akan berfungsi dalam bentuk yang telah selesai. Proses pembuatan prototipe ini disebut *prototyping*”.

Pengguna maupun pengembang menyukai *prototyping* karena alasan-alasan di bawah ini (McLeod, 2011):

1. Membaiknya komunikasi antara pengembang dan pengguna.
2. Pengembang dapat memberikan pekerjaan yang lebih baik dalam menentukan kebutuhan pengguna.
3. Pengguna memainkan peranan yang lebih aktif dalam pengembangan sistem.

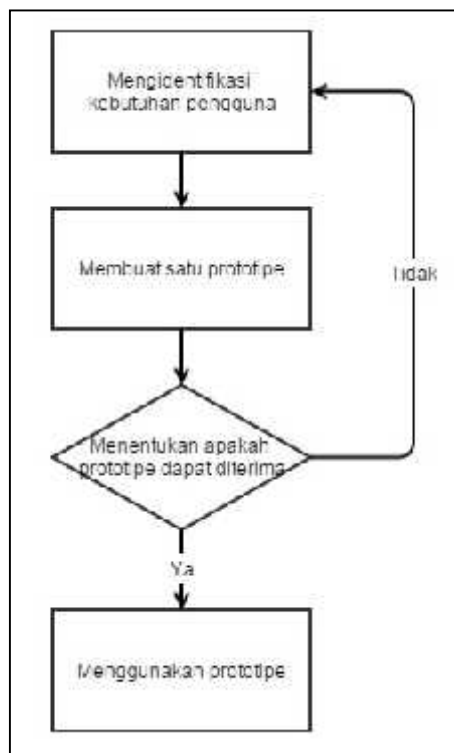
4. Pengembang dan pengguna menghabiskan waktu dan usaha yang lebih sedikit dalam mengembangkan sistem.
5. Implementasi menjadi jauh lebih mudah karena pengguna tahu apa yang diharapkannya.

Keuntungan-keuntungan di atas memungkinkan *prototyping* memangkas biaya pengembangan dan meningkatkan kepuasan pengguna atas sistem yang diserahkan. Walau begitu *prototyping* bukannya tidak memiliki potensi kesulitan. Kesulitan-kesulitan tersebut antara lain (McLeod, 2011):

1. Terburu-buru dalam menyerahkan prototipe dapat menyebabkan diambilnya jalan pintas dalam definisi masalah, evaluasi alternatif, dan dokumentasi. Jalan pintas ini akan menciptakan usaha-usaha yang “cepat dan kotor”.
2. Pengguna dapat terlalu gembira dengan prototipe yang diberikan, yang mengarah pada ekspektasi yang tidak realistis sehubungan dengan sistem produksi nantinya.
3. Prototipe evolusioner bisa jadi tidak terlalu efisien.
4. Antarmuka komputer-manusia yang diberikan oleh beberapa alat *prototyping* tertentu kemungkinan tidak mencerminkan teknik-teknik desain yang baik.

2.7.2 Evolutionary Prototype

Menurut Mcleod (2011) “*Evolutionary prototype* atau prototipe evolusioner adalah salah satu jenis prototipe yang secara terus-menerus disempurnakan sampai memiliki seluruh fungsionalitas yang dibutuhkan pengguna dari sistem yang baru. Prototipe ini kemudian dilanjutkan produksi, jadi satu prototipe evolusioner akan menjadi sistem aktual”.



Gambar II.7 Langkah Pembuatan Prototipe Evolusioner
(Sumber: McLeod, 2011)

Terdapat empat langkah dalam pembuatan suatu prototipe evolusioner yang digambarkan dalam Gambar II.7, antara lain (McLeod, 2011):

1. Mengidentifikasi kebutuhan pengguna

Pengembang mewawancarai pengguna untuk mendapatkan ide mengenai apa yang diminta dari sistem.

2. Membuat satu prototipe

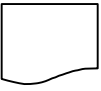
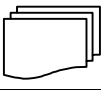
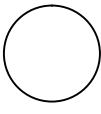
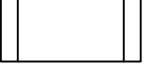
Pengembang mempergunakan satu alat *prototyping* atau lebih untuk membuat prototipe. Contoh dari alat-alat *prototyping* adalah generator aplikasi terintegrasi dan *toolkit prototyping*. Generator aplikasi terintegrasi (*integrated application generator*) adalah sistem peranti lunak siap pakai yang mampu membuat seluruh fitur yang diinginkan dari sistem baru, menu, laporan, tampilan, basis data, dan seterusnya. *Toolkit prototyping* meliputi sistem-sistem peranti lunak terpisah seperti *spreadsheet* elektronik atau sistem manajemen basis data, yang masing-masing mampu membuat sebagian dari fitur-fitur sistem yang diinginkan.

3. Menentukan apakah prototipe dapat diterima
Pengembang mendemonstrasikan prototipe kepada pengguna untuk mengetahui apakah telah memberikan hasil yang memuaskan. Jika ya, langkah 4 akan diambil; jika tidak, prototipe direvisi dengan mengulang kembali langkah 1, 2, dan 3 dengan pemahaman yang lebih baik mengenai kebutuhan pengguna.
4. Menggunakan prototipe
Merupakan lanjutan dari rancangan, prototipe ini langsung disusun sebagai suatu sistem informasi yang akan digunakan.


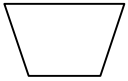




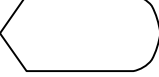

2.8 Flowchart

Bagan alir (*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. Bagan alir digambar dengan menggunakan simbol-simbol yang terdapat pada Tabel II.1 (Jogiyanto, 2005).

Tabel II.1 Simbol-simbol *Flowchart*

Simbol	Nama	Keterangan
	Dokumen	Digunakan untuk semua jenis dokumen yang merupakan formulir untuk merekam transaksi.
	Dokumen rangkap	Menggambarkan dokumen asli dan tembusannya.
	Penghubung pada halaman yang sama	Simbol penghubung yang memungkinkan aliran dokumen berhenti di suatu lokasi pada halaman tertentu dan kembali berjalan pada halaman yang sama.
	Proses terdefinisi	Menunjukkan proses rincian.

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

Simbol	Nama	Keterangan
	Penghubung pada halaman yang berbeda	Untuk menggambarkan bagan alir dokumen suatu sistem diperlukan lebih dari satu halaman.
	Kegiatan manual	Untuk menggambarkan kegiatan manual seperti menerima <i>order</i> , mengisi formulir, membandingkan dan lain-lain.
	Arsip sementara	Menunjukkan tempat penyimpanan dokumen.
	Arsip permanen	Menunjukkan tempat penyimpanan dokumen secara permanen yang tidak akan diproses lagi.
	Proses	Simbol proses yang digunakan untuk mewakili suatu proses dengan komputer.
	Keputusan	Menggambarkan keputusan yang harus dibuat dalam proses pengolahan data.
	Tampilan <i>Output</i>	Menunjukkan <i>output</i> yang ditampilkan pada monitor.
	<i>Input/Output</i>	Menggambarkan data <i>input/output</i> dari proses.

(Sumber: Jogiyanto, 2005)

2.9 Unified Modeling Language (UML)

Menurut Rosa dan Shalahuddin (2014) “*Unified Modeling Language* (UML) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML merupakan

sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek”.

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak.

2.9.1 Tujuan UML

Menurut Suhendar dan Gunadi (2002) bahwa tujuan utama UML adalah:

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

2.9.2 Diagram UML

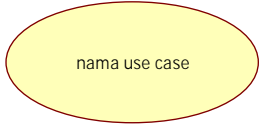




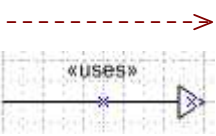
Terdapat beberapa jenis *diagram* UML yang dapat digunakan dengan masing-masing fungsinya yaitu (Rosa dan Shalahuddin, 2014):

1. *Use Case Diagram*

Use Case Diagram merupakan permodelan untuk kejadian-kejadian di sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa dan Shalahuddin, 2014).

Berikut adalah simbol-simbol yang ada pada *use case diagram* yang terdapat pada tabel II.2:

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

No	Simbol	Deskripsi
1	<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> .
2	<p>Aktor/<i>actor</i></p> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, tapi biasanya dinyatakan menggunakan kata benda atau <i>frase</i> nama aktor.
3	<p>Asosiasi/<i>association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4	<p>Ekstensi/<i>Extend</i></p> <p><<extend>></p> 	Relasi <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu.
5	<p>Generalisasi/<i>generalization</i></p> 	Hubungan generalisasi dan spesifikasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya.
6	<p><i>Include</i></p> <p><<include>></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.



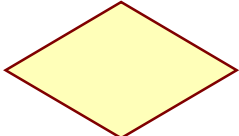


(Sumber: Rosa dan Shalahuddin, 2014)

2. *Activity Diagram*

Diagram aktivitas atau *Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Rosa dan Shalahuddin, 2014).

Berikut adalah simbol-simbol yang ada pada *activity diagram* yang terdapat pada tabel II.3:

Tabel II.3 Simbol-simbol *Activity Diagram*

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


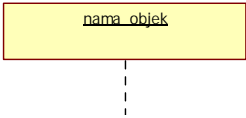

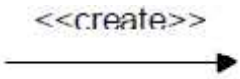
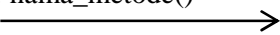
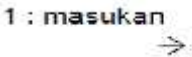
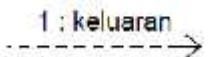
(Sumber: Rosa dan Shalahuddin, 2014)

3. *Sequence Diagram*

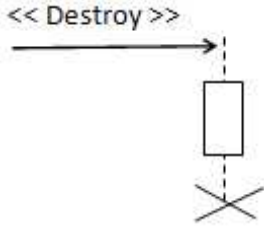
Diagram sekuen menggambarkan perbuatan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu (Rosa dan Shalahuddin, 2014).

Berikut ini adalah komponen-komponen dasar yang ada di dalam *Sequence Diagram* yang terdapat pada tabel II.4:

Tabel II.4 Simbol-simbol *Sequence Diagram*

No	Simbol	Deskripsi
1	Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri.
2	Objek 	Menyatakan objek yang berinteraksi pesan.
3	Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
4	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek lain, arah panah mengarah kepada objek yang dibuat.
5	Pesan tipe <i>call</i> 1: nama_metode() 	Menyatakan suatu objek memanggil operasi atau metode yang ada pada objek lain atau dirinya sendiri.
6	Pesan tipe <i>send</i> 1 : masukan 	Menyatakan bahwa suatu objek mengirimkan data atau masukan atau informasi ke objek lainnya, arah panah, mengarah kepada objek yang dikirim.
7	Pesan tipe <i>return</i> 1 : keluaran 	Menyatakan bahwa suatu objek yang telah dijalankan suatu operasi dan menghasilkan kembalian dari objek tertentu, arah panah mengarah kepada objek yang menerima kembalian.

Tabel II.4 Simbol-simbol *Sequence Diagram* (Lanjutan)

No	Simbol	Deskripsi
8	Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah kepada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

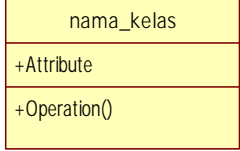
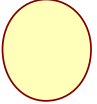
(Sumber: Rosa dan Shalahuddin, 2014)

4. *Class Diagram*






Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa dan Shalahuddin, 2014).

Berikut adalah simbol-simbol yang ada pada *class diagram* yang terdapat pada tabel II.5.

Tabel II.5 Simbol-simbol *Class Diagram*

No	Simbol	Dekripsi
1	Kelas 	Kelas pada struktur sistem + <i>attribute</i> : menggambarkan atribut yang dimiliki <i>class</i> . + <i>Operation()</i> : menunjukkan suatu tindakan yang dapat dilakukan oleh <i>class</i> .
2	Antarmuka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.

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

No	Simbol	Dekripsi
3	Asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4	Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesifikasi (umum-khusus).
6	Kebergantungan/dependency 	Kebergantungan antarkelas.
7	Agregasi / <i>Aggregation</i> 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>).

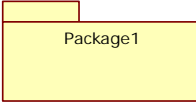
(Sumber: Rosa dan Shalahuddin, 2014)

5. *Component Diagram*

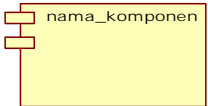



Diagram komponen atau *Component Diagram* dibuat untuk menunjukkan organisasi atau kebergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem (Rosa dan Shalahuddin, 2014).

Berikut adalah simbol-simbol yang ada pada *component diagram* yang terdapat pada tabel II.6:

Tabel II.6 Simbol-simbol *Component Diagram*

No	Simbol	Deskripsi
1	<i>Package</i> 	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih komponen.

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

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

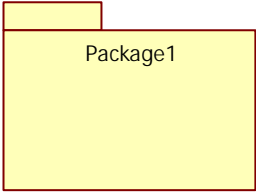
(Sumber: Rosa dan Shalahuddin, 2014)

6. *Deployment Diagram*

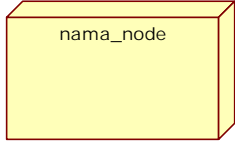


Diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi (Rosa dan Shalahuddin, 2014).

Berikut adalah simbol-simbol yang ada pada *deployment diagram* yang terdapat pada tabel II.7:

Tabel II.7 Simbol-simbol *Deployment Diagram*

No	Simbol	Deskripsi
1	<i>Package</i> 	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih <i>node</i> .

Tabel II.7 Simbol-simbol *Deployment Diagram* (Lanjutan)

No	Simbol	Deskripsi
2	<p><i>Node</i></p> 	Biasanya mengacu kepada perangkat keras, atau perangkat lunak.
3	<p>Kebergantungan/<i>Dependency</i></p> 	Kebergantungan antar node, arah panah mengarah kepada node yang dipakai.
4	<p><i>Link</i></p> 	Relasi antar model.

(Sumber: Rosa dan Shalahuddin, 2014)

2.10 Multiplicity

Multiplicity pada UML memungkinkan untuk menentukan kardinalitas yaitu jumlah elemen dari beberapa kumpulan elemen. *Multiplicity* adalah definisi kardinalitas yaitu jumlah elemen dari beberapa kumpulan elemen dengan memberikan interval termasuk bilangan bulat (*non-negative*) untuk menentukan jumlah contoh elemen dijelaskan (uml-diagrams.org, 2015). Berikut ini pada Tabel II.8 menjelaskan beberapa contoh tipe *multiplicity* (uml-diagrams, 2015):

Tabel II.8 *Multiplicity*

<i>Multiplicity</i>	<i>Option</i>	<i>Cardinalitas</i>
0..0	0	Nol
0..1		Antara nol sampai satu
1..1	1	Tepat satu
0..*	*	Antara nol sampai banyak
1..*		Antara satu sampai banyak
5..5	5	Tepat lima
m..n		Sedikitnya m tapi tidak boleh lebih dari n

(Sumber: uml-diagrams.org, 2015)

2.11 Kamus Data

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

Kamus data dapat berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan di dalam sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses. Berikut ini adalah contoh penulisan kamus data:

1. Spesifikasi tabel pemasok
 - Nama tabel : Pemasok
 - Tipe : File master

Tabel II.9 Contoh Kamus Data Untuk Tabel Pemasok

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

(Sumber: Jogiyanto, 2005)

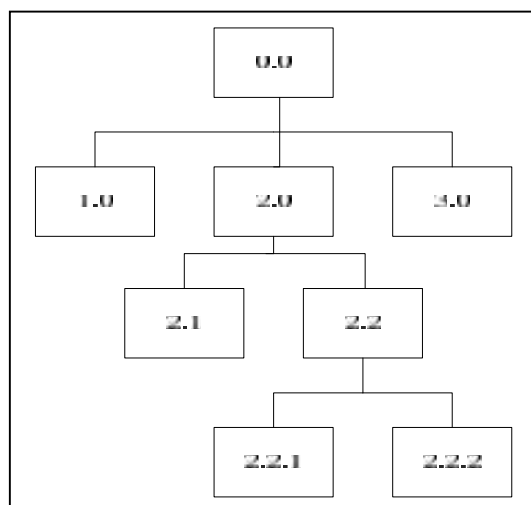
2.12 *Hierarchy plus Input-Proses-Output (HIPO)*

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

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-statementen program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan yang jelas dari *input* yang harus digunakan dan *output* yang harus dihasilkan oleh masing-masing fungsi pada tiap-tiap tingkatan dari diagram-diagram HIPO.
4. Untuk menyediakan *output* yang tepat dan sesuai dengan kebutuhan-kebutuhan pemakai.

Salah satu jenis dari HIPO yaitu *Visual Table Of Contents* (VTOC). *Visual table of contents* menggambarkan hubungan fungsi-fungsi di sistem secara berjenjang, *visual table of contents* menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur (Jogiyanto, 2005). Pada diagram ini nama dan nomor dari program HIPO diidentifikasi. Struktur paket diagram dan hubungan fungsi juga diidentifikasi dalam bentuk hirarki. Keterangan masing-masing fungsi diberikan pada bagian penjelasan yang diikutsertakan dalam diagram ini. *Visual table of contents* ini dapat digambarkan pada Gambar II.8 sebagai berikut:



Gambar II.8 *Visual Table Of Contents*
(Sumber: Jogiyanto, 2005)

2.13 PHP

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

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

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

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

Berikut ini beberapa kelebihan PHP dari bahasa pemrograman lain (Anhar, 2010):

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

6. PHP juga dilengkapi dengan berbagai macam pendukung lain seperti *support* langsung ke berbagai macam *database* yang populer, misal: Oracle, PostgreSQL, dan lain-lain.

2.14 MySQL

Menurut Anhar (2010) “MySQL (*My Structure Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS dari sekian banyak DBMS seperti Oracle, MSSQL, Postgre SQL, dan lain-lain. MySQL merupakan DBMS yang *multithread*, *multi-user* yang bersifat gratis di bawah GNU *General Public Licence* (GPL)”.

Kelebihan dari MySQL dibandingkan dengan DBMS lainnya menurut Anhar (2010):

1. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Bersifat *Open Source*, MySQL didistribusikan secara *open source* (gratis), di bawah lisensi GNU *General Public Licence* (GPL).
3. Bersifat *Multiuser*, MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah.
4. MySQL memiliki kecepatan yang baik dalam menangani *query* (perintah SQL), dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. MySQL memiliki beberapa lapisan keamanan, seperti level *subnet mask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta *password* yang terenkripsi.
6. MySQL memiliki antarmuka terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

MySQL menyediakan berbagai tipe data dengan spesifikasi dan panjang masing-masing. Tipe data untuk *field* dalam MySQL diantaranya ditampilkan pada tabel II.10.

Tabel II.10 Tipe Data pada MySQL

Jenis Tipe	Tipe	Keterangan
<i>NUMERIC</i>	<i>TINYINT</i>	Dari -128 hingga 127 <i>SIGNED</i> dan dari 0 hingga 255 <i>UNSIGNED</i>
	<i>SMALLINT</i>	Dari -32768 hingga 32767 <i>SIGNED</i> dan dari 0 hingga 65535 <i>UNSIGNED</i>
	<i>MEDIUMINT</i>	Dari -8388608 hingga 8388607 <i>SIGNED</i> dan dari 0 hingga 16777215 <i>UNSIGNED</i>
NUMERIC	<i>INT</i>	Dari -2147483648 hingga 2147483647 <i>SIGNED</i> dan dari 0 hingga 4294967295 <i>UNSIGNED</i> .
	<i>BIGINT</i>	Dari -9223372036854775808 hingga 9223372036854775807 <i>SIGNED</i> dan dari 0 hingga 18446744073709551615 <i>UNSIGNED</i>
	<i>FLOAT</i>	Bilangan pecahan presisi tunggal
	<i>DOUBLE</i>	Bilangan pecahan presisi ganda
	<i>DECIMAL</i>	Bilangan dengan decimal
<i>DATE/TIME</i>	<i>DATE</i>	Tanggal dengan format YYYY-MM-DD
<i>DATE/TIME</i>	<i>DATETIME</i>	Tanggal dan waktu dengan format YYYY-MM-DD HH:MM:SS
	<i>TIMESTAMP</i>	Tanggal dan waktu dengan format YYYYMMDDHHMMSS
	<i>YEAR</i>	Tahun dengan format YYYY
<i>STRING</i>	<i>CHAR</i>	0 – 255 karakter
	<i>VARCHAR</i>	0 – 255 karakter
	<i>TINYTEXT</i>	<i>String</i> dengan panjang maksimum 255 karakter
	<i>TEXT</i>	<i>String</i> dengan panjang maksimum 65535 karakter
	<i>BLOB</i>	<i>String</i> dengan panjang maksimum 65535 karakter

(Sumber: Solichin, 2009)

2.15 XAMPP

XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan website berbasis PHP dan menggunakan pengolahan data MySQL di komputer

lokal. XAMPP merupakan paket PHP yang berbasis *open source* yang dikembangkan oleh sebuah komunitas *open source*. XAMPP berperan sebagai *web server* pada komputer (Nugroho, 2008).

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

1. *htdocs* adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan *script* lain.
2. *phpMyAdmin* merupakan bagian untuk mengelola basis data MySQL yang ada di komputer. Untuk membukanya, isi pada kotak alamat (*address bar*) di mesin pencari (*browser*) dengan *http://localhost/phpMyAdmin*, maka akan muncul halaman *phpMyAdmin*.
3. Kontrol Panel yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah suatu prosedur dan alat yang digunakan dalam penelitian. Metodologi penelitian mempelajari cara-cara melakukan pengamatan dengan pemikiran yang tepat secara terpadu melalui tahapan-tahapan yang disusun secara ilmiah untuk mencari, menyusun, serta menganalisis dan menyimpulkan data-data berdasarkan fakta-fakta secara ilmiah (Nazir, 2011).

Untuk menghasilkan penelitian tugas akhir yang lebih lengkap diperlukan suatu metode dalam penelitian yang telah dipersiapkan sesuai dengan masalah yang akan dibahas.

3.2 Jenis dan Sumber Data

Data yang dibutuhkan dalam penelitian ini dikelompokkan menjadi dua jenis, yaitu:

1. Data primer

Data yang diperoleh melalui pengamatan langsung terhadap sistem yang sedang berjalan dan wawancara dengan pegawai Departemen Produksi Seksi *Assembly Engine 2W* sebagai sumber informasinya. Dalam penelitian ini data tersebut berupa data alur proses *update* dan data yang mendukung informasi yang dihasilkan terhadap kualitas *assembly engine 2W* pada PT Suzuki Indomobil Motor.

2. Data Sekunder

Data sekunder merupakan sumber data penelitian yang diperoleh secara tidak langsung melalui media perantara, kepustakaan, buku pengendalian kualitas, internet dan referensi. Dalam penelitian ini data tersebut berupa data umum perusahaan, profil perusahaan dan struktur organisasi perusahaan

serta teori-teori mengenai pengendalian kualitas, pemrograman PHP dan *database* MySQL.

3.3 Metode Pengumpulan Data

Sumber data atau informasi penelitian ini berdasarkan kepada jenis data yang diperlukan. Metode yang digunakan dalam pengumpulan data penelitian ini, diantaranya:

1. Studi Pustaka

Tahap ini merupakan tahap awal pengumpulan data dari sumber-sumber seperti buku yang berhubungan dengan masalah yang sedang dianalisis. Studi pustaka yang dilakukan berkaitan dengan kualitas, pengembangan sistem, analisis dan desain sistem informasi, pemrograman PHP dan *database* MySQL dari berbagai referensi, baik itu referensi elektronik yang didapat dari internet maupun referensi dari buku teks.

2. Observasi

Tahap ini merupakan tahap melakukan pengamatan langsung pada sistem yang berjalan di Departemen Produksi Seksi *Assembly Engine 2W*. Pengamatan yang dilakukan mengenai alur proses pengendalian kualitas *assembly engine 2W*, dokumen yang terkait dengan sistem yang diteliti, laporan kualitas yang digunakan oleh Departemen Produksi Seksi *Assembly Engine 2W*. Hasil dari pengamatan yang dilakukan menjadi landasan dalam melakukan pengembangan sistem yang akan dibuat.

3. Wawancara

Pada tahap ini melakukan wawancara dengan para pengguna sistem tersebut untuk memperoleh keterangan mengenai alur proses *input* data kerusakan *engine* dalam menghasilkan laporan harian dan juga bulanan serta mengenai kebutuhan sistem yang akan dikembangkan selanjutnya. Keterangan yang diperoleh bertujuan agar sistem selanjutnya dapat mudah, cepat dan akurat dalam melakukan pengendalian terhadap kualitas *engine*.

3.4 Metode Pengembangan Sistem

Metode pengembangan sistem informasi pengendalian kualitas *assembly engine 2W* ini menggunakan model *prototype* jenis *evolutionary prototype*. Model *prototype* cocok digunakan untuk menjabarkan kebutuhan *user* secara lebih detail karena *user* sering kali kesulitan menyampaikan kebutuhannya secara *detail* tanpa melihat gambaran yang jelas. Tahapan-tahapan dalam *evolutionary prototype* adalah sebagai berikut (Mcleod, 2011):

1. Mengidentifikasi kebutuhan pengguna
Pengembang mewawancarai pengguna untuk mendapatkan ide mengenai apa yang dibutuhkan dari sistem.
2. Membuat satu prototipe
Pengembang mempergunakan satu alat *prototyping* atau lebih untuk membuat prototipe.
3. Menentukan apakah prototipe dapat diterima
Pengembang mendemonstrasikan prototipe kepada pengguna untuk mengetahui apakah telah memberikan hasil yang memuaskan. Jika ya, langkah 4 akan diambil; jika tidak, prototipe direvisi dengan mengulang kembali langkah 1, 2, dan 3 dengan pemahaman yang lebih baik mengenai kebutuhan pengguna.
4. Menggunakan prototipe
Sistem mulai dikembangkan dengan prototipe yang sudah dibuat.

3.5 Kerangka Penelitian

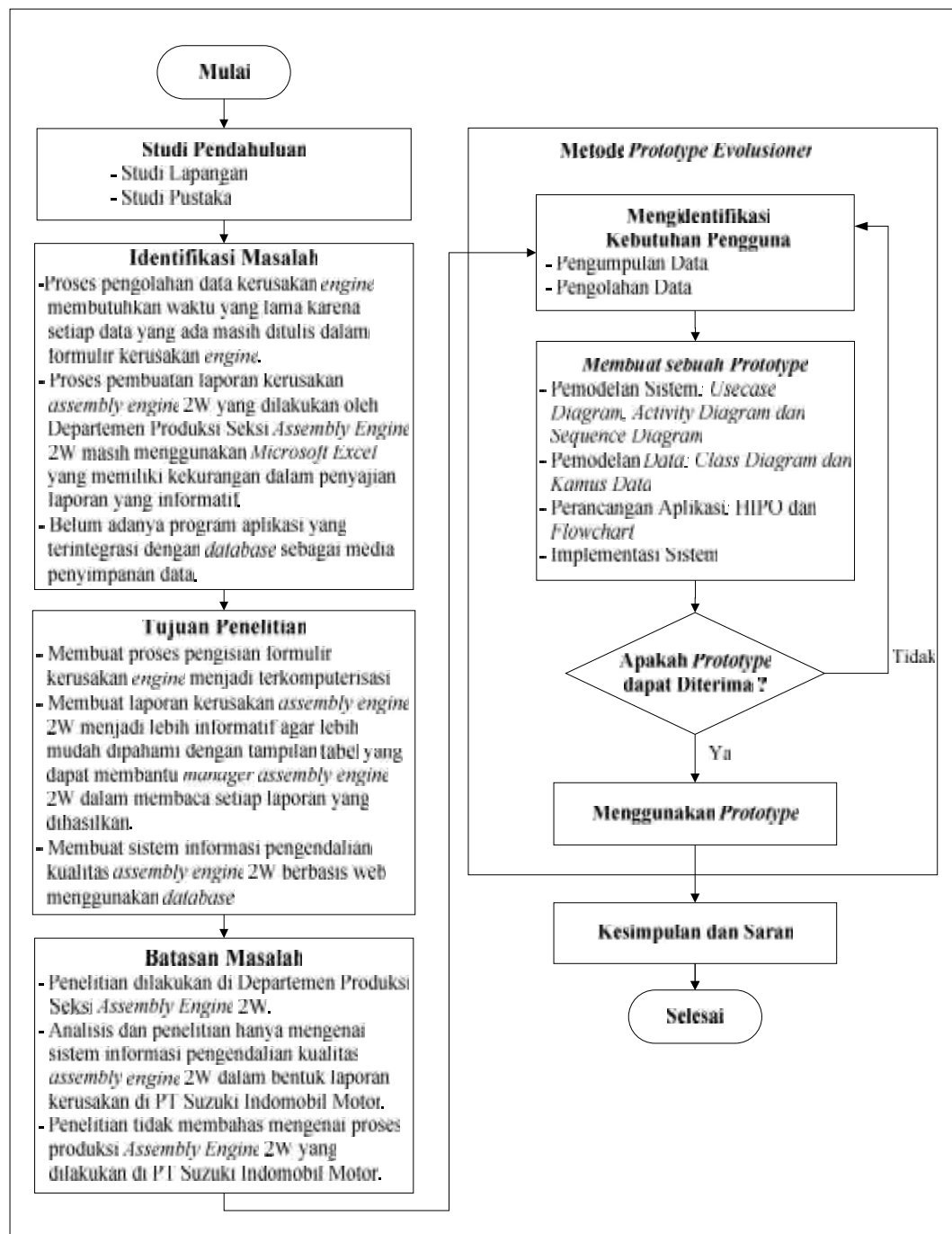
Dalam penelitian yang dibahas dalam tugas akhir ini, dilakukan langkah-langkah atau tahapan dalam penelitian yang ada pada Gambar III.1. Tahapan-tahapan dalam penelitian tersebut adalah sebagai berikut:

1. Melakukan studi pendahuluan. Melakukan studi pendahuluan dengan membaca penelitian yang telah dilakukan di Departemen Produksi Seksi *Assembly Engine 2W* sebelumnya dan literatur-literatur yang berkaitan dengan aktivitas pengendalian kualitas *assembly engine*.

2. Mengidentifikasi dan merumuskan permasalahan yang ditemukan saat dilakukan penelitian pada sistem pengendalian kualitas yang sedang berjalan di Departemen Produksi Seksi *Assembly Engine 2W* PT Suzuki Indomobil Motor.
3. Melakukan studi literatur untuk mendapatkan landasan teori yang dapat mendukung pemecahan masalah yang sedang diteliti. Teori yang didapatkan merupakan langkah awal agar permasalahan lebih dipahami dengan benar sesuai dengan kerangka berpikir ilmiah.
4. Menentukan tujuan penelitian. Menentukan tujuan dari dilakukannya penelitian yaitu merancang bangun sistem informasi pengendalian kualitas yang dapat menyajikan informasi kualitas *assembly engine 2W* dan laporan kerusakan *engine*.
5. Menentukan pendekatan pemrograman sistem, *tools* yang digunakan untuk pemodelan dan metodologi yang akan digunakan dalam pengembangan perangkat lunak. Pendekatan pemrograman sistem yang dipilih untuk rancang bangun sistem informasi pengendalian kualitas *assembly engine 2W* ini adalah pendekatan berorientasi objek dengan *tools* untuk pemodelannya adalah UML dan metodologi pengembangan perangkat lunak yang digunakan adalah *evolutionary prototype*.
6. Mengumpulkan data yang diperlukan dengan cara observasi langsung terhadap sistem yang berjalan, serta wawancara dengan pihak yang mempunyai hubungan dan pengetahuan terhadap sistem yang berjalan.
7. Melakukan analisis terhadap data yang telah diperoleh sebelumnya, sehingga dapat diketahui kebutuhan *user* terhadap aplikasi yang akan dibuat.
8. Membuat pemodelan sistem, pemodelan data dan desain sistem. Pemodelan sistem berbasis objek dengan menggunakan UML (*Unified Modeling Language*) berupa *use case diagram*, *activity diagram*, *sequence diagram*. Sedangkan pemodelan data dengan menggunakan *class diagram* dan kamus data. Untuk desain sistem menggunakan *flowchart* dan HIPO (*Hierarchy plus Input-Proses-Output*).

9. Membuat perancangan *database*, perancangan antarmuka dan perancangan arsitektur sistem.
10. Pengodean (*coding*) sistem yang telah dirancang dengan menggunakan pemrograman PHP dengan *tools* untuk penulisannya notepad++ dan perangkat lunak manajemen basis data MySQL.
11. Pada tahap ini, *prototype* yang diusulkan akan dilihat apakah sudah sesuai dengan keinginan *user* atau belum. Jika belum maka dilakukan analisis data kembali, tetapi apabila sesuai maka tahap selanjutnya adalah implementasi penggunaan *prototype* tersebut.
12. Sebelum melakukan implementasi sistem usulan, ada 2 (dua) hal yang harus dipenuhi yaitu perangkat keras yang akan digunakan yaitu *notebook* atau *Personal Computer (PC)* dan perangkat lunak yaitu *Apache Service* dan *MySQL Service*.
13. Setelah *prototype* sistem usulan diterima maka tahap selanjutnya adalah menarik kesimpulan dari penelitian yang telah dilakukan di PT Suzuki Indomobil Motor dan memberikan saran yang membangun bagi perusahaan dan bagi peneliti selanjutnya dalam penggunaan dan pengembangan sistem pengendalian kualitas *assembly engine 2W*.

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



Gambar III.1 Kerangka Penelitian

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Perusahaan

PT Suzuki Indomobil Motor merupakan sebuah perusahaan Penanaman Modal Asing (PMA) yang berdiri dengan kekuatan lima perusahaan. Perusahaan tersebut adalah sebagai berikut:

1. PT Indohero Steel & Engineering Co
2. PT Indomobil Utama
3. PT Suzuki Indonesia Manufacturing
4. PT Suzuki Engine Industry
5. PT First Chemical Industry

Lima perusahaan tersebut bergabung dengan persetujuan dari Presiden Republik Indonesia melalui surat pernyataan Ketua Badan Koordinasi Penanaman Modal (BKPM) nomor 05/I/PMA/90 tertanggal 1 Januari 1990 menjadi suatu bidang usaha industri komponen dan perakitan kendaraan bermotor merek Suzuki roda dua (sepeda motor) dan roda empat (mobil). Tanggal tersebut kini diperingati sebagai tanggal berdirinya PT Suzuki Indomobil Motor International.

PT Suzuki Indomobil Motor mempunyai beberapa *Overseas Sales Subsidiaries*. Lokasi yang terbesar berada di Perancis, Italia, Inggris, Amerika, Selandia Baru, Jerman, dan Kanada. PT Suzuki Indomobil Motor yang berada di Indonesia memiliki Lokasi kantor pusat di Wisma Indomobil di Jalan MT. Haryono, Kav. 8 Jakarta Timur. Kantor Pusat ini didukung oleh 314 karyawan. Lokasi pabriknya tersebar di beberapa tempat, antara lain di Pulogadung, Cakung, dan Tambun.

4.1.1 Visi dan Misi Perusahaan

Visi dari PT Suzuki Indomobil Motor adalah menjadi perusahaan otomotif yang terhandal dan terpercaya di dalam negeri dan misinya antara lain

mengembangkan seluruh sumber daya yang dimiliki secara berkesinambungan untuk meningkatkan profesionalisme bagi kepuasan pelanggan, memberikan kontribusi penuh bagi pengembangan usaha Indomobil dan memberikan komitmen dan nilai terbaik bagi seluruh pihak yang berkepentingan dengan memperhatikan kepentingan masyarakat.

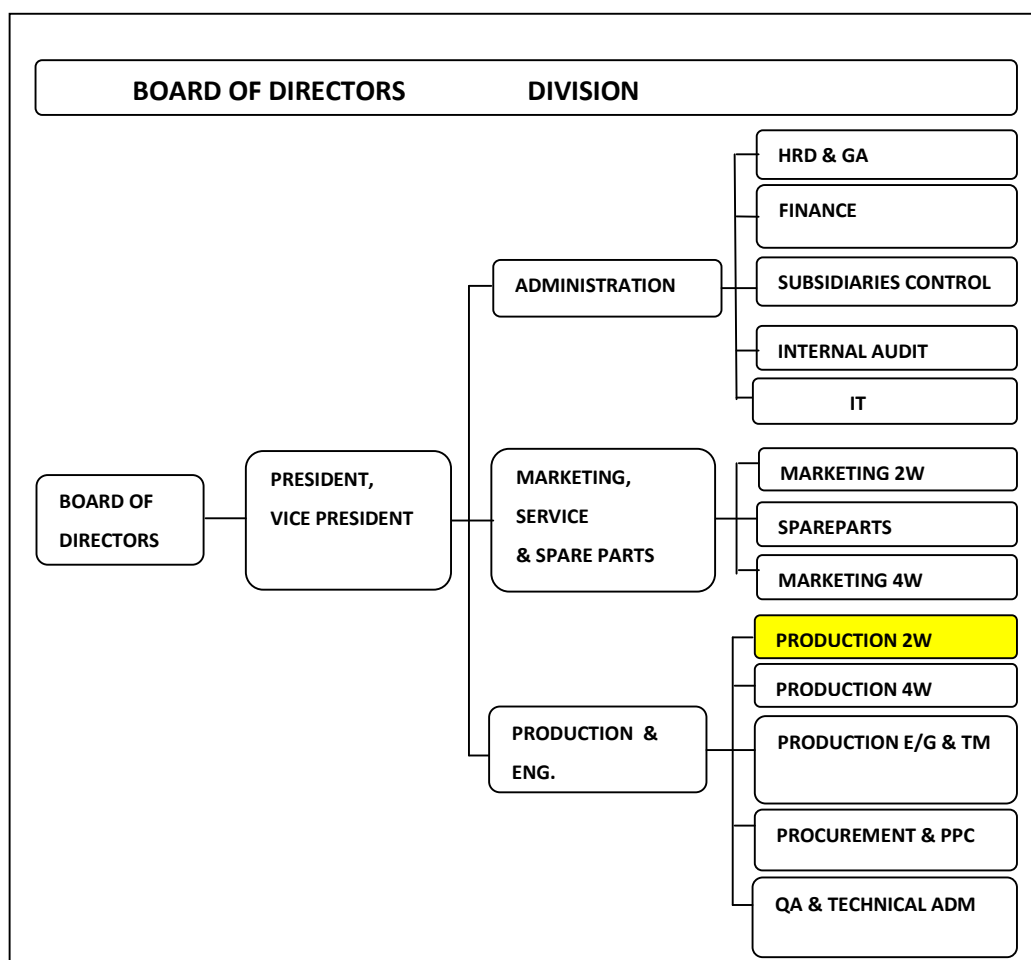
4.1.2 Program Perusahaan

Program-program yang dicanangkan oleh PT Suzuki Indomobil Motor untuk meningkatkan efisiensi dan efektifitas produksi antara lain:

1. **Gerakan Disiplin Suzuki (GDS)**
Merupakan program untuk meningkatkan kedisiplinan untuk seluruh pegawai PT Suzuki Indomobil Motor dengan menerapkan aturan-aturan yang tercantum dalam GDS yang wajib dipatuhi oleh seluruh pegawai PT Suzuki Indomobil Motor.
2. **Gugus Kendali Mutu (GKM)**
Merupakan standar yang menjadi patokan oleh seluruh bagian produksi untuk mengendalikan dan meningkatkan mutu di PT Suzuki Indomobil Motor.
3. **Keselamatan dan Kesehatan Kerja (K3)**
Merupakan panduan-panduan untuk menjaga keselamatan dan kesehatan seluruh pegawai PT Suzuki Indomobil Motor dalam melaksanakan tugas dan kewajibannya selama bekerja.
4. *Kaizen*
Kaizen merupakan teori atau ideologi orang Jepang yang diterapkan oleh PT Suzuki Indomobil Motor untuk terus meningkatkan dan mengembangkan kegiatan produksi untuk mencapai kesuksesan.
5. *Customer Satisfaction (CS)*
Merupakan program PT Suzuki Indomobil Motor untuk selalu mengutamakan kepuasan pelanggan dalam melayani pelanggan dan menjalin relasi bisnis.

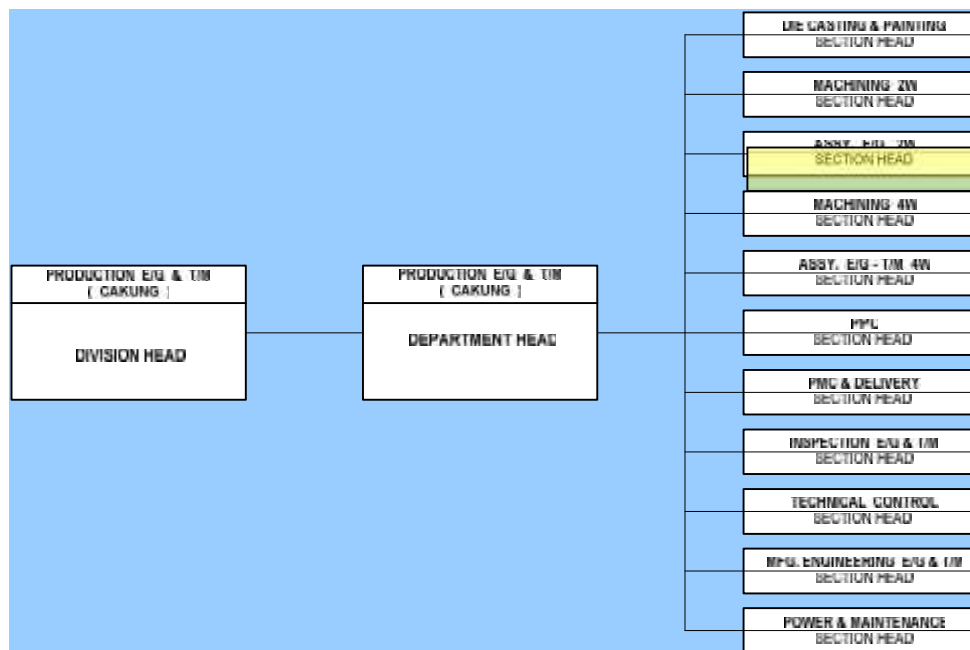
4.1.3 Struktur Organisasi Perusahaan

Struktur Organisasi adalah kerangka kerja untuk menunjukkan pembagian kerja dan mengkoordinasi aktivitas anggota suatu organisasi. Struktur organisasi dibentuk dengan maksud agar setiap anggota organisasi dapat bekerja secara efektif dan efisien. Struktur organisasi PT Suzuki Indomobil Motor diperlihatkan pada gambar IV.1.

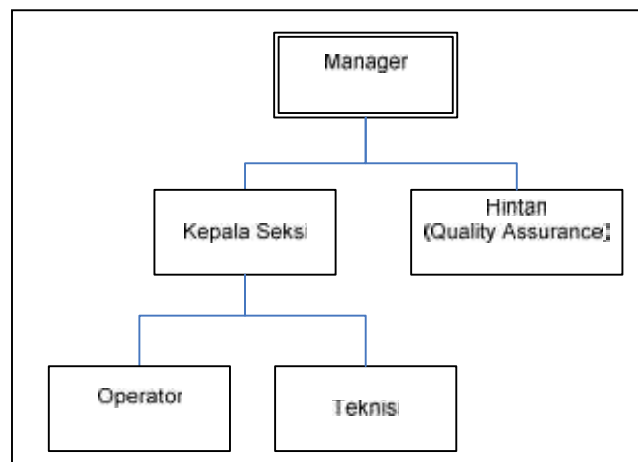


Gambar IV.1 Struktur Organisasi PT Suzuki Indomobil Motor
(Sumber: PT Suzuki Indomobil Motor, 2015)

Struktur organisasi PT Suzuki Indomobil Motor Cakung diperlihatkan pada gambar IV.2.



Gambar IV.2 Struktur Organisasi PT Suzuki Indomobil Motor Cakung
(Sumber: PT Suzuki Indomobil Motor, 2015)



Gambar IV.3 Struktur Organisasi Seksi *Assembly Engine 2W* PT Suzuki Indomobil Motor Cakung
(Sumber: PT Suzuki Indomobil Motor, 2015)

Berikut adalah tugas dan tanggung jawab dari Seksi *Assembly Engine 2W*, antara lain:

1. Merakit setiap komponen yang ada menjadi sebuah *engine 2W*.
2. Memastikan bahwa setiap *engine* yang dibuat tidak ada yang cacat.
3. Setiap terjadi kerusakan pada engine atau komponen harus ditulis dan dibuat bentuk laporannya.

4. Pelaksanaan kegiatan produksi harus selalu memperhatikan aspek kesehatan dan keselamatan kerja.

Berdasarkan tugas dan tanggung jawab Seksi *Assembly Engine 2W* yang telah dijelaskan diatas, berikut ini merupakan tugas dan tanggung jawab dari setiap aktor yang ikut berperan dalam proses produksi, antara lain:

1. *Manager*

Manager bertugas untuk menyusun anggaran produksi, bertanggung jawab terhadap kelancaran produksi, mencari solusi-solusi atas permasalahan yang terjadi, mengawasi pelaksanaan kegiatan produksi, mengelola setiap laporan yang diperoleh dari kepala seksi, dan melakukan evaluasi untuk menunjang proses produksi.

2. Kepala Seksi

Kepala Seksi bertugas untuk mengkoordinir dan mengawasi proses produksi agar dapat dicapai hasil produksi sesuai mutu yang ditetapkan, membuat laporan sesuai dengan sistem pelaporan yang berlaku, melaksanakan tugas-tugas lain yang diberikan oleh *manager* produksi.

3. *Hintan (Quality Assurance)*

Hintan (Quality Assurance) bertugas untuk melakukan inspeksi kualitas terhadap produk yang dihasilkan, membuat prosedur tetap sebagai petunjuk operasional, melakukan evaluasi, melakukan pemantauan terhadap penyimpangan dan turut serta dalam mengatasi permasalahan yang ada.

4. Teknisi

Teknisi bertugas untuk Memeriksa keadaan peralatan, mesin-mesin dan sistem, menganalisis informasi kerusakan pada produk yang sedang diproduksi dan memperbaiki kerusakan yang terjadi kemudian menginformasikan jenis perbaikannya.

4.1.4 Jaringan Pemasaran

Pemasaran merupakan salah satu kegiatan pokok yang dilakukan oleh PT Suzuki Indomobil Motor untuk mempertahankan kelangsungan perusahaannya,

berkembang, dan mendapatkan laba. Kegiatan pemasaran PT Suzuki Indomobil Motor harus juga memberikan kepuasan kepada konsumen.

PT Suzuki Indomobil Motor selalu fokus pada target pengembangan produk yang lebih mutakhir, inovasi teknologi, dan pengembangan yang berkelanjutan terhadap produknya. PT Suzuki Indomobil Motor juga dapat menjual sepeda motor *scooter*, *sport* dan *underbone* yang lebih banyak sesuai dengan permintaan pelanggan. Bisa juga melakukan pengembangan penjualan di tempat lain atau mencari tempat-tempat yang belum pernah di jangkau termasuk ke pedesaan sekalipun.

Di bawah ini merupakan daerah-daerah di Indonesia yang menjadi tempat pemasaran produk sepeda motor PT Suzuki Indomobil Motor diperlihatkan pada gambar IV.3.



Gambar IV.3 Peta Pemasaran Sepeda Motor PT Suzuki Indomobil Motor di Indonesia
(Sumber: PT Suzuki Indomobil Motor, 2015)

PT Suzuki Indomobil Motor juga melakukan pengembangan pemasaran ke seluruh dunia. PT Suzuki Indomobil Motor mempunyai beberapa *Overseas Sales Subsidiaries*. Lokasi yang terbesar berada di Perancis, Italia, Inggris, Amerika, Selandia Baru, Jerman, dan Kanada. PT Suzuki Indomobil Motor di seluruh dunia terdapat kurang lebih 36 pabrik manufaktur dengan jaringan distribusi kurang lebih 23 negara.

Berikut ini merupakan daerah-daerah di dunia yang menjadi tempat pemasaran produk sepeda motor PT Suzuki Indomobil Motor diperlihatkan pada gambar IV.5.



Gambar IV.4 Peta Pemasaran Sepeda Motor PT Suzuki Indomobil Motor di Dunia
(Sumber: PT Suzuki Indomobil Motor, 2015)

4.2 Proses *Assembly Engine*

Proses *assembly engine 2W* di PT Suzuki Indomobil Motor terdiri dari tiga bagian yaitu:

1. Proses *Assy Counter Line*

Proses *assy counter line* merupakan proses *assembly engine 2W* yang melibatkan bagian luar *engine* seperti *crankcase*, *crankshaft*, *cover*, *gear box*, dan *kick starter*. Hasil *assy counter line* diperlihatkan pada gambar IV.5.



Gambar IV.5 Hasil *Assy Counter Line* PT Suzuki Indomobil Motor
(Sumber: PT Suzuki Indomobil Motor, 2015)

2. Proses Assy Main Line 1

Proses *assy main line 1* merupakan proses *assembly engine 2W* yang melibatkan bagian dalam *engine*, seperti *v-belt drive*, *face assy drive*, *face assy driven*, *piston*, *camshaft*, *sensor engine*, dan *arm component valve rocker*. Hasil *assy main line 1* diperlihatkan pada gambar IV.6.



Gambar IV.6 Hasil Assy Main Line 1 PT Suzuki Indomobil Motor
(Sumber: PT Suzuki Indomobil Motor, 2015)

3. Proses Assy Main Line 2

Proses *assy main line 2* merupakan proses *assembly engine 2W* yang melibatkan bagian *finishing engine*, seperti *spark plug*. Hasil *assy main line 2* diperlihatkan pada gambar IV.7.



Gambar IV.7 Hasil Assy Main Line 2 PT Suzuki Indomobil Motor
(Sumber: PT Suzuki Indomobil Motor, 2015)

4.3 Sistem Informasi yang Berjalan

Sistem informasi yang berjalan di Departemen Produksi Seksi *Assembly Engine 2W* meliputi dua kegiatan yaitu kegiatan mengisi *form NG* dan membuat laporan kerusakan berdasarkan *form NG*. Teknisi akan menganalisa kerusakan yang terjadi pada *engine*. Selanjutnya, Teknisi akan mengisi *form NG* untuk mencatat setiap data kerusakan yang ditemukan. Kepala Seksi akan mengisi *form* laporan kerusakan berdasarkan data yang sudah tercatat pada *form NG*. *Form* laporan kerusakan yang sudah diisi akan diberikan kepada *Hintan (Quality Assurance)*. Semua hasil data kerusakan yang tercatat dalam *form* laporan kerusakan perhari diolah dengan menggunakan bantuan *Microsoft Excel* dan dikelompokkan berdasarkan jenis kerusakannya untuk membuat laporan kerusakan.

4.3.1 Arus Dokumen

Dokumen yang terlibat dalam sistem informasi pengendalian kualitas *assembly engine 2W* adalah:

1. *Form NG*

Form NG adalah *form* yang digunakan untuk mencatat setiap kerusakan *engine* yang ditemukan. Contoh dari *form NG* dapat dilihat di Gambar IV.8.

PT. SIM		Jenis Kerusakan :	
SUDAH DIPERBAIKI		Engine Bocor	
OK		Macam Perbaikan :	
		Dibersihkan	
No. Engine : 117641		No. Engine : 117641	
Tgl.	Diperbaiki oleh :	Tgl.	Perbaikan :
16/01/2015	Irwan Syah (Teanrasi)	16/01/2015	

Gambar IV.8 *Form NG*
(Sumber: PT Suzuki Indomobil Motor, 2015)

2. Form Laporan Kerusakan

Form laporan kerusakan adalah lembar kertas yang berisikan kumpulan data kerusakan yang diperoleh dari form NG yang sudah diisi. Contoh dari form laporan kerusakan dapat dilihat di Gambar IV.9.

LAPORAN KERUSAKAN
 BON PERMINTAAN BARANG

To: 25 Mei 2015

NO	TYPE	NO. BARANG	NAMA BARANG	LOKASI KERUSAKAN								QUANTITY	REMARK
				OH	EO	MC	PT	AD	EL	GU	GA		
1	REPAIR		Oil Filter	✓								1	Oil Filter
2	REPAIR		Oil Seal	✓								1	Oil Seal
3	REPAIR		Oil Pan	✓								1	Oil Pan
4	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
5	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
6	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
7	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
8	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
9	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
10	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
11	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
12	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
13	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
14	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
15	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
16	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
17	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
18	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
19	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
20	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
21	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
22	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
23	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
24	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
25	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
26	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
27	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
28	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
29	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
30	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
31	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
32	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
33	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
34	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
35	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
36	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
37	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
38	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
39	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
40	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
41	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
42	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
43	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
44	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
45	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt
46	REPAIR		Oil Pan Nut	✓								1	Oil Pan Nut
47	REPAIR		Oil Pan Washer	✓								1	Oil Pan Washer
48	REPAIR		Oil Pan Seal	✓								1	Oil Pan Seal
49	REPAIR		Oil Pan Gasket	✓								1	Oil Pan Gasket
50	REPAIR		Oil Pan Bolt	✓								1	Oil Pan Bolt

1. FUSION - Fusion Zerkless
 2. DEXUS - DEXUS Zerkless
 3. FUSION ZERKLESS (Khusus untuk mesin 2T)
 4. FUSION ZERKLESS (Khusus untuk mesin 4T)
 5. FUSION ZERKLESS (Khusus untuk mesin 4T)
 6. FUSION ZERKLESS (Khusus untuk mesin 4T)
 7. FUSION ZERKLESS (Khusus untuk mesin 4T)
 8. FUSION ZERKLESS (Khusus untuk mesin 4T)
 9. FUSION ZERKLESS (Khusus untuk mesin 4T)
 10. FUSION ZERKLESS (Khusus untuk mesin 4T)

Gambar IV.9 Form Laporan Kerusakan
(Sumber: PT Suzuki Indomobil Motor, 2015)

3. Laporan Kerusakan

Laporan kerusakan merupakan data form laporan kerusakan yang sudah diolah menggunakan *Microsoft Excel*. Laporan ini akan diberikan kepada *manager* Departemen Produksi Seksi *Assembly Engine 2W* dalam bentuk lembar laporan. Contoh dari laporan kerusakan dapat dilihat pada Gambar IV.10 dan IV.11.

dilakukan oleh bagian *Hintan*. Contoh dari laporan evaluasi dapat dilihat pada Gambar IV.12.

ASSY 2W Evaluasi	
Januari 2015	
Plant :	CAKUNG
Line 1A :	Mesalah terbesar yang masih muncul adalah Engine bocor.
Line 2B :	Mesalah terbesar yang masih muncul adalah Engine bocor.
Line 2A :	Tidak ada aktivitas produksi.
Line 2E :	Tidak ada aktivitas produksi.
Line 3A :	Tidak ada aktivitas produksi.
Line 3F :	Mesalah terbesar yang masih muncul adalah Cover Clutch cacat.
Line 4A :	Tidak ada masalah yang muncul.
Line 1E :	Tidak ada masalah yang muncul.

Gambar IV.12 Laporan Evaluasi
(Sumber: PT Suzuki Indomobil Motor, 2015)

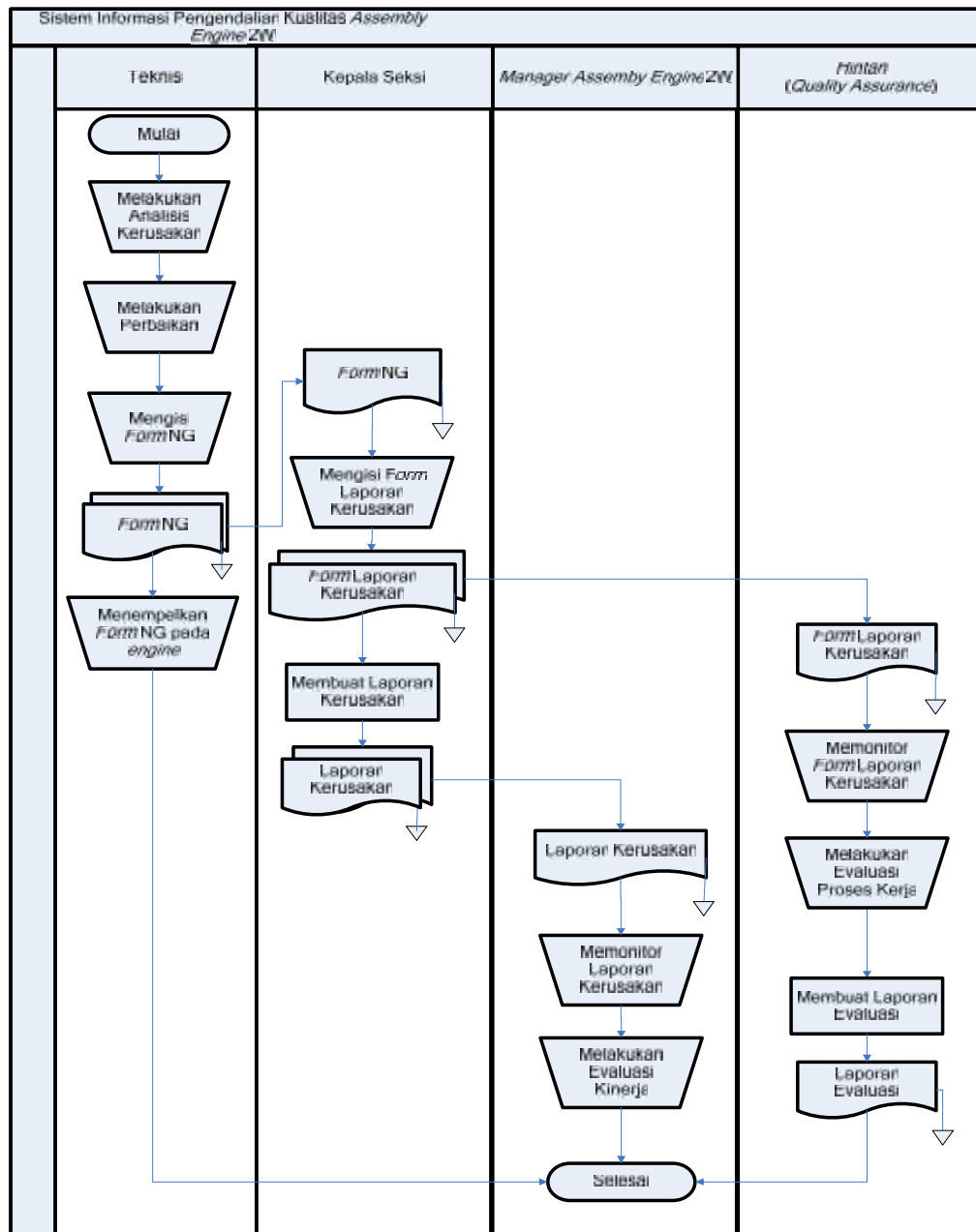
4.3.2 Arus Proses

Proses yang terlibat dalam sistem informasi pengendalian kualitas *assembly engine 2W* saat ini antara lain:

1. Teknisi melakukan analisis kerusakan yang terjadi.
2. Teknisi melakukan perbaikan.
3. Teknisi mengisi *form* NG berdasarkan hasil analisis kerusakan yang terjadi.
4. Teknisi memberikan *form* NG yang sudah diisi kepada Kepala Seksi.
5. Teknisi menempelkan *form* NG pada *engine* yang rusak.
6. Kepala Seksi mengisi *form* laporan kerusakan berdasarkan data yang terdapat pada *form* NG yang sudah diisi.
7. *Form* Laporan kerusakan yang sudah diisi diberikan kepada *Hintan* (*Quality Assurance*).
8. *Hintan* (*Quality Assurance*) memonitor *form* laporan kerusakan yang sudah diisi.
9. *Hintan* (*Quality Assurance*) melakukan evaluasi proses kerja berdasarkan *form* laporan kerusakan yang sudah diisi.

10. *Hintan (Quality Assurance)* membuat laporan hasil evaluasi.
11. Kepala Seksi membuat laporan kerusakan *engine* dengan *Microsoft Excel*.
12. *Manager Assembly Engine 2W* memonitor laporan kerusakan *engine* yang dibuat oleh Kepala Seksi.
13. *Manager Assembly Engine 2W* melakukan evaluasi kinerja.

Berikut diperlihatkan pada gambar IV.13 adalah *flowmap* sistem informasi pengendalian kualitas yang sedang berjalan pada Departemen Produksi Seksi *Assembly Engine 2W* PT Suzuki Indomobil Motor:

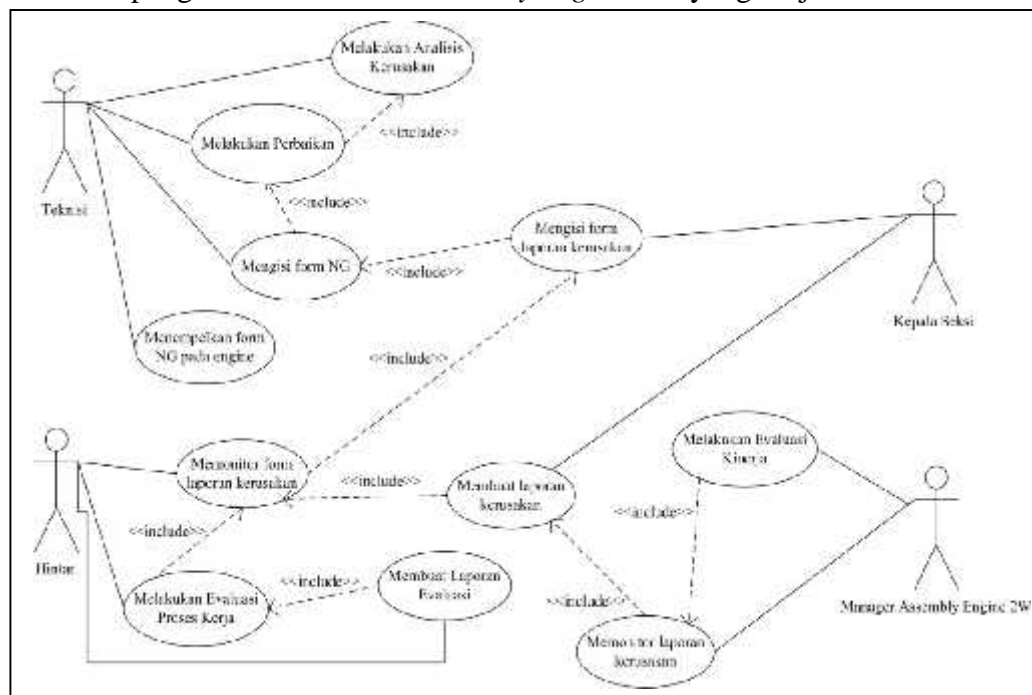


Gambar IV.13 Flowmap Sistem yang berjalan
(Sumber: PT Suzuki Indomobil Motor, 2015)

4.3.3 Diagram Use Case Sistem Informasi yang Berjalan

Guna mempermudah dalam menganalisis sistem informasi yang berjalan dalam Departemen Produksi Seksi *Assembly Engine 2W* PT Suzuki Indomobil Motor, digunakan *tools* UML (*Unified Modeling Language*) dengan diagram yang digunakan adalah diagram *Use case* untuk memodelkan sistem informasi tersebut.

Sistem informasi pengendalian kualitas *assembly engine 2W* PT Suzuki Indomobil Motor memiliki beberapa aktor yang saling terkait, yaitu Teknisi, Kepala Seksi, *Hintan (Quality Assurance)* dan *Manager Assembly Engine 2W*. Setiap aktor memiliki kepentingan-kepentingan yang berkaitan dengan sistem tersebut. Gambar IV.14 menjelaskan tentang penggambaran sistem informasi pengendalian kualitas *assembly engine 2W* yang berjalan.



Gambar IV.14 *Use Case Diagram* Sistem Informasi pengendalian kualitas *assembly engine 2W* yang Berjalan
(Sumber: Pengolahan Data, 2015)

Penjelasan *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* yang berjalan pada Gambar IV.14, dapat dijelaskan sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* dapat dilihat pada tabel IV.1 berikut.

Tabel IV.1 Definisi Aktor *Use Case Diagram*

No	Aktor	Deskripsi
1	Teknisi	Orang yang bertugas menganalisis kerusakan yang terjadi pada <i>engine</i> , melakukan perbaikan, mengisi <i>form</i> NG, dan menempelkan <i>form</i> NG pada <i>engine</i> yang rusak.
2	Kepala Seksi	Orang yang bertugas mengisi <i>form</i> laporan kerusakan dan membuat laporan kerusakan.
3	<i>Hintan (Quality Assurance)</i>	Bagian yang memonitor <i>form</i> laporan kerusakan <i>engine</i> .
4	<i>Manager Assembly Engine 2W</i>	Orang yang memonitor laporan kerusakan <i>engine</i> .

(Sumber: Pengolahan Data , 2015)

2. Definisi *Use Case*

Pendefinisian *Use case* pada *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* yang berjalan dijelaskan pada tabel IV.2.

Tabel IV.2 Definisi *Use case Diagram* Sistem Berjalan

No	<i>Use Case</i>	Deskripsi
1	Melakukan Analisis Kerusakan	<i>Use case</i> ini merupakan <i>use case</i> proses melakukan analisis kerusakan yang dilakukan oleh teknisi.
2	Melakukan Perbaikan	<i>Use case</i> ini merupakan <i>use case</i> proses melakukan perbaikan yang dilakukan oleh teknisi.
3	Mengisi <i>Form</i> NG	<i>Use case</i> ini merupakan <i>use case</i> proses mengisi <i>form</i> NG. Proses ini dilakukan oleh teknisi.
4	Menempelkan <i>Form</i> NG pada <i>Engine</i>	<i>Use case</i> ini merupakan <i>use case</i> proses menempelkan <i>form</i> NG pada <i>engine</i> untuk menandakan bahwa <i>engine</i> tersebut rusak. Proses ini dilakukan oleh Teknisi.

Tabel IV.2 Definisi *Use case Diagram* Sistem Berjalan (Lanjutan)

5	Mengisi Form Laporan Kerusakan	<i>Use case</i> ini merupakan <i>use case</i> proses mengisi form laporan kerusakan. Proses ini dilakukan oleh Kepala Seksi.
6	Memonitor Form Laporan Kerusakan	<i>Use case</i> ini merupakan <i>use case</i> proses memonitor form laporan kerusakan. Proses ini dilakukan oleh Hintan.
7	Melakukan Evaluasi Proses Kerja	<i>Use case</i> ini merupakan <i>use case</i> proses melakukan evaluasi proses kerja untuk mengurangi terjadinya kerusakan pada proses produksi. Proses ini dilakukan oleh Hintan.
8	Membuat Laporan Evaluasi	<i>Use case</i> ini merupakan <i>use case</i> proses membuat laporan evaluasi. Proses ini dilakukan oleh Hintan.
9	Membuat Laporan Kerusakan	<i>Use case</i> ini merupakan <i>use case</i> proses membuat laporan kerusakan. Proses ini dilakukan oleh Kepala Seksi.
10	Memonitor Laporan Kerusakan	<i>Use case</i> ini merupakan <i>use case</i> proses memonitor laporan kerusakan. Proses ini dilakukan oleh <i>Manager Assembly Engine 2W</i> .
11	Melakukan Evaluasi Kinerja	<i>Use case</i> ini merupakan <i>use case</i> proses evaluasi kinerja untuk mengurangi terjadinya kesalahan. Proses ini dilakukan oleh <i>Manager Assembly Engine 2W</i> .

(Sumber: Pengolahan Data, 2015)

3. Skenario *Use Case*

Skenario masing-masing *use case* pada *use case diagram* sistem yang berjalan dijelaskan pada tabel IV.3 sampai tabel IV.12 berikut ini.

Tabel IV.3 Skenario *Use Case* Melakukan Analisis Kerusakan

<i>Use Case Name</i>	Melakukan analisis kerusakan
<i>Primary Actor</i>	Teknisi
<i>Brief Description</i>	Proses melakukan analisis kerusakan yang terjadi.
<i>Normal Flow of Events</i>	Teknisi melakukan analisis kerusakan yang terjadi.

(Sumber: Pengolahan Data , 2015)

Tabel IV.4 Skenario *Use Case* Melakukan Perbaikan

<i>Use Case Name</i>	Melakukan perbaikan
<i>Primary Actor</i>	Teknisi
<i>Brief Description</i>	Proses melakukan perbaikan pada <i>engine</i> .
<i>Normal Flow of Events</i>	Teknisi melakukan perbaikan pada <i>engine</i> .

(Sumber: Pengolahan Data , 2015)

Tabel IV.5 Skenario *Use Case* Mengisi *Form NG*

<i>Use Case Name</i>	Mengisi <i>form NG</i>
<i>Primary Actor</i>	Teknisi
<i>Brief Description</i>	Proses pengisian <i>form NG</i> .
<i>Relationship</i>	<i>Include</i> : melakukan perbaikan
<i>Normal Flow of Events</i>	Teknisi mengisi <i>form NG</i> untuk diserahkan kepada Kepala Seksi

(Sumber: Pengolahan Data, 2015)

Tabel IV.6 Skenario *Use Case* Menempelkan *Form NG* pada *Engine*

<i>Use Case Name</i>	Menempelkan <i>Form NG</i> pada <i>Engine</i>
<i>Primary Actor</i>	Teknisi
<i>Brief Description</i>	Proses menempelkan <i>form NG</i> pada <i>engine</i> .
<i>Relationship</i>	-
<i>Normal Flow of Events</i>	Teknisi menempelkan <i>form NG</i> pada <i>engine</i> .

(Sumber: Pengolahan Data , 2015)

Tabel IV.7 Skenario *Use Case* Mengisi *Form Laporan Kerusakan*

<i>Use Case Name</i>	Mengisi <i>Form Laporan Kerusakan</i>
<i>Primary Actor</i>	Kepala Seksi
<i>Brief Description</i>	Proses mengisi <i>form</i> laporan kerusakan berdasarkan <i>form NG</i> yang sudah diisi oleh Teknisi.
<i>Relationship</i>	<i>Include</i> : mengisi <i>form NG</i>
<i>Normal Flow of Events</i>	Kepala Seksi mengisi <i>form</i> laporan kerusakan berdasarkan <i>form NG</i> yang sudah diisi oleh Teknisi.

(Sumber: Pengolahan Data , 2015)

Tabel IV.8 Skenario *Use Case* Memonitor *Form* Laporan Kerusakan

<i>Use Case Name</i>	Memonitor <i>Form</i> Laporan Kerusakan
<i>Primary Actor</i>	<i>Hintan (Quality Assurance)</i>
<i>Brief Description</i>	Proses memonitor <i>form</i> laporan kerusakan.
<i>Relationship</i>	<i>Include</i> : mengisi <i>form</i> laporan kerusakan
<i>Normal Flow of Events</i>	<i>Hintan (Quality Assurance)</i> memonitor <i>form</i> laporan kerusakan yang telah diisi oleh Kepala Seksi.

(Sumber: Pengolahan Data, 2015)

Tabel IV.9 Skenario *Use Case* Melakukan Evaluasi Proses Kerja

<i>Use Case Name</i>	Melakukan evaluasi proses kerja
<i>Primary Actor</i>	<i>Hintan (Quality Assurance)</i>
<i>Brief Description</i>	Proses melakukan evaluasi proses kerja.
<i>Relationship</i>	<i>Include</i> : memonitor <i>form</i> laporan kerusakan.
<i>Normal Flow of Events</i>	<i>Hintan (Quality Assurance)</i> melakukan evaluasi proses kerja.

(Sumber: Pengolahan Data, 2015)

Tabel IV.10 Skenario *Use Case* Membuat Laporan Evaluasi

<i>Use Case Name</i>	Membuat Laporan Evaluasi
<i>Primary Actor</i>	<i>Hintan (Quality Assurance)</i>
<i>Brief Description</i>	Proses pembuatan laporan evaluasi.
<i>Relationship</i>	<i>Include</i> : Melakukan Evaluasi Proses Kerja
<i>Normal Flow of Events</i>	<i>Hintan (Quality Assurance)</i> membuat laporan evaluasi.

(Sumber: Pengolahan Data, 2015)

Tabel IV.11 Skenario *Use Case* Membuat Laporan Kerusakan

<i>Use Case Name</i>	Membuat Laporan Kerusakan
<i>Primary Actor</i>	Kepala Seksi
<i>Brief Description</i>	Proses pembuatan laporan kerusakan.
<i>Relationship</i>	<i>Include</i> : Memonitor <i>form</i> laporan kerusakan

Tabel IV.11 Skenario *Use Case* Membuat Laporan Kerusakan (Lanjutan)

<i>Normal Flow of Events</i>	Kepala Seksi membuat laporan kerusakan yang akan diserahkan kepada <i>Manager Assembly Engine 2W</i> .
------------------------------	--

(Sumber: Pengolahan Data, 2015)

Tabel IV.12 Skenario *Use Case* Memonitor Laporan Kerusakan

<i>Use Case Name</i>	Memonitor Laporan Kerusakan
<i>Primary Actor</i>	<i>Manager Assembly Engine 2W</i>
<i>Brief Description</i>	Proses memonitor laporan kerusakan yang telah dibuat oleh Kepala Seksi.
<i>Relationship</i>	<i>Include</i> : membuat laporan kerusakan
<i>Normal Flow of Events</i>	<i>Manager Assembly Engine 2W</i> memonitor laporan kerusakan.

(Sumber: Pengolahan Data, 2015)

Tabel IV.13 Skenario *Use Case* Melakukan Evaluasi Kinerja

<i>Use Case Name</i>	Melakukan evaluasi proses kinerja
<i>Primary Actor</i>	<i>Manager Assembly Engine 2W</i>
<i>Brief Description</i>	Proses melakukan evaluasi proses kinerja.
<i>Relationship</i>	<i>Include</i> : memonitor laporan kerusakan.
<i>Normal Flow of Events</i>	<i>Manager Assembly Engine 2W</i> melakukan evaluasi proses kinerja.

(Sumber: Pengolahan Data, 2015)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Sistem Informasi Berjalan

Berdasarkan hasil pengumpulan dan pengolahan data yang dilakukan pada sistem informasi yang sedang berjalan di Departemen Produksi Seksi *Assembly Engine 2W* PT Suzuki Indomobil Motor, dapat disimpulkan bahwa Departemen Produksi Seksi *Assembly Engine 2W* tidak memiliki sistem informasi terkomputerisasi yang khusus untuk menangani kebutuhan informasi pengendalian kualitas *assembly engine 2W*.

5.2 Analisis Kebutuhan Sistem

Berdasarkan hasil analisis sistem informasi yang berjalan, maka diajukan sistem informasi usulan yang khusus untuk menangani kebutuhan pengendalian kualitas di Departemen Produksi Seksi *Assembly Engine 2W*. Berikut ini adalah daftar kebutuhan sistem informasi pengendalian kualitas di Departemen Produksi Seksi *Assembly Engine 2W* PT Suzuki Indomobil Motor yang diusulkan dalam Tabel V.1.

Tabel V.1 Kebutuhan Sistem Informasi Usulan

<i>Project Name</i>	Sistem Informasi Pengendalian Kualitas <i>Assembly Engine 2W</i>
<i>Business Recruitment</i>	Sistem Informasi yang khusus untuk menangani Hasil Analisis kualitas di Departemen Produksi Seksi <i>Assembly Engine 2W</i> .
<i>Business Needs</i>	<ol style="list-style-type: none">1. Terwujudnya sebuah aplikasi yang dapat membantu proses Hasil Analisis kerusakan pada komponen dan <i>engine</i>.2. Terwujudnya sebuah aplikasi yang dapat memberikan informasi terkait dengan laporan kerusakan komponen dan <i>engine</i>.
<i>Business Value</i>	<ol style="list-style-type: none">1. Mengganti penggunaan <i>form</i> laporan kerusakan menjadi terkomputerisasi.

Tabel V.1 Kebutuhan Sistem Informasi Usulan (Lanjutan)

<i>Business Value</i>	<p>2. Data kerusakan komponen dan <i>engine</i> dapat tersimpan permanen dalam basis data berbasis komputer.</p> <p>3. Mempermudah dalam pengolahan laporan kerusakan.</p>
-----------------------	--

(Sumber: Hasil Analisis, 2015)

5.3 Analisis Kebutuhan Rinci Sistem

Analisis kebutuhan rinci sistem untuk sistem informasi pengendalian kualitas di Departemen Produksi Seksi *Assembly Engine 2W* PT Suzuki Indomobil Motor dijelaskan pada Tabel V.2.

Tabel V.2 Kebutuhan Rinci Sistem

No.	Kebutuhan Rinci Sistem	Uraian
1	Mengelola Data <i>Master</i>	<p><i>Input:</i> Data <i>master</i> yang terdiri dari data komponen, data Jenis Kerusakan, data lokasi asal masalah komponen, data tipe <i>engine</i>, data kategori lini produksi dan data petugas.</p> <p><i>Proses:</i> Mengelola data <i>master</i></p> <ul style="list-style-type: none"> – Kepala Seksi dapat menambah, mengubah, menghapus, dan mencari data <i>master</i>. <p><i>Output:</i> Informasi data <i>master</i>.</p>
2	Mengelola Data <i>User</i>	<p><i>Input:</i> Data <i>User</i></p> <p><i>Proses:</i> Mengelola data <i>user</i>.</p> <ul style="list-style-type: none"> – Kepala Seksi memilih data <i>user</i>. – Kepala Seksi dapat menambah, mengubah, menghapus dan mencari data. – <i>Output:</i> Informasi <i>user</i>.
3	Mengolah Laporan Kerusakan	<p><i>Input:</i> Tanggal laporan yang ingin dikelola.</p> <p><i>Proses:</i> Mengelola laporan kerusakan.</p> <ul style="list-style-type: none"> – Kepala Seksi memilih data laporan kerusakan.

Tabel V.2 Kebutuhan Rinci Sistem (Lanjutan)

No.	Kebutuhan Rinci Sistem	Uraian
3	Mengolah Laporan Kerusakan	<ul style="list-style-type: none"> – Kepala Seksi dapat menambah, mengubah, menghapus dan mencari data laporan kerusakan. <p><i>Output:</i> Data Laporan kerusakan.</p>
4	<i>Input Kerusakan Before Process</i>	<p><i>Input:</i> Data laporan kerusakan <i>before process</i>.</p> <p>Proses: <i>Input</i> laporan kerusakan <i>before process</i>.</p> <ul style="list-style-type: none"> – Teknisi mengisi <i>form</i> laporan kerusakan <i>before process</i>. – Teknisi menyimpan data yang sudah diisi di dalam <i>form</i> laporan kerusakan <i>before process</i>. <p><i>Output:</i> Laporan kerusakan <i>before process</i></p>
5	<i>Input Kerusakan In Process</i>	<p><i>Input:</i> Data laporan kerusakan <i>in process</i>.</p> <p>Proses: <i>Input</i> laporan kerusakan <i>in process</i></p> <ul style="list-style-type: none"> – Teknisi mengisi <i>form</i> laporan kerusakan <i>in process engine</i> rusak. - Teknisi menyimpan data yang sudah diisi. Setelah dilakukan perbaikan, Teknisi mengubah status rusak menjadi OK di <i>form in process engine</i> diperbaiki. – <i>Output:</i> Laporan kerusakan.
6	Mencetak Laporan Kerusakan	<p><i>Input:</i> Tanggal laporan yang ingin dicetak.</p> <p>Proses: Mencetak laporan kerusakan yaitu:</p> <ul style="list-style-type: none"> – Kepala Seksi mencari data laporan kerusakan berdasarkan tanggal yang ditentukan. – Kepala Seksi mencetak laporan. – <i>Output:</i> Laporan kerusakan.

Tabel V.2 Kebutuhan Rinci Sistem (Lanjutan)

No.	Kebutuhan Rinci Sistem	Uraian
7	Melihat Laporan Kerusakan	<p><i>Input:</i> Tanggal laporan yang ingin dilihat.</p> <p>Proses Melihat laporan kerusakan</p> <ul style="list-style-type: none"> – Kepala Seksi, <i>Hintan</i> (<i>Quality Assurance</i>), dan <i>Manager Assembly Engine 2W</i> mencari data laporan kerusakan berdasarkan tanggal yang ditentukan. <p><i>Output:</i> Laporan kerusakan.</p>
8	<i>Input</i> Hasil Evaluasi	<p><i>Input:</i> Data hasil evaluasi.</p> <p>Proses: <i>Input</i> hasil evaluasi</p> <ul style="list-style-type: none"> – <i>Hintan</i> mengisi <i>form</i> hasil evaluasi. - <i>Hintan</i> menyimpan data yang sudah diisi. – <i>Output:</i> Laporan hasil evaluasi.
9	Melihat dan Mencetak Laporan Evaluasi	<p><i>Input:</i> Bulan dan tahun laporan yang ingin dicetak.</p> <p>Proses: Melihat dan mencetak laporan evaluasi yaitu:</p> <ul style="list-style-type: none"> – <i>Hintan</i> mencari data laporan evaluasi berdasarkan bulan dan tahun yang ditentukan. – <i>Hintan</i> melihat laporan kemudian mencetak laporan. – <i>Output:</i> Laporan evaluasi.

(Sumber: Hasil Analisis, 2015)

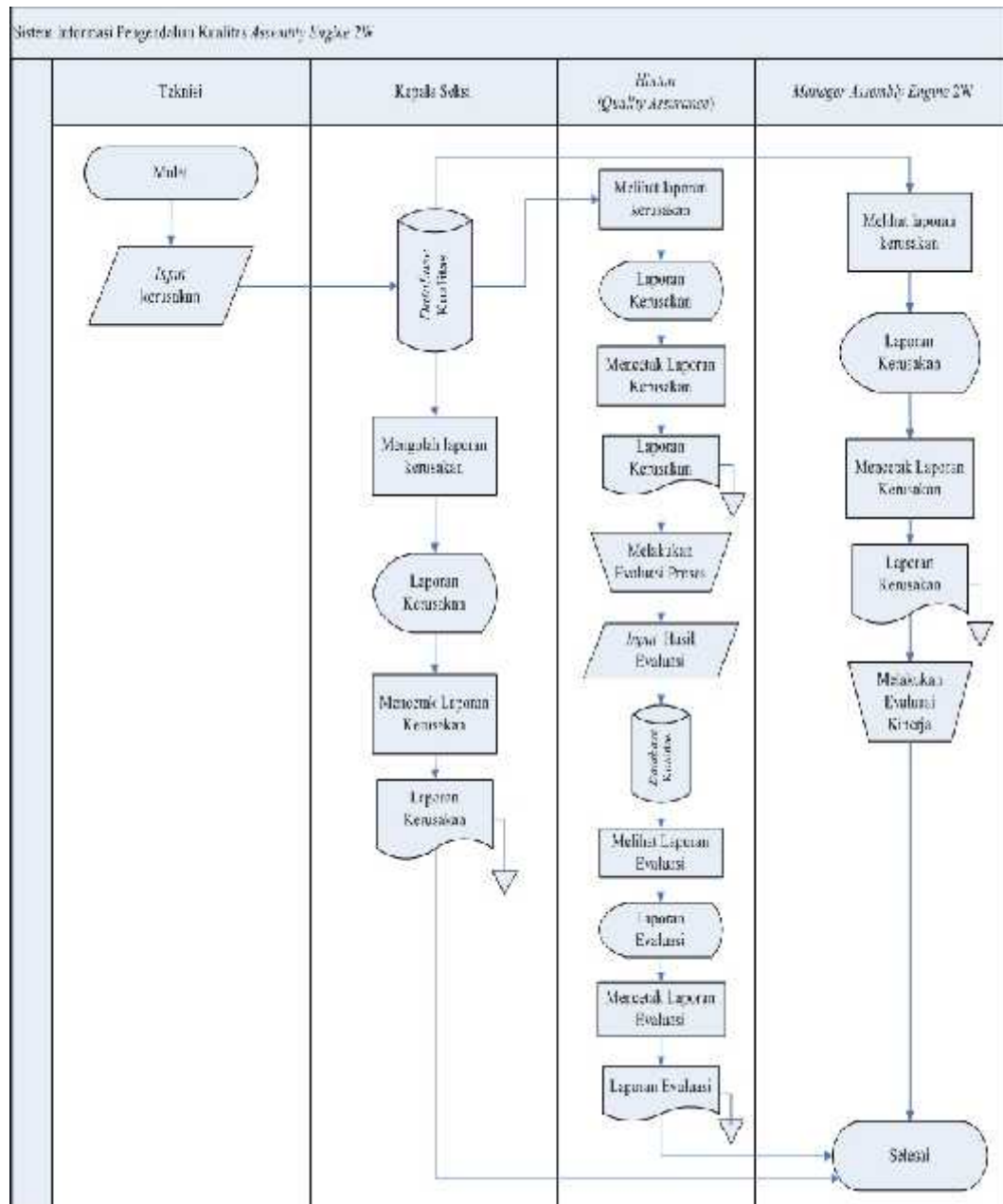
5.4 Perancangan *Flowmap* Sistem Usulan

Flowmap sistem informasi pengendalian kualitas *assembly engine 2W* yang diusulkan sebagai berikut:

- Teknisi melakukan *input* data laporan kerusakan.
- Data yang telah diinput masuk ke dalam *database* kualitas.

- Kepala Seksi mengolah laporan kerusakan dan mencetak laporan kerusakan untuk diarsipkan.
- *Hintan (Quality Assurance)* melihat laporan kerusakan secara keseluruhan berdasarkan data laporan yang telah diinput oleh Kepala Seksi.
- *Hintan (Quality Assurance)* melakukan evaluasi proses kerja berdasarkan laporan kerusakan yang sudah dibuat.
- *Hintan (Quality Assurance)* memasukan data hasil evaluasi kemudian membuat laporan evaluasi dan mencetaknya.
- *Manager Assembly Engine 2W* melihat laporan kerusakan yang telah dibuat.
- *Manager Assembly Engine 2W* melakukan evaluasi kinerja Departemen Produksi seksi *Assembly Engine 2W* berdasarkan dari laporan kerusakan yang telah dibuat oleh Kepala Seksi.

Flowmap sistem informasi pengendalian kualitas *assembly engine 2W* usulan digambarkan dalam Gambar V.1.



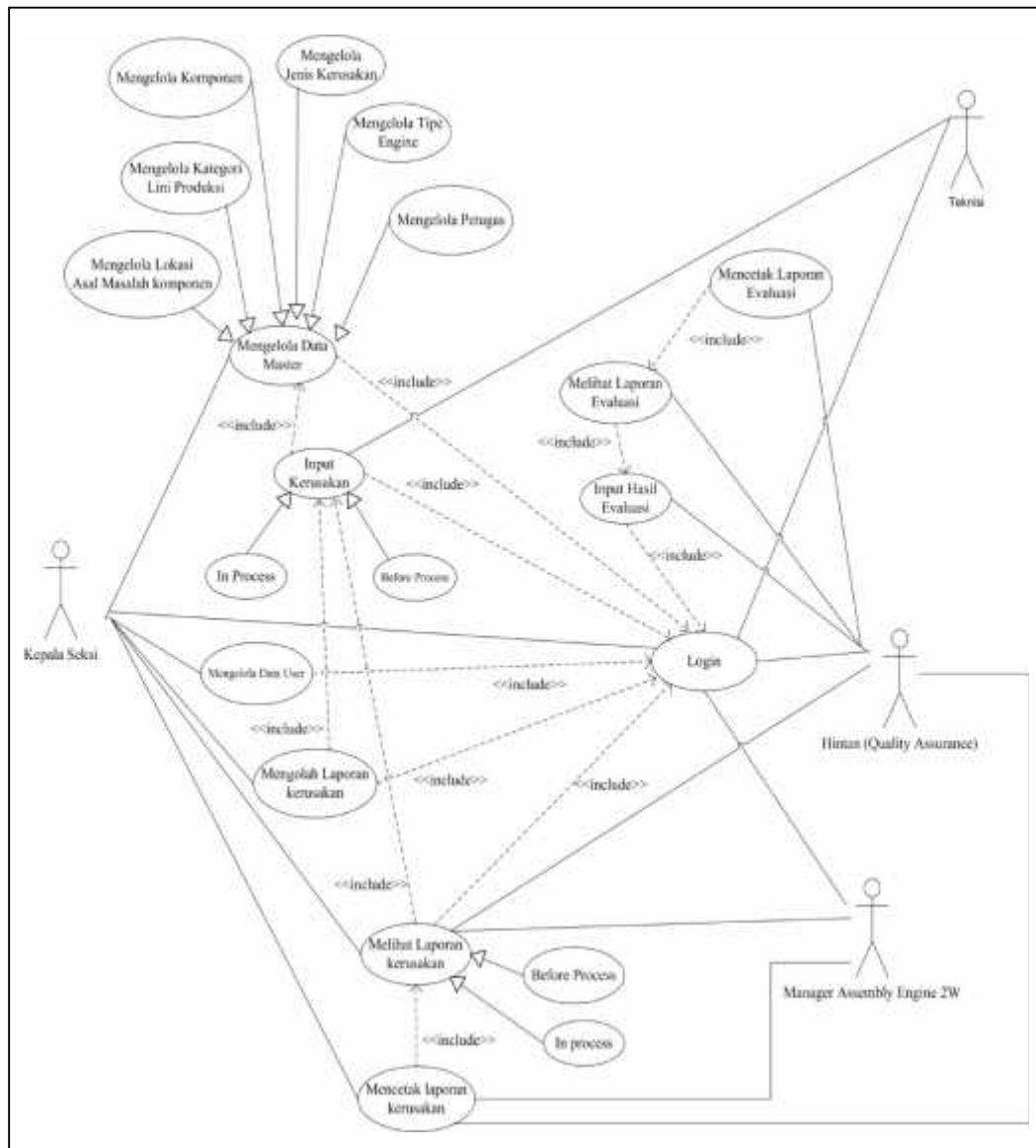
Gambar V.1 *Flowmap* Sistem Informasi Pengendalian Kualitas *Assembly Engine 2W*
(Sumber: Hasil Analisis, 2015)

5.5 Pemodelan Sistem Usulan

Pemodelan sistem informasi pengendalian kualitas *assembly engine 2W* menggunakan *tools* pemodelan sistem UML (*unified modeling language*) yang meliputi pembuatan *use case diagram*, *activity diagram*, *sequence diagram*. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun.

5.5.1 Use Case Diagram

Berdasarkan analisis yang telah dilakukan, maka *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* usulan dapat dilihat pada Gambar V.2 sebagai berikut ini:



Gambar V.2 Use Case Diagram Sistem Informasi Pengendalian Kualitas *Assembly Engine 2W* Usulan
(Sumber: Hasil Analisis, 2015)

Penjelasan *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* usulan dijelaskan sebagai berikut ini:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* usulan dijelaskan pada Tabel V.3.

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

No	Aktor	Deskripsi
1	Teknisi	Teknisi bertugas dalam melakukan <i>input</i> kerusakan.
2	Kepala Seksi	Kepala Seksi bertugas dalam mengelola data <i>master</i> yaitu data komponen, data Jenis Kerusakan, data lokasi asal masalah komponen, data tipe <i>engine</i> , data kategori lini produksi dan data petugas. Kepala Seksi juga bertugas dalam mengolah laporan kerusakan.
3	<i>Hintan (Quality Assurance)</i>	<i>Hintan (Quality Assurance)</i> memiliki hak akses untuk input hasil evaluasi, melihat laporan kerusakan dan melihat laporan evaluasi.
4	<i>Manager Assembly Engine 2W</i>	<i>Manager Assembly Engine 2W</i> memiliki hak akses untuk melihat laporan kerusakan.

(Sumber: Hasil Analisis, 2015)

2. Definisi *Use case*

Pendefinisian *use case* pada *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* usulan dijelaskan pada Tabel V.4.

Tabel V.4 Definisi *Use Case Diagram* Sistem Informasi Usulan

No	<i>Use Case</i>	Deskripsi
1	<i>Login</i>	<i>Use case</i> ini merupakan <i>use case</i> proses <i>login</i> . Proses ini mengharuskan <i>user</i> untuk melakukan <i>login</i> terlebih dahulu sebelum menggunakan aplikasi.
2	Mengelola Data <i>User</i>	<i>Use case</i> ini menerangkan proses mengelola data <i>user</i> yang dilakukan oleh Kepala Seksi.

Tabel V.4 Definisi *Use Case Diagram* Sistem Informasi Usulan (Lanjutan)

No	<i>Use Case</i>	Deskripsi
3	Mengelola Data <i>Master</i>	<i>Use case</i> ini berfungsi untuk mengelola data <i>master</i> yang terdiri dari data komponen, data Jenis Kerusakan, data lokasi asal masalah komponen, data tipe <i>engine</i> , data kategori lini produksi dan data petugas.
4	<i>Input</i> Kerusakan	<i>Use case</i> ini berfungsi pada saat melakukan proses <i>input</i> kerusakan. Proses <i>Input</i> kerusakan memiliki <i>generalisasi</i> yaitu <i>before process</i> dan <i>in process</i> .
5	Melihat Laporan Kerusakan	<i>Use case</i> ini menerangkan proses melihat laporan kerusakan yang dilakukan oleh Kepala Seksi, <i>Hintan</i> dan <i>Manager Assembly Engine 2W</i> . Laporan yang dapat dilihat adalah laporan kerusakan <i>before process</i> dan laporan kerusakan <i>in process</i> .
6	Mengolah Laporan Kerusakan	<i>Use case</i> ini menerangkan proses mengolah laporan kerusakan yang dilakukan oleh Kepala Seksi.
7	Mencetak Laporan Kerusakan	<i>Use case</i> ini menerangkan proses mencetak laporan kerusakan.
8	<i>Input</i> Hasil evaluasi	<i>Use case</i> ini berfungsi pada saat melakukan proses <i>input</i> hasil evaluasi.
9	Melihat Laporan evaluasi	<i>Use case</i> ini menerangkan proses melihat laporan evaluasi yang dilakukan oleh <i>Hintan</i> .
10	Mencetak Laporan Evaluasi	<i>Use case</i> ini menerangkan proses mencetak laporan evaluasi.

(Sumber: Hasil Analisis, 2015)

3. Skenario *Use Case*

Pendefinisian masing-masing *use case* pada *use case diagram* sistem informasi pengendalian kualitas *assembly engine 2W* usulan dijelaskan pada Tabel V.5 sampai Tabel V.14 berikut ini.

Tabel V.5 Skenario *Use Case login*

<i>Use Case Name</i>	<i>Login</i>
<i>Primary Actor</i>	Teknisi, Kepala Seksi, <i>Hintan (Quality Assurance)</i> , <i>Manager Assembly Engine 2W</i>
<i>Brief Description</i>	- <i>Input: Username dan password</i> - <i>Output: Menu utama</i>
<i>Normal Flow of Events</i>	1. <i>User membuka form login.</i> 2. <i>User memasukkan username dan password.</i> 3. <i>Jika username dan password benar, maka aplikasi akan menampilkan menu utama.</i> 4. <i>Jika username dan password salah, maka akan muncul pesan kesalahan username atau password.</i>

(Sumber: Hasil Analisis, 2015)

Tabel V.6 Skenario *Use Case Mengelola Data User*

<i>Use Case Name</i>	Mengelola Data <i>User</i>
<i>Primary Actor</i>	Kepala Seksi
<i>Brief Description</i>	- <i>Input: Data user</i> - <i>Output: Informasi user</i>
<i>Relationship</i>	<i>Include: Login</i>
<i>Normal Flow of Events</i>	1. Kepala Seksi masuk ke tampilan menu utama. 2. Kepala Seksi memilih menu pengaturan. 3. Kepala Seksi memilih submenu <i>user</i> . 4. Kepala Seksi melakukan proses tambah, ubah, dan hapus data.

(Sumber: Hasil Analisis, 2015)

Tabel V.7 Skenario *Use Case Mengelola Data Master*

<i>Use Case Name</i>	Mengelola Data <i>Master</i>
<i>Primary Actor</i>	Kepala Seksi

Tabel V.7 Skenario *Use Case* Mengelola Data *Master* (Lanjutan)

<i>Brief Description</i>	<ul style="list-style-type: none"> - <i>Input</i>: Data <i>master</i> yang terdiri dari data komponen, data jenis kerusakan, data lokasi asal masalah komponen, data tipe <i>engine</i>, data kategori lini produksi dan data petugas. - <i>Output</i>: Data <i>Master</i>
<i>Relationship</i>	<p><i>Generalization</i>: mengelola data komponen, data jenis kerusakan, data lokasi asal masalah komponen, data tipe <i>engine</i>, data kategori lini produksi dan data petugas.</p> <p><i>Include</i>: <i>Login</i>.</p>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Seksi masuk ke tampilan menu utama. 2. Kepala Seksi memilih menu data <i>master</i>. 3. Kepala Seksi memilih data komponen, data jenis kerusakan, data lokasi asal masalah komponen, data tipe <i>engine</i>, data kategori lini produksi dan data petugas. 4. Kepala Seksi melakukan proses tambah, cari, ubah dan hapus data <i>master</i> ke <i>database</i>.
<i>Precondition</i>	Kepala Seksi telah melakukan <i>login</i> terlebih dahulu.

(Sumber: Hasil Analisis, 2015)

Tabel V.8 Skenario *Use Case* *Input* Kerusakan

<i>Use Case Name</i>	<i>Input</i> Kerusakan
<i>Primary Actor</i>	Teknisi
<i>Brief Description</i>	<ul style="list-style-type: none"> - <i>Input</i>: Data kerusakan yang terdiri dari <i>before process</i> dan <i>in process</i>. - <i>Output</i>: Laporan Kerusakan.
<i>Relationship</i>	<p><i>Generalization</i>: <i>before process</i> dan <i>in process</i>.</p> <p><i>Include</i>: <i>Login</i>, Mengelola Data <i>Master</i>.</p>

Tabel V.8 Skenario *Use Case Input* Kerusakan (Lanjutan)

<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Teknisi masuk ke tampilan menu utama. 2. Teknisi memilih <i>input</i> kerusakan. 3. Teknisi memilih submenu kerusakan sesuai dengan proses yang dijalani. 4. Teknisi mengisi <i>form</i> kerusakan dan menyimpan data laporan kerusakan.
<i>Precondition</i>	Teknisi telah melakukan <i>login</i> terlebih dahulu dan Kepala seksi sudah mengelola data <i>master</i> .

(Sumber: Hasil Analisis, 2015)

Tabel V.9 Skenario *Use Case* Melihat Laporan Kerusakan

<i>Use Case Name</i>	Melihat Laporan Kerusakan
<i>Primary Actor</i>	Kepala Seksi, <i>Hintan (Quality Assurance)</i> , <i>Manager Assembly Engine 2W</i>
<i>Brief Description</i>	<ul style="list-style-type: none"> - <i>Input</i>: Tanggal laporan yang ingin dilihat. - <i>Output</i>: Laporan kerusakan berdasarkan tanggal.
<i>Relationship</i>	<p><i>Generalization</i>: <i>before process</i> dan <i>in process</i></p> <p><i>Include</i>: <i>Login</i>, <i>Input</i> kerusakan</p>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> masuk ke tampilan menu utama. 2. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> memilih menu lihat laporan. 3. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> memilih submenu <i>before process</i> atau <i>in process</i>. 4. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> mencari data laporan kerusakan dengan mengisi rentang tanggal. 5. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> dapat melihat dan mencetak laporan.

Tabel V.9 Skenario *Use Case* Melihat Laporan Kerusakan (Lanjutan)

<i>Precondition</i>	Kepala Seksi, <i>Hintan (Quality Assurance)</i> , <i>Manager Assembly Engine 2W</i> telah melakukan <i>login</i> terlebih dahulu.
---------------------	---

(Sumber: Hasil Analisis, 2015)

Tabel V.10 Skenario *Use Case* Mengolah Laporan Kerusakan

<i>Use Case Name</i>	Mengolah Laporan Kerusakan
<i>Primary Actor</i>	Kepala Seksi
<i>Brief Description</i>	- <i>Input</i> : Tanggal laporan yang ingin diolah. - <i>Output</i> : Laporan kerusakan berdasarkan tanggal.
<i>Relationship</i>	<i>Include</i> : <i>Login</i> , <i>Input</i> kerusakan
<i>Normal Flow of Events</i>	1. Kepala Seksi masuk ke tampilan menu utama. 2. Kepala Seksi memilih menu lihat laporan. 3. Kepala Seksi memilih submenu <i>before process</i> atau <i>in process</i> . 4. Kepala Seksi mencari data laporan kerusakan dengan mengisi rentang tanggal. 5. Kepala Seksi dapat melihat serta mencetak laporan kerusakan.
<i>Precondition</i>	Kepala Seksi telah melakukan <i>login</i> terlebih dahulu.

(Sumber: Hasil Analisis, 2015)

Tabel V.11 Skenario *Use Case* Mencetak Laporan Kerusakan

<i>Use Case Name</i>	Mencetak Laporan Kerusakan
<i>Primary Actor</i>	Kepala Seksi, <i>Hintan (Quality Assurance)</i> , <i>Manager Assembly Engine 2W</i>
<i>Brief Description</i>	- <i>Input</i> : Tanggal laporan yang ingin dilihat. - <i>Output</i> : Laporan kerusakan berdasarkan tanggal.
<i>Relationship</i>	<i>Include</i> : Melihat laporan kerusakan

Tabel V.11 Skenario *Use Case* Mencetak Laporan Kerusakan (Lanjutan)

<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> masuk ke tampilan menu utama. 2. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> memilih menu lihat laporan. 3. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> memilih submenu <i>before process</i> atau <i>in process</i>. 4. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> mencari data laporan kerusakan dengan mengisi rentang tanggal. 5. Kepala Seksi, <i>Hintan (Quality Assurance)</i>, <i>Manager Assembly Engine 2W</i> mencetak laporan.
<i>Precondition</i>	Kepala Seksi, <i>Hintan (Quality Assurance)</i> , <i>Manager Assembly Engine 2W</i> telah melakukan <i>login</i> terlebih dahulu.

(Sumber: Hasil Analisis, 2015)

Tabel V.12 Skenario *Use Case Input* Hasil Evaluasi

<i>Use Case Name</i>	<i>Input</i> Hasil Evaluasi
<i>Primary Actor</i>	<i>Hintan (Quality Assurance)</i>
<i>Brief Description</i>	<ul style="list-style-type: none"> - <i>Input</i>: Hasil evaluasi. - <i>Output</i>: Laporan evaluasi.
<i>Relationship</i>	<i>Include: Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>Hintan</i> masuk ke tampilan menu utama. 2. <i>Hintan</i> memilih <i>input</i> hasil evaluasi. 3. <i>Hintan</i> mengisi <i>form</i> hasil evaluasi dan menyimpan data evaluasi.
<i>Precondition</i>	<i>Hintan</i> telah melakukan <i>login</i> terlebih dahulu.

(Sumber: Hasil Analisis, 2015)

Tabel V.13 Skenario *Use Case* Melihat Laporan Evaluasi

<i>Use Case Name</i>	Melihat Laporan Evaluasi
<i>Primary Actor</i>	<i>Hintan (Quality Assurance)</i>
<i>Brief Description</i>	- <i>Input</i> : bulan atau tahun. - <i>Output</i> : Laporan evaluasi.
<i>Relationship</i>	<i>Include</i> : <i>Input</i> Hasil Evaluasi
<i>Normal Flow of Events</i>	1. <i>Hintan</i> masuk ke tampilan menu utama. 2. <i>Hintan</i> memilih menu laporan, lalu pilih submenu evaluasi. 3. <i>Hintan</i> mencari data laporan dengan memilih bulan atau tahun yang ingin dilihat.
<i>Precondition</i>	<i>Hintan</i> telah melakukan <i>login</i> terlebih dahulu.

(Sumber: Hasil Analisis, 2015)

Tabel V.14 Skenario *Use Case* Mencetak Laporan Evaluasi

<i>Use Case Name</i>	Mencetak Laporan Evaluasi
<i>Primary Actor</i>	<i>Hintan (Quality Assurance)</i>
<i>Brief Description</i>	- <i>Input</i> : bulan atau tahun. - <i>Output</i> : Laporan evaluasi.
<i>Relationship</i>	<i>Include</i> : Melihat Laporan Evaluasi
<i>Normal Flow of Events</i>	1. <i>Hintan</i> masuk ke tampilan menu utama. 2. <i>Hintan</i> memilih menu laporan, lalu pilih submenu evaluasi. 3. <i>Hintan</i> mencari data laporan dengan memilih bulan atau tahun yang ingin dicetak. 4. <i>Hintan</i> mencetak laporan.
<i>Precondition</i>	<i>Hintan</i> telah melakukan <i>login</i> terlebih dahulu.

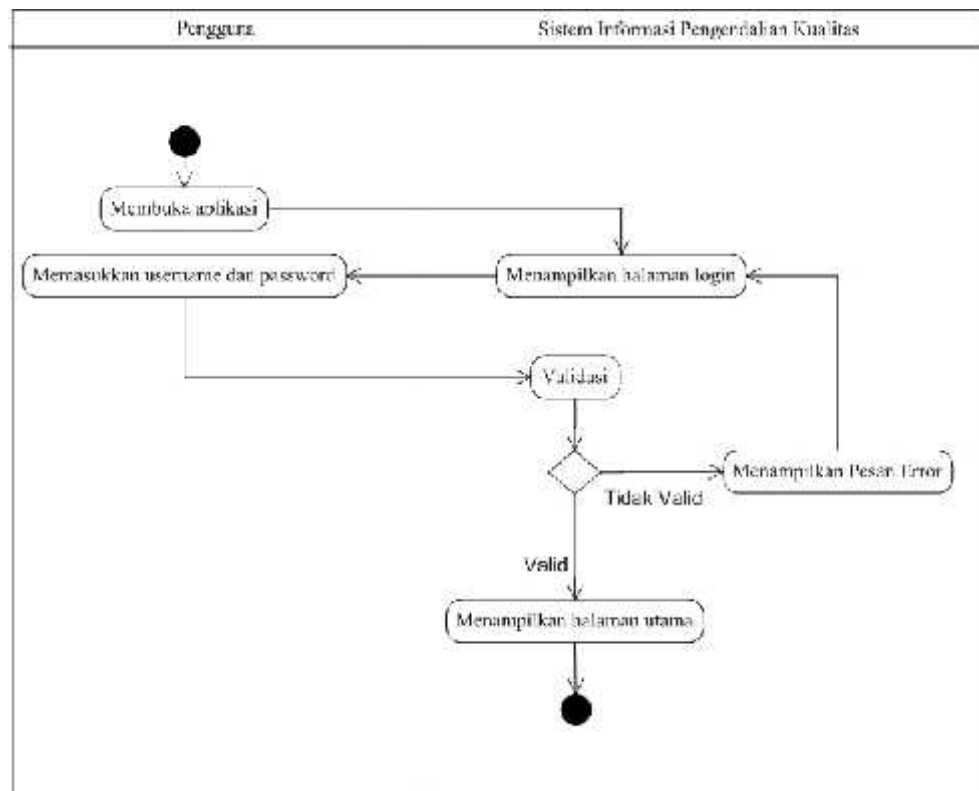
(Sumber: Hasil Analisis, 2015)

5.5.2 Activity Diagram

Activity diagram digunakan untuk menggambarkan aliran kerja tiap *use case* pada sistem informasi pengendalian kualitas *assembly engine 2W*. *Activity diagram* masing-masing *use case* digambarkan pada gambar-gambar berikut ini.

1. Activity Diagram Proses Login

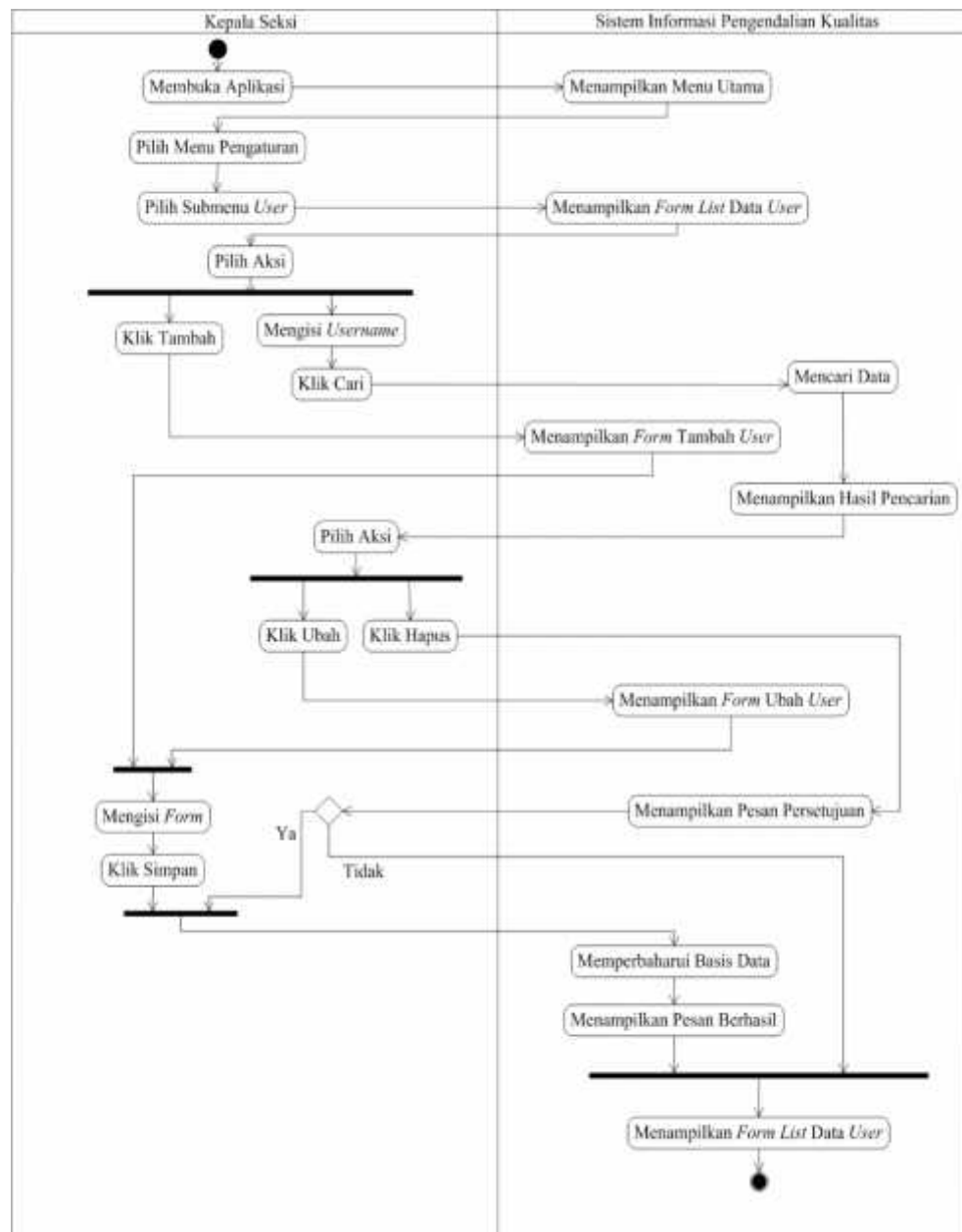
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan melakukan proses *login*, dimana pengguna mengisi *username* dan *password* untuk dapat masuk ke dalam sistem. *Activity diagram* proses *login* digambarkan pada Gambar V.3.



Gambar V.3 Activity Diagram Proses Login
(Sumber: Hasil Analisis, 2015)

2. Activity Diagram Proses Mengelola Data User

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan melakukan mengelola data *user*, dimana Kepala Seksi dapat menambah, mencari, mengubah dan menghapus data *user*. *Activity diagram* proses mengelola data *user* digambarkan pada Gambar V.4.

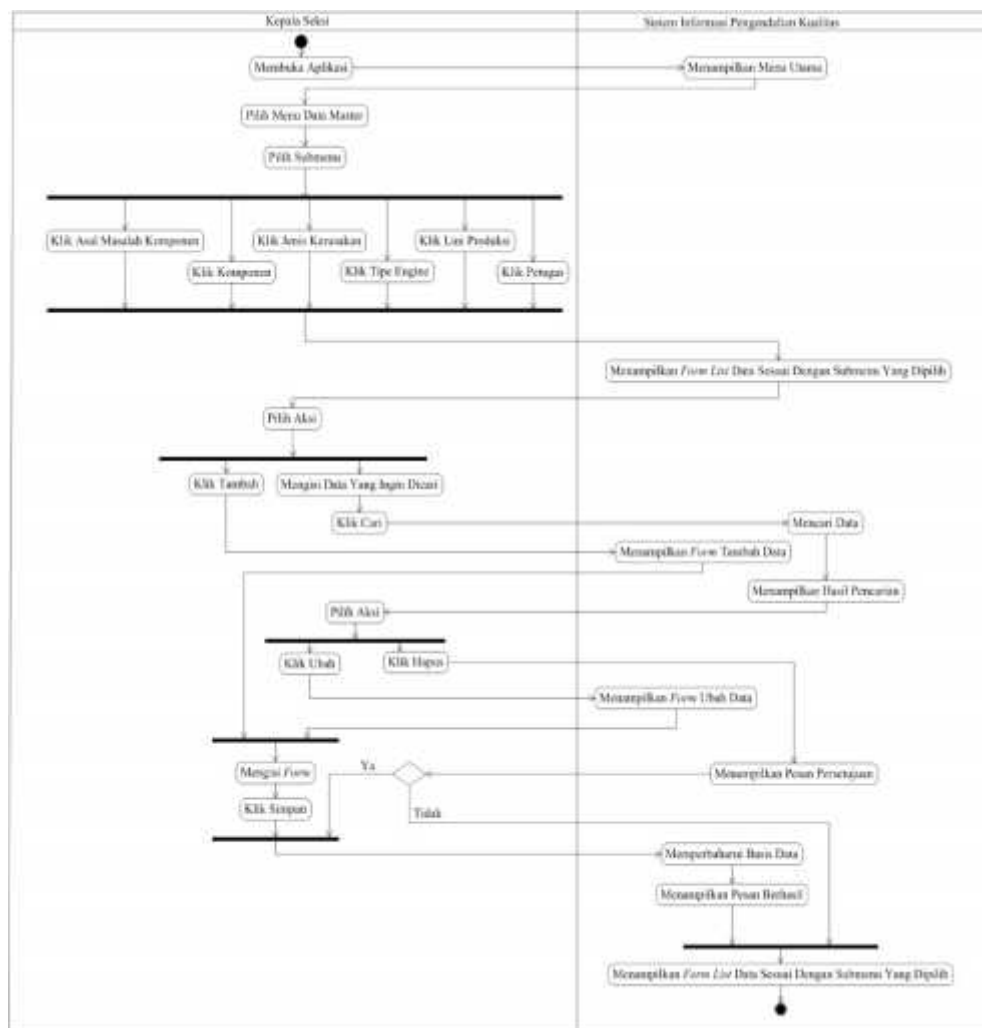


Gambar V.4 Activity Diagram Proses Mengelola Data User
(Sumber: Hasil Analisis, 2015)

3. Activity Diagram Proses Mengelola Data Master

Activity diagram berikut ini menjelaskan aktivitas yang dilakukan dalam mengelola data master. Data master tersebut terdiri dari data komponen, data jenis kerusakan, data lokasi asal masalah komponen, data tipe engine, data kategori lini produksi dan data petugas. Adapun cara mengelola data master yaitu pada menu utama memilih menu data master, lalu memilih

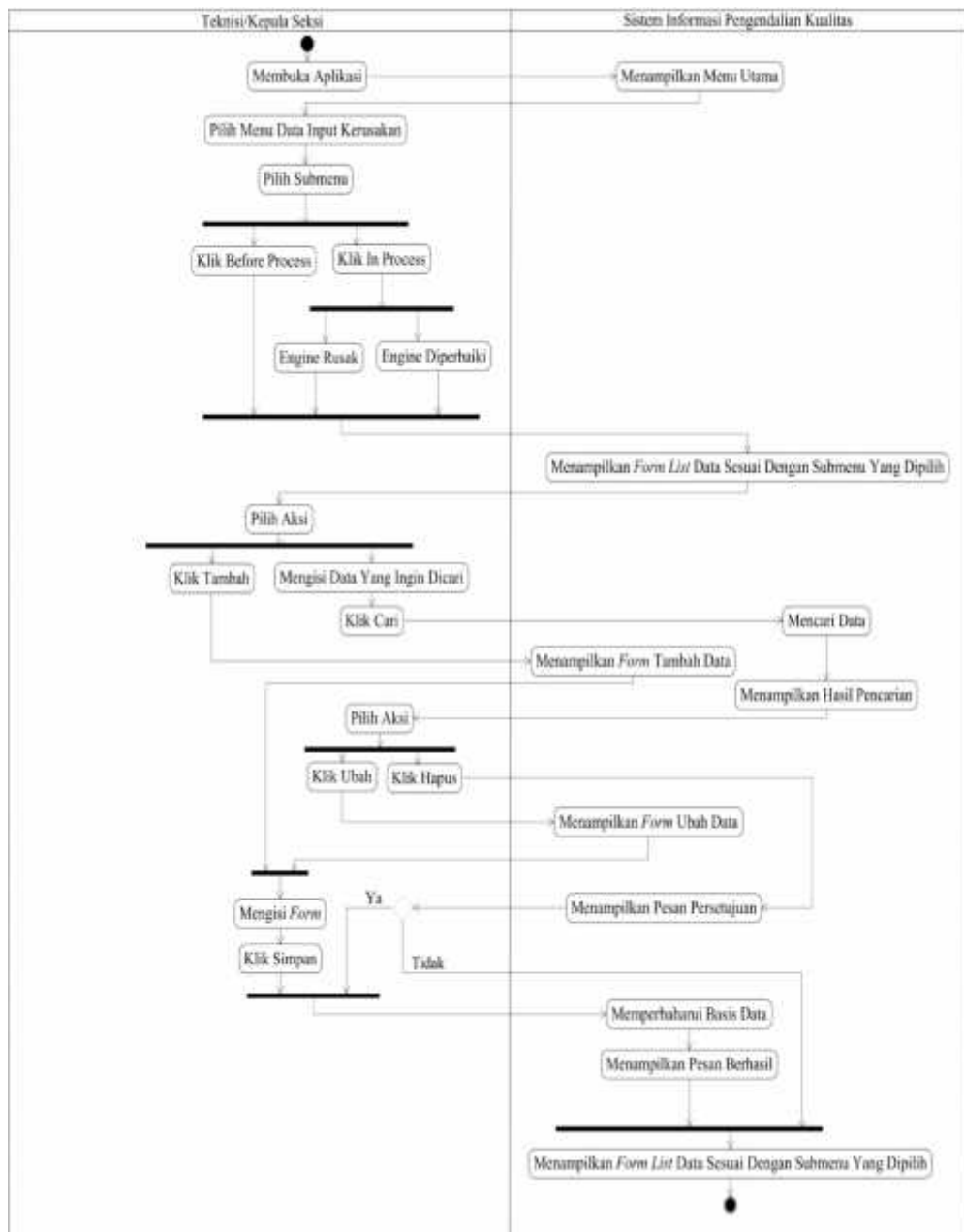
submenu komponen/jenis kerusakan/lokasi asal masalah komponen/tipe *engine*/kategori lini produksi/petugas. Kepala Seksi yang mempunyai akses untuk menjalankan menu ini dapat tambah, ubah dan hapus data. *Activity diagram* mengelola data *master* ditampilkan pada Gambar V.5.



Gambar V.5 *Activity Diagram* Proses Mengelola Data *Master*
(Sumber: Hasil Analisis, 2015)

4. *Activity Diagram* Proses *Input* Kerusakan

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan *input* kerusakan, dimana Teknisi/Kepala Seksi dapat melakukan tambah data, cari, ubah dan hapus data. *User* dapat memilih menu *input* kerusakan, kemudian memilih submenu *before process* atau *in process*. *Activity diagram* proses *input* kerusakan digambarkan pada Gambar V.6.

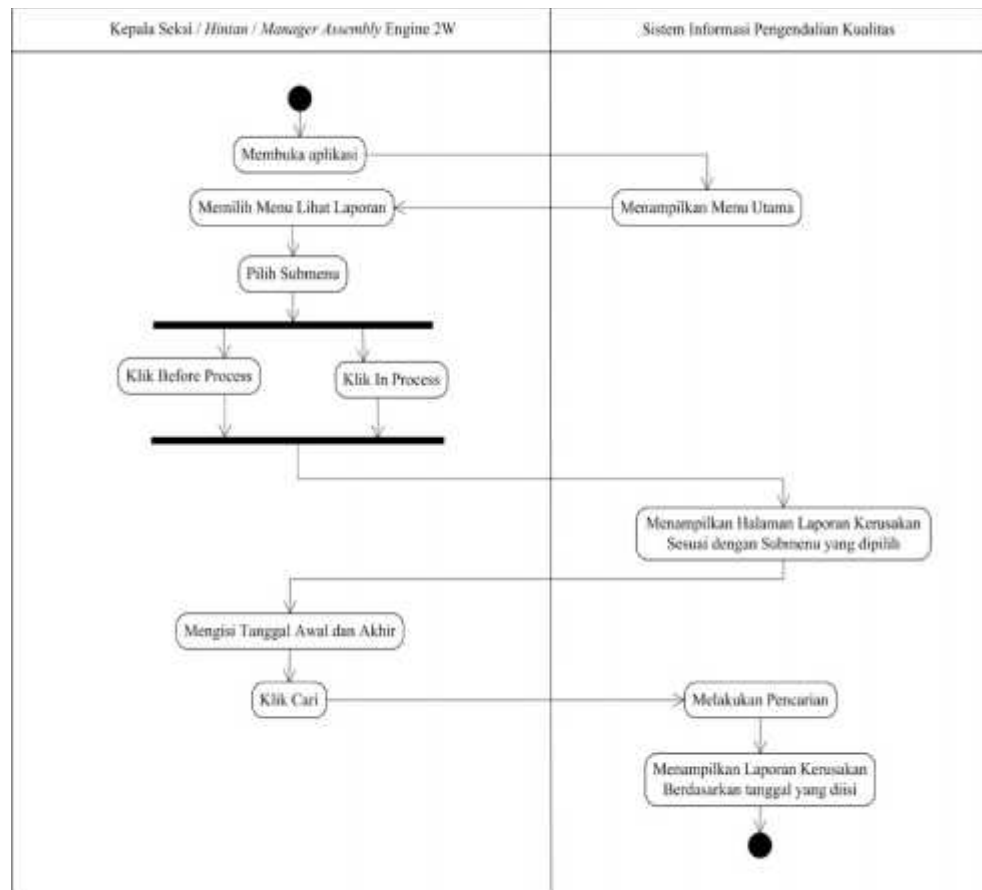


Gambar V.6 Activity Diagram Proses Input Kerusakan
(Sumber: Hasil Analisis, 2015)

5. Activity Diagram Proses Melihat Laporan Kerusakan

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan melihat laporan kerusakan, dimana Kepala Seksi, *Hintan* dan *Manager Assembly Engine 2W* dapat melihat laporan kerusakan. Laporan kerusakan yang dapat dilihat terdiri dari laporan kerusakan *before process* dan laporan kerusakan *in process*. Adapun cara melihat laporan

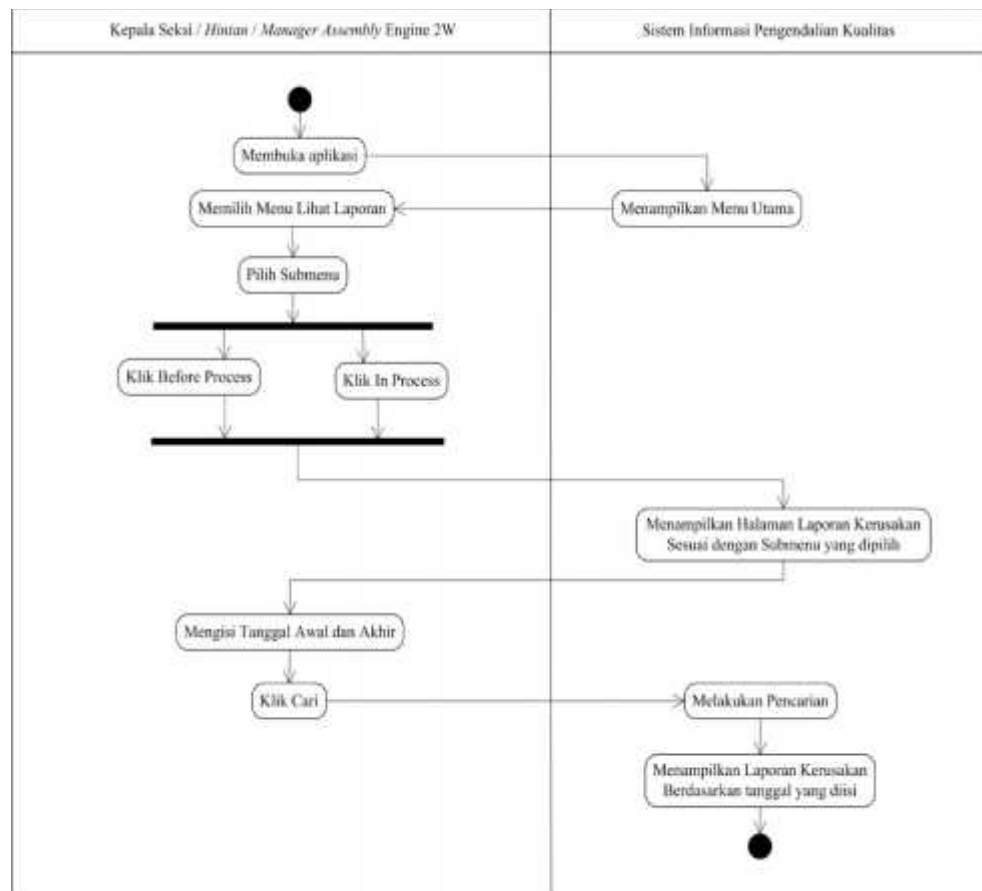
kerusakan yaitu pada menu utama memilih menu lihat laporan, lalu memilih submenu *before process* atau *in process*. *Activity diagram* proses melihat laporan kerusakan digambarkan pada Gambar V.7.



Gambar V.7 *Activity Diagram* Proses Melihat Laporan Kerusakan
(Sumber: Hasil Analisis, 2015)

6. *Activity Diagram* Proses Mengolah Laporan Kerusakan

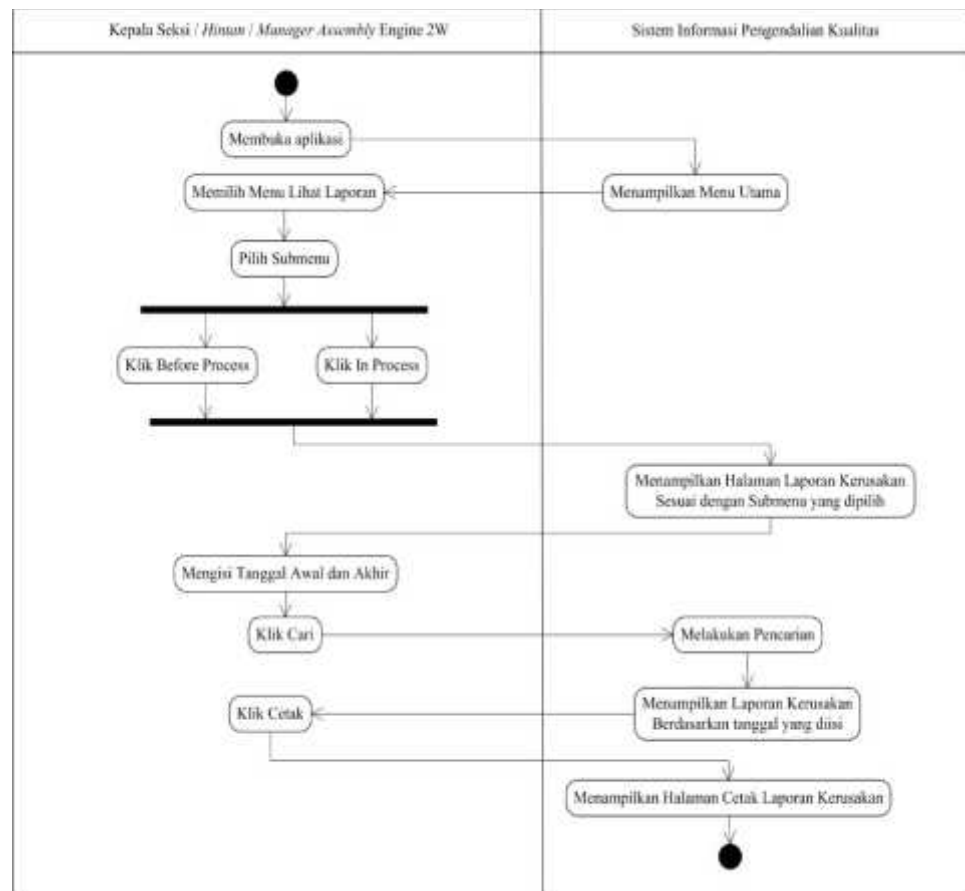
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan melihat mengolah laporan kerusakan, dimana Kepala Seksi dapat mengolah laporan kerusakan. Laporan kerusakan yang dapat diolah terdiri dari laporan kerusakan *before process* dan laporan kerusakan *in process*. Adapun cara mengolah laporan kerusakan yaitu pada menu utama memilih menu lihat laporan, lalu memilih submenu *before process* atau *in process*. *Activity diagram* proses mengolah laporan kerusakan digambarkan pada Gambar V.8.



Gambar V.8 Activity Diagram Proses Mengolah Laporan Kerusakan
(Sumber: Hasil Analisis, 2015)

7. Activity Diagram Proses Mencetak Laporan Kerusakan

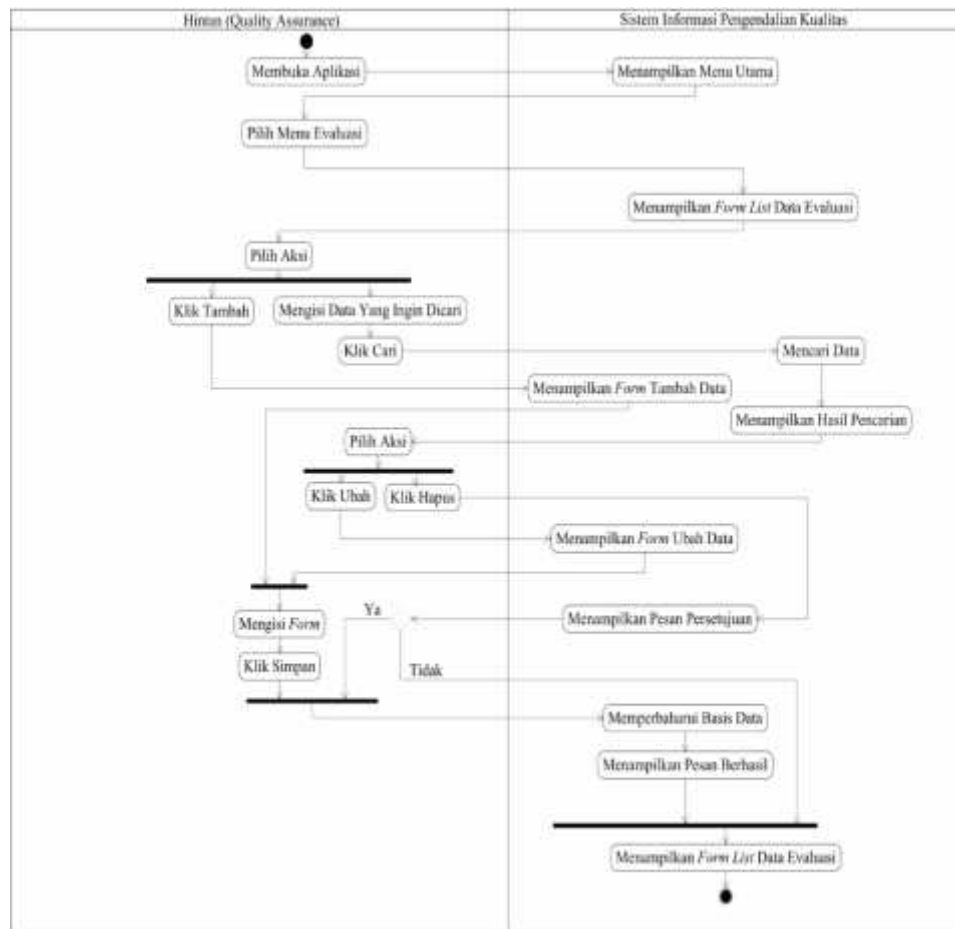
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mencetak laporan kerusakan, dimana Kepala Seksi, *Hintan* dan *Manager Assembly Engine 2W* mencari laporan kerusakan berdasarkan tanggal. Activity diagram proses mencetak laporan kerusakan digambarkan pada Gambar V.9.



Gambar V.9 Activity Diagram Proses Mencetak Laporan Kerusakan
(Sumber: Hasil Analisis, 2015)

8. Activity Diagram Proses Input Hasil Evaluasi

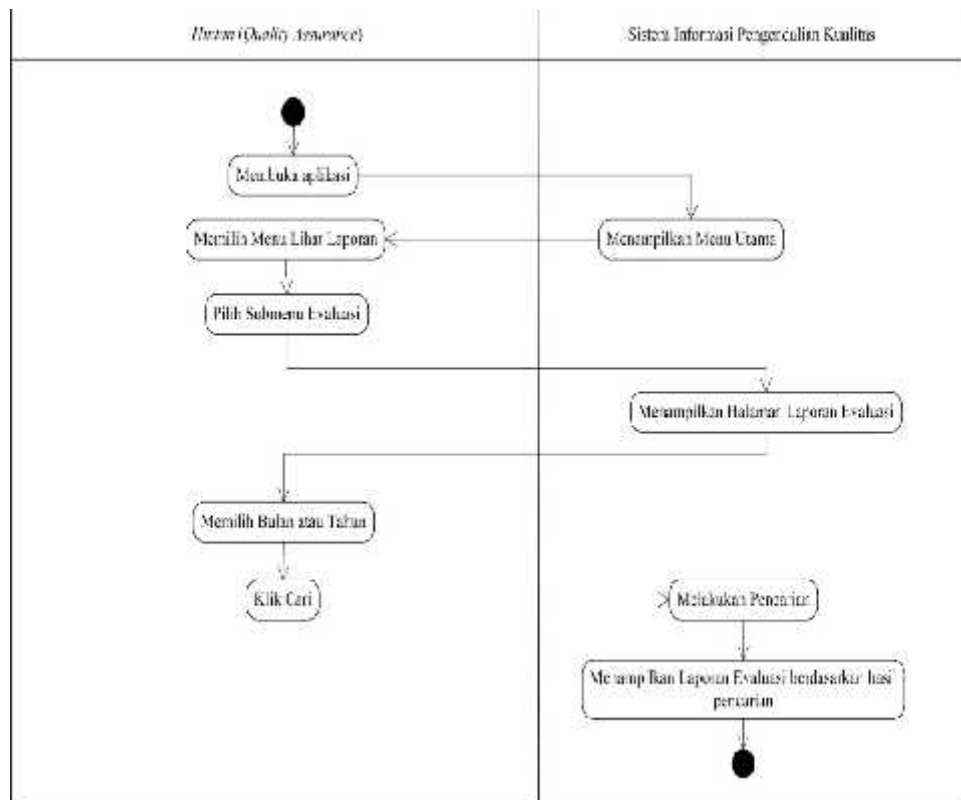
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan input hasil evaluasi, dimana *Hintan (Quality Assurance)* dapat melakukan tambah data, cari, ubah dan hapus data. *User* dapat memilih menu evaluasi. Activity diagram proses input kerusakan digambarkan pada Gambar V.10.



Gambar V.10 Activity Diagram Proses Input hasil Evaluasi
(Sumber: Hasil Analisis, 2015)

9. Activity Diagram Proses Melihat Laporan Evaluasi

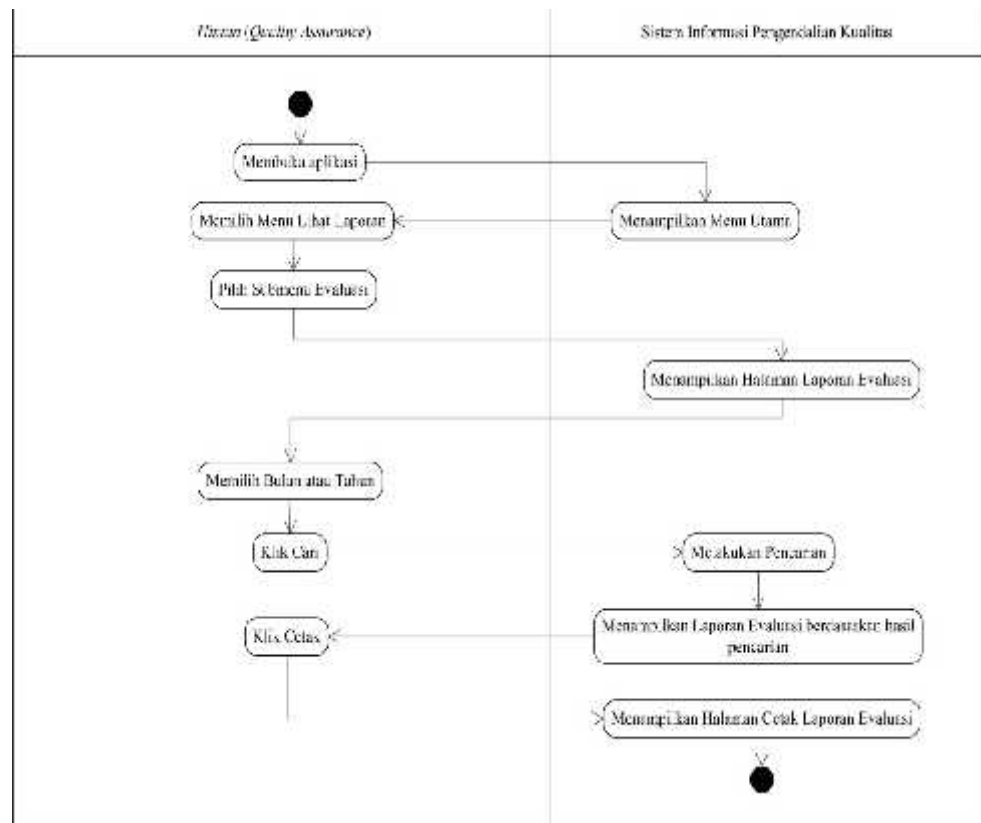
Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan melihat laporan evaluasi, dimana *Hintan* dapat melihat laporan evaluasi. Adapun cara melihat laporan evaluasi yaitu pada menu utama memilih menu lihat laporan, lalu memilih submenu evaluasi. Activity diagram proses melihat laporan evaluasi digambarkan pada Gambar V.11.



Gambar V.11 *Activity Diagram* Proses Melihat Laporan Evaluasi
(Sumber: Hasil Analisis, 2015)

10. *Activity Diagram* Proses Mencetak Laporan Evaluasi

Activity diagram berikut ini menggambarkan aktivitas yang dilakukan saat akan mencetak laporan evaluasi, dimana *Hintan* mencari laporan evaluasi berdasarkan bulan atau tahun. *Activity diagram* proses mencetak laporan evaluasi digambarkan pada Gambar V.12.



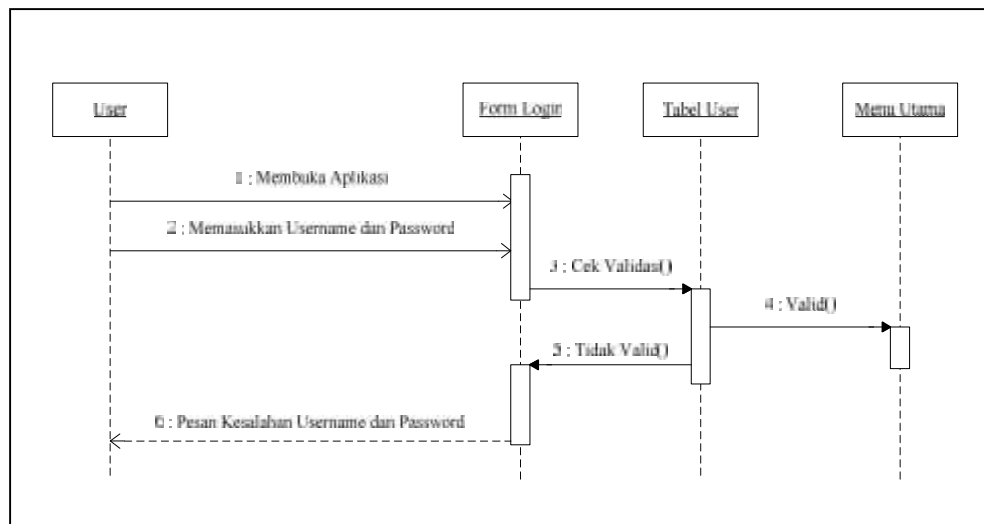
Gambar V.12 Activity Diagram Proses Mencetak Laporan Evaluasi
(Sumber: Hasil Analisis, 2015)

5.5.3 Sequence Diagram

Sequence diagram menggambarkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu, di mana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada diagram. Hubungan yang ada pada gambar-gambar di bawah ini adalah proses yang dilakukan oleh sistem sesuai dengan objek pada *use case diagram*, berikut adalah *sequence diagram* pada sistem informasi pengendalian kualitas *assembly engine 2W*:

1. Sequence Diagram pada Proses Login

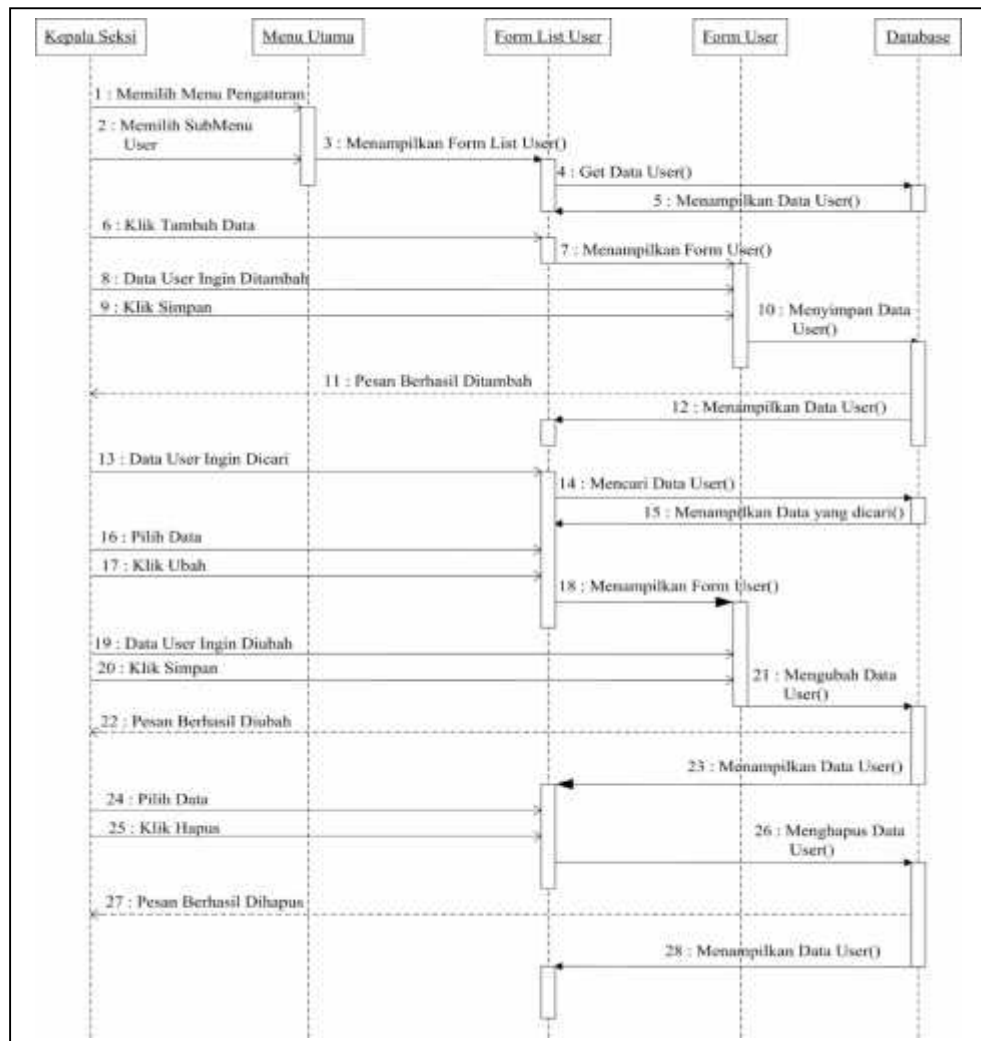
Sequence diagram login menggambarkan interaksi yang terjadi pada proses *login*. Proses ini dilakukan oleh seluruh aktor dalam *use case diagram*. *Sequence diagram* proses *login* digambarkan pada Gambar V.13.



Gambar V.13 *Sequence Diagram* Proses Login
(Sumber: Hasil Analisis, 2015)

2. *Sequence Diagram* Mengelola User

Sequence diagram mengelola user menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data user yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola user digambarkan pada Gambar V.14.



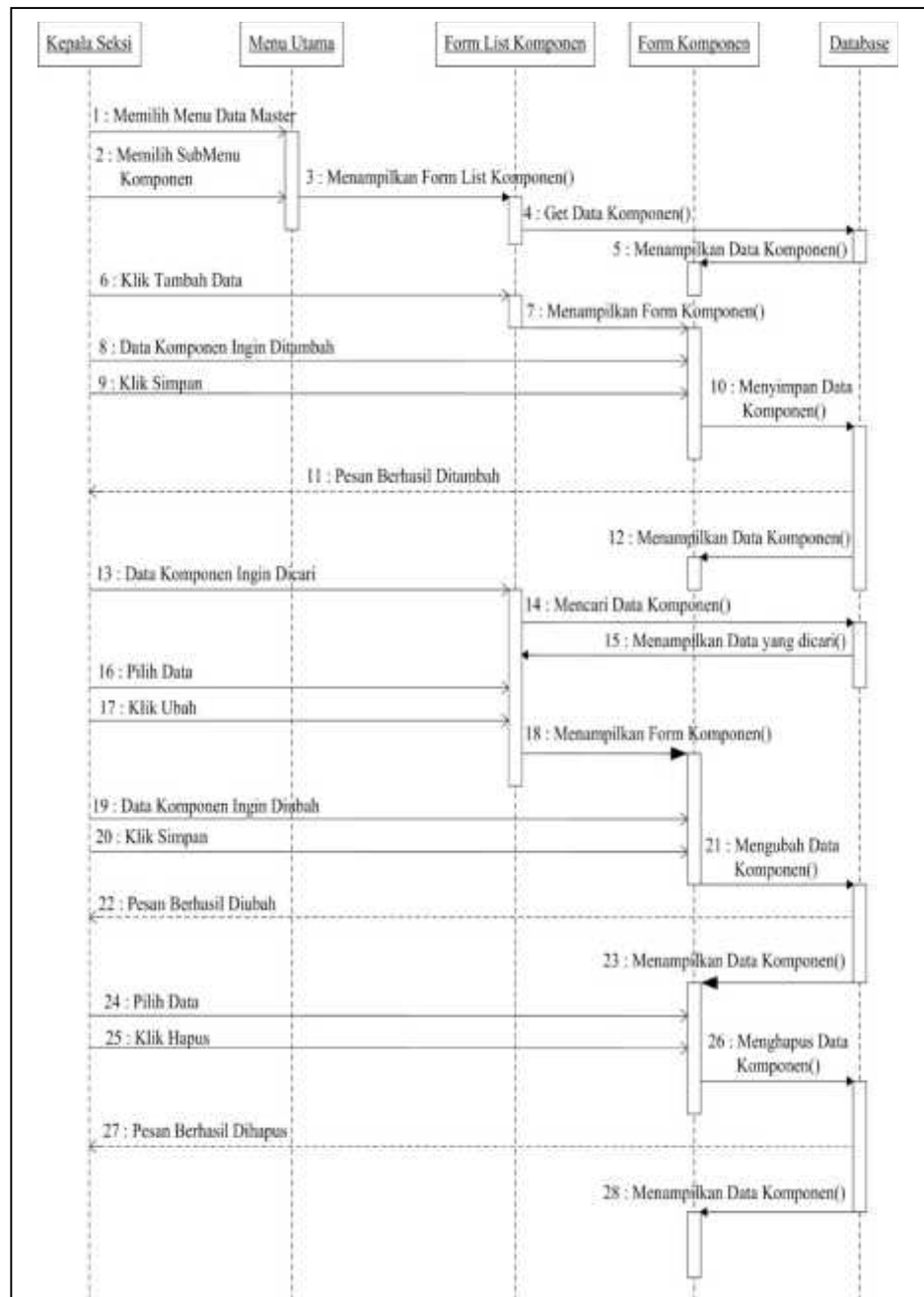
Gambar V.14 *Sequence Diagram* Mengelola User
(Sumber: Hasil Analisis, 2015)

3. *Sequence Diagram* Mengelola Data Master

Sequence diagram mengelola Data Master menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola data master digambarkan pada Gambar V.15 sampai Gambar V.17.

a. *Sequence Diagram* Mengelola Data Master Komponen

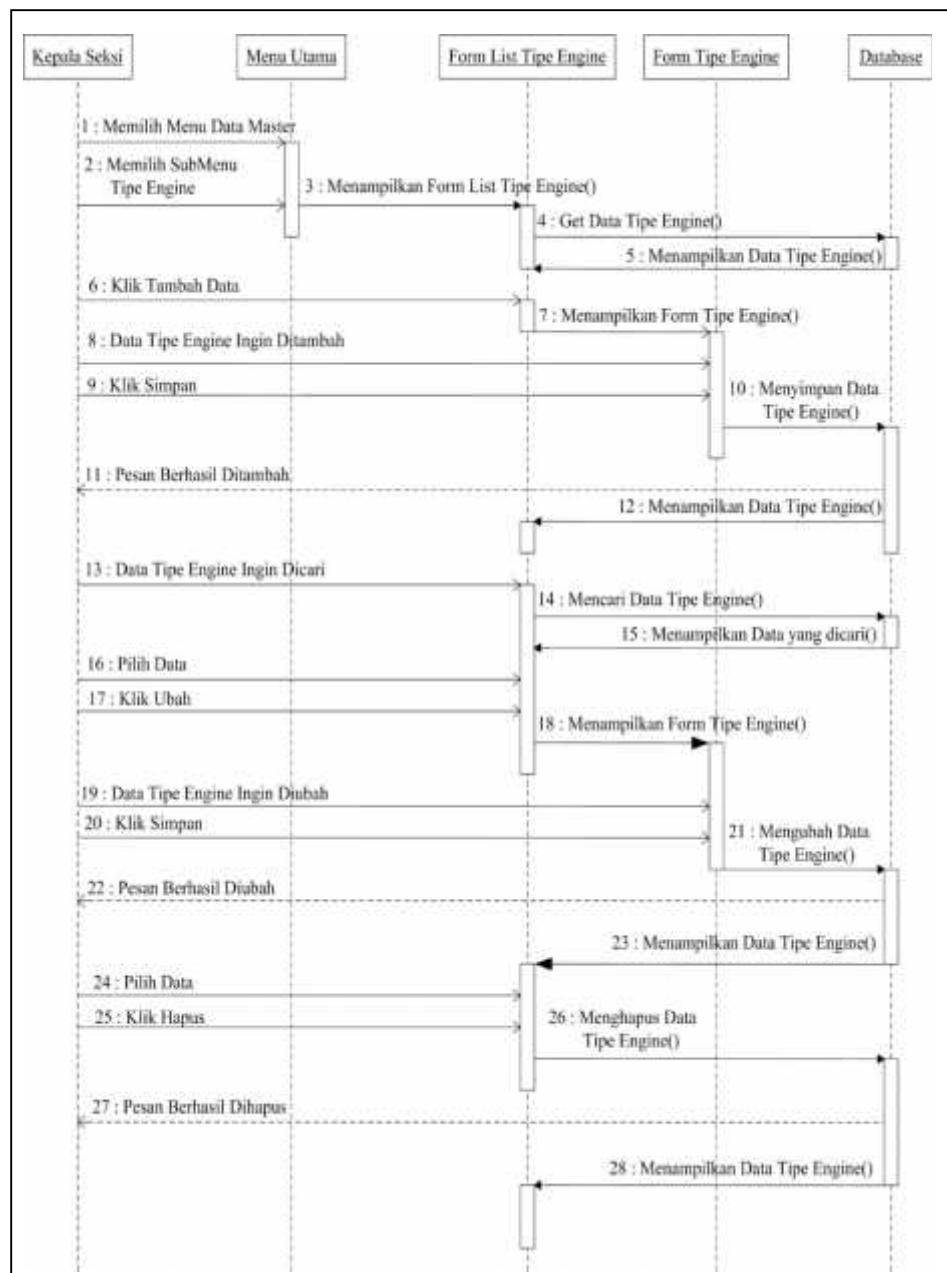
Sequence diagram mengelola data master komponen menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola data master komponen digambarkan pada Gambar V.15.



Gambar V.15 *Sequence Diagram* Mengelola Data Master Komponen
(Sumber: Hasil Analisis, 2015)

b. *Sequence Diagram* Mengelola Data Master Tipe Engine

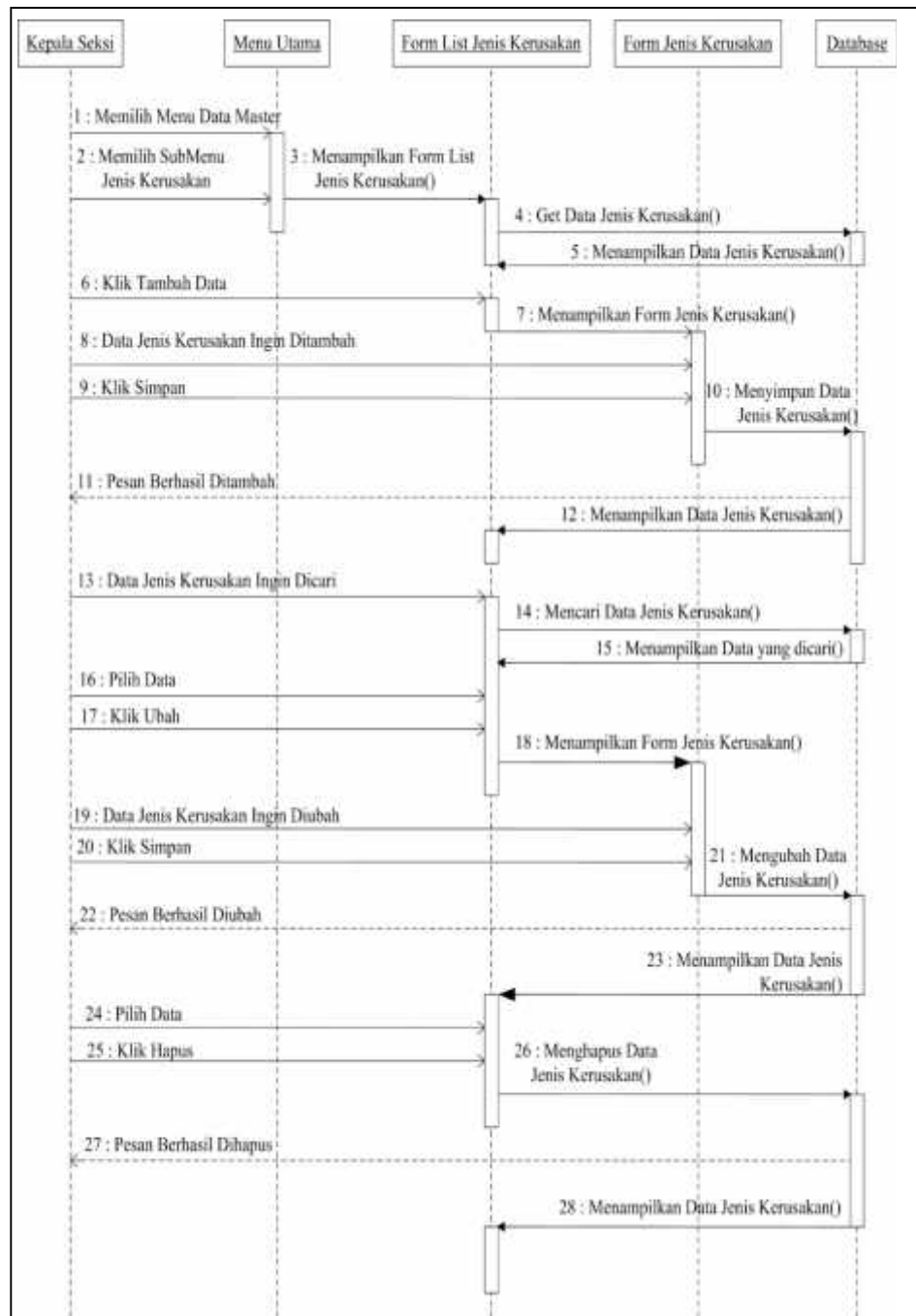
Sequence diagram mengelola data master tipe engine menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola data master tipe engine digambarkan pada Gambar V.16.



Gambar V.16 *Sequence Diagram* Mengelola Data Master Tipe Engine
(Sumber: Hasil Analisis, 2015)

c. *Sequence Diagram* Mengelola Data Master Jenis Kerusakan

Sequence diagram mengelola data master jenis kerusakan menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola data master jenis kerusakan digambarkan pada Gambar V.17.

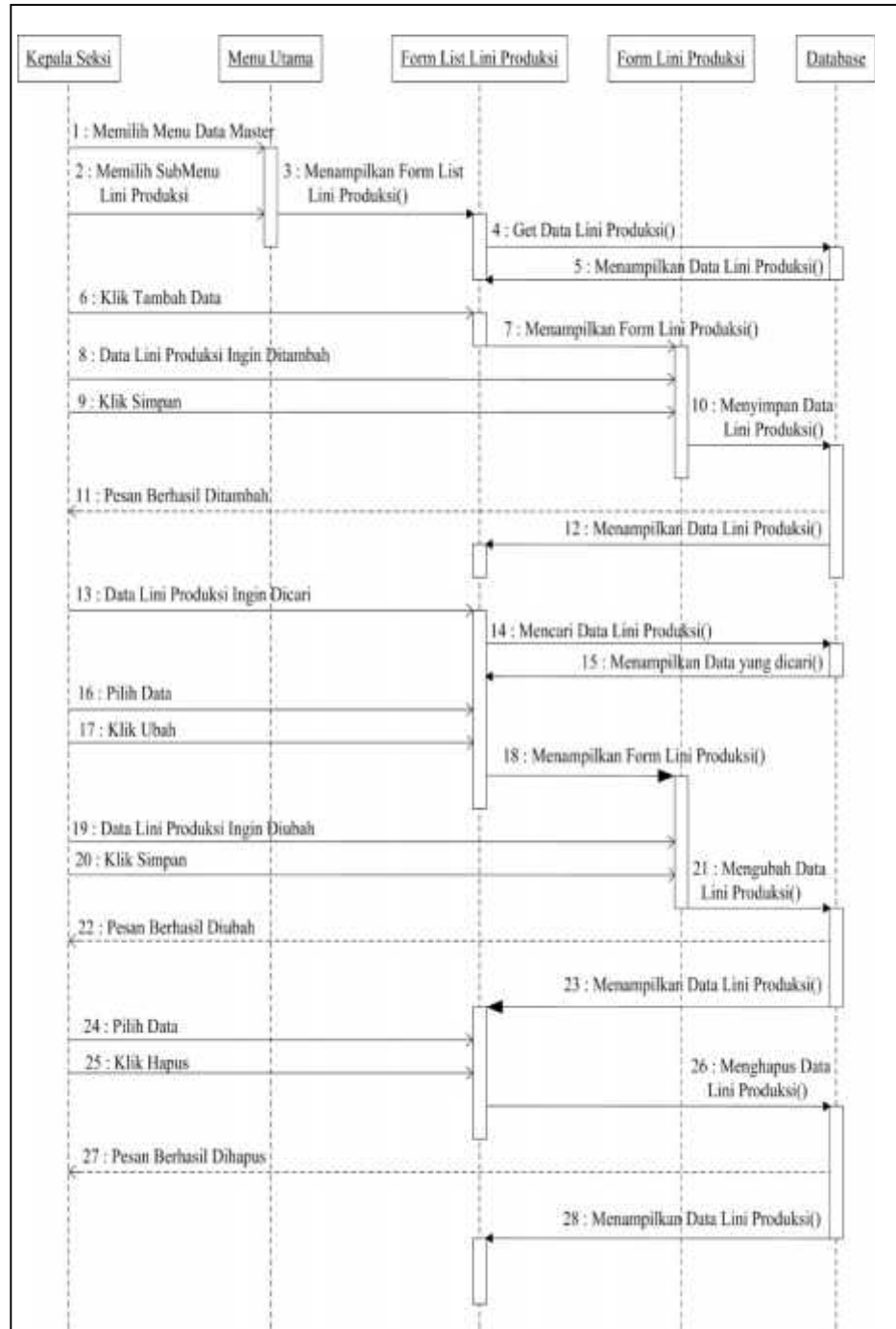


Gambar V.17 *Sequence Diagram* Mengelola Data *Master* Jenis Kerusakan
(Sumber: Hasil Analisis, 2015)

d. *Sequence Diagram* Mengelola Data *Master* Lini Produksi

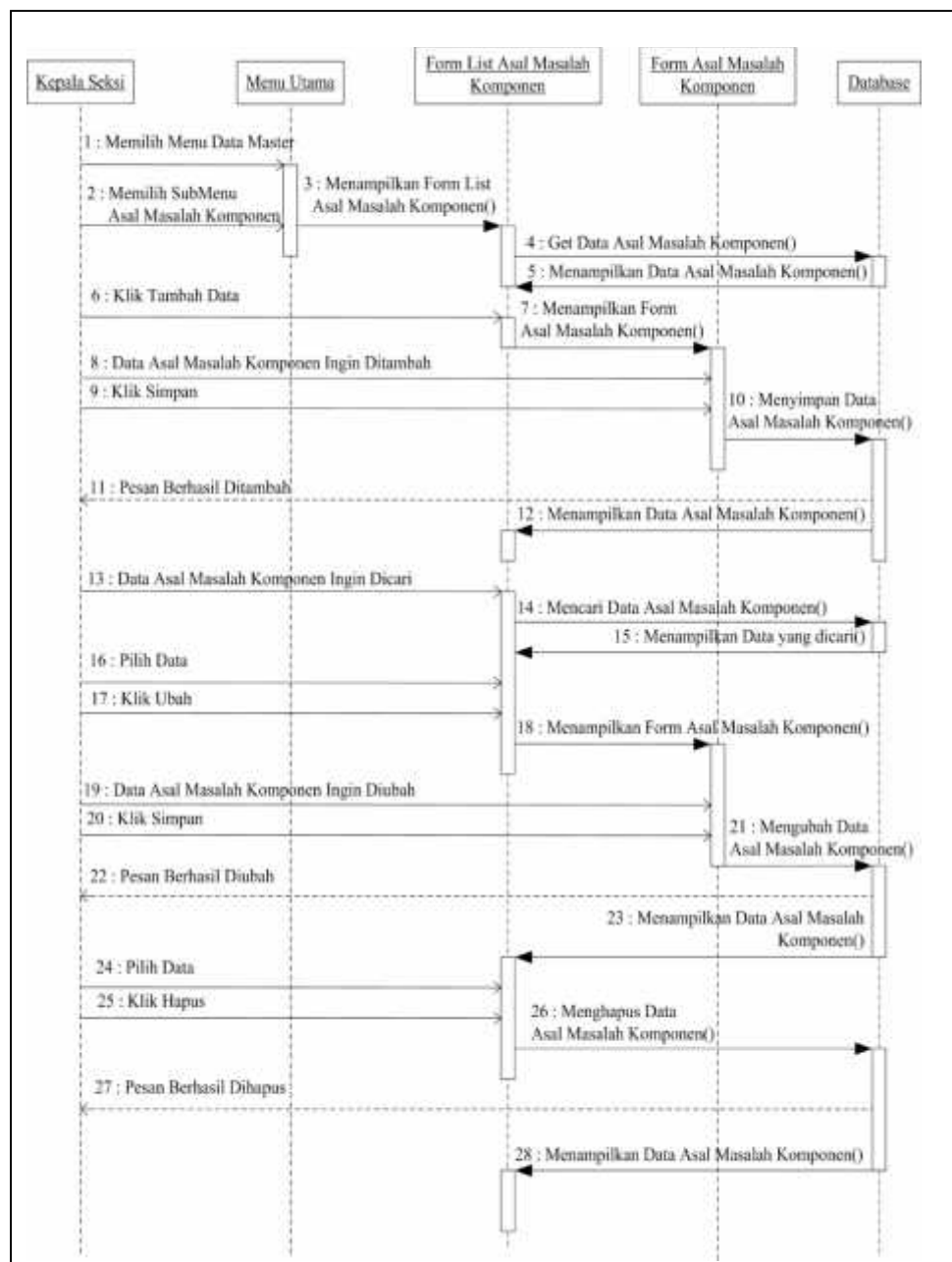
Sequence diagram mengelola data *master* lini produksi menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan

hapus data yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola data *master* lini produksi digambarkan pada Gambar V.18.



Gambar V.18 *Sequence Diagram* Mengelola Data *Master* Lini Produksi
(Sumber: Hasil Analisis, 2015)

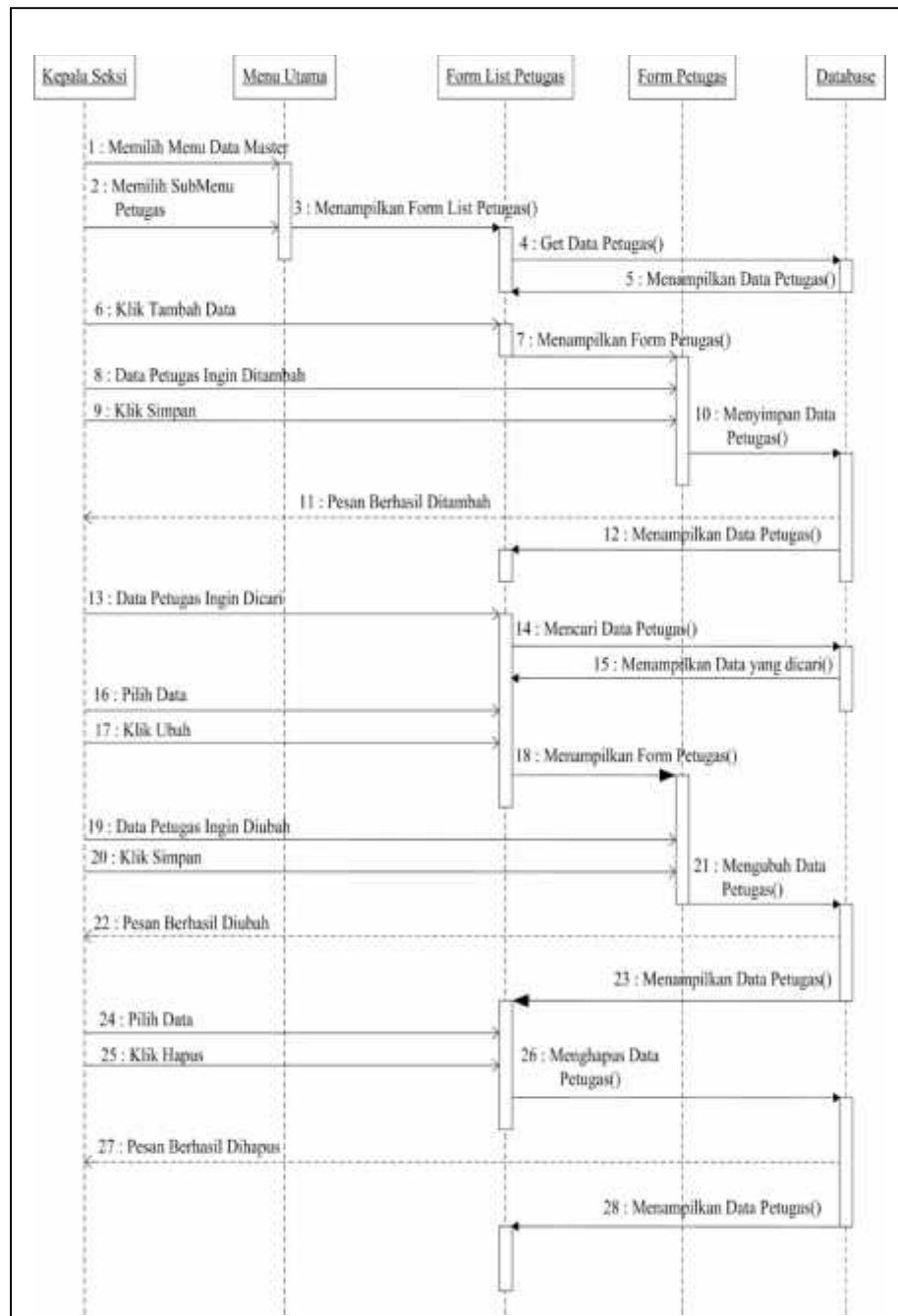
- e. *Sequence Diagram* Mengelola Data *Master* Asal Masalah Komponen
- Sequence diagram* mengelola data *master* asal masalah komponen menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola data *master* asal masalah komponen digambarkan pada Gambar V.19.



Gambar V.19 *Sequence Diagram* Data *Master* Asal Masalah Komponen
(Sumber: Hasil Analisis, 2015)

f. *Sequence Diagram* Mengelola Data *Master* Petugas

Sequence diagram mengelola data *master* petugas menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Kepala Seksi. *Sequence diagram* mengelola data *master* petugas digambarkan pada Gambar V.20.



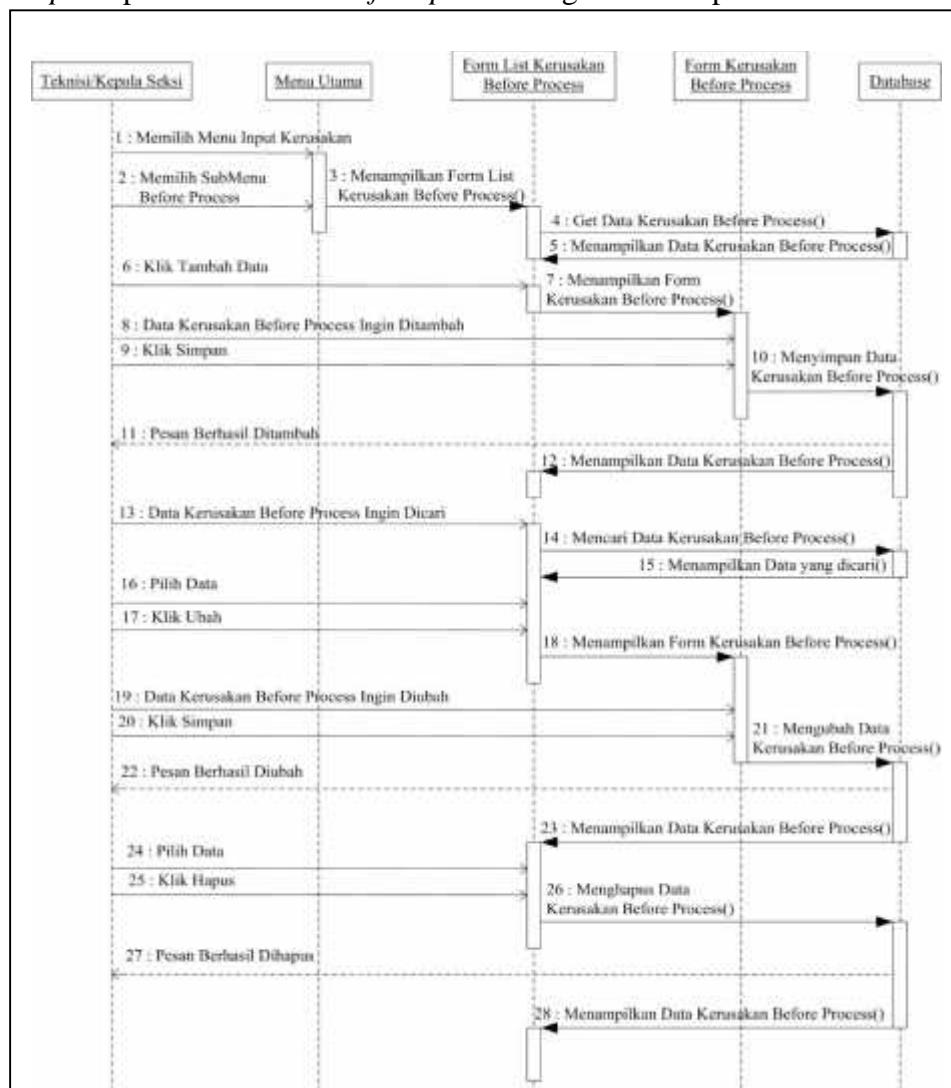
Gambar V.20 *Sequence Diagram* Mengelola Data *Master* Petugas
(Sumber: Hasil Analisis, 2015)

4. *Sequence Diagram Input Kerusakan*

Sequence diagram input kerusakan menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Teknisi/Kepala Seksi. *Sequence diagram input* kerusakan digambarkan pada Gambar V.21 dan V.22.

a. *Sequence Diagram Input Kerusakan Before Process*

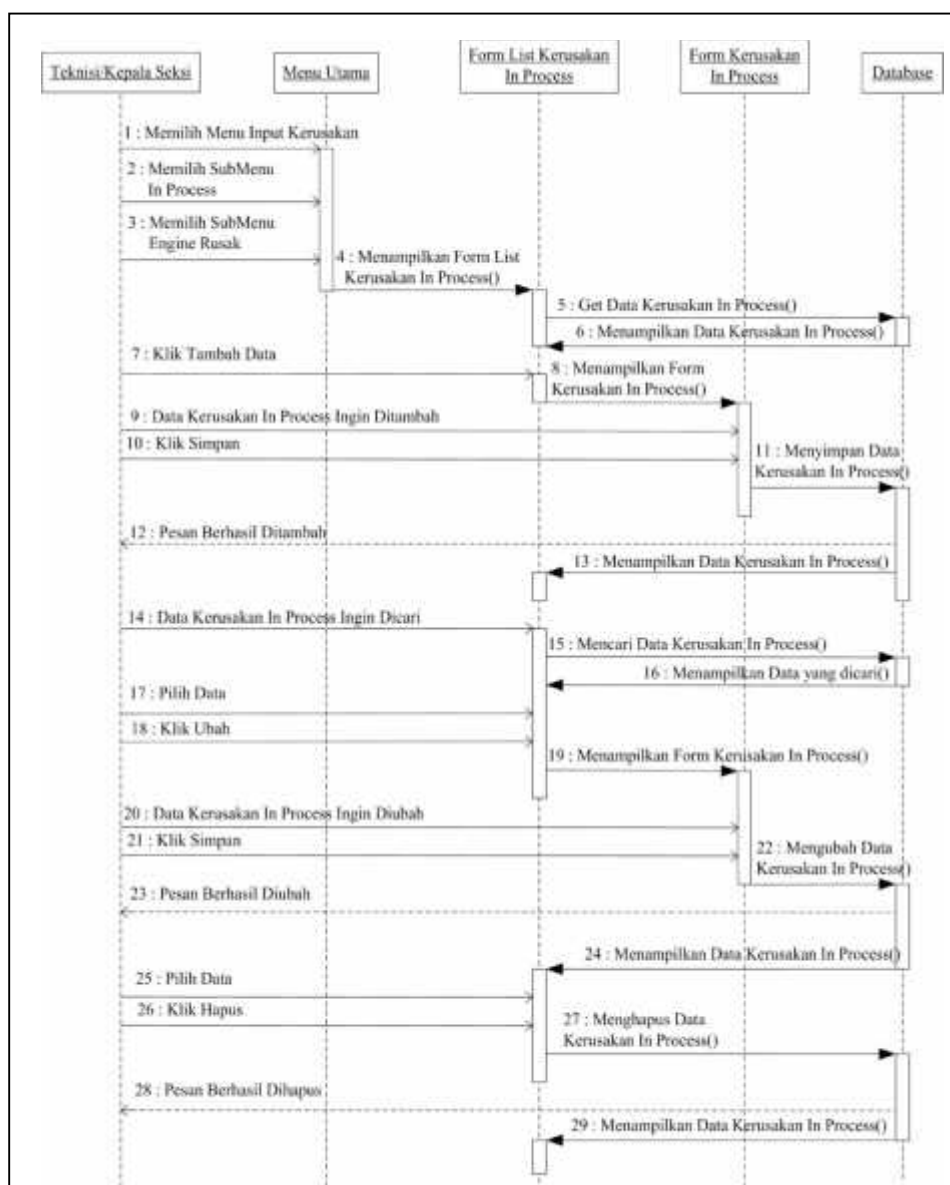
Sequence diagram input kerusakan *before process* menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Teknisi/Kepala Seksi. *Sequence diagram input* laporan kerusakan *before process* digambarkan pada Gambar V.21.



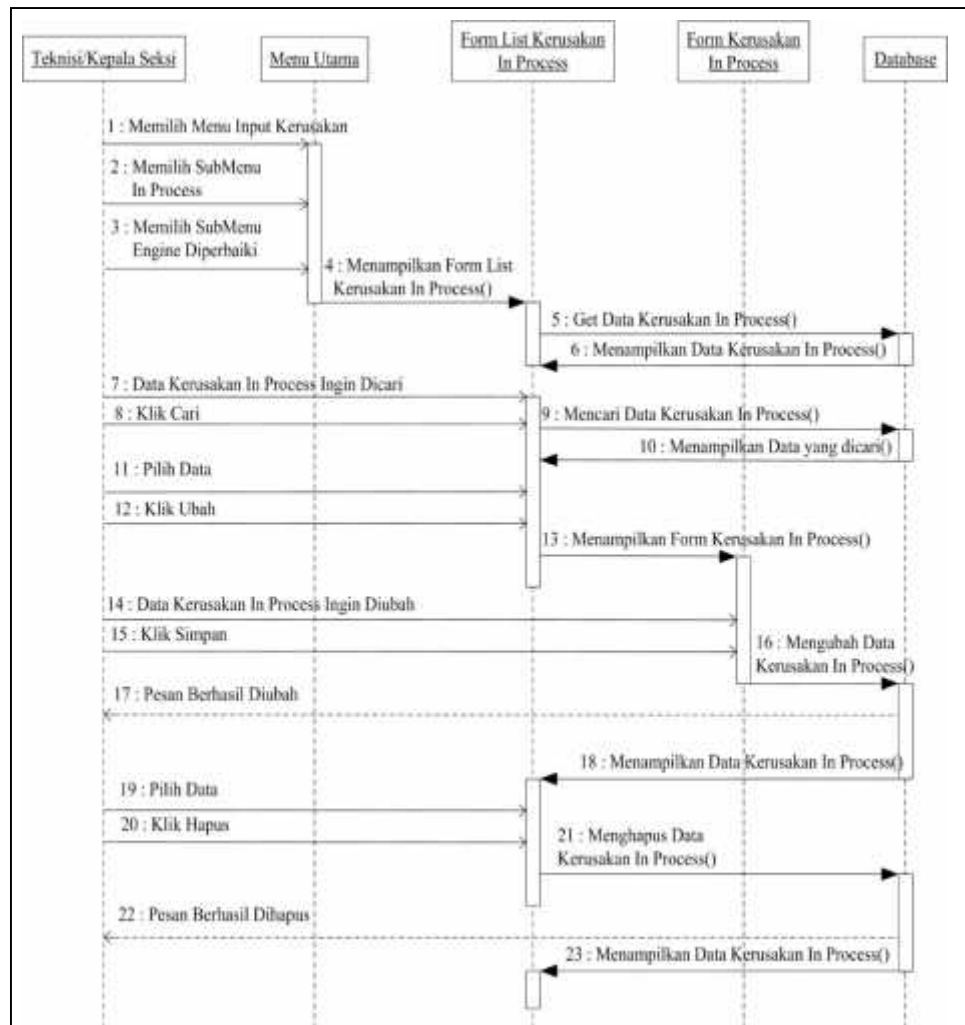
Gambar V.21 *Sequence Diagram Input Kerusakan Before Process*
(Sumber: Hasil Analisis, 2015)

b. *Sequence Diagram Input Kerusakan In Process*

Sequence diagram input laporan kerusakan *in process* menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh Teknisi/Kepala Seksi. Pengolahan data kerusakan In Process terbagi menjadi dua yaitu *engine* rusak dan *engine* diperbaiki. *Sequence diagram input* kerusakan *in process* untuk *engine* rusak digambarkan pada Gambar V.22 dan untuk *engine* diperbaiki digambarkan pada Gambar V.23.



Gambar V.22 *Sequence Diagram Input Kerusakan In Process engine rusak*
(Sumber: Hasil Analisis, 2015)



Gambar V.23 *Sequence Diagram* Input Kerusakan In Process engine diperbaiki (Sumber: Hasil Analisis, 2015)

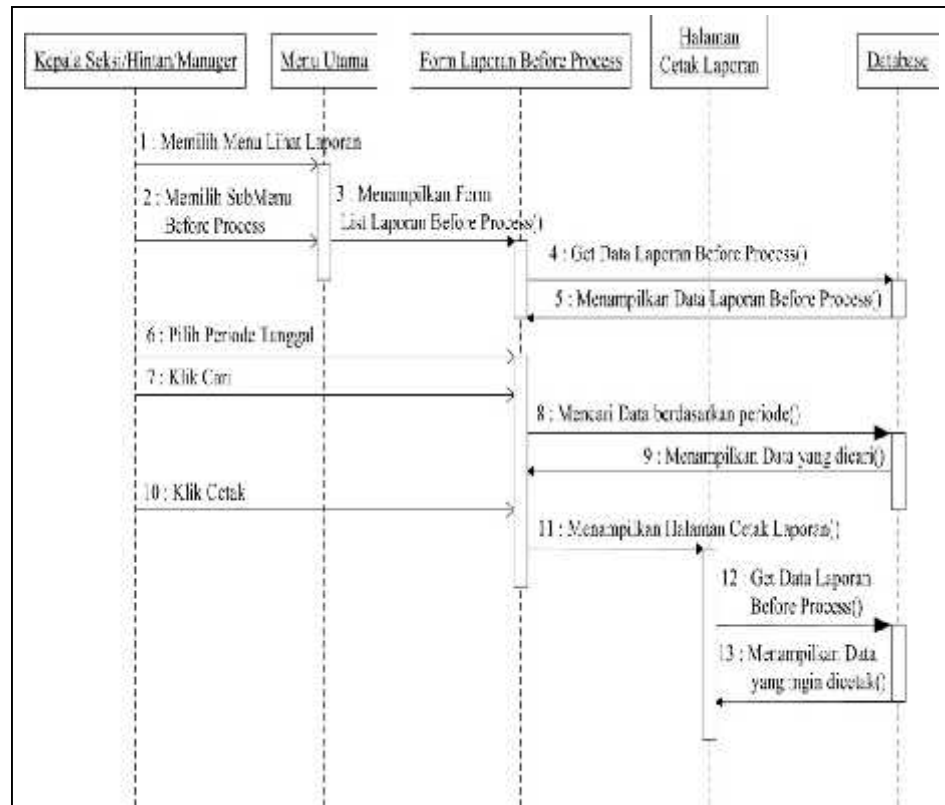
5. *Sequence Diagram* Melihat dan Mencetak Laporan Kerusakan

Sequence diagram melihat dan mencetak laporan kerusakan menggambarkan interaksi yang terjadi saat melakukan proses melihat data yang dilakukan oleh Kepala Seksi/*Hintan/Manager Assembly Engine 2W*. *Sequence diagram* melihat dan mencetak laporan kerusakan digambarkan pada Gambar V.24 dan V.25.

a. *Sequence Diagram* Melihat dan Mencetak Laporan *Before Process*

Sequence diagram melihat laporan kerusakan *before process* menggambarkan interaksi yang terjadi saat melakukan proses melihat data yang dilakukan oleh Kepala Seksi/*Hintan/Manager Assembly*

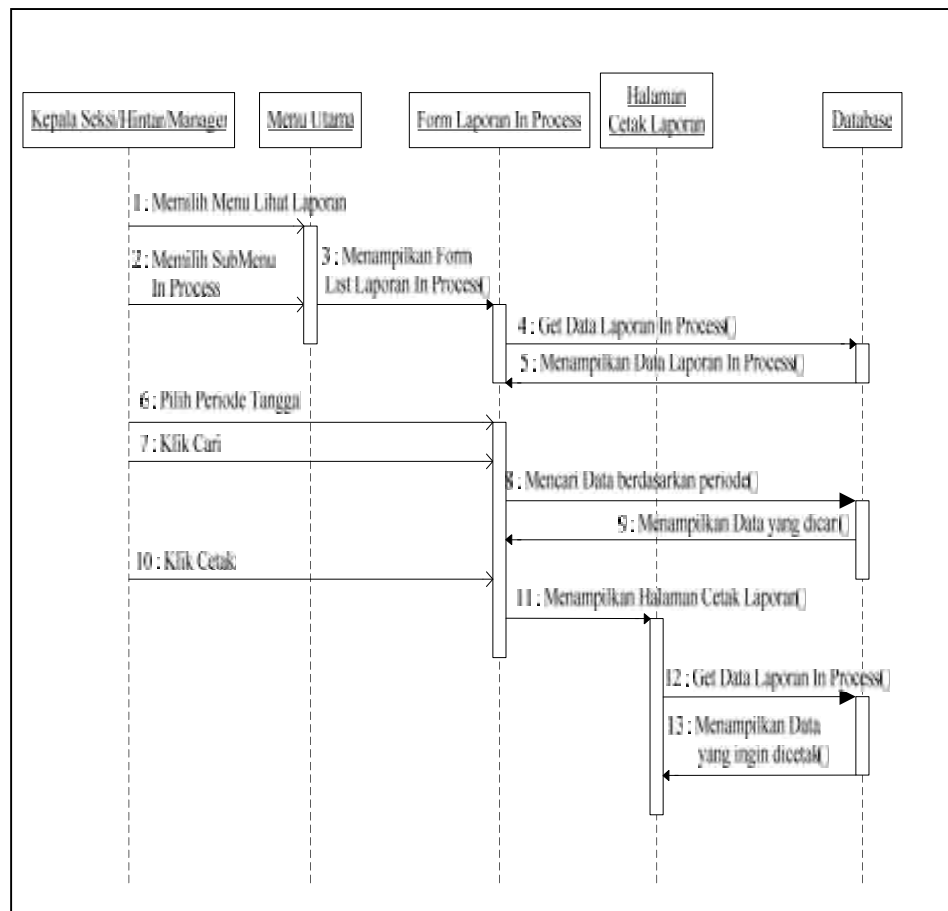
Engine 2W. *Sequence diagram* melihat laporan kerusakan *before process* digambarkan pada Gambar V.24.



Gambar V.24 *Sequence Diagram* Melihat dan Mencetak Laporan *Before Process*

(Sumber: Hasil Analisis, 2015)

- b. *Sequence Diagram* Proses Melihat dan Mencetak Laporan *In Process*
Sequence diagram melihat dan mencetak laporan kerusakan *in process* menggambarkan interaksi yang terjadi saat melakukan proses melihat data yang dilakukan oleh Kepala Seksi/*Hintan/Manager Assembly Engine 2W*. *Sequence diagram* melihat dan mencetak laporan kerusakan *in process* digambarkan pada Gambar V.25.

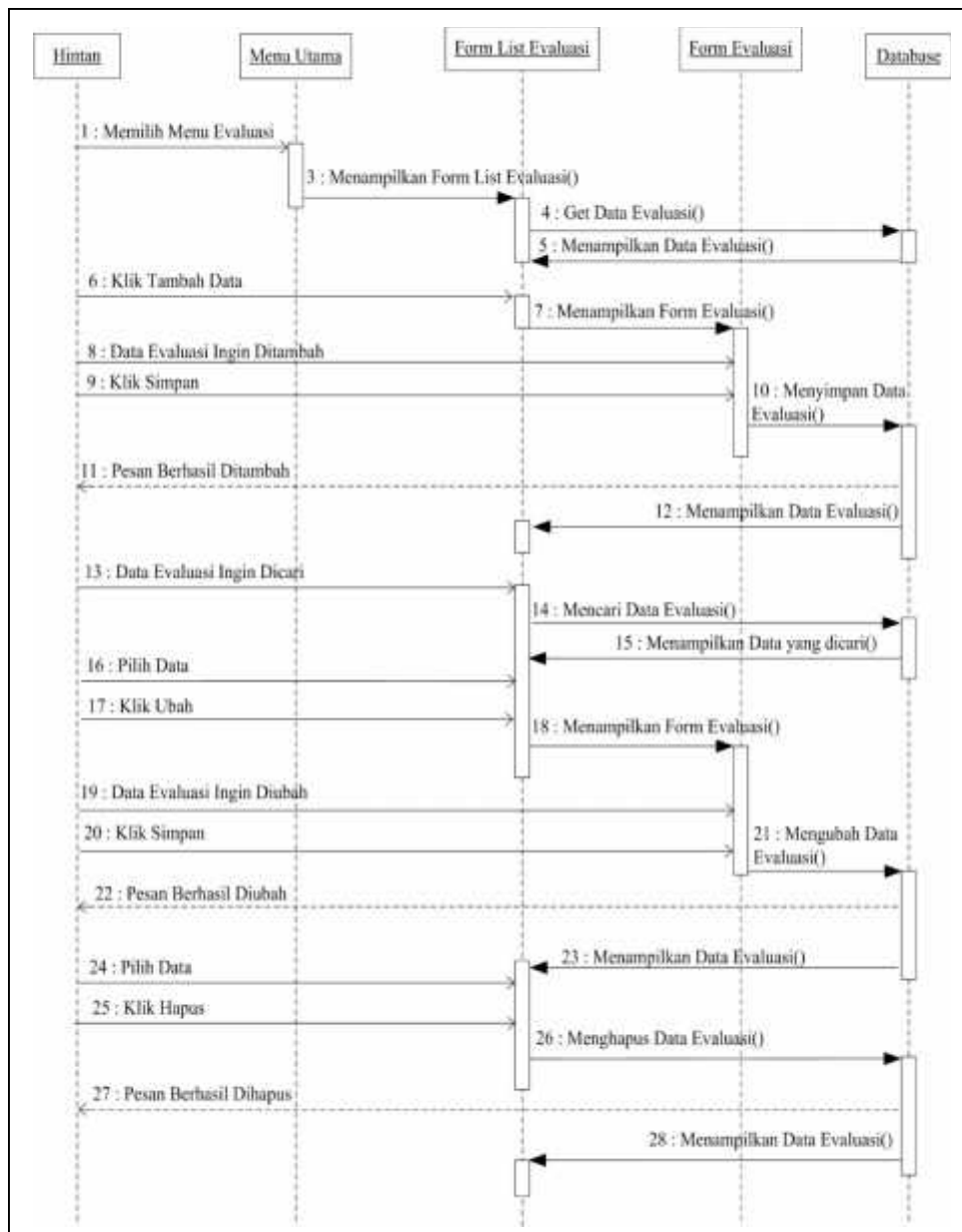


Gambar V.25 *Sequence Diagram* Melihat dan Mencetak Laporan Kerusakan *In Process*

(Sumber: Hasil Analisis, 2015)

6. *Sequence Diagram* Input Hasil Evaluasi

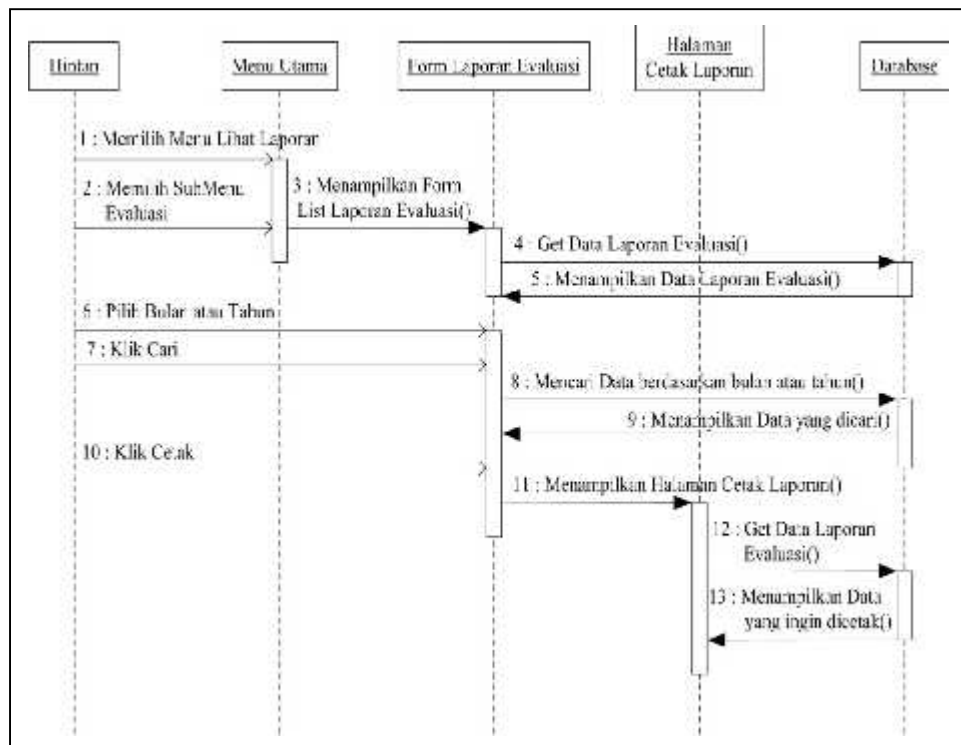
Sequence diagram input hasil evaluasi menggambarkan interaksi yang terjadi saat melakukan proses tambah, cari, ubah dan hapus data yang dilakukan oleh *Hintan* (*Quality Assurance*). *Sequence diagram* input hasil evaluasi digambarkan pada Gambar V.26.



Gambar V.26 *Sequence Diagram Input Hasil Evaluasi*
(Sumber: Hasil Analisis, 2015)

7. *Sequence Diagram* Melihat dan Mencetak Laporan Evaluasi

Sequence diagram melihat dan mencetak laporan evaluasi menggambarkan interaksi yang terjadi saat melakukan proses melihat data yang dilakukan oleh *Hintan (Quality Assurance)*. *Sequence diagram* melihat dan mencetak laporan evaluasi digambarkan pada Gambar V.27.



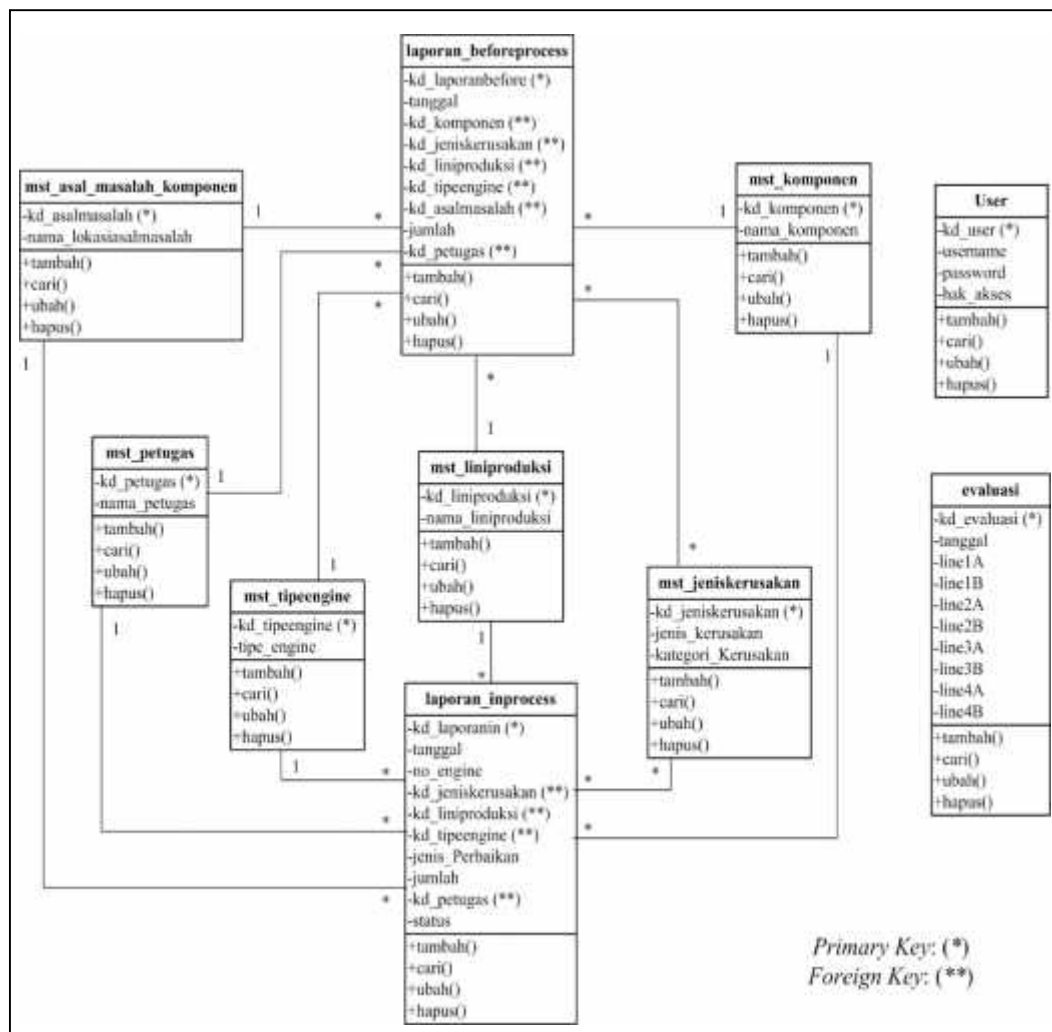
Gambar V.27 *Sequence Diagram* Melihat dan Mencetak Laporan Evaluasi
(Sumber: Hasil Analisis, 2015)

5.6 Pemodelan Data

Pemodelan data sistem informasi pengendalian kualitas *assembly engine* 2W meliputi pembuatan *class diagram* dan kamus data. Tahapan ini akan memberikan gambaran mengenai data yang terdapat pada sistem informasi yang akan dibangun.

5.6.1 *Class Diagram*

Class diagram digunakan untuk memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi pengendalian kualitas *assembly engine* 2W yang diusulkan dapat dilihat pada Gambar V.28 berikut ini:



Gambar V.28 Class Diagram Sistem Informasi Pengendalian Kualitas
(Sumber: Hasil Analisis, 2015)

Primary key untuk setiap kelas ditunjukkan oleh atribut yang memiliki tanda (*) dan untuk *foreign key* ditunjukkan oleh atribut yang memiliki tanda (**).

5.6.2 Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Berikut adalah kamus data sistem informasi pengendalian kualitas *assembly engine 2W*:

1. Spesifikasi Tabel *user*

Nama Tabel : *user*

Fungsi : Untuk menyimpan data pengguna

Tipe : Data Pengguna

Tabel V.15 Tabel *user*

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_user	<i>Tinyint</i>	1	<i>Primary Key</i>	Kode <i>user</i>
username	<i>Varchar</i>	10		Nama <i>user</i>
password	<i>Varchar</i>	10		<i>Password user</i>
hak_akses	<i>Varchar</i>	8		Hak akses berdasarkan jabatan

(Sumber: Hasil Analisis, 2015)

2. Spesifikasi Tabel *mst_komponen*

Nama Tabel : *mst_komponen*

Fungsi : Untuk menyimpan data komponen.

Tipe : Data *Master*

Tabel V.16 Tabel *mst_komponen*

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_komponen	<i>Char</i>	4	<i>Primary Key</i>	kode komponen
nama_komponen	<i>Varchar</i>	25		Nama komponen

(Sumber: Hasil Analisis, 2015)

3. Spesifikasi Tabel *mst_tipeengine*

Nama Tabel : *mst_tipeengine*

Fungsi : Untuk menyimpan data tipe *engine*

Tipe : Data *Master*

Tabel V.17 Tabel *mst_tipeengine*

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_tipeengine	<i>Tinyint</i>	2		Kode tipe <i>engine</i>
tipe_engine	<i>Varchar</i>	10		Tipe <i>engine</i>

(Sumber: Hasil Analisis, 2015)

4. Spesifikasi Tabel mst_jeniskerusakan

Nama Tabel : mst_jeniskerusakan

Fungsi : Untuk menyimpan data jenis kerusakan

Tipe : Data *Master*

Tabel V.18 Tabel mst_jeniskerusakan

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_jeniskerusakan	<i>Tinyint</i>	4	<i>Primary Key</i>	kode jenis kerusakan
jenis_kerusakan	<i>Varchar</i>	35		Jenis kerusakan
kategori_kerusakan	<i>Varchar</i>	14		

(Sumber: Hasil Analisis, 2015)

5. Spesifikasi Tabel mst_asal_masalah_komponen

Nama Tabel : mst_asal_masalah_komponen

Fungsi : Untuk menyimpan data asal masalah komponen

Tipe : Data *Master*

Tabel V.19 Tabel mst_asal_masalah_komponen

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_asalmasalah	<i>Char</i>	2	<i>Primary Key</i>	kode asal masalah
nama_lokasiasal masalah	<i>Varchar</i>	20		Nama lokasi asal masalah

(Sumber: Hasil Analisis, 2015)

6. Spesifikasi Tabel mst_petugas

Nama Tabel : mst_petugas

Fungsi : Untuk menyimpan data petugas

Tipe : Data *Master*

Tabel V.20 Tabel mst_Petugas

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_petugas	<i>Char</i>	7	<i>Primary Key</i>	Kode petugas
nama_petugas	<i>Varchar</i>	20		Nama petugas

(Sumber: Hasil Analisis, 2015)

7. Spesifikasi Tabel *mst_liniproduksi*

Nama Tabel : *mst_liniproduksi*

Fungsi : Untuk menyimpan data lini produksi

Tipe : *Data Master*

Tabel V.21 Tabel *mst_liniproduksi*

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
<i>kd_liniproduksi</i>	<i>Char</i>	2	<i>Primary Key</i>	Kode lini produksi
<i>nama_liniproduksi</i>	<i>Char</i>	7		Nama lini produksi

(Sumber: Hasil Analisis, 2015)

8. Spesifikasi Tabel *laporan_beforeprocess*

Nama Tabel : *laporan_beforeprocess*

Fungsi : Untuk menyimpan data laporan *before process*

Tipe : *Data Transaksi*

Tabel V.22 Tabel *laporan_beforeprocess*

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
<i>kd_laporanbefore</i>	<i>SmallInt</i>	1	<i>Primary Key</i>	kode transaksi setiap laporan <i>before process</i>
Tanggal	<i>Date</i>			Tanggal terjadinya kerusakan
<i>kd_komponen</i>	<i>Char</i>	4	<i>Foreign Key</i>	Kode komponen
<i>kd_jeniskerusakan</i>	<i>Tinyint</i>	4	<i>Foreign Key</i>	Kode jenis kerusakan
<i>kd_liniproduksi</i>	<i>Char</i>	2	<i>Foreign Key</i>	Kode lini produksi
<i>kd_tipeengine</i>	<i>Tinyint</i>	2	<i>Foreign Key</i>	Kode tipe <i>engine</i>
<i>kd_asalmasalah</i>	<i>Char</i>	3	<i>Foreign Key</i>	Kode asal masalah
Jumlah	<i>Tinyint</i>	2		Jumlah kerusakan
<i>kd_petugas</i>	<i>Char</i>	7	<i>Foreign Key</i>	Kode petugas

(Sumber: Hasil Analisis, 2015)

9. Spesifikasi Tabel *laporan_inprocess*

Nama Tabel : *laporan_inprocess*

Fungsi : Untuk menyimpan data laporan *in process*

Tipe : Data Transaksi

Tabel V.23 Tabel laporan_inprocess

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_laporanin	<i>Smallint</i>	1	<i>Primary Key</i>	kode transaksi setiap laporan <i>in process</i>
Tanggal	<i>Date</i>			Tanggal terjadinya kerusakan
no_engine	<i>Char</i>	6		Nomor <i>engine</i>
kd_jeniskerusakan	<i>Tinyint</i>	4	<i>Foreign Key</i>	Kode jenis kerusakan
kd_liniproduksi	<i>Char</i>	2	<i>Foreign Key</i>	Kode lini produksi
kd_tipeengine	<i>Tinyint</i>	2	<i>Foreign Key</i>	Kode tipe <i>engine</i>
jenis_perbaikan	<i>Varchar</i>	20		Jenis perbaikan
jumlah	<i>Tinyint</i>	4		Jumlah kerusakan
kd_petugas	<i>Char</i>	7	<i>Foreign Key</i>	Kode petugas
status	<i>Varchar</i>	15		Status kerusakan

(Sumber: Hasil Analisis, 2015)

10. Spesifikasi Tabel evaluasi

Nama Tabel : evaluasi

Fungsi : Untuk menyimpan data evaluasi

Tipe : Data evaluasi

Tabel V.24 Tabel evaluasi

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
kd_evaluasi	<i>Tinyint</i>	2	<i>Primary Key</i>	kode evaluasi
tanggal	<i>Date</i>			Tanggal terjadinya kerusakan
line1A	<i>Text</i>			Evaluasi line 1A
line1B	<i>Text</i>			Evaluasi line 1B
line2A	<i>Text</i>			Evaluasi line 2A

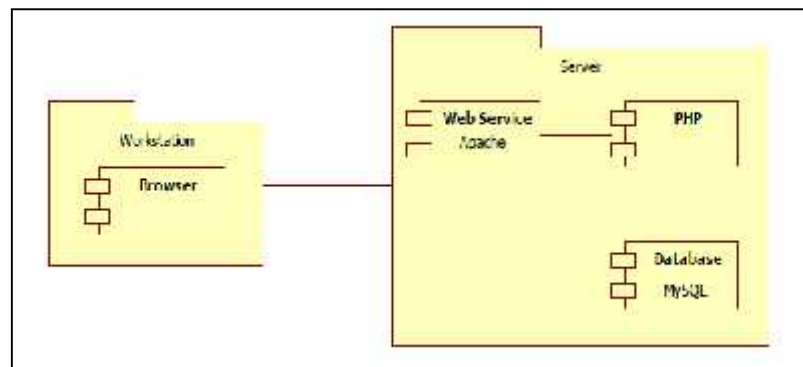
Tabel V.24 Tabel evaluasi (Lanjutan)

Nama Field	Tipe	Panjang	Keterangan	Deskripsi
line2B	Text			Evaluasi line 2B
line3A	Text			Evaluasi line 3A
line3B	Text			Evaluasi line 3B
line4A	Text			Evaluasi line 4A
line4B	Text			Evaluasi line 4B

(Sumber: Hasil Analisis, 2015)

5.6.3 Deployment Diagram

Deployment diagram menggambarkan arsitektur fisik dari sistem, seperti *web server* dan semua perangkat lunak tambahan pendukung, seperti yang dapat dilihat pada Gambar V.29 sebagai berikut:



Gambar V.29 *Deployment Diagram* Sistem Informasi Pengendalian Kualitas *Assembly Engine 2W*

(Sumber: Hasil Analisis, 2015)

Berikut adalah penjelasan Gambar V.29 *deployment diagram* sistem informasi pengendalian kualitas *assembly engine 2W*:

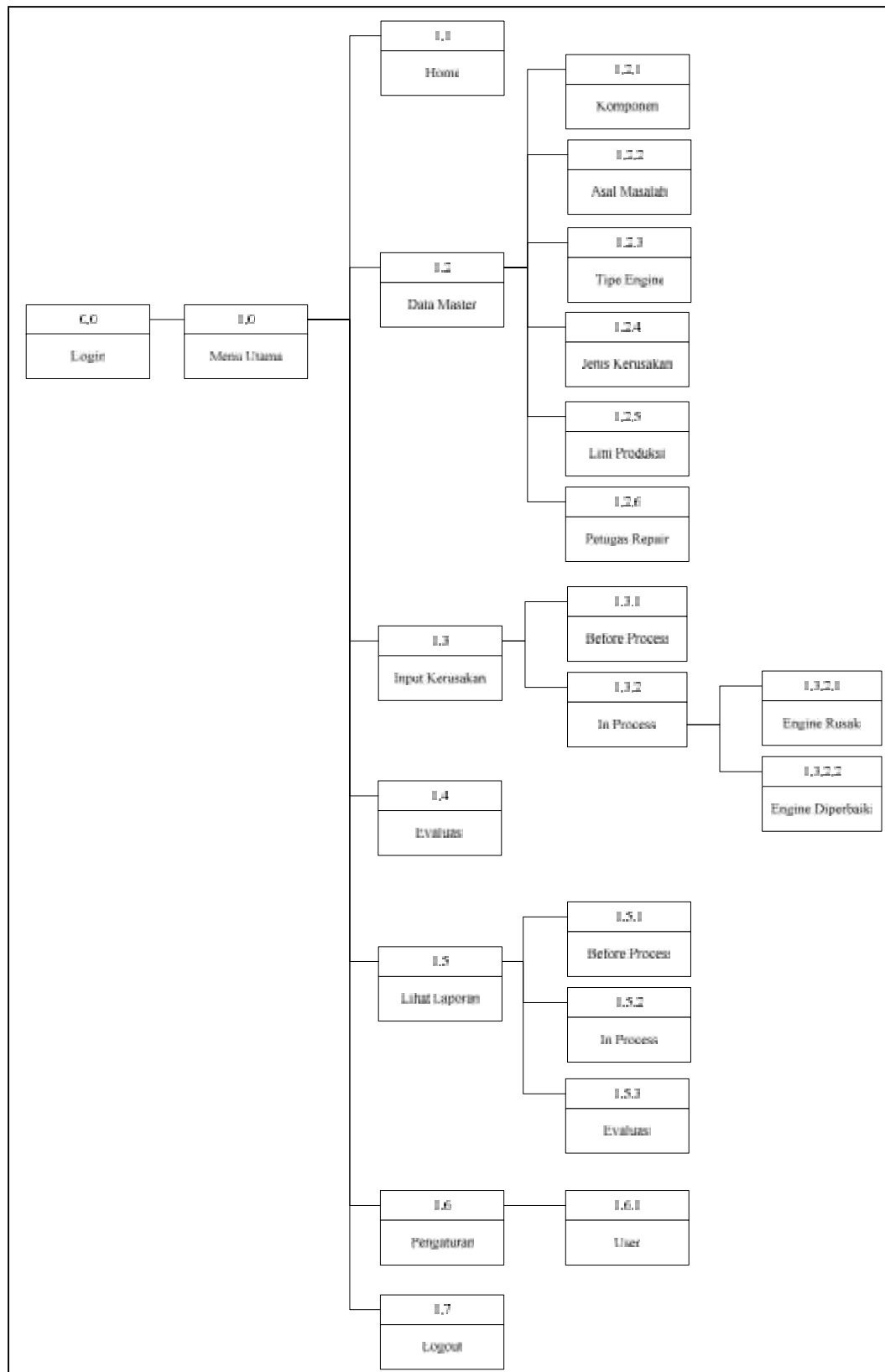
- *Workstation* adalah komputer *client* yang harus terinstal sebuah *browser* untuk menjalankan aplikasi sistem informasi pengendalian kualitas *assembly engine 2W* dan terhubung dengan *server*.
- *Server* aplikasi sistem informasi pengendalian kualitas *assembly engine 2W* terdiri dari *web service* Apache, bahasa pemrograman PHP dan perangkat lunak manajemen basis data MySQL.

5.7 Perancangan Aplikasi

Perancangan aplikasi sistem informasi pengendalian kualitas *assembly engine 2W* meliputi pembuatan HIPO (*Hierarchy plus Input-Process-Output*), *flowchart*, dan perancangan *interface* usulan. Tahapan ini akan memberikan gambaran mengenai rancangan menu dan fitur yang terdapat pada sistem informasi yang akan dibangun.

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

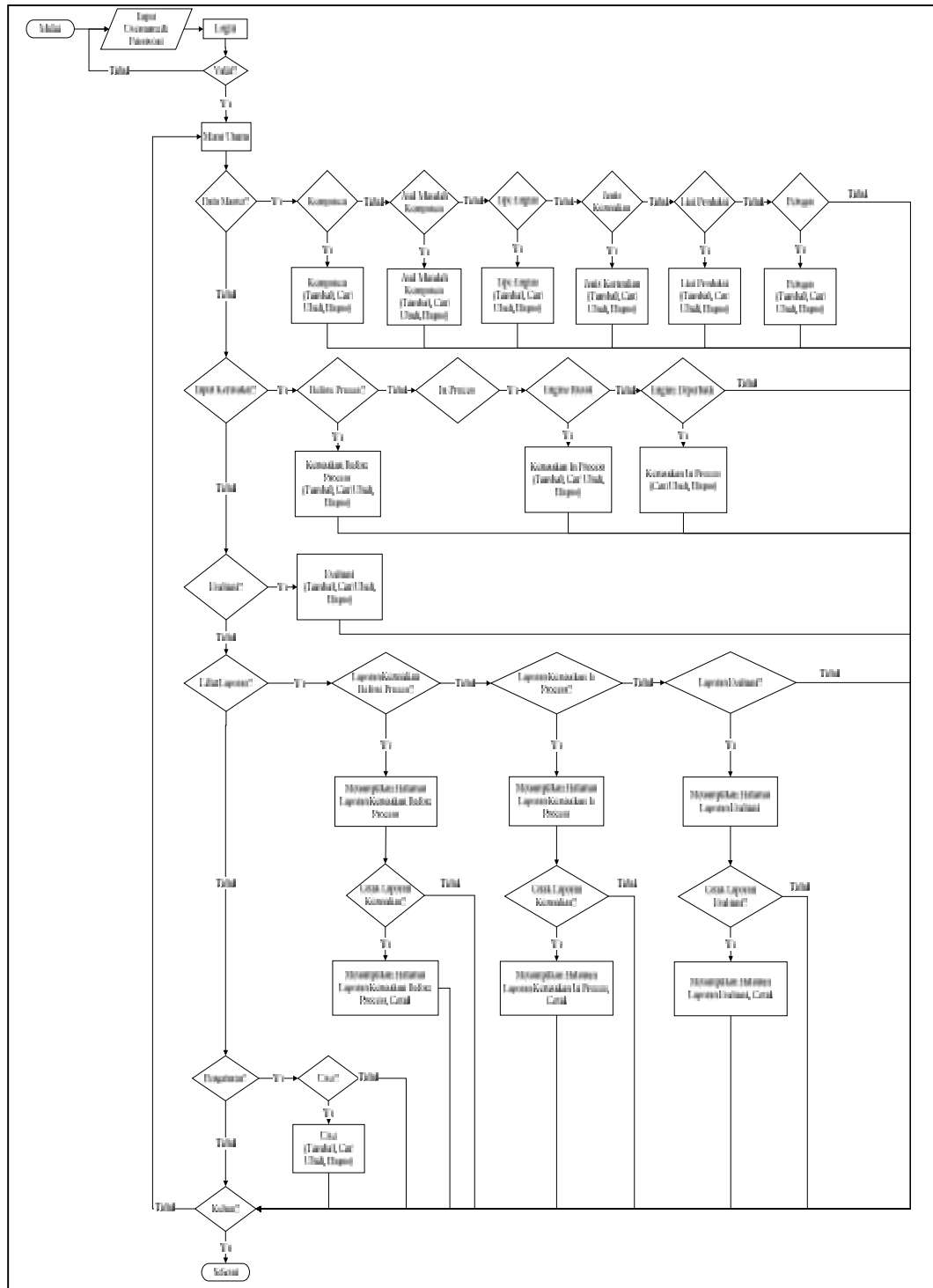
Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hirarkis. Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya. Berikut adalah Gambar V.30 perancangan HIPO sistem informasi pengendalian kualitas *assembly engine 2W* sebagai berikut:



Gambar V.30 HIPO Sistem Informasi Pengendalian Kualitas *Assembly Engine 2W*
(Sumber: Hasil Analisis, 2015)

5.7.2 Flowchart Aplikasi Usulan

Berikut adalah *flowchart* aplikasi sistem informasi pengendalian kualitas *assembly engine 2W* yang diusulkan terdapat pada Gambar V.31:



Gambar V.31 *Flowchart* Sistem Informasi Pengendalian Kualitas *Assembly Engine 2W* (Sumber: Hasil Analisis, 2015)

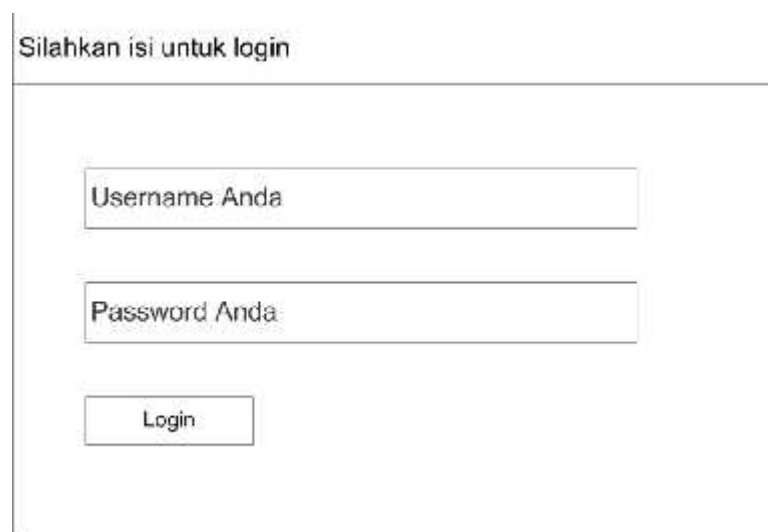
5.7.3 Perancangan *Interface* Aplikasi Usulan

Interface aplikasi pada sistem informasi pengendalian kualitas *assembly engine 2W* ini berupa rancangan tampilan yang akan dibuat, mulai dari *form login* hingga tampilan laporan yang akan dihasilkan. Perancangan *interface* aplikasi usulan adalah sebagai berikut:

1. Tampilan *login*

Form login adalah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan *username* dan *password* dengan benar. Rancangan *form login* dapat dilihat pada Gambar V.32. Adapun penjelasan tombol-tombol di *form login* sebagai berikut:

- *Login* : digunakan user untuk *login*.



The image shows a login form with a title bar that says "Silahkan isi untuk login". Below the title bar, there are three input fields: "Username Anda", "Password Anda", and a "Login" button.

Gambar V.32 Rancangan *Form Login*
(Sumber: Hasil Analisis, 2015)

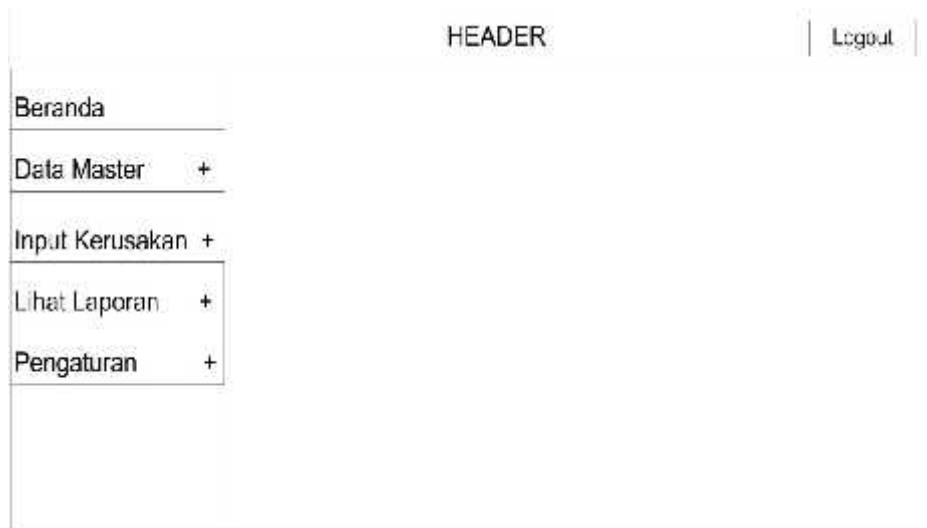
2. Tampilan menu utama

Menu utama adalah tampilan awal pada saat user berhasil *login*. Berdasarkan hak aksesnya, tampilan menu utama dibedakan menjadi empat jenis menu utama, yaitu:

a. Menu Utama Kepala Seksi

Menu utama Kepala Seksi adalah tampilan awal pada aplikasi yang berisi lima menu yaitu beranda, menu data *master*, *input* kerusakan, lihat

laporan, dan pengaturan. Setiap menu yang ada memiliki submenu. Rancangan menu utama Kepala Seksi dapat dilihat pada Gambar V.33.



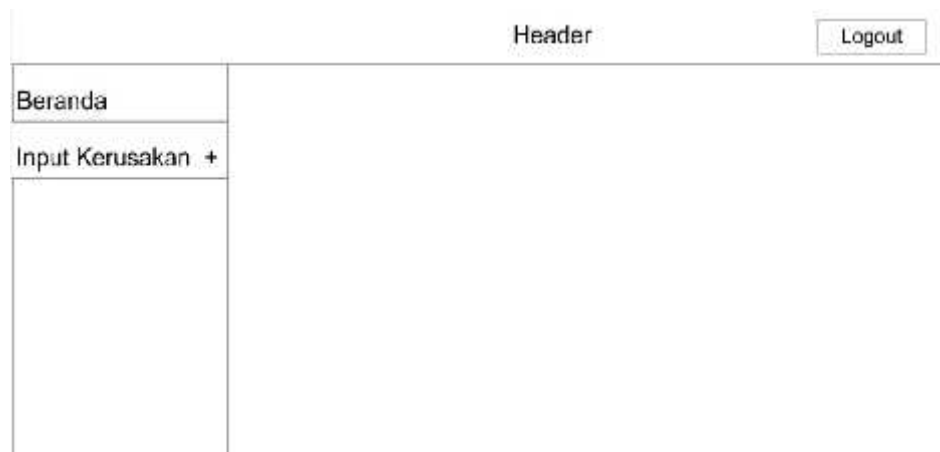
Gambar V.33 Tampilan Menu Utama Kepala Seksi
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di menu utama Kepala Seksi sebagai berikut:

- Beranda : menampilkan halaman beranda.
- *Data Master* : tombol menu data *master*.
- *Input Kerusakan* : menampilkan halaman *input* kerusakan.
- Lihat Laporan : tombol menu lihat laporan.
- *Logout* : tombol keluar dari aplikasi.

b. Menu Utama Teknisi

Menu utama Teknisi adalah tampilan awal pada aplikasi yang berisi dua menu yaitu beranda, menu *input* kerusakan. Setiap menu yang ada memiliki submenu. Rancangan menu utama Teknisi dapat dilihat pada Gambar V.34.



Gambar V.34 Tampilan Menu Utama Teknisi
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol yang diperlihatkan pada gambar V.34 adalah sebagai berikut:

- Beranda : menampilkan halaman beranda.
- *Input Kerusakan* : tombol menu *input* kerusakan.
- *Logout* : tombol keluar dari aplikasi.

c. Menu utama *Hintan* dan *Manager Assembly Engine 2W*

Menu utama *Hintan (Quality Assurance)* dan *Manager Assembly Engine 2W* adalah tampilan awal pada aplikasi yang berisi dua menu yaitu menu beranda dan lihat laporan. Masing-masing menu memiliki submenu. Rancangan menu utama *Hintan (Quality Assurance)* dapat dilihat pada Gambar V.35.



Gambar V.35 Tampilan Menu Utama *Hintan* dan *Manager Assembly Engine 2W*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol pada gambar V.35 adalah sebagai berikut:

- Beranda : menampilkan halaman beranda.
- Lihat laporan : tombol menu lihat laporan.
- *Logout* : tombol keluar dari aplikasi.

3. Tampilan mengelola data *user*

Form data *user* adalah halaman yang berisi seluruh data *user* yang dapat digunakan pada menu pengaturan lalu pilih submenu *user*. Rancangan *form* data *user* dapat dilihat pada Gambar V.36.

Username	Password	Hak Akses	Aksi
			Ubah Hapus

Gambar V.36 Rancangan Tampilan Mengelola Data *User*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol pada Gambar V.36 adalah sebagai berikut:

- Cari : mencari data *user*.
- Tambah : membuka *form* tambah *user*.
- Ubah : membuka *form* ubah data *user* yang dipilih.
- Hapus : menghapus data *user* yang dipilih.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah user yang dapat dilihat pada gambar V.37.

The image shows a web form titled "Tambah User". It contains three input fields: "Username", "Password", and "Hak Akses". The "Hak Akses" field is a dropdown menu. Below the input fields is a "Simpan" button.

Gambar V.37 Rancangan Tampilan *Form* Tambah *User*
(Sumber: Hasil Analisis, 2015)

Pada saat mengisi *form* tambah *user* terdapat beberapa pilihan hak akses yaitu Teknisi, *Hintan*, Kepala Seksi atau *Manager Assembly Engine 2W*. Adapun penjelasan tombol-tombol pada Gambar V.37 adalah sebagai berikut:

- Simpan : menyimpan data *user* yang telah diisi.

4. Mengelola data *master*

Mengelola data *master* adalah salah satu menu yang dapat diakses oleh Kepala Seksi. Aktivitas yang dilakukan dalam mengelola data master adalah sebagai berikut:

a. Tampilan mengelola komponen

Form data komponen adalah halaman yang berisi seluruh data komponen. Rancangan *form* data komponen dapat dilihat pada Gambar V.38.



The image shows a web form titled "Data Komponen". It features a "Tambah" button at the top left. Below it is a search section with an input field and a "Cari" button. The main part of the form is a table with three columns: "Kode Komponen", "Nama Komponen", and "Aksi". The "Aksi" column contains two buttons: "Ubah" and "Hapus".

Kode Komponen	Nama Komponen	Aksi
		Ubah Hapus

Gambar V.38 Rancangan Tampilan *Form* Data Komponen
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol pada gambar V.38 adalah sebagai berikut:

- Cari : mencari data komponen.
- Tambah : membuka *form* komponen.
- Ubah : membuka *form* ubah data komponen yang dipilih.
- Hapus : menghapus data komponen yang dipilih.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah komponen yang dapat dilihat pada gambar V.39.



The image shows a web form titled "Tambah Komponen". It contains two input fields: "Kode Komponen" and "Nama Komponen". At the bottom of the form is a "Simpan" button.

Gambar V.39 Rancangan Tampilan *Form* Tambah Komponen
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* tambah komponen sebagai berikut:

- Simpan : menyimpan data komponen yang telah diisi.

b. Tampilan mengelola jenis kerusakan

Form data jenis kerusakan adalah halaman yang berisi seluruh data jenis kerusakan. Rancangan *form* data jenis kerusakan dapat dilihat pada Gambar V.40.

Jenis Kerusakan	Kategori Kerusakan	Aksi
		Ubah Hapus

Gambar V.40 Rancangan Tampilan *Form* data Jenis Kerusakan
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* data jenis kerusakan sebagai berikut:

- Cari : mencari data jenis kerusakan.
- Tambah : membuka *form* tambah jenis kerusakan.
- Ubah : membuka *form* ubah data jenis kerusakan yang dipilih.
- Hapus : menghapus data jenis kerusakan yang dipilih.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah jenis kerusakan yang dapat dilihat pada gambar V.41.

Gambar V.41 Rancangan Tampilan *Form* Tambah Jenis Kerusakan
(Sumber: Hasil Analisis, 2015)

Pada saat *form* jenis kerusakan, terdapat dua jenis kategori yang dapat dipilih yaitu *in process* atau *before process*. Adapun penjelasan tombol-tombol di *form* tambah jenis kerusakan sebagai berikut:

- Simpan : menyimpan data jenis kerusakan yang telah diisi.

c. Tampilan mengelola tipe *engine*

Form data tipe *engine* adalah halaman yang berisi seluruh data tipe *engine*. Rancangan *form* data tipe *engine* dapat dilihat pada Gambar V.42.

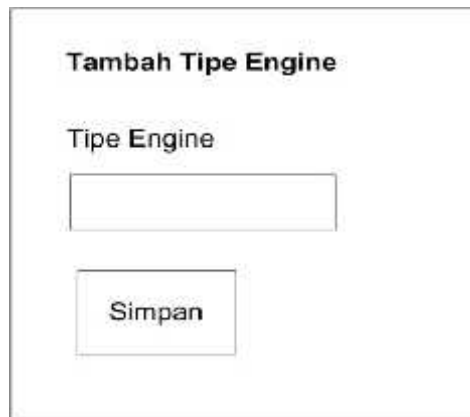
Gambar V.42 Rancangan Tampilan *Form* Data Tipe *Engine*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* data tipe *engine* sebagai berikut:

- Cari : mencari data tipe *engine*.

- Tambah : membuka *form* tambah tipe *engine*.
- Ubah : membuka *form* ubah data tipe *engine* yang dipilih.
- Hapus : menghapus data tipe *engine* yang dipilih.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah komponen yang dapat dilihat pada gambar V.43.

The image shows a rectangular form titled "Tambah Tipe Engine". Below the title, there is a label "Tipe Engine" followed by a horizontal text input field. At the bottom of the form, there is a rectangular button labeled "Simpan".

Gambar V.43 Rancangan Tampilan *Form* Tambah Tipe *Engine*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* tambah tipe *engine* sebagai berikut:

- Simpan : menyimpan data tipe *engine* yang telah diisi.

d. Tampilan *form* data lini produksi

Form data lini produksi adalah halaman yang berisi seluruh data lini produksi. Rancangan *form* data lini produksi dapat dilihat pada Gambar V.44.

Data Lini Produksi

Tambah

_____ Cari

Kode lini produksi	Nama Lini Produksi	Aksi
		Ubah Hapus

Gambar V.44 Rancangan Tampilan *Form* Data Lini Produksi
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* data lini produksi sebagai berikut:

- Cari : mencari data lini produksi.
- Tambah : membuka *form* tambah lini produksi.
- Ubah : membuka *form* ubah data lini produksi yang dipilih.
- Hapus : menghapus data lini produksi yang dipilih.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah lini produksi yang dapat dilihat pada gambar V.45. Adapun penjelasan tombol-tombol di *form* tambah lini produksi sebagai berikut:

- Simpan : menyimpan data lini produksi yang telah diisi.

Tambah Lini Produksi

Kode Lini Produksi

Nama Lini Produksi

Simpan

Gambar V.45 Rancangan Tampilan *Form* Tambah Lini Produksi
(Sumber: Hasil Analisis, 2015)

e. Tampilan *form* data asal masalah komponen

Form data asal masalah komponen adalah halaman yang berisi seluruh data asal masalah komponen. Rancangan *form* data asal masalah komponen dapat dilihat pada Gambar V.46.

Kode Asal Masalah	Nama Lokasi Asal Masalah	Aksi
		Ubah Hapus

Gambar V.46 Rancangan Tampilan *Form* Data Asal Masalah Komponen
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* data asal masalah komponen sebagai berikut:

- Cari : mencari data asal masalah komponen.
- Tambah : membuka *form* tambah asal masalah komponen.
- Ubah : membuka *form* ubah data asal masalah komponen
- Hapus : menghapus data asal masalah komponen.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah asal masalah komponen yang dapat dilihat pada gambar V.47.

Gambar V.47 Rancangan Tampilan *Form* Tambah Asal Masalah Komponen
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* tambah asal masalah komponen sebagai berikut:

- Simpan : menyimpan data asal masalah komponen yang telah diisi.

f. Tampilan *form* data petugas

Form data petugas adalah halaman yang berisi seluruh data petugas.

Rancangan *form* data petugas dapat dilihat pada Gambar V.48.

Data Petugas		
Tambah		
[Input Field]		Cari
Kode Petugas	Nama Petugas	Aksi
		Ubah Hapus

Gambar V.48 Rancangan Tampilan *Form* Data Petugas
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* data petugas sebagai berikut:

- Cari : mencari data petugas.
- Tambah : membuka *form* tambah petugas.
- Ubah : membuka *form* ubah data petugas yang dipilih.
- Hapus : menghapus data petugas yang dipilih.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah petugas yang dapat dilihat pada gambar V.49.

Gambar V.49 Rancangan Tampilan *Form* Petugas
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* tambah petugas sebagai berikut:

- Simpan : menyimpan data petugas yang telah diisi.

5. Tampilan *input* kerusakan

Menu *input* kerusakan memiliki dua submenu yaitu *before process* dan *in process*. Halaman yang berisi seluruh data laporan kerusakan *before process*. *Input* kerusakan dapat dilakukan oleh Kepala Seksi dan Teknisi. Namun, dalam hal ini teknisi hanya bisa melakukan proses tambah sedangkan ubah dan hapus hanya bisa dilakukan oleh kepala seksi.

a. *Input* Kerusakan *Before Process*

Rancangan *form* data kerusakan *before process* dapat dilihat pada Gambar V.50.

Tanggal	Kode Kerusakan	Nama Komponen	Jenis Kerusakan	Lrt Produk	Tipe Engine	Lokasi Reper	Jumlah	Nama Petugas	Mhs
									Uchir / Hagus

Gambar V.50 Rancangan Tampilan *Form* Data Kerusakan *Before Process*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* data kerusakan *before process* sebagai berikut:

- Cari : mencari data kerusakan *before process*.
- Tambah : membuka *form* tambah kerusakan *before process*.
- Ubah : membuka *form* ubah data kerusakan *before process*.
- Hapus : menghapus data laporan kerusakan *before process*.

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah kerusakan *before process* yang dapat dilihat pada gambar V.51.

Gambar V.51 Rancangan Tampilan *Form* Tambah Kerusakan *Before Process*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* tambah kerusakan *before process* sebagai berikut:

- Simpan : menyimpan data kerusakan yang telah diisi.

b. *Input Kerusakan In Process*

Rancangan *form* data kerusakan *in process* terbagi menjadi dua yaitu *engine* rusak dan *engine* diperbaiki, dapat dilihat pada Gambar V.52 dan Gambar V.54.

Tanggal	No Engine	Jenis kerusakan	Unit Produksi	Tipe Engine	Status	Aksi
						Hapus

Gambar V.52 Rancangan Tampilan *Form Data Kerusakan In Process*
(Sumber: Hasil Analisis, 2015)

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah kerusakan *in process* yang dapat dilihat pada gambar V.53.

Gambar V.53 Rancangan Tampilan *Form Input Laporan In Process*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* tambah kerusakan *in process* sebagai berikut:

- Simpan : menyimpan data kerusakan yang telah diisi.

Setelah Teknisi melakukan perbaikan *engine*, selanjutnya Teknisi akan memasukan data hasil perbaikan dan mengubah status kerusakan.

Rancangan *form* data kerusakan *in process* untuk *engine* yang sudah diperbaiki dapat dilihat pada Gambar V.54.

Perbaikan Kerusakan In Process

Masukkan nomor Engine

Tanggal	No Engine	Jenis Kerusakan	No Perbaikan	Tipe Engine	Jenis Perbaikan	Status	Peragas	Aspi
								Utah Kerusak

Gambar V.54 Rancangan Tampilan *Form* Data Kerusakan *In Process engine* diperbaiki
(Sumber: Hasil Analisis, 2015)

Untuk mengolah status kerusakan, Teknisi harus melakukan pencarian data dengan mengisi nomor *engine*, setelah data ditemukan teknisi akan melakukan perubahan data dengan klik ubah.

6. Tampilan *input* evaluasi

Menu *input* evaluasi merupakan menu yang berisi seluruh data hasil evaluasi. *Input* evaluasi dapat dilakukan oleh *Hintan (Quality Assurance)*. Rancangan *form* data evaluasi dapat dilihat pada Gambar V.55.

Data Evaluasi

Pilih Bulan atau Tahun Untuk Melakukan Perbaikan

Bulan Tahun

Bulan Tahun	Line 1A	Line 1B	Line 2A	Hasil Evaluasi	Line 2B	Line 3A	Line 3B	Line 4A	Line 4B	Detail
										Ubah Hapus

Gambar V.55 Rancangan Tampilan *Form* Data Evaluasi
(Sumber: Hasil Analisis, 2015)

Ketika tombol tambah di klik maka aplikasi akan menampilkan *form* tambah hasil evaluasi yang dapat dilihat pada gambar V.56.

Gambar V.56 Rancangan Tampilan *Form* Tambah hasil Evaluasi
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* tambah kerusakan *in process* sebagai berikut:

- Simpan : menyimpan data kerusakan yang telah diisi.

7. Tampilan melihat laporan

Menu lihat laporan memiliki tiga submenu yaitu *before process*, *in process* dan evaluasi. Menu ini dapat diakses oleh tiga aktor yaitu Kepala Seksi, *Hintan* dan *Manager*.

a. Lihat laporan *before process*

Data laporan *before process* terdapat perbedaan untuk data yang ditampilkan Kepala Seksi, *Manager* dan *Hintan*. Kepala Seksi dan *Manager* dapat melihat secara detail data dari keseluruhan laporan *before process*, sedangkan *Hintan* hanya melihat data yang ditampilkan sesuai dengan formulir laporan kerusakan yang berjalan. Adapun rancangan tampilan lihat laporan *before process* yang dilakukan *Hintan* diperlihatkan pada Gambar V.57 dan rancangan tampilan lihat laporan

before process yang dilakukan *Manager* dan Kepala Seksi ditampilkan pada Gambar V.58.

Laporan Before Process

Tanggal Awal Tanggal Akhir CARI CETAK

Tanggal	Tipe Engine	Kode Komponen	Nama Komponen	Lokasi Recept	Jumlah	Jenis Kerusakan

Gambar V.57 Lihat Laporan Kerusakan *Before Process* oleh *Hintan*
(Sumber: Hasil Analisis, 2015)

Laporan Before Process

Tanggal Awal Tanggal Akhir CARI CETAK

Tanggal	Kode Komponen	Nama Komponen	Jenis kerusakan	Lini Produk	Tipe Engine	Lokasi Recept	Jumlah	Nama Pelugas

Gambar V.58 Lihat Laporan Kerusakan *Before Process* oleh *Manager* dan Kepala Seksi
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di *form* laporan kerusakan *in process* sebagai berikut:

- Cari : mencari laporan berdasarkan periode tanggal.
- Cetak : mencetak laporan kerusakan yang telah dicari.

b. Lihat laporan *in process*

Data laporan kerusakan *in process* adalah laporan yang berisi setiap data *engine* yang rusak pada saat proses produksi. Rancangan melihat laporan kerusakan *in process* dapat dilihat pada Gambar V.59.

The screenshot shows a web interface titled "Laporan In Process". At the top, there are two date pickers labeled "Tanggal Awal" and "Tanggal Akhir", followed by two buttons labeled "CARI" and "CETAK". Below this is a table with the following columns: "Tanggal", "No Engine", "Jenis kerusakan", "Lm. Produksi", "Tipe Engine", "Jenis Pekerjaan", "Jumlah", and "Nama Pelugas". The table is currently empty.

Gambar V.59 Rancangan Tampilan Melihat Laporan Kerusakan *In Process*
(Sumber: Hasil Analisis, 2015)

Adapun penjelasan tombol-tombol di melihat laporan kerusakan *in process* sebagai berikut:

- Cari : mencari laporan berdasarkan periode tanggal.
- Cetak : mencetak laporan kerusakan yang telah dicari.

c. Lihat laporan Evaluasi

Data laporan evaluasi adalah laporan yang berisi setiap data evaluasi yang dilakukan oleh *Hintan*. Rancangan melihat laporan *evaluasi* dapat dilihat pada Gambar V.60.

Laporan Evaluasi

Pilih Dulan atau Tahun: Untuk Melakukan Pencarian:

Bulan Tahun Can Cetak

Dulan	Hasil Evaluasi							
	Line 1A	Line 1B	Line 2A	Line 2E	Line 3A	Line 3B	Line 4A	Line 4B

Gambar V.60 Rancangan Tampilan Melihat Laporan Evaluasi
(Sumber: Hasil Analisis, 2015)

5.8 Implementasi Sistem

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

1. Kebutuhan *Hardware*

- *Processor* : Minimal *Processor Pentium IV*.
- *RAM* : Minimal *RAM 512 MB*.
- *Harddisk* : Minimal *Harddisk 64 GB*.
- *Mouse, Keyboard, Monitor* sebagai peralatan antar muka.
- *Printer* sebagai media *output*.

2. Kebutuhan *Software*

- *Sistem Operasi* : *Microsoft Windows 7*
- *Web Server* : *Apache*
- *Database Server* : *MySQL versi 5.5*
- *Web Browser* : *Mozilla firefox, Google Chrome*

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil pengumpulan, pengolahan, analisis data dan perancangan sistem usulan maka dapat diambil kesimpulan bahwa:

1. Sistem informasi pengendalian kualitas *assembly engine 2W* berbasis web ini dapat mempercepat dan mempermudah proses pengolahan data kerusakan *engine* karena pengisian formulir kerusakan *engine* sudah dilakukan secara terkomputerisasi.
2. Sistem informasi pengendalian kualitas *assembly engine 2W* ini dapat mempermudah *manager assembly engine 2W* dalam membaca setiap laporan kerusakan *assembly engine* karena sudah disajikan dalam bentuk tabel yang lebih informatif dan mudah dipahami.
3. Sistem informasi pengendalian kualitas *assembly engine 2W* ini dapat mempermudah proses penyimpanan data kerusakan *engine* dan mencegah terjadinya kehilangan data karena setiap kerusakan *engine* sudah disimpan dengan menggunakan *database MySQL*.

6.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem informasi pengendalian kualitas *assembly engine 2W* ini selanjutnya adalah sebagai berikut:

1. Dalam penerapan sistem yang lama ke sistem yang baru sebaiknya dilakukan secara bertahap, hal ini untuk menghindari terjadinya kegagalan terhadap sistem yang baru.
2. Perlunya pemeliharaan terhadap sistem yang telah dibuat agar tetap berjalan dengan baik.
3. Diharapkan bagi pembaca tugas akhir ini dapat membuat sistem informasi pengendalian kualitas *assembly engine 2W* ini menjadi sistem informasi yang berbasis *SMS gateway*.

DAFTAR PUSTAKA

- Anhar. 2010. *Panduan Menguasai PHP & MySQL Secara Otodidak*. Cetakan Pertama. Jakarta: Mediakita.
- Anonim. 2014. *Profile Book Departemen Produksi Seksi Assembly Engine 2W*. Edisi kedua. Jakarta: PT Suzuki Indomobil Motor.
- Anonim. 2014. *Sejarah dan Produksi PT Suzuki Indomobil Motor*. edisi pertama. Jakarta: PT Suzuki Indomobil Motor.
- Assauri, Sofjan. 2004. *Manajemen Operasi dan Produksi*. Jakarta: LP FE UI Dwiwinarno.
- Davis, G.B. 1991. *Kerangka Dasar Sistem Informasi Manajemen Bagian 1*. Jakarta: PT Pustaka Binamas Pressindo.
- Gaspersz, Vincent. 2005. *Total Quality Management*. Jakarta: PT Gramedia Pusaka Utama.
- Ginting, Rosnani. 2007. *Sistem Produksi*. Yogyakarta: Graha Ilmu.
- Hasibuan, Malayu S.P. 1996. *Manajemen Dasar, Pengertian dan Masalah*. Jakarta: PT Bumi Aksara.
- Indrajit. 2001. *Analisis dan Perancangan Sistem Berorientasi Object*. Bandung: Informatika.
- Jogiyanto. 2005. *Analisis & Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*. Cetakan Ketiga. Yogyakarta: Andi.
- Lani, Sidharta. 1995. *Pengantar Sistem Informasi Bisnis*. Jakarta: PT Elex Media Komputindo.

- McLeod, Jr., Raymond; Schell, George P. 2011. *Sistem Informasi Manajemen*. Edisi 10. Jakarta: Salemba Empat.
- Murdick, R.G. 1991. *Sistem Informasi untuk Manajemen Modern*. Jakarta: Erlangga.
- Nazir, Moh. 2011. *Metode Penelitian*. Bogor: Ghalia Indonesia.
- Nugroho, Bunafit. 2008. *Aplikasi Pemrograman Web Dinamis dengan PHP dan MySQL*. Cetakan Pertama. Yogyakarta: Gava Media.
- Purnomo, Hari. 2004. *Pengantar Teknik Industri*. Yogyakarta: Graha Ilmu.
- Rosa, A. S., dan M. Shalahuddin. 2014. *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Cetakan Kedua. Bandung: Informatika.
- Solichin, Achmad. 2009. *Pemrograman Web dengan PHP & MySQL*. Jakarta: Universitas Budi Luhur.
- Suhendar, A., dan Gunadi, H. 2002. *Visual Modelling Menggunakan UML dan Relational Rose*. Bandung: Informatika.
- Sutabri, Tata. 2004. *Analisa Sistem Informasi*. Yogyakarta: Andi.
- Sutanta, Edhy. 2003. *Sistem Informasi manajemen*. Yogyakarta: Graha Ilmu.
- Tim Penyusun KBBI. 2008. *Kamus Besar Bahasa Indonesia*. Edisi Keempat. Jakarta: Gramedia Pustaka Utama.
- Welsch, Glenn A, Ronald W. Hilton. 2003. *Anggaran, Perencanaan dan Pengendalian, alih bahasa Purwatiningsih*. Jakarta: Salemba Empat.
- Yamit, Zulian. 2013. *Manajemen Kualitas Produk dan Jasa*. edisi keenam. Yogyakarta: Ekonisia.

Uml-diagrams, <http://www.uml-diagrams.org/multiplicity.html>, (Tanggal akses: 26 Oktober 2015).

TAMPILAN PROGRAM

1. Login



2. Menu Utama



3. Data Komponen

Data Komponen

→ tambah

Nama Komponen

Kode Komponen	Nama Komponen	Aksi	
0001	Cover Cylinder Head	<input type="button" value="Ubah"/>	<input type="button" value="hapus"/>
0002	Cover Crank Box	<input type="button" value="Ubah"/>	<input type="button" value="hapus"/>
0003	Crank Case RT	<input type="button" value="Ubah"/>	<input type="button" value="hapus"/>
0004	Cap Crank Case L	<input type="button" value="Ubah"/>	<input type="button" value="hapus"/>
0005	Cover Magnat	<input type="button" value="Ubah"/>	<input type="button" value="hapus"/>
0006	Cover Oilpan	<input type="button" value="Ubah"/>	<input type="button" value="hapus"/>
0007	Crank Case LH	<input type="button" value="Ubah"/>	<input type="button" value="hapus"/>

4. Data Asal Masalah Komponen



Kode Asal Masalah	Nama Luaran Masalah	Aksi
AS	ASSEMBLY	CRUD
DE	DEFECT	CRUD
LE	LINE LISTING	CRUD
MT	MANUFACTURING	CRUD
CT	COST CENTER	CRUD
PT	PLANT	CRUD
QA	QUALITY ASSURANCE	CRUD

5. Data Tipe Engine



Tipe Engine	Aksi
XDS11	CRUD
XDS1100	CRUD
XDS53RD	CRUD
XSD17000	CRUD
XSD17001	CRUD

6. Data Jenis Kerusakan



Jenis Kerusakan	Kategori Kerusakan	Aksi
Oil Terlempar	Defect Process	CRUD
Cover mesin rusak	in process	CRUD
Engine tidak	in process	CRUD
Engine tidak bisa berjalan	in process	CRUD
Flak dalam tidak rata	in process	CRUD
Flak crank tidak halus	in process	CRUD
Flak crank tidak lurus	in process	CRUD

7. Data Lini Produksi



Kode Lini Produksi	Nama Lini Produksi	Aksi
1A	Line 1A	CRUD
1B	Line 1B	CRUD
2A	Line 2A	CRUD
2B	Line 2B	CRUD
3A	Line 3A	CRUD

8. Data Petugas

Data Petugas

➔ Tambah

Filter:

No. Petugas	Nama Petugas	Status	Aksi
123456789	Andriyanto	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
123456789	Andriyanto	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
123456789	Andriyanto	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
123456789	Andriyanto	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
123456789	Andriyanto	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
123456789	Andriyanto	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

9. Data User

Data User

➔ Tambah

Filter:

Username	Password	Role User	Aksi
u1	123456	ADMIN	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
u2	123456	TRUCK	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
u3	123456	IBRAS	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
u4	123456	PASIR	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

10. Kerusakan Before Process

Kerusakan Before Process

➔ Tambah

Filter: Tanggal Untuk Melakukan Pencarian:

Tanggal	No. Kolaborasi	Nama Kolaborasi	Jenis Kerusakan	Unit Produk	Tipe Engine	Status	Uraian	Area
2015-03-18	12345	Andriyanto	Andriyanto	12345	12345	aktif	Andriyanto	Andriyanto
2015-03-18	12345	Andriyanto	Andriyanto	12345	12345	aktif	Andriyanto	Andriyanto
2015-03-18	12345	Andriyanto	Andriyanto	12345	12345	aktif	Andriyanto	Andriyanto

11. Kerusakan In Process

Kerusakan In Process

➔ Tambah

Filter: Tanggal Untuk Melakukan Pencarian:

Tanggal	No. Engine	Jenis Kerusakan	Unit Produk	Tipe Engine	Status	Aksi
2015-03-18	12345	Andriyanto	12345	12345	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>
2015-03-18	12345	Andriyanto	12345	12345	aktif	<input type="button" value="Edit"/> <input type="button" value="Hapus"/>

15. Laporan Evaluasi

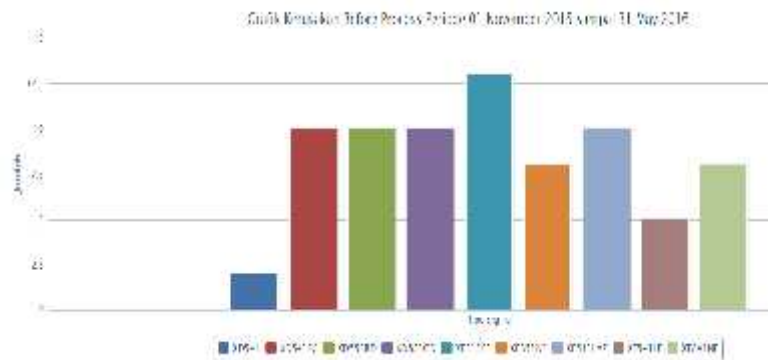
Laporan Evaluasi

REKAM JEKIF UMUM - RENCANA KEMAJUAN PERKANTORAN

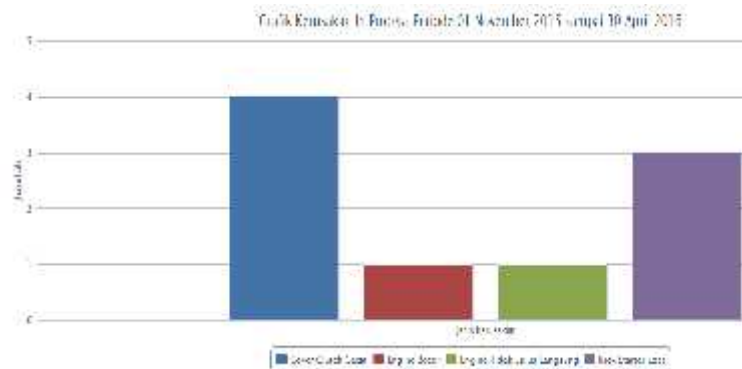
Tahun: 2015

Kategori	Tahap Pelaksanaan				Tahap Penilaian			
	1. Perencanaan	2. Pelaksanaan	3. Monitoring	4. Evaluasi	1. Perencanaan	2. Pelaksanaan	3. Monitoring	4. Evaluasi
Indikator Kinerja Utama (IKU)	1. Perencanaan	2. Pelaksanaan	3. Monitoring	4. Evaluasi	1. Perencanaan	2. Pelaksanaan	3. Monitoring	4. Evaluasi
Indikator Kinerja Tambahan (IKT)	1. Perencanaan	2. Pelaksanaan	3. Monitoring	4. Evaluasi	1. Perencanaan	2. Pelaksanaan	3. Monitoring	4. Evaluasi

16. Grafik Before Process



17. Grafik In Process



KODE PROGRAM

1. Login

```

<?php
include 'class.php';
?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-
scale=1.0" />
  <title>Login Sistem</title>
  <!-- BOOTSTRAP STYLES-->
  <link href="assets/css/bootstrap.css" rel="stylesheet" />
  <!-- FONTAWESOME STYLES-->
  <link href="assets/css/font-awesome.css" rel="stylesheet" />
  <!-- CUSTOM STYLES-->
  <link href="assets/css/custom.css" rel="stylesheet" />
  <!-- GOOGLE FONTS-->
  <link href='http://fonts.googleapis.com/css?family=Open+Sans'
rel='stylesheet' type='text/css' />

</head>
<body>
  <div class="container">
    <div class="row text-center ">
      <div class="col-md-12">
        <br /><br />
        <h2>Sistem Informasi Pengendalian Kualitas</h2>
        <br />
      </div>
    </div>
    <div class="row ">

      <div class="col-md-4 col-md-offset-4 col-sm-6 col-sm-offset-3
col-xs-10 col-xs-offset-1">
        <div class="panel panel-default">
          <div class="panel-heading">
            <strong> Silahkan isi untuk login </strong>
          </div>
          <div class="panel-body">
            <form role="form" method="POST">
              <br />
              <div class="form-group input-group">

```

```

        <span class="input-group-addon"><i class="fa
fa-tag" ></i></span>
        <input type="text" class="form-control"
placeholder="Username anda " name="uid" />
    </div>
    <div class="form-group input-group">
        <span class="input-group-addon"><i class="fa
fa-lock" ></i></span>
        <input type="password" class="form-control"
placeholder="Password anda" name="pass" />
    </div>
    <button type="submit" name="melogin" class="btn
btn-primary">login</button>
    <hr />
</form>
<?php
if(isset($_POST['melogin']))
{
    $coballogin = $pengguna-
>login_pengguna($_POST['uid'],$_POST['pass']);
    if($coballogin)
    {
        echo"<script>alert('login sukses');</script>";

echo"<script>>window.location='index.php';</script>";
    }
    else
    {
        echo"<script>alert('login gagal');</script>";

echo"<script>>window.location='login.php';</script>";

    }
}
?>
</div>

</div>
</div>

</div>
<div class="row text-center ">
    <div class="col-md-12">
        <br /><br />

```

```

    <h5>PT Suzuki Indomobil Motor</h5>
    <br />
  </div>
</div>

```

```

  <!-- SCRIPTS -AT THE BOTTOM TO REDUCE THE LOAD TIME-->
  <!-- JQUERY SCRIPTS -->
  <script src="assets/js/jquery-1.10.2.js"></script>
  <!-- BOOTSTRAP SCRIPTS -->
  <script src="assets/js/bootstrap.min.js"></script>
  <!-- METISMENU SCRIPTS -->
  <script src="assets/js/jquery.metisMenu.js"></script>
  <!-- CUSTOM SCRIPTS -->
  <script src="assets/js/custom.js"></script>

</body>
</html>

```

2. Menu Utama

```

<?php
include 'class.php';
if(empty($_SESSION['kd_user']))
{
    echo "<script>alert('login dulu');</script>";
    echo "<script>>window.location='login.php';</script>";
}

if(isset($_GET['aksi']))
{
    if($_GET['aksi']=='logout')
    {
        $pengguna->logout_pengguna();
        echo "<script>alert('anda telah logout');</script>";
        echo "<script>>window.location='login.php';</script>";
    }
}

?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0" />

```

```

<title>Sistem Informasi Pengendalian Kualitas</title>
  <!-- BOOTSTRAP STYLES-->
<link href="assets/css/bootstrap.css" rel="stylesheet" />
  <!-- FONTAWESOME STYLES-->
<link href="assets/css/font-awesome.css" rel="stylesheet" />
  <!-- MORRIS CHART STYLES-->
<link href="assets/js/morris/morris-0.4.3.min.css" rel="stylesheet" />
  <!-- CUSTOM STYLES-->
<link href="assets/css/custom.css" rel="stylesheet" />
  <!-- GOOGLE FONTS-->
<link href='http://fonts.googleapis.com/css?family=Open+Sans'
rel='stylesheet' type='text/css' />
  <!-- AUTO COMPLETE-->
  <link rel="stylesheet" href="js/jquery-ui.css">
  <script src="js/jquery-1.9.1.js"></script>
  <script src="js/jquery-ui.js"></script>
</head>
<body>
  <div id="wrapper">
    <nav class="navbar navbar-default navbar-cls-top " role="navigation"
style="margin-bottom: 0">
      <div class="navbar-header">
        <button type="button" class="navbar-toggle" data-
toggle="collapse" data-target=".sidebar-collapse">
          <span class="sr-only">Toggle navigation</span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href="index.php">SUZUKI</a>
      </div>
      <div style="color: white;
padding: 15px 50px 5px 50px;
float: right;
font-size: 16px;"> <a href="index.php?aksi=logout" class="btn btn-danger
square-btn-adjust">Logout</a> </div>
    </nav>
    <!-- /. NAV TOP -->
    <nav class="navbar-default navbar-side" role="navigation">
      <div class="sidebar-collapse">
        <ul class="nav" id="main-menu">
          <li>
            <a href="index.php?halaman=home"><i class="fa fa-
desktop fa-3x"></i> Beranda </a>
          </li>
          <?php

```

```

        if(($_SESSION['hak_akses']=="KPLSEKSI")){
        ?>
        <li>
            <a href="#"><i class="fa fa-desktop fa-3x"></i> Data
Master<span class="fa arrow"></span></a>
            <ul class="nav nav-second-level">
                <li>
                    <a
href="index.php?halaman=komponen">Komponen</a>
                </li>
                <li>
                    <a href="index.php?halaman=dataasalmasalah">Asal
Masalah Komponen</a>
                </li>
                <li>
                    <a href="index.php?halaman=tipe">Tipe Engine</a>
                </li>
                <li>
                    <a href="index.php?halaman=jeniskerusakan">Jenis
Kerusakan </a>
                </li>
                <li>
                    <a href="index.php?halaman=lini">Lini Produksi</a>
                </li>
                <li>
                    <a href="index.php?halaman=petugas">Petugas
Repair</a>
                </li>
            </ul>
        </li>
        <?php
        }
        ?>
        <?php
        if(($_SESSION['hak_akses']=="TEKNISI")){
        ?>
        <li>
            <a href="#"><i class="fa fa-desktop fa-3x"></i>Input
Kerusakan<span class="fa arrow"></span></a>
            <ul class="nav nav-second-level">
                <li>
                    <a
href="index.php?halaman=laporanbeforetek">Before Process</a>
                </li>
                <li>

```

```

Process</a>
    <a href="index.php?halaman=laporanintek">In
Process</a>
    </li>
</ul>
</li>
<?php
}
?>
<?php
if(($_SESSION['hak_akses']=="KPLSEKSI")){
?>
<li>
    <a href="#"><i class="fa fa-desktop fa-3x"></i>Input
Kerusakan<span class="fa arrow"></span></a>
    <ul class="nav nav-second-level">
        <li>
            <a href="index.php?halaman=laporanbefore">Before
Process</a>
        </li>
        <li>
            <a href="index.php?halaman=laporanin">In
Process</a>
        </li>
    </ul>
</li>
</ul>
</li>
<?php
}
?>
<?php
if(($_SESSION['hak_akses']=="MANAGER" )){
?>
<li>
    <a href="#"><i class="fa fa-desktop fa-3x"></i>Lihat
Laporan<span class="fa arrow"></span></a>
    <ul class="nav nav-second-level">
        <li>
            <a href="index.php?halaman=lihatbefore">Before
Process</a>
        </li>
        <li>
            <a href="index.php?halaman=lihatin">In Process</a>
        </li>
    </ul>
</li>
</ul>
<?php
}

```

```

?>
<?php
if($_SESSION['hak_akses']=="HINTAN"){
?>
<li>
  <a href="#"><i class="fa fa-desktop fa-3x"></i>Lihat
Laporan<span class="fa arrow"></span></a>
  <ul class="nav nav-second-level">
    <li>
      <a href="index.php?halaman=lihatbeforehin">Before
Process</a>
    </li>
    <li>
      <a href="index.php?halaman=lihatin">In Process</a>
    </li>
  </ul>
</li>
<?php
}
?>
<?php
if($_SESSION['hak_akses']=="KPLSEKSI"){
?>
<li>
  <a href="#"><i class="fa fa-desktop fa-3x"></i>Lihat
Laporan<span class="fa arrow"></span></a>
  <ul class="nav nav-second-level">
    <li>
      <a href="index.php?halaman=lihatbefore">Before
Process</a>
    </li>
    <li>
      <a href="index.php?halaman=lihatin">In Process</a>
    </li>
  </ul>
</li>
<?php
}
?>
<?php
if($_SESSION['hak_akses']=="KPLSEKSI"){
?>
<li>
  <a href="#"><i class="fa fa-desktop fa-
3x"></i>Pengaturan<span class="fa arrow"></span></a>
  <ul class="nav nav-second-level">

```

```

        <li>
            <a href="index.php?halaman=datauser">User</a>
        </li>

    </ul>
</li>
<?php
}
?>
</ul>

</div>

</nav>
<!-- /. NAV SIDE -->
<div id="page-wrapper" >
    <div id="page-inner">
        <?php
            include "connect.php";

            if(isset($_GET['halaman']))
            {
                if($_GET['halaman']=='komponen')
                {
                    include 'datakomponen.php';
                }
                elseif($_GET['halaman']=='save')
                {
                    include 'komponen/save.php';
                }
                elseif($_GET['halaman']=='hapuskomponen')
                {
                    include 'komponen/hapus.php';
                }
                elseif ($_GET['halaman']=='tambahkomponen')
                {
                    include 'komponen/tambahkomponen.php';
                }
                elseif ($_GET['halaman']=='ubahkomponen')
                {
                    include 'komponen/ubahkomponen.php';
                }
                elseif ($_GET['halaman']=='ubah')
                {
                    include 'komponen/ubah.php';
                }
            }

```

```
elseif($_GET['halaman']=='jeniskerusakan')
{
    include 'datajeniskerusakan.php';
}
elseif ($_GET['halaman']=='tambahkerusakan')
{
    include 'jeniskerusakan/tambahkerusakan.php';
}
elseif($_GET['halaman']=='savekerusakan')
{
    include 'jeniskerusakan/save.php';
}
elseif ($_GET['halaman']=='ubahjeniskerusakan')
{
    include 'jeniskerusakan/ubahkerusakan.php';
}
elseif ($_GET['halaman']=='ubahkerusakan')
{
    include 'jeniskerusakan/ubah.php';
}
elseif($_GET['halaman']=='hapusjeniskerusakan')
{
    include 'jeniskerusakan/hapus.php';
}
elseif($_GET['halaman']=='dataasalmasalah')
{
    include 'datalokasi.php';
}
elseif ($_GET['halaman']=='tambahformlokasi')
{
    include 'asalmasalah/tambahlokasi.php';
}
elseif($_GET['halaman']=='savelokasi')
{
    include 'asalmasalah/save.php';
}
elseif ($_GET['halaman']=='ubahformlokasi')
{
    include 'asalmasalah/ubahlokasi.php';
}
elseif ($_GET['halaman']=='ubahlokasi')
{
    include 'asalmasalah/ubah.php';
}
elseif($_GET['halaman']=='hapuslokasi')
{
```

```
        include 'asalmasalah/hapus.php';
    }
elseif($_GET['halaman']=='lini')
{
    include 'datalini.php';
}
elseif ($_GET['halaman']=='tambahformlini')
{
    include 'liniproduksi/tambahlini.php';
}
elseif($_GET['halaman']=='savelini')
{
    include 'liniproduksi/save.php';
}
elseif ($_GET['halaman']=='ubahformlini')
{
    include 'liniproduksi/ubahlini.php';
}
elseif ($_GET['halaman']=='ubahlini')
{
    include 'liniproduksi/ubah.php';
}
elseif($_GET['halaman']=='hapuslini')
{
    include 'liniproduksi/hapus.php';
}
elseif($_GET['halaman']=='petugas')
{
    include 'datapetugas.php';
}
elseif ($_GET['halaman']=='tambahformpetugas')
{
    include 'petugas/tambahpetugas.php';
}
elseif($_GET['halaman']=='savepetugas')
{
    include 'petugas/save.php';
}
elseif ($_GET['halaman']=='ubahformpetugas')
{
    include 'petugas/ubahpetugas.php';
}
elseif ($_GET['halaman']=='ubahpetugas')
{
    include 'petugas/ubah.php';
}
}
```

```
elseif($_GET['halaman']=='hapuspetugas')
{
    include 'petugas/hapus.php';
}
elseif($_GET['halaman']=='datauser')
{
    include 'datauser.php';
}
elseif($_GET['halaman']=='tambahformuser')
{
    include 'user/tambahuser.php';
}
elseif($_GET['halaman']=='saveuser')
{
    include 'user/save.php';
}
elseif($_GET['halaman']=='ubahformuser')
{
    include 'user/ubahuser.php';
}
elseif($_GET['halaman']=='ubahuser')
{
    include 'user/ubah.php';
}
elseif($_GET['halaman']=='hapususer')
{
    include 'user/hapus.php';
}
elseif($_GET['halaman']=='tipe')
{
    include 'datatipe.php';
}
elseif($_GET['halaman']=='tambahformtipe')
{
    include 'tipeengine/tambahtipe.php';
}
elseif($_GET['halaman']=='savetipe')
{
    include 'tipeengine/save.php';
}
elseif($_GET['halaman']=='ubahformtipe')
{
    include 'tipeengine/ubahtipe.php';
}
elseif($_GET['halaman']=='ubahtipe')
{
```

```
        include 'tipeengine/ubah.php';
    }
elseif($_GET['halaman']=='hapus tipe')
{
    include 'tipeengine/hapus.php';
}
elseif($_GET['halaman']=='laporanin')
{
    include 'datainprocess.php';
}
elseif ($_GET['halaman']=='tambahformlaporanin')
{
    include 'transaksi/in/tambahin.php';
}
elseif($_GET['halaman']=='saveinprocess')
{
    include 'transaksi/in/save.php';
}
elseif ($_GET['halaman']=='ubahforminprocess')
{
    include 'transaksi/in/ubahin.php';
}
elseif ($_GET['halaman']=='ubahinprocess')
{
    include 'transaksi/in/ubah.php';
}
elseif($_GET['halaman']=='hapusinprocess')
{
    include 'transaksi/in/hapus.php';
}
elseif($_GET['halaman']=='laporanintek')
{
    include 'dataintek.php';
}
elseif ($_GET['halaman']=='tambahformlaporanintek')
{
    include 'transaksi/in/tambahintek.php';
}
elseif($_GET['halaman']=='saveinprocesstek')
{
    include 'transaksi/in/savetek.php';
}
elseif($_GET['halaman']=='laporanbeforetek')
{
    include 'databeforetek.php';
}
}
```

```
elseif ($_GET['halaman']=='tambahformlaporanbeforetek')
{
    include 'transaksi/before/tambahtek.php';
}
elseif($_GET['halaman']=='savebeforeprocesstek')
{
    include 'transaksi/before/savetek.php';
}
elseif($_GET['halaman']=='laporanbefore')
{
    include 'databeforeprocess.php';
}
elseif ($_GET['halaman']=='tambahformlaporanbefore')
{
    include 'transaksi/before/tambahbefore.php';
}
elseif($_GET['halaman']=='savebeforeprocess')
{
    include 'transaksi/before/save.php';
}
elseif ($_GET['halaman']=='ubahformbeforeprocess')
{
    include 'transaksi/before/ubahbefore.php';
}
elseif ($_GET['halaman']=='ubahbeforeprocess')
{
    include 'transaksi/before/ubah.php';
}
elseif($_GET['halaman']=='hapusbeforeprocess')
{
    include 'transaksi/before/hapus.php';
}
elseif($_GET['halaman']=='lihatbefore')
{
    include 'laporan/before/lapbefore.php';
}
elseif($_GET['halaman']=='lihatbeforehin')
{
    include 'laporan/before/lapbefhin.php';
}
elseif($_GET['halaman']=='lihatin')
{
    include 'laporan/in/lapin.php';
}
elseif ($_GET['halaman']=='home')
{

```

```
        include 'home.php';
    }
}
else
{
    include 'home.php';
}
?>
</div>
<!-- /. PAGE INNER -->
</div>
<!-- /. PAGE WRAPPER -->
</div>
<!-- /. WRAPPER -->
<!-- SCRIPTS -AT THE BOTTOM TO REDUCE THE LOAD TIME-->
<!-- JQUERY SCRIPTS -->
<script src="assets/js/jquery-1.10.2.js"></script>
<!-- BOOTSTRAP SCRIPTS -->
<script src="assets/js/bootstrap.min.js"></script>
<!-- METISMENU SCRIPTS -->
<script src="assets/js/jquery.metisMenu.js"></script>
<!-- MORRIS CHART SCRIPTS -->
<script src="assets/js/morris/raphael-2.1.0.min.js"></script>
<script src="assets/js/morris/morris.js"></script>
<!-- CUSTOM SCRIPTS -->
<script src="assets/js/custom.js"></script>

</body>
</html>
```