

**RANCANG BANGUN SISTEM INFORMASI PENGENDALIAN
KUALITAS *ACCUMULATOR* DIVISI *PAINTING* BERBASIS
WEB MENGGUNAKAN PHP 5.4.27 DAN MYSQL 5.5.36 PADA
PT KURNIA MANUNGAL SEJAHTERA**

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Penyelesaian
Program Diploma Empat (D-IV) Program Studi Sistem Infomasi
pada Politeknik STMI Jakarta

OLEH

EGANA PERTIWI

1312021



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

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir : Rancang Bangun Sistem Informasi Pengendalian
Kualitas *Accumulator* Divisi *Painting* Berbasis Web
Menggunakan PHP 5.4.27 dan MySQL 5.5.36 Pada
PT Kurnia Manunggal Sejahtera

Disusun Oleh :

Nama : Egana Pertiwi
NIM : 1312021
Program Studi : Sistem Informasi

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem
Informasi Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia
pada Hari Kamis Tanggal 10 November 2016.

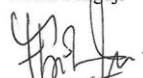
Jakarta, 15 November 2016

Dosen Pembimbing



Noveriza Yuliasari, SSi, MT
NIP: 197811212009012003

Ketua Penguji



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

Dosen Penguji



Ulil Hamida ST, MT
NIP. 198103272005022001

Dosen Penguji



Drs. Jacob Saragih, MM
NIP: 195404281986031002



POLITEKNIK STMI JAKARTA

d.h. SEKOLAH TINGGI MANAJEMEN INDUSTRI

Jl. Letjen Suprapto No. 26 Cempaka Putih, Jakarta 10510
Telp: (021) 42886064 Fax: (021) 42888206
www.stmi.ac.id



LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Egana Pertiwi
NIM : 1312021
Judul TA : Rancang Bangun Sistem Informasi Pengendalian Kualitas Accumulator
Divisi Painting Berbasis Web Menggunakan PHP 5.4.27 dan MySQL
5.5.36 Pada PT Kurnia Manunggal Sejahtera
Pembimbing : Noveriza Yuliasari, SSI, MT

Tanggal	Keterangan	Paraf
28/06/2016	Proposal	<i>J</i>
29/06/2016	Bab I	<i>J J</i>
18/07/2016	Bab I	<i>J J</i>
22/07/2016	Revisi Bab I dan Bab II	<i>J J</i>
29/07/2016	Revisi Bab I dan Bab II	<i>J J</i>
03/08/2016	Revisi Bab I, Bab II dan Bab III	<i>J J</i>
10/08/2016	Revisi Bab I, Bab II dan Bab III	<i>J J</i>
12/08/2016	Revisi Bab I dan Bab IV	<i>J J</i>
23/08/2016	Revisi Bab IV	<i>J J</i>
14/09/2016	Revisi Bab IV dan Bab V	<i>J J</i>
26/09/2016	Revisi Bab IV dan Bab V	<i>J J</i>
29/09/2016	Revisi Bab V	<i>J J</i>
05/10/2016	Revisi Bab V	<i>J J</i>
10/10/2016	Revisi Bab V	<i>J J</i>
13/10/2016	Bab II, Daftar Pustaka dan Demo Program	<i>J J</i>
18/10/2016	Demo Program dan Bawa Keseluruhan	<i>J J</i>

Mengetahui,
Ketua Prodi
Sistem Informasi Industri Otomotif

(Drs. Jacob Saragih, MM.)
NIP: 195404281986031002

Dosen Pembimbing

(Noveriza Yuliasari, SSI, MT)
NIP: 197811212009012003



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



LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Egana Pertiwi
 NIM : 1312021
 Judul TA : Rancang Bangun Sistem Informasi Pengendalian Kualitas *Accumulator*
 Divisi *Painting* Berbasis Web Menggunakan PHP 5.4.27 dan MySQL
 5.5.36 Pada PT Kurnia Manunggal Sejahtera
 Pembimbing : Ahmad Juniar, S.Kom, MT

Tanggal	Keterangan	Paraf
19/08/2016	Bab I	A
22/08/2016	Bab I Latar Belakang dan Tujuan	A
20/09/2016	Bab II Landasan Teori, Perbaiki Halaman 9, 24 dan 28	A
21/09/2016	Bab III dan Revisi Bab II	A
22/09/2016	Bab III Metodologi Penelitian	A
13/10/2016	Bab IV Pengumpulan dan Pengolahan Data, Perbaiki Halaman 56 dan 69 (Note Saran belum ada)	A
14/10/2016	Bab V Usulan Use Case, Activity Diagram, Sequence Diagram dan Class Diagram	A
18/10/2016	Revisi Bab V	A
19/10/2016	Bab V Kamus Data, HIPO , Flowchart Program dan Rancangan Interface	A
20/10/2016	Kesimpulan, Daftar Pustaka dan Lampiran	A

Mengetahui,
Ketua Prodi
Sistem Informasi Industri Otomotif



(Drs. Jacob Saragih, MM.)
NIP: 195404281986031002

Asisten Dosen Pembimbing



(Ahmad Juniar, S.Kom, MT)
NIP: 197906052006041002



POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN R.I.

TANDA PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : **Rancang Bangun Sistem Informasi Pengendalian
Kualitas *Accumulator* Divisi *Painting* Berbasis
Web Menggunakan PHP 5.4.27 dan MySQL
5.5.36 Pada PT Kurnia Manunggal Sejahtera**

Disusun Oleh :

Nama : Egana Pertiwi
NIM : 1312021
Program Studi : Sistem Informasi
Tanggal Seminar : Jumat, 26 Oktober 2016
Tanggal Sidang : Rabu, 10 November 2016
Tanggal Lulus : Rabu, 10 November 2016

Jakarta, 15 November 2016

Menyetujui,
Dosen Pembimbing



Noveriza Yuliasari, SSi, MT
NIP. 197811212009012003

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN R.I.

TANDA PERSETUJUAN ASISTEN DOSEN PEMBIMBING

Judul Tugas Akhir : **Rancang Bangun Sistem Informasi Pengendalian
Kualitas *Accumulator* Divisi *Painting* Berbasis
Web Menggunakan PHP 5.4.27 dan MySQL
5.5.36 Pada PT Kurnia Manunggal Sejahtera**

Disusun Oleh :

Nama : Egana Pertiwi
NIM : 1312021
Program Studi : Sistem Informasi
Tanggal Seminar : Jumat, 26 Oktober 2016
Tanggal Sidang : Rabu, 10 November 2016
Tanggal Lulus : Rabu, 10 November 2016

Jakarta, 15 November 2016

Menyetujui,
Asisten Dosen Pembimbing



Ahmad Juniar, S.Kom, MT
NIP: 197906052006041002

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Egana Pertiwi

Nim : 1312021

Program Studi : Sistem Informasi

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

“Rancang Bangun Sistem Informasi Pengendalian Kualitas *Accumulator* Divisi *Painting* Berbasis Web Menggunakan PHP 5.4.27 DAN MySQL 5.5.36 Pada PT Kurnia Manunggal Sejahtera”.

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

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

Jakarta, 15 November 2016

METERAL
TEMPEL
No.
A3CB5AEF300555829
6000
ENAM RIBURUPAH
Egana Pertiwi

ABSTRAK

PT Kurnia Manunggal Sejahtera adalah perusahaan yang memproduksi komponen otomotif dan elektronik industri, yang mengkhususkan diri dalam 3 divisi yaitu divisi *Stamping*, divisi *Painting* dan divisi Konstruksi Baja. PT Kurnia Manunggal Sejahtera membutuhkan pengembangan sebuah sistem informasi pengendalian kualitas *accumulator* mengenai *quality monthly report*. Dalam menjalankan kegiatannya, pengolahan data yang ada pada bagian *Quality Control* divisi *Painting* masih dilakukan secara manual. Proses pengolahan data dikerjakan dengan cara melakukan pengecekan data produk dan menulisnya di laporan *loading-unloading*. Proses penginputan data tersebut dilakukan secara berulang-ulang sehingga memakan waktu kurang lebih dari satu sampai dua jam. Selain itu dalam proses pencarian data *accumulator* masih sulit dilakukan karena harus mencari data ditumpukkan kertas dan harus melakukan pencarian data dengan cara mencari ke *spreadsheet-spreadsheet* sebelumnya pada *Microsoft Excel*. Rancang bangun sistem informasi pengendalian kualitas *accumulator* sangat diperlukan dalam perusahaan untuk melakukan proses pengolahan data *loading-unloading* dan *quality monthly report*. Sistem informasi yang diusulkan akan mempermudah dan mengintergrasi proses pengolahan data sehingga membantu bagian di dalam sistem untuk saling bertukar informasi dan mengambil keputusan dengan cepat. Untuk perancangan sistem informasi yang diusulkan, metode pengembangan sistem yang digunakan ialah *evolutionary prototype* dengan melalui tahap-tahap yang digunakan mulai dari pemodelan sistem menggunakan *Unified Modeling Language (UML)*, perancangan data menggunakan *class diagram*, perancangan HIPO, perancangan *flowchart* program yang diusulkan, perancangan *interface* program, dan pembuatan *prototype* menggunakan bahasa pemrograman PHP 5.4.27 serta *MySQL 5.5.36* sebagai basis datanya. Sistem informasi pengendalian kualitas *accumulator* diharapkan dapat memudahkan perusahaan dalam proses pengolahan data terutama dalam penginputan laporan *loading-unloading* dan pembuatan *quality monthly report*. Untuk penerapan sistem baru, disarankan untuk melakukan sosialisasi kepada bagian terkait dan pemeliharaan aplikasi agar dapat berjalan dengan baik.

Kata Kunci : *Sistem Informasi, Pengendalian Kualitas, Accumulator, Unified Modeling Language (UML), PHP 5.4.27, MySQL 5.5.36*

KATA PENGANTAR

Alhamdulillah segala puji syukur penulis panjatkan kepada Allah SWT atas limpahan nikmat, rahmat serta karunia yang senantiasa diberikan sehingga penulis mampu menyelesaikan Tugas Akhir yang berjudul **“Rancang Bangun Sistem Informasi Pengendalian Kualitas *Accumulator* Divisi *Painting* Berbasis Web Menggunakan PHP 5.4.27 dan MySQL 5.5.36 Pada PT Kurnia Manunggal Sejahtera”**.

Penyusunan Tugas Akhir ini dilakukan guna memenuhi salah satu persyaratan akademis untuk menyelesaikan Program Diploma Empat (D-IV) Program Studi Sistem Informasi pada Politeknik STMI Jakarta Kementerian Perindustrian R.I.

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

1. Allah SWT, yang telah memberikan rahmat dan karunia-Nya.
2. Bapak, Mama, Umar dan Haura yang selalu mendoa'akan dan memberi semangat dan kasih sayang yang penuh agar laporan dan kuliah dapat diselesaikan dengan baik.
3. Bapak Dr. Mustofa, ST, MT. Selaku Direktur Politeknik STMI Jakarta.
4. Bapak Drs. Jacob Saragih, MM. Selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
5. Ibu Noveriza, SSi, MT. Selaku Dosen Pembimbing yang telah bersedia membantu memberikan bimbingan serta pengarahan kepada penulis untuk menyelesaikan Tugas Akhir ini.
6. Bapak Ahmad Juniar, S.Kom, MT. Selaku Asisten Dosen Pembimbing yang telah memberikan arahan, bimbingan serta penjelasan teknis dalam penyusunan Tugas Akhir.

7. Bapak Fay Muhamad NR dan Bapak Hendry selaku pembimbing di PT Kurnia Manunggal Sejahtera yang telah membimbing dan memberikan informasi yang berguna kepada penulis dalam penyusunan Tugas Akhir.
8. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
9. Sumboro, yang selalu memberikan motivasi, do'a dan sangat membantu selama proses penyelesaian Tugas Akhir.
10. Para sahabat Andini Maharani, Annisa Setyawati, Fitri Chaerani Nopriyadi dan Rizki Dwi Lestari atas dukungan dan motivasinya selama ini.
11. Sahabat SMA Hilda Ayuanita, Jean Elma Fitri, Lutfiyana Latifa dan Cindy Aprilia atas dukungan dan motivasinya selama ini.
12. Rekan-rekan mahasiswa Politeknik STMI Jakarta Program Studi Sistem Informasi 2012 atas kebersamaan dan motivasinya selama ini.
13. Serta semua pihak yang baik langsung maupun tidak langsung memberikan kritik, saran dan bantuan dalam pembuatan laporan ini yang tidak bisa kami sebutkan satu per satu.

Penulis menyadari bahwa penulisan dalam penyusunan 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, 16 Oktober 2016

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING	
LEMBAR PENGESAHAN	
LEMBAR BIMBINGAN DENGAN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN DENGAN ASISTEN DOSEN PEMBIMBING	
LEMBAR PERNYATAAN KEASLIAN	
ABSTRAK	viii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xv
DAFTAR TABEL	xviii
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.....	Sistemat
ika Penulisan	4
BAB II LANDASAN TEORI	7

2.1.....	Pengerti	
an Sistem		7
2.1.1	Elemen	
Sistem		7
2.2.....	Pengerti	
an Informasi		9
2.2.1	Siklus	
Informasi		9
2.2.2	Kualitas	
Informasi		10
2.2.3	Karakter	
istik Informasi		11
2.3.....	Pengerti	
an Sistem Informasi		12
2.3.1	Kompon	
en Sistem Informasi		12
2.4.....	Pengerti	
an Pengendalian		14
2.5.....	Pengerti	
an Kualitas		15
2.6.....	Pengerti	
an Pengendalian Kualitas		15
2.6.1.....	Tujuan	
Pengendalian Kualitas		16
2.7.....	Alat	
Pengendalian Kualitas		16
2.7.1.....	Diagram	
Pareto		17
2.7.2.....	Histogra	
m		17

2.7.3.....	Lembar	
Pengecekan (<i>Check Sheet</i>)		18
2.7.4.....	Diagram	
Sebab Akibat		19
2.7.5.....	Diagram	
Penyebaran (<i>Scatter Diagram</i>)		19
2.7.6.....	Diagram	
Alur		20
2.7.7.....	Peta	
Pengendali (<i>Control Chart</i>)		21
2.8.....	<i>System</i>	
<i>Development Life Cycle (SDLC)</i>		21
2.9.....	Metodol	
ogi Pengembangan Sistem		23
2.9.1.....	<i>Prototyp</i>	
<i>e Evolutioner</i>		23
2.9.2.....	<i>Prototyp</i>	
<i>e Requirement</i>		24
2.10.....	<i>Object</i>	
<i>Oriented Analysis and Design (OOAD)</i>		25
2.11.....	<i>Flowcha</i>	
<i>rt</i>		26
2.12.....	<i>Unified</i>	
<i>Modelling Language (UML)</i>		29
2.12.1.....	<i>Use</i>	
<i>Case Diagram</i>		30
2.12.2.....	<i>Activity</i>	
<i>Diagram</i>		31
2.12.3.....	<i>Sequenc</i>	
<i>e Diagram</i>		32

2.12.4.....	<i>Class</i>	
<i>Diagram</i>		34
2.12.5.....	<i>Deploy</i>	
<i>ment Diagram</i>		35
2.12.6.....	<i>Compon</i>	
<i>ent Diagram</i>		36
2.13.....	Kamus	
Data		37
2.14.....	HIPO	
(<i>Hierarchy plus Input-Process-Output</i>)		39
2.15.....	<i>Hyperte</i>	
<i>xt Pre-Processor (PHP)</i>		41
2.16.....	MySQL	
.....		42
2.16.1.....	Keungg	
ulan MySQL		43
2.16.2.....	Tipe	
Data My SQL		44
BAB III	METODOLOGI PENELITIAN	46
3.1	Metodologi Penelitian	46
3.2	Jenis dan Metode Pengumpulan Data	46
3.2.1	Sumber Data Primer	46
3.2.2	Sumber Data Sekunder	47
3.3	Metode Pengembangan Sistem	47
3.4	Kerangka Penelitian	48
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA	52
4.1	Sekilas Perusahaan	52
4.2	Profil Perusahaan	53
4.3	Visi dan Misi Perusahaan	53
4.4	Tujuan Perusahaan	54
4.5	Struktur Organisasi Perusahaan	54

4.6	Produk	61
4.7	<i>Accumulator</i>	61
4.8	Komponen <i>Accumulator</i> dan Fungsinya	62
4.9	Proses Produksi dan <i>Painting Accumulator</i>	63
4.10	Proses Pengendalian Kualitas <i>Accumulator</i>	64
4.11	Jenis-Jenis Kerusakan <i>Accumulator</i>	68
4.12	Dokumen Pengendalian Kualitas	69
4.13	Prosedur Pengendalian Kualitas	71
4.14	Pemodelan Sistem Pengendalian Kualitas dengan <i>Use Case Diagram</i>	73
BAB V	ANALISIS DAN PEMBAHASAN	77
5.1	Analisis Kebutuhan Sistem	77
5.2	Prosedur Sistem Informasi Pengendalian Kualitas <i>Accumulator Usulan</i>	78
5.3	Analisis dan Perancangan Sistem Usulan	79
5.4	Pemodelan Sistem Pengendalian Kualitas Usulan	80
5.4.1	<i>Use Case Diagram</i>	80
5.4.2 <i>Activity Diagram</i>	89
5.4.3 <i>Sequence Diagram</i>	100
5.5Pemodelan Data Pengendalian Kualitas Usulan	108
5.5.1 <i>Class Diagram</i>	108
5.5.2Kamus Data	109
5.6 <i>Deployment Diagram</i>	112
5.7Perancangan Aplikasi Pengendalian Kualitas	113

5.7.1.....	<i>Flowchart Program</i>	113
5.7.2.....	<i>Hierarchy plus Input-Process-Output (HIPO)</i>	115
5.7.3.....	Perancangan <i>Interface Program</i>	115
5.8.....	Implementasi Sistem	122
BAB VI KESIMPULAN DAN SARAN		124
6.1	Kesimpulan	116
6.2	Saran	116
DAFTAR PUSTAKA		117
LAMPIRAN		

DAFTAR GAMBAR

	Halaman
Gambar II.1	Elemen Sistem 9
Gambar II.2	Siklus Informasi 10
Gambar II.3	Interaksi Blok Sistem Informasi 14
Gambar II.4	Diagram Pareto 17
Gambar II.5	Histogram 18
Gambar II.6	<i>Check Sheet</i> 18
Gambar II.7	<i>Fish Bone Diagram</i> 19
Gambar II.8	<i>Scatter Diagram</i> 20
Gambar II.9	<i>Flowchart</i> 20
Gambar II.10	Peta Kendali 21
Gambar II.11	Pengembangan <i>Prototype Evolusioner</i> 24
Gambar II.12	Pengembangan <i>Prototype Requirement</i> 25
Gambar II.13	Diagram UML 29
Gambar II.14	<i>Visual Table Of Contents</i> 40
Gambar II.15	<i>Overview Diagram</i> 40
Gambar II.16	Siklus PHP dalam <i>Web Server</i> 42
Gambar III.1	Kerangka Penelitian 51
Gambar IV.1	Struktur Organisasi 55
Gambar IV.2	<i>Accumulator White</i> 62
Gambar IV.3	<i>Flow Proses Painting Accumulator</i> 63
Gambar IV.4	<i>Flowchart</i> Proses Pengendalian Kualitas <i>Accumulator</i> 64
Gambar IV.5	Laporan <i>Loading-Unloading</i> 69
Gambar IV.6	<i>Quality Monthly Report</i> 70
Gambar IV.7	<i>Note Saran</i> 71
Gambar IV.8	Aliran Dokumen Sistem yang Berjalan 73
Gambar IV.9	<i>Use Case Diagram</i> Sistem Informasi Pengendalian Kualitas 76

Gambar V.1	<i>Flowmap</i> Sistem Informasi Pengendalian Kualitas <i>Accumulator</i> divisi <i>Painting</i> Usulan	79
Gambar V.2	<i>Use Case Diagram</i> Sistem Informasi Pengendalian Kualitas <i>Accumulator</i> Usulan	82
Gambar V.3	<i>Activity Diagram</i> <i>Login</i>	90
Gambar V.4	<i>Activity Diagram</i> Mengelola Data Master	90
Gambar V.5	<i>Activity Diagram</i> Memasukkan Data <i>Loading-Unloading</i> ...	92
Gambar V.6	<i>Activity Diagram</i> Menampilkan Data <i>Loading-Unloading</i> ...	93
Gambar V.7	<i>Activity Diagram</i> Memvalidasi Data <i>Loading-Unloading</i>	94
Gambar V.8	<i>Activity Diagram</i> Mencetak <i>Quality Monthly Report</i>	95
Gambar V.9	<i>Activity Diagram</i> View <i>Quality Monthly Report</i>	96
Gambar V.10	<i>Activity Diagram</i> Menampilkan <i>Quality Monthly Report</i>	97
Gambar V.11	<i>Activity Diagram</i> Menampilkan Grafik	98
Gambar V.12	<i>Activity Diagram</i> Melakukan Evaluasi Target <i>Reject Max 3%</i>	99
Gambar V.13	<i>Activity Diagram</i> Membuat <i>Note</i> Saran	100
Gambar V.14	<i>Sequence Diagram</i> Proses <i>Login</i>	101
Gambar V.15	<i>Sequence Diagram</i> Mengolah Data Master	102
Gambar V.16	<i>Sequence Diagram</i> Memasukkan Data <i>Loading-Unloading</i>	103
Gambar V.17	<i>Sequence Diagram</i> Menampilkan Data <i>Loading-Unloading</i>	104
Gambar V.18	<i>Sequence Diagram</i> Memvalidasi Data <i>Loading-Unloading</i>	104
Gambar V.19	<i>Sequence Diagram</i> Mencetak <i>Quality Monthly Report</i>	105
Gambar V.20	<i>Sequence Diagram</i> Menampilkan <i>Quality Monthly Report</i> ..	106
Gambar V.21	<i>Sequence Diagram</i> View <i>Quality Monthly Report</i>	106
Gambar V.22	<i>Sequence Diagram</i> Menampilkan Grafik	107
Gambar V.23	<i>Sequence Diagram</i> Melakukan Evaluasi Target <i>Reject Max 3%</i>	107
Gambar V.24	<i>Sequence Diagram</i> Membuat <i>Note</i> Saran	108

Gambar V.25	<i>Class Diagram Usulan</i>	109
Gambar V.26	<i>Deployment Diagram</i>	113
Gambar V.27	<i>Program Logic Flowchart Aplikasi Sistem Informasi</i> <i>Pengendalian Kualitas Accumulator</i>	114
Gambar V.28	<i>HIPO Sistem Informasi Pengendalian Kualitas</i> <i>Accumulator Usulan</i>	115
Gambar V.29	<i>Interface Form Login</i>	116
Gambar V.30	<i>Interface Form Menu Utama</i>	117
Gambar V.31	<i>Interface Form Data User</i>	117
Gambar V.32	<i>Interface Form Tambah Data User</i>	118
Gambar V.33	<i>Interface Form Data Produk</i>	118
Gambar V.34	<i>Interface Form Tambah Data Produk</i>	119
Gambar V.35	<i>Interface Form Loading-Unloading</i>	119
Gambar V.36	<i>Interface Form Tambah Loading-Unloading</i>	120
Gambar V.37	<i>Interface Form Detail Loading-Unloading</i>	120
Gambar V.38	<i>Interface Quality Monthly Report</i>	121
Gambar V.39	<i>Interface Form Note Saran</i>	121
Gambar V.40	<i>Interface Form Grafik</i>	122
Gambar V.41	<i>Interface Form Grafik</i>	122

DAFTAR TABEL

	Halaman
Tabel II.1	Simbol-Simbol <i>Flowchart</i> 26
Tabel II.2	Simbol <i>Flowchart</i> 27
Tabel II.3	Simbol-Simbol <i>Use Case Diagram</i> 30
Tabel II.4	Simbol-Simbol <i>Activity Diagram</i> 31
Tabel II.5	Simbol-Simbol <i>Sequence Diagram</i> 32
Tabel II.6	Simbol-Simbol <i>Class Diagram</i> 34
Tabel II.7	Tipe <i>Multiplicity</i> 35
Tabel II.8	Simbol-Simbol <i>Deployment Diagram</i> 36
Tabel II.9	Simbol-Simbol <i>Component Diagram</i> 37
Tabel II.10	Contoh Kamus Data Untuk Tabel Pemasok 38
Tabel II.11	Jenis Data Pada MySQL 44
Tabel IV.1	Komponen <i>Accumulator</i> dan Fungsinya 62
Tabel IV.2	Jenis Kerusakan <i>Accumulator</i> 68
Tabel IV.3	Definisi Aktor Sistem Informasi Pengendalian Kualitas 73
Tabel IV.4	Definisi <i>Use Case</i> Sistem Informasi Pengendalian Kualitas ... 74
Tabel V.1	Kebutuhan Sistem Informasi Pengendalian Kualitas <i>Accumulator</i> 77
Tabel V.2	Definisi Aktor <i>Use Case Diagram</i> Sistem Usulan 80
Tabel V.3	Definisi <i>Use Case Diagram</i> Sistem Usulan 81
Tabel V.4	Skenario <i>Use Case Login</i> 83
Tabel V.5	Skenario <i>Use Case</i> Mengelola Data Master..... 83
Tabel V.6	Skenario <i>Use Case</i> Memasukkan Data <i>Loading-Unloading</i> ... 84
Tabel V.7	Skenario <i>Use Case</i> Menampilkan Data <i>Loading-Unloading</i> 85
Tabel V.8	Skenario <i>Use Case</i> Memvalidasi Data <i>Loading-Unloading</i> 85
Tabel V.9	Skenario <i>Use Case</i> Mencetak <i>Quality Monthly Report</i> 86
Tabel V.10	Skenario <i>Use Case View Quality Monthly Report</i> 86
Tabel V.11	Skenario <i>Use Case</i> Menampilkan <i>Quality Monthly Report</i> 87

Tabel V.12	Skenario <i>Use Case</i> Menampilkan Grafik	88
Tabel V.13	Skenario <i>Use Case</i> Melakukan Evaluasi Target <i>Reject Max 3%</i>	88
Tabel V.14	Skenario <i>Use Case</i> Membuat <i>Note</i> Saran	89
Tabel V.15	Tabel <i>User</i>	110
Tabel V.16	Tabel Produk	110
Tabel V.17	Tabel Detail <i>Loading-Unloading</i>	111
Tabel V.18	Tabel <i>Loading-Unloading</i>	111
Tabel V.19	Tabel <i>Note</i>	112

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan industri di era globalisasi ini sangat pesat seiring dengan kemajuan teknologi saat ini. Kemajuan teknologi ini mendorong juga perkembangan dari sistem informasi. Sistem informasi merupakan hal yang penting dalam sebuah perusahaan. Sistem informasi bisa diterapkan dalam segala aspek yang ada di dalam perusahaan, baik dibagian perencanaan, keuangan, produksi dan salah satunya dalam pengendalian kualitas. Dengan adanya sistem informasi, perusahaan dapat menjamin kualitas informasi dapat disajikan dengan baik, cepat dan mudah dipahami. Penerapan sistem informasi merupakan salah satu cara perusahaan dapat bersaing dengan perusahaan lainnya.

Untuk mampu bersaing dalam dunia bisnis, perusahaan dituntut untuk bisa memenuhi segala aspek yang dibutuhkan oleh konsumen. Demi memenuhi tuntutan konsumen tersebut maka kualitas produk yang dihasilkan harus benar-benar sesuai standar. Maka dari itu sangat dibutuhkan suatu sistem pengendalian kualitas untuk mempertahankan mutu dan kualitas dari barang yang dihasilkan, agar sesuai dengan spesifikasi produk yang telah ditetapkan oleh perusahaan.

Pengendalian kualitas produk yang dilakukan secara terus menerus dapat meningkatkan mutu suatu produk, sehingga akan menciptakan kepuasan konsumen. Dengan demikian fungsi pengendalian kualitas memegang peranan yang sangat penting bagi perusahaan dalam memperbaiki dan meningkatkan kualitas produk agar sesuai dengan yang konsumen butuhkan. Karena kualitas suatu produk merupakan salah satu faktor yang menentukan perkembangan suatu perusahaan.

PT Kurnia Manunggal Sejahtera adalah perusahaan yang memproduksi komponen otomotif dan elektronik industri dan terbagi dari 3 divisi yaitu divisi

Stamping, divisi *Painting* dan divisi Konstruksi Baja. Produk yang dihasilkan oleh PT Kurnia Manunggal Sejahtera dalam komponen otomotif adalah blok mesin vario dan beat, *stay mirror* besar dan *stay mirror* kecil sedangkan elektronik adalah rangka *air conditioner*, kipas angin dan *accumulator*. PT Kurnia Manunggal Sejahtera sendiri merupakan pemasok untuk beberapa perusahaan industri, antara lain PT Sanyo, PT Astra Honda Motors dan beberapa perusahaan besar lainnya.

Dalam menjalankan kegiatannya, pengolahan data yang ada pada bagian *Quality Control* divisi *Painting* masih dilakukan secara manual. Proses pengolahan data dikerjakan oleh Operator dimana Operator melakukan pengecekan data produk dengan cara menuliskannya di laporan *loading-unloading*. Kemudian laporan *loading-unloading* diserahkan ke bagian *Quality Control Office* untuk di pindahkan ke dalam *Microsoft Excel* dan dibuat *quality monthly report*. Proses penginputan data tersebut dilakukan secara berulang-ulang sehingga memakan waktu kurang lebih dari satu sampai dua jam. Selain itu dalam proses pencarian data masih sulit dilakukan karena bagian *Quality Control Office* harus mencari data ditumpukkan kertas dan melakukan pencarian data dengan cara mencari ke *spreadsheet-spreadsheet* sebelumnya pada *Microsoft Excel*.

Untuk mengatasi masalah tersebut, diperlukan aplikasi yang dapat membantu perusahaan mengelola setiap data yang ada menjadi suatu laporan yang berguna bagi perusahaan. Adapun judul Tugas Akhir ini adalah "RANCANG BANGUN SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR DIVISI PAINTING BERBASIS WEB MENGGUNAKAN PHP 5.4.27 DAN MYSQL 5.5.36 PADA PT KURNIA MANUNGGAL SEJAHTERA".

1.2 Pokok Permasalahan

Permasalahan yang terjadi pada PT Kurnia Manunggal Sejahtera pada bagian *Quality Control* divisi *Painting* adalah sebagai berikut:

1. Proses penginputan laporan *loading-unloading part accumulator* yang dilakukan secara berulang-ulang sehingga membutuhkan waktu yang cukup lama.
2. Sulitnya pencarian data *accumulator* bila diperlukan karena media penyimpanan laporan yang masih menggunakan *Microsoft Excel*.
3. Proses pembuatan *quality monthly report accumulator* yang belum dilakukan secara otomatis menyebabkan waktu yang dibutuhkan untuk membuat laporan cukup lama karena media penyimpanan data kerusakan yang masih dilakukan pada *Microsoft Excel*.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah membangun sistem informasi pengendalian kualitas *accumulator* divisi *Painting* yang mampu:

1. Menginput dan mengolah laporan *loading-unloading part accumulator* secara otomatis.
2. Memiliki media penyimpanan berbasis *database* yang dapat memudahkan proses pencarian data *accumulator*.
3. Membuat laporan *quality monthly report* menjadi lebih cepat, mudah dan akurat sehingga dapat digunakan oleh perusahaan dalam mengambil keputusan.

1.4 Batasan Masalah

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

1. Penelitian dilakukan pada bagian *Quality Control* divisi *Painting accumulator* di PT Kurnia Manunggal Sejahtera selama satu bulan mulai dari tanggal 03 Maret sampai dengan 03 April 2016.
2. Pada sistem informasi yang akan dibangun tidak mengolah proses produksi barang sebelum memasuki bagian pengendalian kualitas divisi *Painting*.

1.5 Manfaat Tugas Akhir

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

1. Bagi perusahaan
 - a. Sebagai bahan untuk pengambilan keputusan yang membantu kinerja perusahaan dalam mengawasi kualitas produk.
 - b. Mempercepat dalam memberikan laporan terhadap kualitas *accumulator* dan mendapatkan informasi yang lengkap terhadap setiap laporan pengendalian kualitas *accumulator*.
2. Bagi mahasiswa
 - a. Memberikan kemampuan dalam mengaplikasikan teori secara jelas terhadap masalah yang diamati.
 - b. Memberikan wawasan dan pengalaman kepada mahasiswa dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi permasalahan.

3. Bagi pihak lain

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

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada didalamnya. Adapun sistematika penulisan Tugas Akhir ini diuraikan dalam enam bab, yaitu:

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Bab ini membahas tentang berbagai teori yang diperoleh dari buku-buku literature ataupun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan adalah pengertian sistem, sistem informasi, pengendalian kualitas, metode pengembangan sistem informasi, *Unified Modelling Language* (UML), pemrograman PHP sebagai alat bantu untuk membuat rancangan konseptual, dan teori-teori lain yang berhubungan dengan pengembangan sistem informasi.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metode ilmiah mencari, mengembangkan, dan menguji kebenaran tentang suatu pengetahuan. Selain itu dijelaskan pula kerangka pemecahan masalah yang menguraikan tahap-tahap untuk mengatasi permasalahan yang telah diidentifikasi.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini akan menguraikan tentang data yang telah diperoleh, analisis sistem informasi pengendalian kualitas *accumulator* pada divisi *Painting* di PT Kurnia Manunggal Sejahtera dan memodelkan sistem tersebut menggunakan *use case*.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis kebutuhan sistem, proses bisnis usulan, memodelkan sistem dengan *Unified Modelling Language* (UML), perancangan sistem menggunakan *use case*, *activity diagram* dan *sequence diagram*, memodelkan data menggunakan *class diagram* dan kamus data, perancangan aplikasi yang terdiri dari tahapan perancangan *flowchart* program, *Hierarchy plus Input-Process-*

Output (HIPO) dan tampilan antar muka, sampai kebutuhan *software* dan *hardware* yang diperlukan.

BAB VI PENUTUP

Bab ini berisi kesimpulan secara keseluruhan yang diperoleh dari hasil penelitian serta saran-saran dalam penerapan Sistem Informasi Pengendalian Kualitas *Accumulator* divisi *Painting* untuk perusahaan dan pengembangan selanjutnya.

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Pengertian sistem menurut beberapa ahli yaitu, menurut Sutabri (2012) pada buku Analisis Sistem Informasi, pada dasarnya sistem adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Selanjutnya menurut McLeod dikutip oleh Yakub (2012) dalam buku Pengantar Sistem Informasi mendefinisikan sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan. Sistem juga merupakan suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu.

Sedangkan menurut Jogiyanto (2005) pada buku Analisis dan Desain Sistem Informasi mendefinisikan 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.

2.1.1 Elemen Sistem

Menurut Kadir (2014) elemen yang membentuk sebuah sistem yaitu:

1. Tujuan

Setiap sistem memiliki tujuan (*goal*), entah hanya satu atau mungkin banyak. Tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tidak terarah dan tidak terkendali. Tentu saja tujuan antara satu sistem dengan sistem lain berbeda-beda.

2. Masukan

Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan dapat berupa hal-hal berwujud (tampak secara fisik) maupun yang tidak berwujud. Contoh masukan yang berwujud adalah bahan mentah, sedangkan contoh yang tidak berwujud adalah informasi (misalnya permintaan jasa dari pelanggan).

3. Proses

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran berguna, misalnya berupa informasi dan produk, tetapi juga bisa berupa hal-hal yang tidak berguna, misalnya saja sisa pembuangan atau limbah. Pada pabrik kimia, proses dapat berupa pemanasan bahan mentah. Pada rumah sakit, proses dapat berupa aktivitas pembedahan pasien.

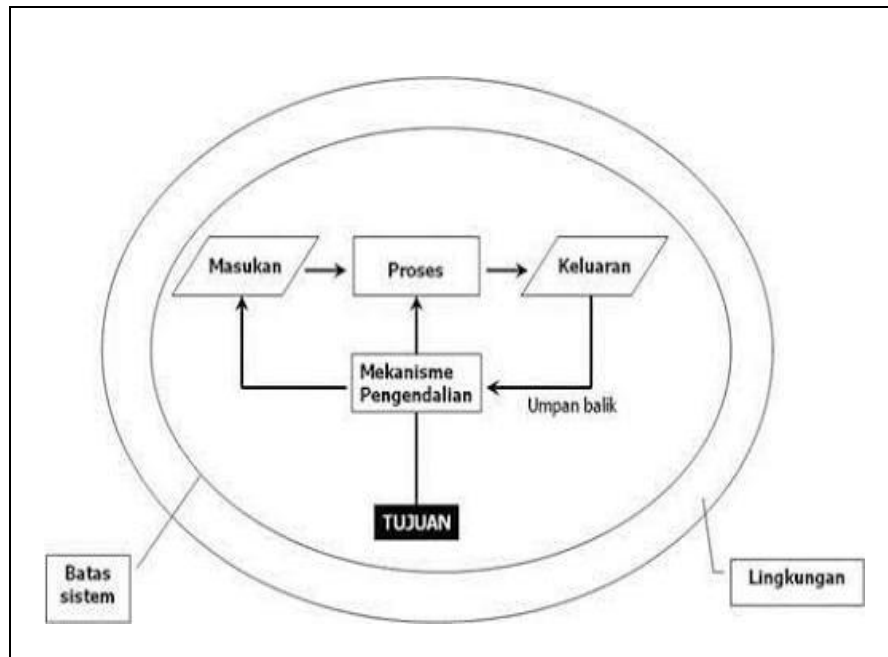
4. Keluaran

Keluaran (*output*) merupakan hasil dari pemrosesan. Pada sistem informasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan dan sebagainya.

5. Metode Pengendalian dan Umpan Balik

Mekanisme pengendalian (*control mechanism*) diwujudkan dengan menggunakan umpan balik (*feed back*), yang mencuplik keluaran. Umpan balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya adalah untuk mengatur agar sistem berjalan sesuai dengan tujuan.

Selain itu, sistem juga berinteraksi dengan lingkungan dan memiliki batas. Gambar II.1 memperlihatkan hubungan antara elemen dan juga kaitannya dengan lingkungan.



Gambar II.1 Elemen Sistem

(Sumber: Kadir, 2014)

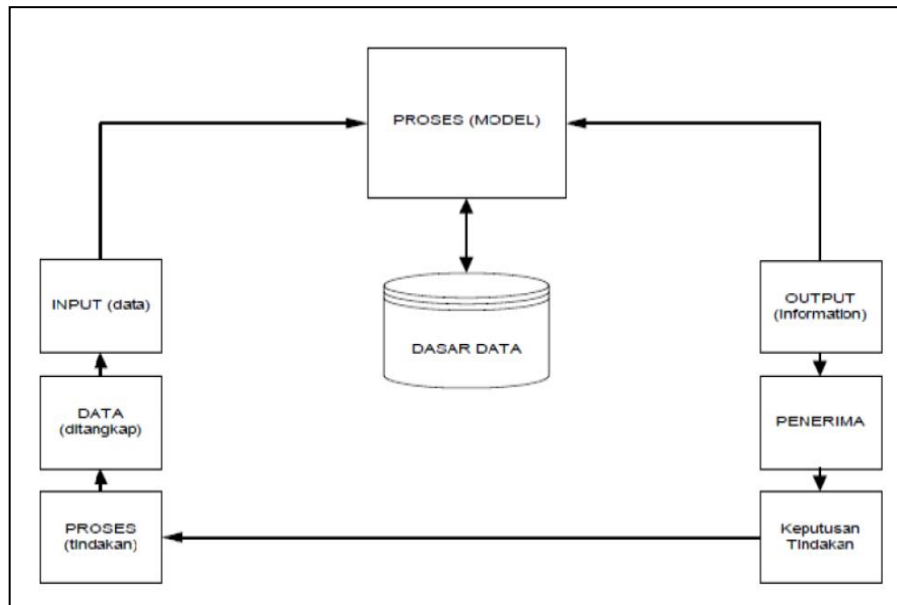
2.2 Pengertian Informasi

Menurut McLeod dikutip oleh Yakub (2012) pada buku Pengantar Sistem Informasi, informasi adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya. Sedangkan menurut Sutabri (2012) pada buku Analisis Sistem Informasi, informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

2.2.1 Siklus Informasi

Siklus informasi (*information cycle*) atau siklus pengolahan data (*data processing cycle*) adalah gambaran secara umum mengenai proses terhadap data

sehingga menjadi informasi yang bermanfaat bagi pengguna. Informasi yang menghasilkan informasi berikutnya, demikian seterusnya proses pengolahan data menjadi informasi. Untuk memperoleh informasi yang bermanfaat bagi penerimanya, perlu untuk dijelaskan bagaimana siklus yang terjadi atau dibutuhkan dalam menghasilkan informasi. Siklus informasi digambarkan pada Gambar II.4 sebagai berikut (Yakub, 2012).



Gambar II.2 Siklus Informasi

Sumber: Yakub (2012)

2.2.2 Kualitas Informasi

Menurut Sutabri (2012) pada buku Analisis Sistem Informasi, kualitas dari suatu informasi tergantung dari 3 hal, yaitu informasi harus akurat (*accurate*), tepat waktu (*timeliness*) dan relevan (*relevance*).

1. Akurat (*Accuracy*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya.

2. Tepat Waktu (*Time Lines*)

Informasi yang datang kepada penerima tidak boleh terlambat. Informasi yang sudah usang tidak mempunyai nilai lagi, karena informasi merupakan suatu landasan dalam mengambil sebuah keputusan dimana bila pengambilan keputusan terlambat maka akan berakibat fatal untuk organisasi.

3. Relevan (*Relevance*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk setiap orang berbeda. Menyampaikan informasi tentang penyebab kerusakan mesin produksi kepada akuntan perusahaan tentunya kurang relevan. Akan lebih relevan bila ditujukan kepada ahli teknik perusahaan. Sebaliknya informasi mengenai harga pokok produksi disampaikan untuk ahli teknik merupakan informasi yang kurang relevan, tetapi akan sangat relevan untuk seorang akuntan perusahaan.

2.2.3 Karakteristik Informasi

Menurut Yakub (2012) pada buku Pengantar Sistem Informasi, untuk tiap-tiap tingkatan manajemen dengan kegiatan yang berbeda, dibutuhkan informasi dengan karakteristik yang berbeda pula. Karakteristik dari informasi yaitu:

1. Kepadatan Informasi, untuk manajemen tingkat bawah karakteristik informasinya adalah terperinci dan kurang padat, karena digunakan untuk pengendalian operasi. Sedangkan untuk manajemen yang lebih tinggi tingkatannya, mempunyai karakteristik informasi yang semakin tersaring, lebih ringkas dan padat.
2. Luas Informasi, manajemen tingkat bawah karakteristik informasinya adalah terfokus pada suatu masalah tertentu, karena digunakan oleh manajer bawah yang mempunyai tugas khusus. Sedangkan untuk manajemen yang lebih tinggi tingkatannya, mempunyai karakteristik informasi yang semakin luas, karena manajemen atas berhubungan dengan masalah yang luas.
3. Frekuensi Informasi, manajemen tingkat bawah frekuensi informasi yang diterimanya adalah rutin, karena digunakan oleh manajer bawah yang

mempunyai tugas terstruktur dengan pola yang berulang-ulang dari waktu ke waktu. Manajemen yang lebih tinggi tingkatannya frekuensi informasinya adalah tidak rutin, karena manajemen tingkat atas berhubungan dengan pengambilan keputusan tidak terstruktur yang pola dan waktunya tidak jelas.

4. Akses Informasi, level bawah membutuhkan informasi yang periodenya berulang-ulang sehingga dapat disediakan oleh bagian sistem informasi yang memberikan dalam bentuk laporan periodik. Dengan demikian akses informasi tidak dapat secara online tetapi dapat secara *off line*. sebaliknya untuk level tinggi, periode informasi yang dibutuhkan tidak jelas sehingga manajer-manajer tingkat atas perlu disediakan akses *online* untuk mengambil informasi kapan pun mereka membutuhkan.
5. Waktu Informasi, manajemen tingkat bawah, informasi yang dibutuhkan adalah informasi historis, karena digunakan dalam pengendalian operasi yang memeriksa tugas rutin yang sudah terjadi. Untuk manajemen tingkat tinggi waktu informasi lebih ke masa depan berupa informasi prediksi karena digunakan untuk pengambilan keputusan strategik yang menyangkut nilai masa depan.
6. Sumber Informasi, karena manajemen tingkat bawah lebih berfokus pada pengendalian internal perusahaan. Maka manajer tingkat bawah lebih memerlukan informasi dengan data yang bersumber dari internal perusahaan sendiri. Manajer tingkat atas lebih berorientasi pada masalah perencanaan strategik yang berhubungan dengan lingkungan luar perusahaan. Karena itu membutuhkan informasi dengan data yang bersumber pada eksternal perusahaan.

2.3 Pengertian Sistem Informasi

Menurut O'Brian dikutip oleh Yakub (2012) pada buku Pengantar Sistem Informasi, sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras, perangkat lunak, jaringan komunikasi, dan

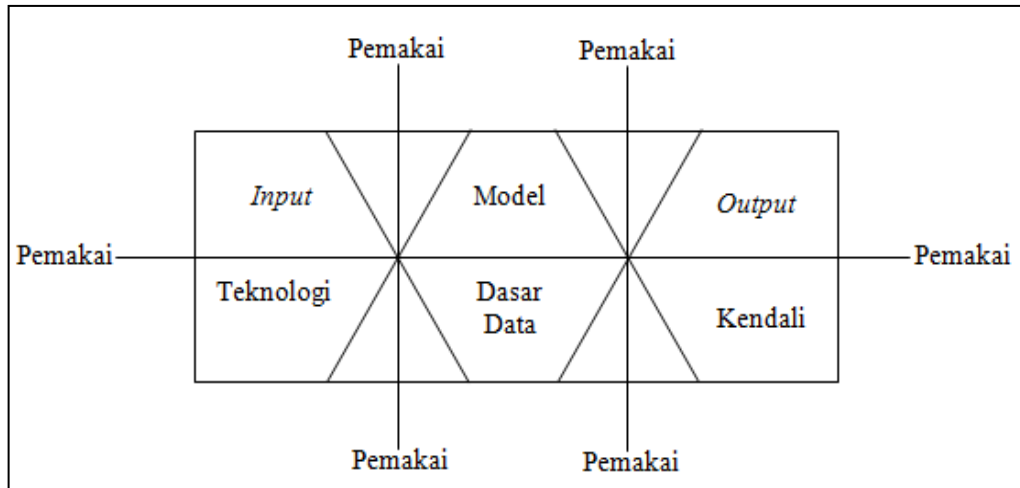
sumber daya data yang mengumpulkan, mengubah, menyebarkan informasi dalam sebuah organisasi.

2.3.1 Komponen Sistem Informasi

Menurut Yakub (2012) sistem informasi merupakan sebuah susunan yang terdiri dari beberapa komponen atau elemen. Komponen-komponen dari sistem informasi ini dapat digambarkan sebagai berikut ini:

1. Blok Masukan (*Input Block*), *input* memiliki data yang masuk ke dalam sistem informasi, juga metode-metode untuk menangkap data yang dimasukkan.
2. Blok Model (*Model Block*), blok ini terdiri dari kombinasi prosedur, logika, dan model matematis yang akan memanipulasi data *input* dan data yang tersimpan di basis data.
3. Blok Keluaran (*Output Block*), produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Blok Teknologi (*Technology Block*), blok teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu: teknisi (*brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*).
5. Basis Data (*Database Block*), basis data merupakan kumpulan dari data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

Sistem informasi memiliki lima blok utama yang saling tersusun yaitu blok masukan, model, keluaran, teknologi dan basis data. Untuk mempermudah dalam memahami blok-blok ini, berikut adalah Gambar II.3 yang menunjukkan adanya interaksi antara blok sistem informasi.



Gambar II.3 Interaksi Blok Sistem Informasi

(Sumber: Yakub, 2012)

2.4 Pengertian Pengendalian

Pada dasarnya rencana dan pelaksanaan merupakan satu kesatuan tindakan, walaupun hal ini jarang terjadi. Pengendalian diperlukan untuk melihat sejauh mana hasil yang telah tercapai, apakah telah sesuai dengan rencana atau malah terjadi kesenjangan akibat adanya penyimpangan-penyimpangan. Untuk lebih jelasnya mengenai pengertian pengendalian ada beberapa menurut para ahli:

1. Pengendalian (*controlling*), para manajer berusaha sedapat mungkin agar organisasi bergerak ke arah tujuannya. Apabila ada salah satu bagian organisasi bergerak ke arah yang salah, maka para manajer berusaha untuk mencari sebabnya dan kemudian mengarahkan kembali ke tujuan yang benar (Yakub, 2012).
2. *Controlling is the measuring and correcting of activities of subordinates to assure that events conform to plans* atau pengendalian adalah berhubungan dengan pembandingan kejadian-kejadian dengan rencana-rencana dan melakukan tindakan-tindakan koreksi yang perlu terhadap kejadian-kejadian yang menyimpang dari rencana-rencana (Koontz and cyrill O'donell, 2007).

3. Proses pengamatan dari pada pelaksanaan seluruh kegiatan organisasi untuk menjamin agar semua pekerjaan yang sedang dilakukan berjalan sesuai dengan rencana yang telah ditetapkan (Sondang P. Siagian, 2007).

Dari beberapa pendapat diatas, dapat disimpulkan bahwa pengendalian merupakan salah satu dari fungsi manajemen yang tujuannya untuk memberikan arahan agar pelaksanaan rencana dapat sesuai dengan yang telah ditentukan. Kegiatan pengendalian dimaksudkan untuk mencegah penyimpangan-penyimpangan dalam pelaksanaan kegiatan.

2.5 Pengertian Kualitas

Kualitas secara garis besar merupakan kepuasan pelanggan yang merupakan salah satu tujuan penting sebagian besar perusahaan. Menurut Render dan Heizer (2001) yang dialihbahasakan oleh Kresnohadi Ariyoto, kualitas adalah totalitas bentuk dan karakteristik barang atau jasa yang menunjukkan kemampuannya untuk memuaskan kebutuhan-kebutuhan yang tampak jelas maupun yang tersembunyi. Sedangkan menurut Assauri (2004), kualitas diartikan sebagai faktor-faktor yang terdapat dalam suatu barang/hasil yang menyebabkan barang/hasil tersebut sesuai dengan tujuan untuk apa barang/hasil tersebut dimaksudkan atau dibutuhkan.

Berdasarkan definisi diatas bahwa kualitas adalah suatu tingkatan spesifikasi produk yang dapat memuaskan konsumen dan hasil produksi sesuai dengan standar produksi yang telah ditetapkan.

2.6 Pengertian Pengendalian Kualitas

Pengendalian kualitas perlu dilaksanakan, karena pengendalian kualitas sangat erat hubungannya dengan peningkatan kualitas. Menurut Assauri (2004) dalam buku Manajemen Produksi dan Operasi dikatakan bahwa "Pengendalian kualitas adalah kegiatan memastikan apakah kebijakan dalam hal kualitas (standar) dapat tercermin dalam hasil akhir, atau dengan kata lain usaha untuk mempertahankan mutu atau kualitas dari barang-barang yang dihasilkan agar

sesuai dengan spesifikasi produk yang telah ditetapkan berdasarkan kebijakan pimpinan”. Dalam mewujudkan pelaksanaan dari pengendalian kualitas, kegiatan ini dilakukan oleh operator dan manajemen dari departemen yang bersangkutan dengan melakukan pengukuran pencapaian standar yang telah ditetapkan sebelumnya.

2.6.1 Tujuan Pengendalian Kualitas

Perusahaan agar dapat menghasilkan produk yang berkualitas maka harus dilakukan pengendalian kualitas, tetapi sebelumnya harus ditetapkan terlebih dahulu standar kualitas yang harus dicapai oleh suatu produk. Kegiatan pengendalian kualitas merupakan salah satu fungsi yang terpenting dari suatu perusahaan karena dengan adanya pengendalian kualitas, produk yang dihasilkan berkualitas baik dan sesuai dengan apa yang telah direncanakan. Pelaksanaan pengendalian kualitas dalam suatu perusahaan dimaksudkan untuk mencerminkan spesifikasi standar yang telah ditetapkan dalam produk atau hasil akhir.

Menurut Assauri (2004) tujuan dari pengendalian kualitas adalah sebagai berikut:

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.

2.7 Alat Pengendalian Kualitas

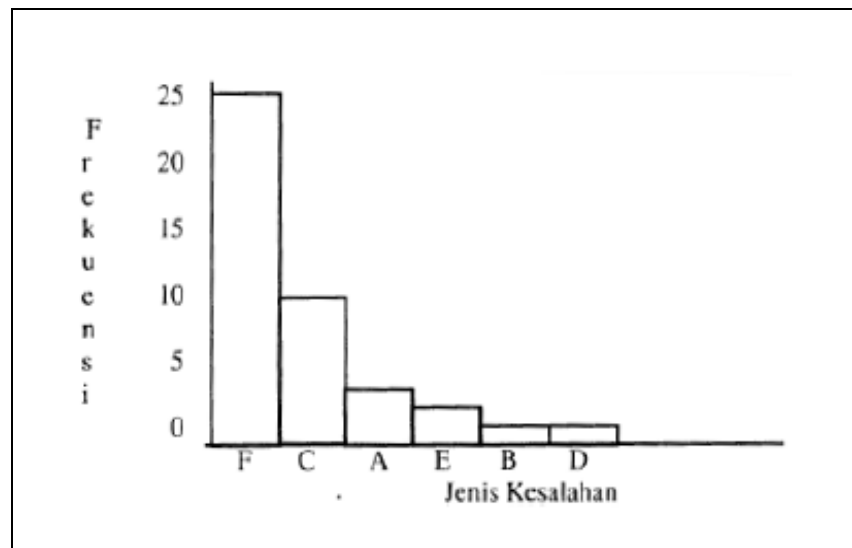
Alat pengendalian kualitas adalah 7 macam alat perbaikan mutu yang digunakan pada kegiatan siklus *quality control*. QC 7 Tools pertama kali ditegaskan oleh Kaoru Ishikawa, seorang profesor *engineering* di Universitas Tokyo pada tahun 1968 yang juga merupakan bapak *Quality Circles* (lingkaran kualitas), menegaskan bahwa “*As much as 95% of quality related problems in the factory can be solved with seven fundamental quantitative tools*” yaitu beliau

percaya bahwa statistik mampu menyelesaikan 95% persoalan kualitas. Ketujuh alat pengendalian kualitas ini antara lain (Heizer dan Render, 2006):

1. Diagram Pareto
2. Histogram
3. Lembar Pengecekan (*Check Sheet*)
4. Diagram Sebab-Akibat
5. Diagram Penyebaran (*Scatter Diagram*)
6. Diagram Alur
7. Peta Pengendali (*Control Chart*)

2.7.1 Diagram Pareto

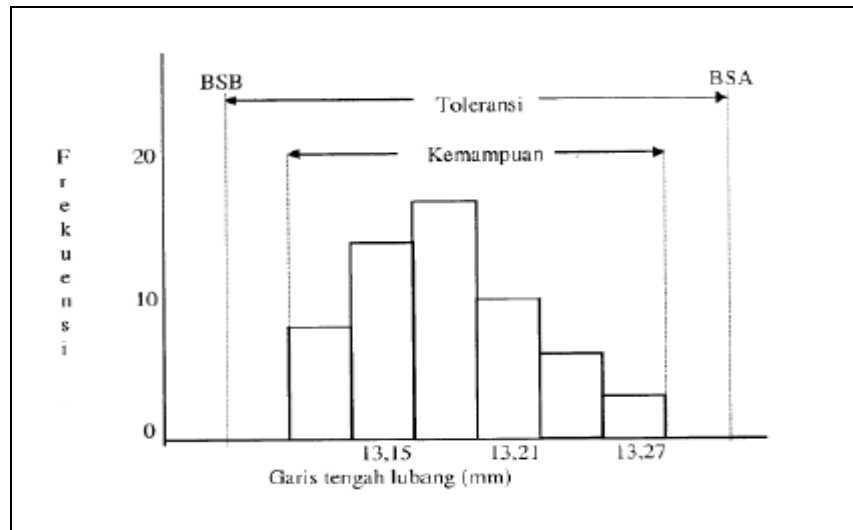
Diagram pareto pertama kali diperkenalkan oleh Alfredo Pareto (1848-1923). Diagram pareto merupakan suatu gambar yan mengurutkan klasifikasi data dari kiri ke kanan menurut urutan rangking tertinggi hingga terendah. Hal ini dapat membantu menemukan permasalahan yang paling penting untuk segera diselesaikan (rangking tertinggi) sampai dengan masalah yang tidak harus diselesaikan (rangking terendah).



Gambar II.4 Diagram Pareto
(Sumber: Tim Dosen Univ. Wijaya Putra Surabaya, 2009)

2.7.2 Histogram

Histogram menjelaskan variasi proses, namun belum mengurutkan ranking dari variasi terbesar sampai dengan yang terkecil. Histogram juga menunjukkan kemampuan proses, dan apabila memungkinkan, histogram dapat menunjukkan hubungan dengan spesifikasi proses dan angka-angka nominal, misalnya rata-rata. Dalam histogram, garis vertikal menunjukkan banyaknya observasi tiap-tiap kelas.



Gambar II.5 Histogram
(Sumber: Tim Dosen Univ. Wijaya Putra Surabaya, 2009)

2.7.3 Lembar Pengecekan (*Check Sheet*)

Tujuan pembuatan lembar pengecekan adalah menjamin bahwa data dikumpulkan secara teliti dan akurat oleh karyawan operasional untuk diadakan pengendalian proses dan penyelesaian masalah. Data dalam lembar pengecekan tersebut nantinya akan digunakan dan dianalisis secara tepat dan mudah.

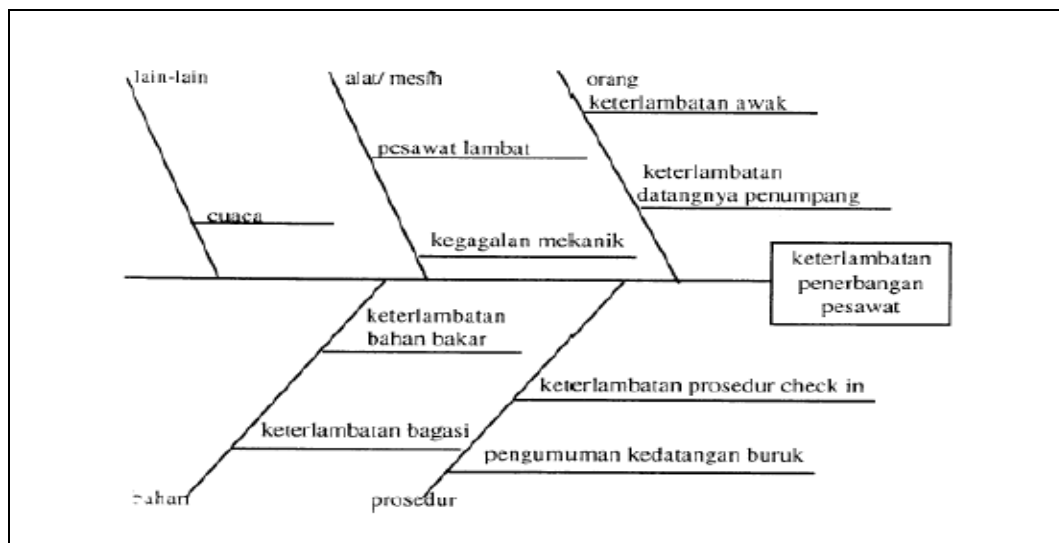
Jenis Kesalahan	Minggu -ke				
	1	2	3	4	5
Cara mengajar	v		v		
Pembimbingan akademik		v			v
Pembimbingan skripsi		v	v		
Pelayanan perpust.	v			v	v
Buku teks kuno	v	v	v	v	v
Penataan rak buku di perpustakaan	v			v	
Pelayanan registrasi			v		v
Pelayanan administrasi perkuliahan	v	v			
Pengaturan jadwal kuliah	v	v			v
Pengaturan ujian			v		
Penataan ruang kelas		v		v	v
Tidak ada dukungan					v

Gambar II.6 *Check Sheet*

(Sumber: Tim Dosen Univ. Wijaya Putra Surabaya, 2009)

2.7.4 Diagram Sebab-Akibat

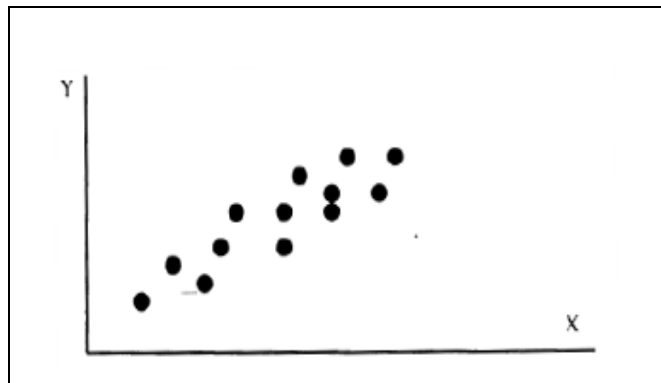
Diagram sebab-akibat dikembangkan oleh Dr. Kaoru Ishikawa pada tahun 1943, sehingga sering disebut dengan diagram Ishikawa. Diagram sebab-akibat menggambarkan garis dan simbol-simbol yang menunjukkan hubungan antara akibat dan penyebab suatu masalah. Diagram tersebut memang digunakan untuk mengetahui akibat dari suatu masalah untuk selanjutnya diambil tindakan perbaikan. Dari akibat tersebut kemudian dicari beberapa kemungkinan penyebabnya. Penyebab masalah ini pun berasal dapat berasal dari berbagai sumber utama, misalnya metode kerja, bahan, pengukuran, karyawan, lingkungan dan seterusnya.



Gambar II.7 *Fish Bone Diagram*
(Sumber: Tim Dosen Univ. Wijaya Putra, 2009)

2.7.5 Diagram Penyebaran (*Scatter Diagram*)

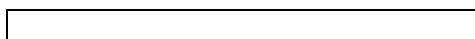
Scatter diagram merupakan cara yang paling sederhana untuk menentukan hubungan antara sebab dan akibat dalam dua variabel. Langkah-langkah yang diambil pun sederhana. Data dikumpulkan dalam bentuk pasangan titik (x , y). Dari titik-titik tersebut dapat diketahui hubungan antara variabel x dan variabel y , apakah terjadi hubungan positif atau negatif.

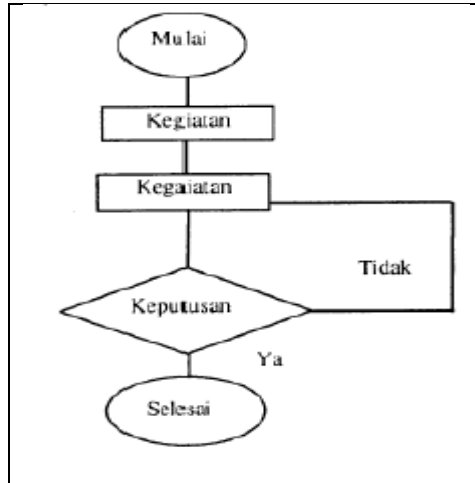


Gambar II.8 *Scater Diagram*
(Sumber: Tim Dosen Univ. Wijaya Putra Surabaya, 2009)

2.7.6 Diagram Alur

Diagram alur merupakan diagram yang menunjukkan aliran atau urutan suatu proses atau peristiwa. Diagram tersebut akan memudahkan dalam menggambarkan suatu sistem, mengidentifikasi masalah dan melakukan tindakan pengendalian. Diagram alur juga menunjukkan siapa pelanggan masing-masing tahapan proses. Diagram tersebut akan lebih baik apabila disusun oleh suatu tim, sehingga dapat diketahui serangkaian proses secara jelas dan tepat. Tindakan perbaikan dapat dicapai dengan pengurangan atau penyederhanaan tahapan proses, pengkombinasian proses, atau membuat frekuensi terjadinya langkah atau proses lebih efisien.



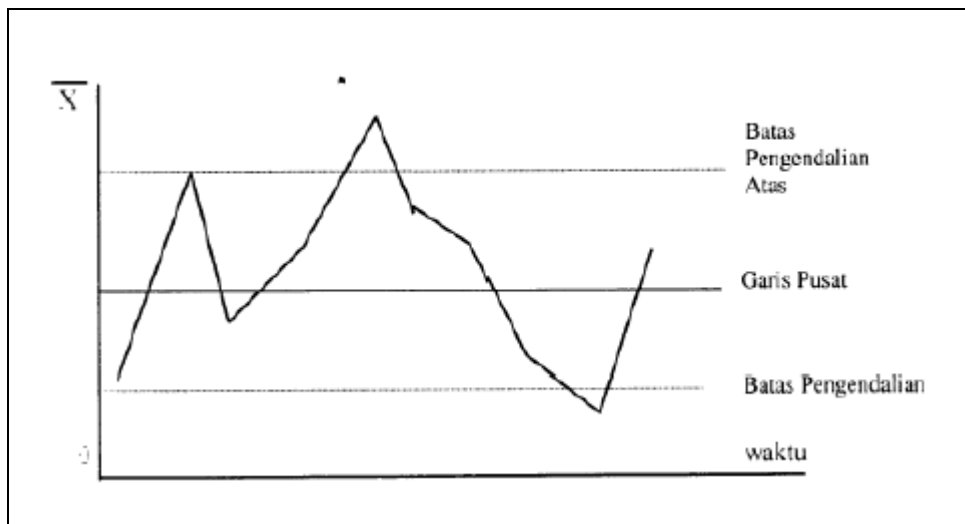


Gambar II.9 Flowchart

(Sumber: Tim Dosen Univ. Wijaya Putra Surabaya, 2009)

2.7.7 Peta Pengendali (*Control Chart*)

Peta pengendali menggambarkan perbaikan kualitas. Perbaikan kualitas terjadi pada dua situasi. Situasi pertama adalah ketika peta kendali dibuat, proses dalam kondisi tidak stabil. Kondisi yang diluar batas kendali terjadi karena sebab khusus (*assignable cause*), kemudian dicari tindakan perbaikan sehingga proses menjadi stabil. Hasilnya adalah adanya perbaikan proses.



Gambar II.10 Peta Kendali

(Sumber: Tim Dosen Univ. Wijaya Putra Surabaya, 2009)

2.8 *System Development Life Cycle (SDLC)*

SDLC atau *Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Rosa dan Shalahuddin, 2015).

Tahapan yang ada pada SDLC secara global adalah sebagai berikut:

1. Inisiasi (*initiation*)

Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

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

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

3. Perencanaan (*planning*)

Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.

4. Analisis kebutuhan (*requirements analysis*)

Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.

5. Desain (*design*)

Mentransformasikan kebutuhan *detail* menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

6. Pengembangan (*development*)

Mengkonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang

dibutuhkan, membuat basis data dan mempersiapkan prosedur kasus pengujian, mempersiapkan berkas atau *file* pengujian, pengkodean, pengkompilasian, memperbaiki dan membersihkan program serta peninjauan pengujian.

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

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

8. Implementasi (*implementation*)

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

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

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

10. Disposisi (*disposition*)

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

2.9 Metodologi Pengembangan Sistem

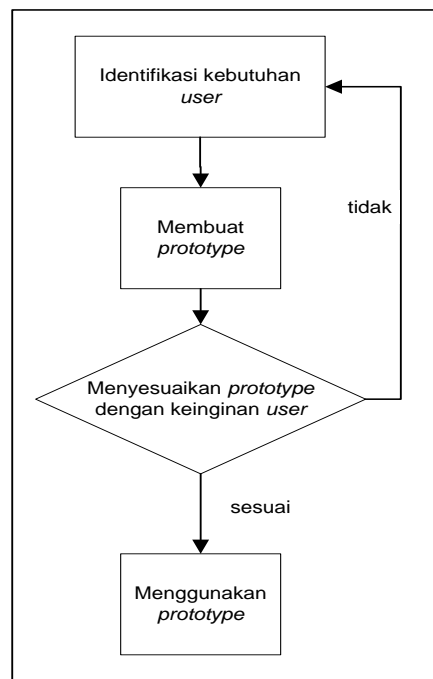
Metodologi pengembangan sistem adalah pendekatan formal untuk menerapkan SDLC. Ada berbagai metodologi pengembangan sistem dan setiap metodologi memiliki keunikan tersendiri. SDLC memiliki beberapa metode dalam penerapan tahapan prosesnya, salah satunya adalah metode *prototyping*.

Menurut McLeod (2011) prototipe adalah suatu versi sistem potensial yang disediakan bagi pengembang dan calon pengguna yang dapat memberikan

gambaran bagaimana kira-kira sistem tersebut akan berfungsi bila telah disusun dalam bentuk yang lengkap. Proses dalam memproduksi suatu prototipe disebut *prototyping*. Tujuannya adalah menghasilkan prototipe secepat mungkin dan memperoleh umpan balik dari pengguna yang akan memungkinkan prototipe untuk ditingkatkan sampai sistem dianggap sempurna. Adapun jenis *prototype* terdapat dua jenis prototipe yaitu prototipe evolusioner (*evolutionary prototype*) dan prototipe requirement (*requirement prototype*) (Mc Leod, 2011).

2.9.1 *Prototype Evolusioner*

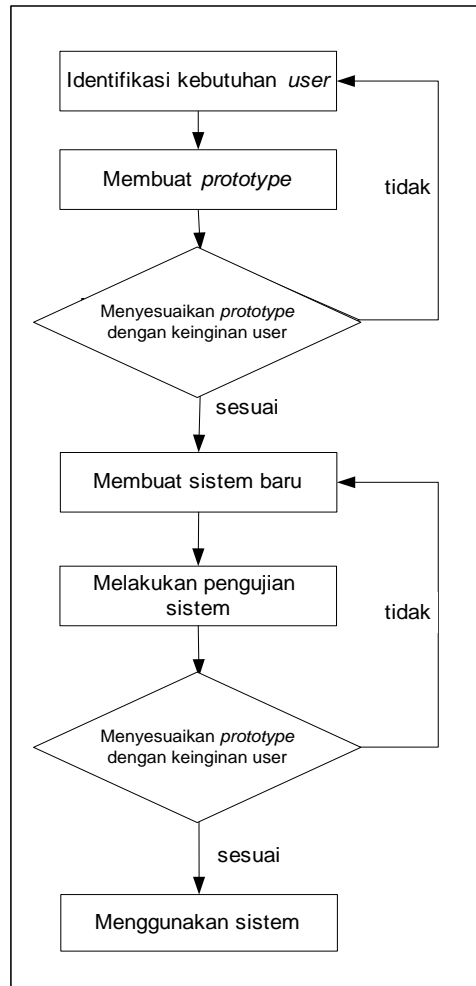
Prototype evolusioner adalah prototipe yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi. Ada empat langkah yang diambil dalam mengembangkan suatu *prototype evolusioner* yaitu identifikasi kebutuhan pengguna, mengembangkan prototipe, menentukan prototipe dapat diterima atau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



Gambar II.11 Pengembangan *Prototype Evolusioner*
(Sumber: McLeod, 2011)

2.9.2 *Prototype Requirement*

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



Gambar II.12 Pengembangan *Prototype Requirement*
(Sumber: McLeod, 2011)

2.10 *Object Oriented Analysis and Design (OOAD)*

OOA (*Object Oriented Analysis*) dan OOD (*Object Oriented Design*) dalam proses yang berulang-ulang seringkali memiliki batasan yang samar, sehingga kedua tahapan ini sering juga disebut OOAD (*Object Oriented Analysis and Design*) atau dalam bahasa Indonesia berarti Analisis dan Desain Berorientasi Objek (Rosa dan Shalahuddin, 2015).

1. Analisis berorientasi objek (*object oriented analysis*)

Adalah tahapan untuk menganalisis spesifikasi atau kebutuhan akan sistem yang akan dibangun dengan konsep berorientasi objek, apakah benar kebutuhan yang ada dapat diimplementasikan menjadi sebuah sistem berorientasi objek.

2. Desain berorientasi objek (*object oriented design*)

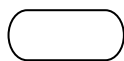


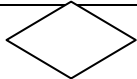
Adalah tahapan perantara untuk memetakan spesifikasi atau kebutuhan sistem yang akan dibangun dengan konsep berorientasi objek ke desain pemodelan yang lebih mudah diimplementasikan dengan pemrograman berorientasi objek.


2.11 Flowchart

Untuk menggambarkan sebuah algoritma yang terstruktur dan mudah dipahami oleh orang lain maka dibutuhkan alat bantu yang berbentuk diagram alir (*flowchart*). *Flowchart* menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga *flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu. Diagram alir ini selain dibutuhkan sebagai alat komunikasi, juga diperlukan sebagai dokumentasi (Sitorus, 2015).





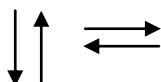
Tujuan dari *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas menggunakan simbol-simbol seperti yang dijelaskan pada Tabel II.1 berikut ini.

Tabel II.1 Simbol-Simbol *Flowchart*

Simbol	Nama	Keterangan
	Terminal	Menyatakan permulaan atau akhir suatu program.
	<i>Input-Output</i>	Meyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
	<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
	<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang

		akan menghasilkan dua kemungkinan jawaban: ya atau tidak.
	<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.



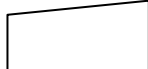
Tabel II.1 Simbol-Simbol *Flowchart* (lanjutan)

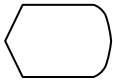

Simbol	Nama	Keterangan
	<i>Off-line Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
	<i>Predafined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
	<i>Punched Card</i>	Menyatakan <i>input berasal dari kartu atau output</i> ditulis ke kartu.
	<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>).
	<i>Flow</i>	Menyatakan jalannya arus suatu proses.

(Sumber: Sitorus, 2015)


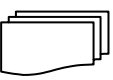






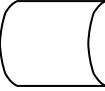

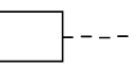
Selain simbol-simbol flowchart diagram di atas terdapat simbol-simbol flowchart diagram yang digunakan yang dapat dilihat pada tabel II.2.


Tabel II.2 Simbol *Flowchart*

Simbol	Nama	Keterangan
	Simbol Drum Magnetik	Menunjukkan <i>input/output</i> menggunakan drum magnetik
	Simbol Pita Kertas Berlubang	Menunjukkan <i>input/output</i> menggunakan pita kertas berlubang.
	Simbol Keyboard	Menunjukkan <i>input</i> yang menggunakan <i>online keyboard</i>

	Display	Menunjukkan <i>output</i> yang ditampilkan di monitor
	Simbol Kegiatan Manual	Menunjukkan pekerjaan manual

Tabel II.2 Simbol *Flowchart* (lanjutan)

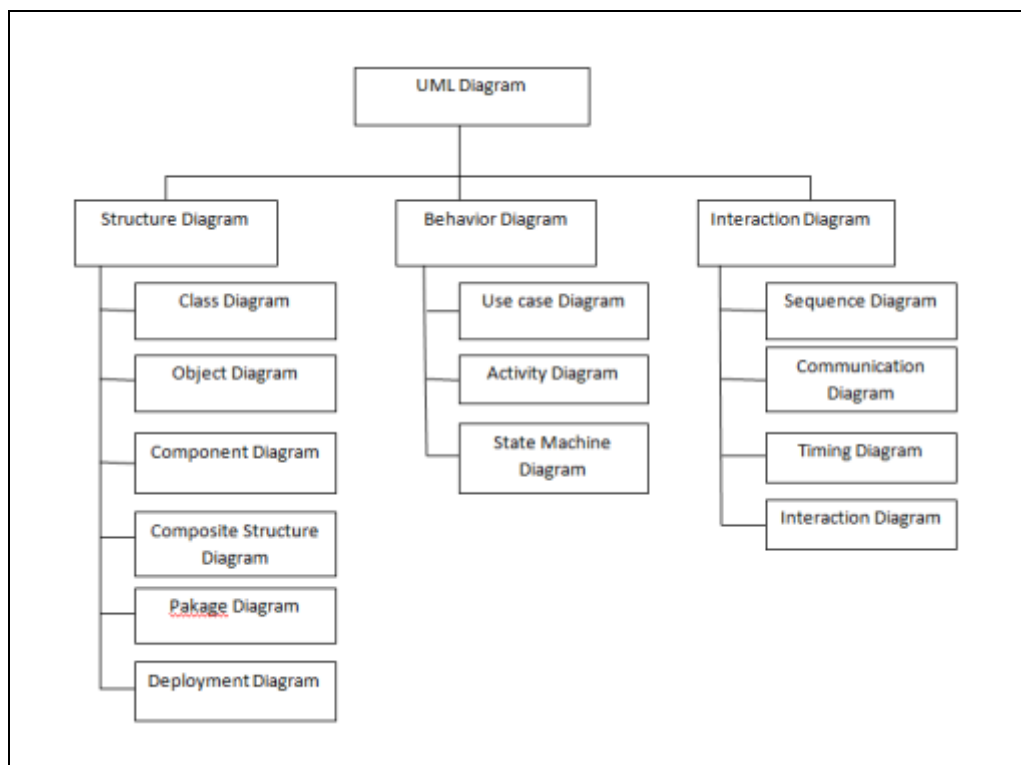
Simbol	Nama	Keterangan
	Simbol Proses Terdefinisi (<i>Predefined Process Symbol</i>)	Menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
	Dokumen Rangkap	Menggambarkan dokumen asli dan tembusannya
	Operasi Luar	Menunjukkan operasi yang dilakukan di luar proses operasi komputer
	Arsip Sementara	Menunjukkan tempat penyimpanan dokumen.
	Arsip Permanen	Menunjukkan tempat penyimpanan dokumen secara permanen yang tidak akan diproses lagi.
	Simbol Pengurutan <i>Offline</i>	Menunjukkan proses pengurutan data di luar proses komputer
	Simbol Pita Magnetik	Menunjukkan <i>input/output</i> menggunakan pita magnetik
	Simbol Hard Disk	Menunjukkan <i>input/output</i> menggunakan <i>hard disk</i>
	Simbol <i>Diskette</i>	Menunjukkan <i>input/output</i> menggunakan <i>diskette</i>
	Simbol <i>Data Storage</i>	Menunjukkan penyimpanan data
	Penjelasan	Menunjukkan penjelasan dari suatu proses

	Hubungan Komunikasi	Menunjukkan proses transmisi data melalui <i>channel</i> komunikasi
---	------------------------	--

(Sumber: Jogiyanto, 2005)

2.12 Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks tertentu. Pada Gambar II.13 dijelaskan bahwa UML terdiri dari 13 macam diagram yang dikelompokkan dalam tiga kategori (Rosa dan Shalahuddin, 2015).



Gambar II.13 Diagram UML
(Sumber: Rosa dan Shalahudin, 2015)

Berikut ini penjelasan singkat dari pembagian kategori UML tersebut (Rosa dan Shalahuddin, 2015):


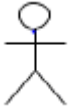


1. *Structure diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.


2. *Behavior diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan proses sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

2.12.1 Use Case Diagram


Use case adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Tabel II.3 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 frase 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 yang akan dibuat itu sendiri, jadi walaupun <i>symbol</i> dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase 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> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

	<<extend>>	
5.	<i>Generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.

Tabel II.3 Simbol-Simbol *Use Case Diagram* (lanjutan)


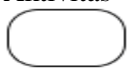
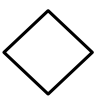
No.	Simbol	Deskripsi
6.	<<Include>> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.



(Sumber: Rosa dan Shalahuddin, 2015)

2.12.2 *Activity Diagram*

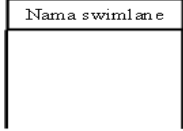
Activity diagram menggambarkan *work flow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Tabel II.4 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 sebuah status akhir.

Tabel II.4 Simbol-Simbol *Activity Diagram* (lanjutan)

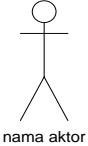
No.	Simbol	Deskripsi
6.	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.


(Sumber: Rosa dan Shalahuddin, 2014)

2.12.3 *Sequence Diagram*

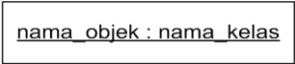

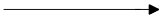
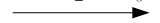


Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang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

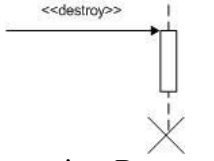
Tabel II.5 Simbol-Simbol *Sequence Diagram*

No.	Simbol	Deskripsi
1.	Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang,

		tapi aktor belum tentu merupakan orang.
2.	<p>Garis Hidup/<i>lifeline</i></p> 	Menyatakan kehidupan suatu objek.

Tabel II.5 Simbol-Simbol *Sequence Diagram* (lanjutan)

No.	Simbol	Deskripsi
3.	<p>Objek</p> 	Menyatakan objek yang berinteraksi pesan.
4.	<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.	<p><<create>></p>  <p>Pesan tipe <i>create</i></p>	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
6.	<p>1 : Nama_Metode ()</p>  <p>Pesan tipe <i>call</i></p>	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
7.	<p>1 : Masukan</p>  <p>Pesan tipe <i>send</i></p>	Menyatakan bahwa suatu objek mengirimkan data/masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8.	<p>1 : Keluaran</p>  <p>Pesan tipe <i>return</i></p>	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek

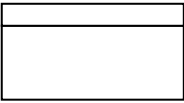



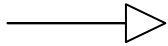
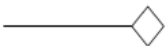
		tertentu, arah panah mengarah pada objek yang menerima kembalian.
9.	 <p>Pesan tipe <i>Destroy</i></p>	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

(Sumber: Rosa dan Shalahuddin, 2015)

2.12.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. (Rosa dan Shalahudin, 2014).

Tabel II.6 Simbol-Simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	Kelas 	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2.	Antarmuka/ <i>Interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	Asosiasi/ <i>Association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4.	Asosiasi Berarah/ <i>Directed Association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	Generalisasi/ <i>Generalization</i> 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.	Agregasi/ <i>Aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>).

(Sumber: Rosa dan Shalahuddin, 2015)

Dalam kelas terdapat beberapa *keys* yang berfungsi untuk membedakan semua basis data dalam tabel secara unik. Pengertian *keys* menurut Yanto (2016) yaitu sebagai berikut:

1. *Candidate key*

Candidate key adalah sejumlah kecil *attribute* yang secara unik mengidentifikasi setiap kejadian dari setiap tipe *entity*.

2. *Primary key*

Primary key adalah *candidate key* yang terpilih untuk mendefinisikan secara unik pada setiap kejadian dari sebuah tipe *entity*.

3. *Composite key*

Composite key adalah sebuah *candidate key* yang terdiri dari dua atau banyak *attribute*.

4. *Foreign key*

Foreign key adalah himpunan *attribute* dalam data *relationship* yang cocok dengan *candidate key* dari beberapa *relationship* lainnya.

Pada relasi *class diagram* terdapat suatu penanda yang disebut *multiplicity*. *Multiplicity* ini akan mengindikasikan berapa banyak obyek dari suatu kelas terelasi ke obyek lain. Berikut Tabel II.7 tentang notasi UML untuk *multiplicity*:

Tabel II.7 Tipe *Multiplicity*

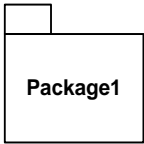
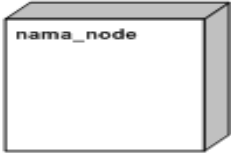


<i>Multiplicity</i>	Keterangan
*	Banyak
0	Nol
1	Satu
0..*	Antara nol sampai banyak
1..*	Antara satu sampai banyak
0..1	Nol atau satu
1..1	Tepat satu

(Sumber: Britton dan Doake, 2005)

2.12.5 *Deployment Diagram*

Deployment diagram merupakan gambaran proses-proses berbeda pada suatu sistem yang berjalan dan bagaimana relasi di dalamnya. Hal inilah yang mempermudah *user* dalam pemakaian sistem yang telah dibuat dan diagram tersebut merupakan diagram yang statis. *Deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi (Rosa dan Shalahuddin, 2015). Berikut mengenai simbol-simbol *deployment diagram* yang akan dijelaskan pada Tabel II.8:

Tabel II.8 Simbol-Simbol *Deployment Diagram*

No.	Simbol	Deskripsi
1.	<p><i>Package</i></p> 	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih <i>node</i> .
2.	<p><i>Node</i></p> 	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika di dalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka, komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.
3.	<p><i>Dependency</i></p> 	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
4.	<p><i>Link</i></p> 	Relasi antar objek.

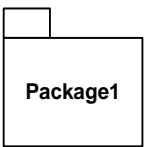
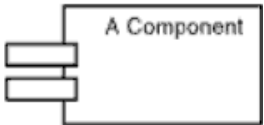
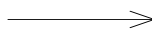

(Sumber: Rosa dan Shalahuddin, 2015)

2.12.6 Component Diagram

Component diagram dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem. Diagram komponen juga dapat digunakan untuk memodelkan hal-hal berikut:

1. *Source code* program perangkat lunak
2. Komponen *executable* yang dilepas ke *user*
3. Basis data secara fisik
4. Sistem yang harus beradaptasi dengan sistem lain
5. *Framework* sistem, *framework* pada perangkat lunak merupakan kerangka kerja yang dibuat untuk memudahkan pengembangan dan pemeliharaan aplikasi.

Tabel II.9 Simbol-simbol *Component Diagram*

No.	Simbol	Deskripsi
1.	<p><i>Package</i></p> 	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih komponen.
2.	<p>Komponen</p> 	Komponen <i>system</i> .
3.	<p>Kebergantungan/<i>dependency</i></p> 	Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai.
4.	<p>Antarmuka/<i>interface</i></p> 	Sama dengan konsep <i>interface</i> pada pemograman berorientasi objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen.
5.		Relasi antar komponen.

	<i>Link</i> _____	
--	----------------------	--

(Sumber: Rosa dan Shalahuddin, 2015)

2.13 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.

Keuntungan penggunaan kamus data adalah (Sommerville, 2003):

1. Kamus data merupakan mekanisme untuk manajemen nama. Banyak orang yang harus menciptakan nama untuk entitas dan relasi ketika mengembangkan model sistem yang besar. Nama-nama ini harus dipakai secara konsisten dan tidak boleh bentrok. Kamus data dapat memeriksa keunikan nama dan memberitahu analis persyaratan sekiranya terjadi duplikasi nama.
2. Kamus data sebagai tempat penyimpanan informasi yang dapat menghubungkan analis, desain, implementasi dan evolusi. Sementara sistem dikembangkan, informasi diambil untuk memberitahu perkembangan informasi baru ditambahkan pada sistem. Semua informasi mengenai entitas berada pada satu tempat. Berikut adalah contoh penulisan kamus data:

Spesifikasi Tabel pemasok

Nama Tabel : Pemasok

Tipe : File master

Tabel II.10 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)

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

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

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

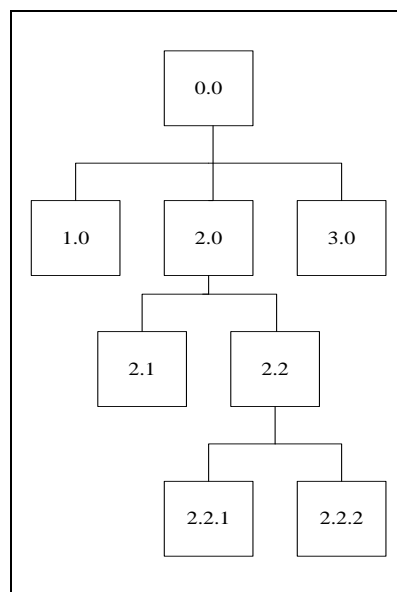
1. Untuk menyediakan suatu struktur guna memahami fungsi-fungsi dari sistem.
2. Untuk lebih menekankan fungsi-fungsi yang harus diselesaikan oleh program, bukannya menunjukkan statemen-statemen program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan yang jelas dari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

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

menggunakan tiga macam diagram untuk masing-masing tingkatannya, yaitu sebagai berikut:

1. *Visual Table Of Contents (VTOC)*

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

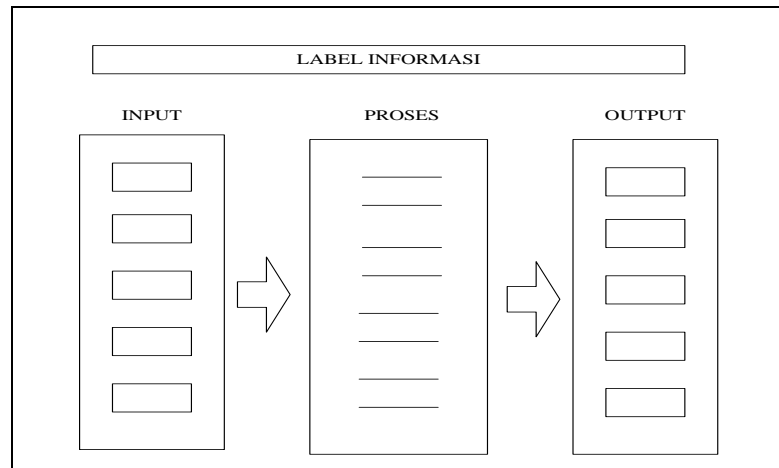


Gambar II.14 *Visual Table Of Contents*
(Sumber: Jogiyanto, 2005)

2. *Overview Diagram*

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

Berikut adalah Gambar II.15 *Overview diagram*.



Gambar II.15 *Overview Diagram*
(Sumber: Jogiyanto, 2005)

3. *Detail Diagram*

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

2.15 *Hypertext Preprocessor (PHP)*

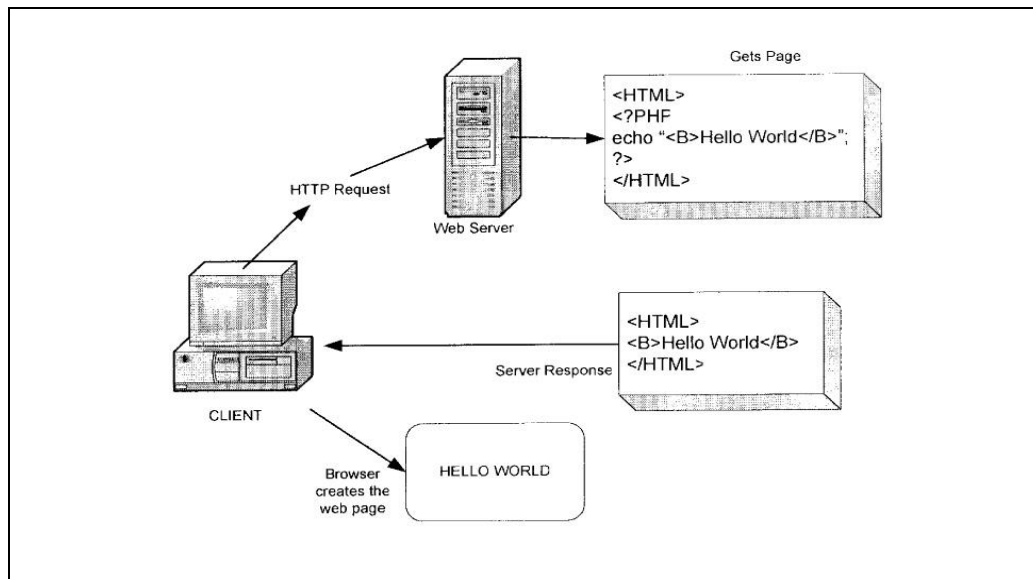
PHP adalah singkatan dari *hypertext preprocessor* yaitu bahasa pemrograman yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis. PHP dapat dikatakan sebagai sebuah *server-side embedded script language* yang artinya semua sintaks dan perintah program yang ditulis akan sepenuhnya dijalankan oleh *server* dan dapat juga diletakkan pada *script hyper text markup language (HTML)*.

Pada prinsipnya *server* akan bekerja apabila ada permintaan dari *client*, maka dalam hal ini *client* menggunakan kode-kode PHP untuk mengirim permintaan ke *server* ketika menggunakan PHP sebagai *server-side embedded script language*, maka *server* akan melakukan beberapa hal sebagai berikut (Wahana Komputer, 2010):

1. Membaca permintaan dengan *script PHP* yang berasal dari *browser*.
2. Mencari halaman/*page* di *server (server pages)*.
3. Melakukan processing melalui instruksi yang diberikan oleh PHP untuk melakukan modifikasi pada halaman/*page*

4. Mengirim kembali halaman tersebut kepada *client* melalui *internet* atau *intranet* yang merupakan proses *echo/print*.

Siklus PHP dalam *web server* dapat dijelaskan pada Gambar II.17.



Gambar II.16 Siklus PHP dalam *Web Server*
(Sumber: Wahana Komputer, 2010)

2.16 MySQL

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

MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam basis data sejak lama, yaitu SQL (*Structure Query Language*). SQL adalah sebuah konsep pengoperasian basis data terutama untuk proses seleksi, pemasukan, perubahan, dan penghapusan data yang dimungkinkan dapat dikerjakan dengan mudah dan otomatis (Sutaji, 2012).

2.16.1 Keunggulan MySQL

Berikut keunggulan dari MySQL, di antaranya adalah (Sutaji, 2012):

1. *Portability*

Dapat berjalan stabil pada berbagai sistem operasi, di antaranya: Windows, Linux, FreeBSD, MacOS X Server, Solaris, Asigma.

2. *Open source*

Didistribusikan secara gratis di bawah lisensi dari *General Public License* (GPL), dimana setiap orang bebas untuk menggunakannya tetapi tidak boleh untuk dijadikan program induk turunan bersifat *close source* (komersial).

3. *Multi User*

Dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan.

4. *Performance tuning*

Mempunyai kecepatan yang tinggi dalam menangani *query*.

5. *Column types*

Memiliki tipe data yang sangat kompleks, seperti: *signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year* dan *enum*.

6. *Command dan function*

Memiliki operator dan fungsi penuh yang mendukung *select* dan *where* dalam *query*.

7. *Security*

Memiliki beberapa lapisan keamanan seperti tingkat *subnet mask*, *hostname*, *privilege user* dengan sistem perijinan yang mendetail serta *password* yang ter-enkripsi.

8. *Scalability* dan *limits*

Mampu menangani basis data dalam jumlah besar, dengan jumlah *field* lebih dari 50 juta, 60 ribu tabel dan 5 miliar *record*. Batas indeks mencapai 32 buah per tabel.

9. *Localization*

Dapat mendeteksi pesan kesalahan (*error code*) pada klien dengan menggunakan lebih dari 20 bahasa.

10. *Connectivity*

Dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix Socket*, *Named Pipes*.

11. *Interface*

Memiliki antarmuka terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan API.

12. *Client* dan *tools*

Dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi basis data sekaligus dokumen petunjuk *online*.

13. Struktur tabel

Memiliki struktur tabel yang lebih fleksibel dalam menangani *alter* tabel dibandingkan dengan PostgreSQL dan Oracle.

2.16.2 Tipe Data MySQL

MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam merancang *table*.

Beberapa jenis data yang tersedia pada MySQL dapat dilihat pada Tabel II.11 (Sutaji, 2012).

Tabel II.11 Jenis Data Pada MySQL

Jenis Data	Keterangan
<i>CHAR</i>	Sebuah <i>string</i> dengan panjang tetap. Sisa Jumlah karakter yang belum terisi akan diisi dengan spasi, akan tetapi spasi ini dibuang jika data dipanggil. Jangkauan nilai M adalah 1-255 karakter.
<i>VARCHAR</i>	<i>String</i> dengan panjang berupa variabel. M bisa mencapai 65535.
<i>DATE</i>	Data berupa tanggal. Format tanggal dalam bentuk ‘YYYY-MM-DD’.

Tabel II.11 Jenis Data Pada MySQL (lanjutan)

Jenis Data	Keterangan
<i>TIME</i>	Data berupa waktu. Format waktu dalam bentuk ‘HH:MM:SS’.
<i>TINYINT</i>	Bilangan antara -128 sampai dengan +127.
<i>SMALLINT</i>	Bilangan antara -32768 sampai dengan +32767.
<i>MEDIUMINT</i>	Bilangan antara -8388608 sampai dengan +8388607.
<i>INT</i>	Bilangan antara -2147683648 sampai dengan +2147683647
<i>FLOAT</i>	Bilangan <i>floating point</i> yang kecil (presisi tunggal). Jangkauan nilai yang diizinkan adalah -3.402823466E+38 s/d -1.175494351e-38,0 dan 1.175494351E-38 s/d 3.402823466E+38.
<i>DOUBLE</i>	Bilangan <i>floating point</i> dengan ukuran normal (presisi ganda). Jangkauan nilai yang diizinkan adalah -1.7976931348623157E+308 s/d -2.225073858507201E-308,0 dan 2.225073858507201E-308 s/d 1.7976931348623157E+308.
<i>ENUM</i>	Sebuah <i>enumeration</i> . Sebuah obyek string yang hanya boleh memiliki satu nilai, yang terambil dari ‘value1’, ‘value2’, ..., NULL atau nilai spesial “” <i>error</i> . Sebuah <i>enum</i> dapat menampung 65535 pilihan nilai.

<i>TEXT, BLOB</i>	Sebuah <i>TEXT</i> atau <i>BLOB</i> dengan panjang karakter maksimum 65535 karakter.
-------------------	--

(Sumber: Sutaji, 2012)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian yang penulis gunakan dimulai dari mengidentifikasi masalah sampai dengan kesimpulan yang membentuk sebuah alur yang sistematis. Metodologi penelitian ini juga digunakan sebagai pedoman dalam pelaksanaan penelitian Tugas Akhir agar hasil yang dicapai tidak menyimpang dari tujuan yang telah ditentukan sebelumnya.

3.2 Jenis dan Metode Pengumpulan Data

Metode pengumpulan data merupakan cara yang dilakukan oleh penulis dalam mengumpulkan data-data yang diperlukan untuk kegiatan penelitian.

3.2.1 Sumber Data Primer

Data primer merupakan data yang didapatkan langsung dari sumber yang diamati, dalam hal ini adalah pengamatan data dan informasi langsung dari perusahaan. Dalam mendapatkan data primer, teknik pengumpulan data dan informasi yang dilakukan penulis, yaitu:

1. Observasi

Tahap ini merupakan tahap melakukan pengamatan langsung pada sistem yang berjalan divisi *Painting*. Pengamatan yang dilakukan mengenai alur proses pengendalian kualitas *accumulator* pada divisi *Painting* serta dokumen yang terkait dengan sistem yang diteliti, seperti laporan *loading-unloading* dan *quality monthly report*. Hasil dari pengamatan yang dilakukan menjadi landasan dalam melakukan pengembangan sistem yang akan dibuat.

2. Wawancara

Wawancara adalah metode pengumpulan data melalui tanya jawab secara langsung dengan pihak-pihak yang berkaitan dengan sistem pengendalian kualitas pada divisi *Painting*. Pihak yang diwawancarai adalah Manajer divisi *Painting*.

3.2.2 Sumber Data Sekunder

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

3.3 Metode Pengembangan Sistem

Menurut McLeod (2011) *evolutionary prototype* yaitu *prototype* yang secara terus menerus diperbaiki sampai semua kriteria yang baru terpenuhi. Pada pendekatan evolusioner, suatu *prototype* dibangun berdasarkan pada kebutuhan dan pemahaman secara umum. *Prototype* kemudian diubah dan dievolusikan dari pada dibuang. *Prototype* yang dibuang biasanya digunakan dengan aspek sistem yang dimengerti secara luas dan dibangun atas kekuatan tahapan *evolutionary prototype*.

Langkah-langkah dalam pembuatan suatu *evolutionary prototype* adalah sebagai berikut:

1. Mengidentifikasi kebutuhan pengguna. Pengembangan mewawancarai pengguna untuk mendapatkan ide mengenai apa yang diminta dari sistem.
2. Membuat satu *prototype*. Pengembang merancang *prototype* yang sesuai dengan kebutuhan pengguna.
3. Menentukan apakah *prototype* kepada para pengguna untuk mengetahui apakah telah memberikan hasil yang memuaskan. Jika ya, langkah 4 akan diambil, jika tidak, *prototype* direvisi dengan mengulang kebutuhan pengguna.
4. Menggunakan *prototype*.

3.4 Kerangka Penelitian

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

1. Studi Pendahuluan

Melakukan studi pendahuluan terlebih dahulu dengan mencari dan menentukan judul penelitian, mencari literatur dan menentukan apa yang melatar belakangi masalah yang terjadi. Tujuan melakukan studi pendahuluan adalah untuk mendapatkan pengetahuan umum mengenai sistem apa yang sedang diteliti.

2. Pokok Permasalahan

Pokok dari permasalahan yang menjadi objek penelitian ini adalah mengenai persoalan solusi pengendalian kualitas *accumulator* pada divisi *Painting* pada PT Kurnia Manunggal Sejahtera. Identifikasi dari permasalahan tersebut adalah kesulitan dalam memberikan laporan yang berhubungan dengan kualitas *accumulator* dikarenakan masih menggunakan aplikasi *spreadsheet* yang dalam proses pembuatan *quality monthly report* diperlukan waktu yang lama serta masih terdapat beberapa laporan atau dokumen pengendalian kualitas *accumulator* yang hanya disimpan di dalam arsip.

3. Tujuan Penelitian

Pada tahap ini dijelaskan tujuan penelitian yang dilakukan penulis, yaitu untuk merancang sebuah aplikasi sistem pengendalian kualitas *accumulator* yang mampu mendata, menyimpan, dan mengelola setiap *quality monthly report* sehingga para pengguna akan mendapatkan informasi laporan yang lebih cepat, mudah dan akurat.

4. Batasan masalah

Pada tahap ini dijelaskan batasan masalah yang dilakukan penulis, yaitu melakukan penelitian dan analisis mengenai sistem informasi pengendalian kualitas *accumulator* agar penelitian yang dilakukan lebih fokus dan lebih terarah.

5. Pengumpulan Data

Dalam mengumpulkan data, metode yang digunakan yaitu dengan metode wawancara, dan observasi (pengamatan). Metode wawancara dilakukan dengan kegiatan tanya jawab terhadap karyawan divisi *Painting*, pertanyaan yang diajukan yaitu seputar pengendalian kualitas *accumulator* yang berlangsung saat ini. Pengamatan dilakukan dengan cara sebagai berikut:

- a. Mempelajari dokumen organisasi, seperti profil perusahaan dan bagan struktur organisasi.
- b. Mempelajari sistem informasi pengendalian kualitas *accumulator* yang sedang berjalan di PT Kurnia Manunggal Sejahtera, termasuk mempelajari aliran dokumen seperti laporan *loading-unloading* dan *quality monthly report*.

6. Pengembangan Sistem Menggunakan *Evolutionary Prototype*

a. Identifikasi Kebutuhan Pengguna

Fase ini dilakukan untuk mengidentifikasi apa saja yang dibutuhkan oleh pengguna pada sistem pengendalian kualitas *accumulator*.

b. Pembuat *Prototype*

Membuat sebuah *prototype* sesuai dengan analisis yang dilakukan.

Pengamatan dilakukan dengan cara sebagai berikut:

1. Memodelan sistem yang menggunakan UML, *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*.
2. Memodelkan data menggunakan *Class Diagram* dan kamus data.
3. Merancang sistem menggunakan *flowchart*, HIPO dan antarmuka.

c. Menentukan apakah *prototype* dapat diterima

Pengembang mendemonstrasikan *prototype* kepada para pengguna untuk mengetahui apakah telah memberikan hasil yang memuaskan. Jika ya, langkah d akan diambil, jika tidak, *prototype* direvisi dengan mengulang kembali langkah a dengan pemahaman yang lebih baik mengenai kebutuhan pengguna.

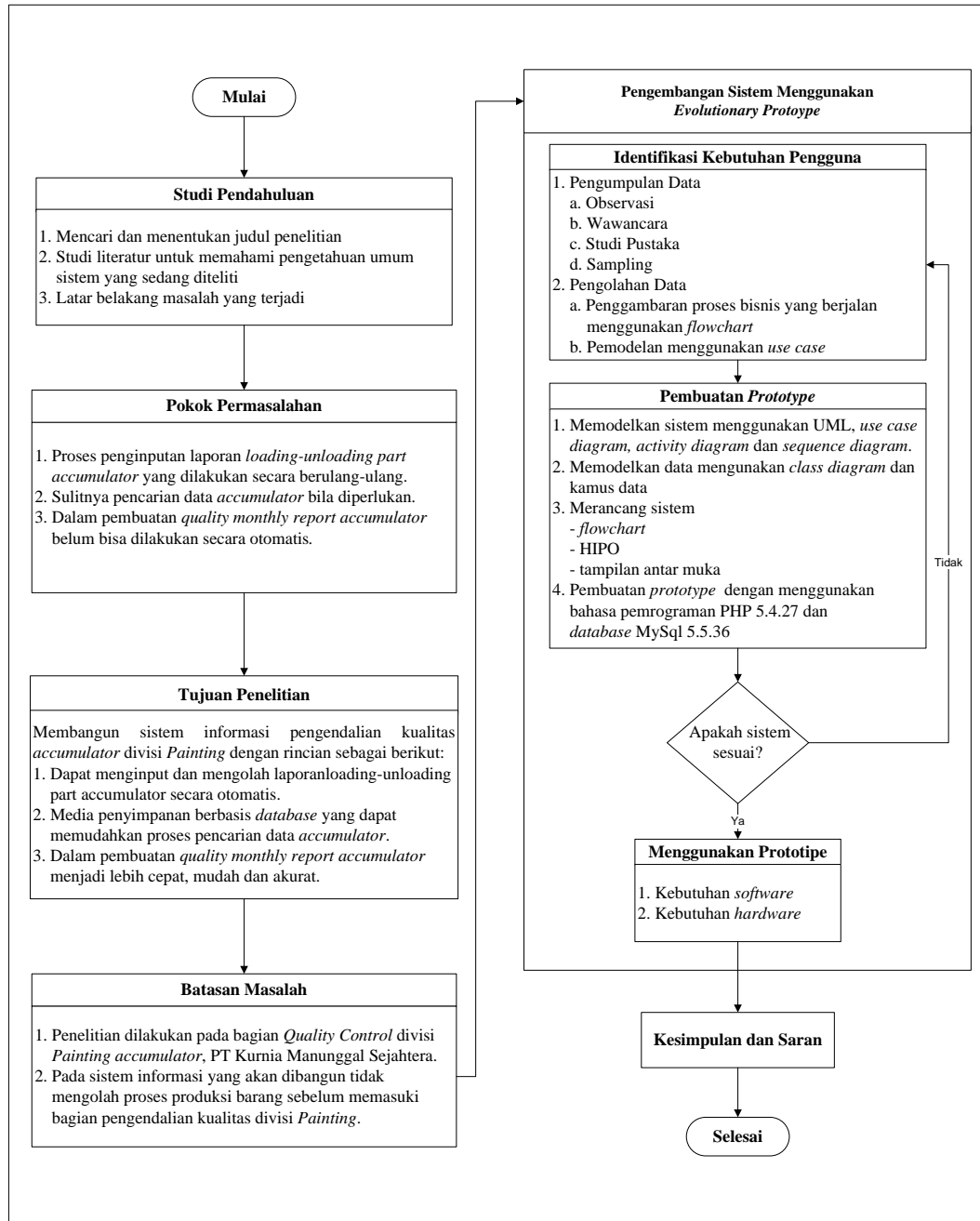
d. Menggunakan *prototype*

Pada tahap ini yang dilakukan pengumpulan dan menganalisis data kembali dan mencari tahu kebutuhan apa yang dibutuhkan oleh pengguna sistem, serta melakukan perancangan sistem ulang sesuai dengan kebutuhan pengguna.

7. Kesimpulan dan saran

Mengambil kesimpulan dari hasil analisis sistem berjalan, sistem yang diusulkan dan aplikasi yang dirancang serta memberikan saran.

Kerangka penelitian dibuat dalam bentuk *flowchart* yang menggambarkan tahap-tahap kegiatan mulai dari awal hingga akhir dapat dilihat pada Gambar III.1.



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

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Perusahaan

PT Kurnia Manunggal Sejahtera atau biasa disingkat dengan PT Kumase didirikan pada tahun 2001. PT Kumase merupakan anak perusahaan dari PT Baja Kurnia. PT Baja Kurnia memulai usahanya pada tahun 1978. PT Baja Kurnia adalah perusahaan *joint venture* antara Bapak Razak Musa dengan Ahmad Kalla dan Yusuf Kalla. Pada tahun 1978 PT Baja Kurnia mulai berdiri. Tahun 1985 PT Baja Kurnia menjadi perseroan terbatas dan menempati lokasi baru seluas 6000 meter. Tahun 1993 menjadi mitra Astra Mitra Ventura. Tahun 1999 menambah fasilitas peleburan dengan menggunakan tungku peleburan induksi listrik. Tahun 2002 menggunakan fasilitas peleburan *kupola continue* kapasitas 5 ton per jam. Tahun 2005 penambahan fasilitas peleburan induksi kapasitas 1 ton dan *mixer continue* untuk cetakan *furans*. Kemudian tahun 2010 perluasan lokasi pabrik menjadi 15.000 meter persegi. Tahun 2012 penambahan peralatan produksi antara lain induksi kapasitas 6 ton, *sand plant* untuk *green sand*, mesin perkakas CNC, mesin perkakas konvensional. Tahun 2001 PT Baja Kurnia mendirikan anak perusahaan yaitu PT Kurnia Manunggal Sejahtera.

PT Kurnia Manunggal Sejahtera adalah perusahaan yang memproduksi komponen otomotif dan elektronik industri, yang mengkhususkan diri dalam 3 divisi yaitu divisi *Stamping*, divisi *Painting* dan divisi Konstruksi Baja. Adapun kapasitas produksi pada PT Kurnia Manunggal Sejahtera ini mencapai 238.000 unit/tahun. PT Kurnia Manunggal Sejahtera juga telah menerima sertifikasi ISO-9001 untuk *management quality control*. Pada bagian *painting*, produk yang dihasilkan yaitu *accumulator*, *stay mirror* besar, *stay mirror* kecil, blok mesin vario, blok mesin beat dan rangka *air conditioner*. PT Kurnia Manunggal Sejahtera sendiri merupakan pemasok untuk beberapa perusahaan

industri, diantaranya adalah PT Panasonic, PT Sanyo, PT Astra Honda Motors dan beberapa perusahaan besar lainnya.

Komitmen PT Kurnia Manunggal Sejahtera untuk memenuhi permintaan pelanggan dengan memberikan produk yang berkualitas. PT Kurnia Manunggal Sejahtera percaya bahwa kepuasan pelanggan menghasilkan kepercayaan pelanggan dan dengan kepercayaan PT Kurnia Manunggal Sejahtera bisa sukses.

4.2 Profil Perusahaan

Profil singkat PT Kurnia Manunggal Sejahtera sebagai berikut:

Nama Perusahaan	: PT Kurnia Manunggal Sejahtera
Status Perusahaan	: Perseroan Terbatas
Pendiri	: Bapak Rajak Musa, Ahmad Kalla, dan Jusuf Kalla
Tahun Berdiri	: 2001
Alamat	: Jl. Pancasila V No.25 Gunung Putri, Bogor, Jawa Barat
Kapasitas Produksi	: 238.000 unit/tahun
Standar Produk	: SII (<i>Standar Industri Indonesia</i>) SNI (<i>Standar Nasional Indonesia</i>) HES (<i>Honda Engineering Standar</i>) ROHS (<i>Restriction of Hazardous Substances Directive</i>) ISO 9001 (<i>Management Quality Control</i>)

4.3 Visi dan Misi Perusahaan

Setiap perusahaan memiliki tujuan yang akan dicapai. Begitu juga PT Kurnia Manunggal Sejahtera yang bergerak di bidang industri komponen. Untuk mencapai tujuan tersebut perusahaan merencanakan visi dan misi. Visi dan misi inilah yang akan menjadi landasan bagi PT Kurnia Manunggal Sejahtera dalam menjalankan usahanya dengan 7 pengelolaan dan pengembangan, 2 *point* ekspansi dan 1 *point* pengayoman. Adapun visi dan misi perusahaan sebagai berikut:

1. Visi:

Menjadi perusahaan terkemuka dan terpercaya untuk industri otomotif dan komponen elektronik.

2. Misi:
 - a. Perbaiki terus menerus untuk kinerja terbaik dan berkelanjutan dalam semua aspek.
 - b. Membangun hubungan yang saling menguntungkan untuk semua mitra bisnis.
 - c. Membangun sumber daya manusia yang berkualitas dan terintegritas.

4.4 Tujuan Perusahaan

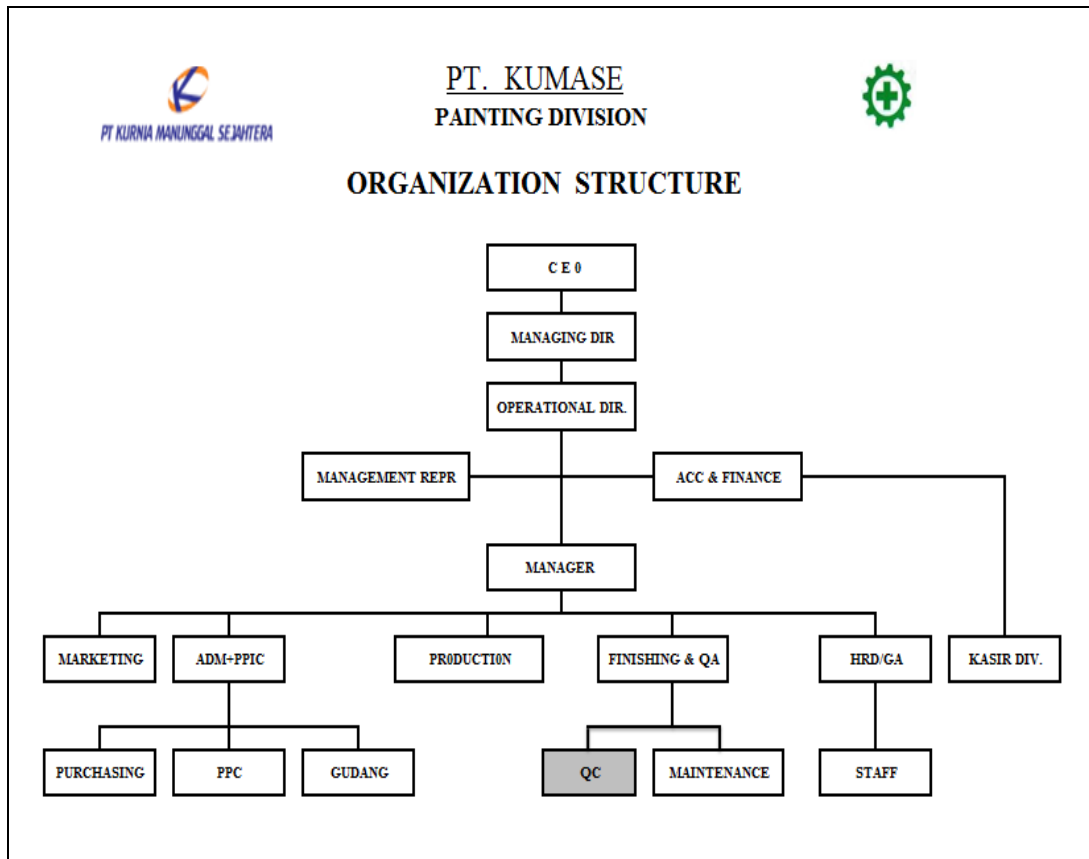
Tujuan perusahaan yang bergerak khusus pada divisi *stamping*, divisi *painting* dan divisi konstruksi baja adalah memenuhi kebutuhan konsumen akan komponen-komponen yang dibutuhkan. Karena komponen-komponen tersebut berperan sangat penting dalam pembuatan suatu produk. Adapun komponen yang dipesan seperti *accumulator* dan *stay mirror*.

Kedua komponen tersebut dipasok untuk beberapa perusahaan manufaktur, yaitu PT Panasonic, PT Sanyo dan PT Astra Honda Motors. Ketiga perusahaan besar ini memberikan standar kualitasnya masing-masing untuk setiap komponen yang akan diproduksi di PT Kurnia Manunggal Sejahtera.

Untuk *accumulator* bahan baku dikirim langsung oleh PT Panasonic. Dan *stay mirror* juga dikirim langsung oleh PT Astra Honda Motors untuk kemudian dikerjakan sesuai dengan standar yang berlaku di perusahaan tersebut.

4.5 Struktur Organisasi Perusahaan

Struktur organisasi adalah suatu susunan dalam unit kerja yang ada di sebuah organisasi. Fungsi dari organisasi yaitu memberikan kejelasan tanggung jawab, kejelasan kedudukan/jabatan, kejelasan mengenai hubungan antar jabatan, serta kejelasan uraian pekerjaan. Dengan adanya struktur organisasi tersebut, perusahaan dapat mengetahui dan menganalisa pelaksanaan tugas dan tanggung jawab antar bagian/divisi. Struktur organisasi PT Kurnia Manunggal Sejahtera dapat dilihat pada gambar berikut.



Gambar IV.1 Struktur Organisasi
(Sumber: PT Kurnia Manunggal Sejahtera, 2016)

Gambaran jelas mengenai struktur organisasi divisi *Painting* pada PT Kurnia Manunggal Sejahtera, diuraikan beberapa posisi dan jabatan masing-masing tugas dan tanggung jawabnya di dalam perusahaan sebagai berikut:

1. CEO (*Chief Executive Officer*)

Adapun tugas dan wewenang yang diberikan kepada CEO adalah sebagai berikut:

- a. Mengkoordinasikan dan mengendalikan kegiatan pengadaan dan peralatan.
- b. Mengkoordinasikan dan mengendalikan kegiatan dibidang administrasi keuangan, kepegawaian dan kesekretariatan.
- c. Bertindak sebagai perwakilan organisasi dalam hubungan dengan dunia luar.

d. Mengambil keputusan pada situasi tertentu yang dianggap perlu.

2. *Managing Director*

Adapun tugas dan wewenang yang dapat dilakukan oleh *Managing Director* adalah sebagai berikut:

- a. Membuat rencana pengembangan dan usaha perusahaan dalam jangka pendek & panjang.
- b. Memberikan laporan pertanggungjawaban kepada rapat umum pemegang saham.
- c. Bertanggung jawab penuh atas tugasnya untuk kepentingan perseroan dalam mencapai maksud dan tujuannya.
- d. Mengawasi serta mengurus kekayaan perusahaan.
- e. Menandatangani permintaan pengeluaran kas yang jumlahnya besar dan sifatnya penting.
- f. Menetapkan pencapaian tujuan untuk jangka panjang.
- g. Mengambil keputusan dan strategi bagi perusahaan.

3. *Operational Director*

Adapun tugas dan wewenang yang dapat dilakukan oleh *Operational Director* adalah sebagai berikut:

- a. Membantu tugas-tugas direktur.
- b. Mewakili direktur apabila berhalangan hadir.
- c. Merencanakan pengembangan dan usaha perusahaan dalam jangka pendek dan jangka panjang.
- d. Bertanggung jawab penuh atas tugasnya untuk kepentingan perseroan dalam mencapai maksud dan tujuannya.
- e. Mengawasi kinerja perusahaan secara keseluruhan.
- f. Mengawasi dan mengurus harta kekayaan perusahaan.
- g. Menetapkan pencapaian tujuan untuk jangka panjang.
- h. Mengambil keputusan dan strategi bagi perusahaan.

4. *Management Representative*

Tugas dan wewenang yang dapat dilakukan oleh *Management*

Representative menganalisa dan memilih *customer* yang datang membawa jasa yang menguntungkan.

5. *Acc & Finance*

Adapun tugas dan wewenang yang dapat dilakukan oleh *Acc & Finance* adalah sebagai berikut:

- a. Melakukan analisa keuangan termasuk masalah pajak.
- b. Melakukan verifikasi ulang atas semua bukti kas, penerimaan dan pengeluaran kas.
- c. Melakukan verifikasi atas semua buku penjualan tunai, faktur penjualan dan nota pembelian.
- d. Bertanggung jawab pada uraian piutang, utang, penagihan dan pembukuan.
- e. Memeriksa kebenaran penulisan, perhitungan kwitansi dan surat jalan.
- f. Memeriksa limit piutang perusahaan.
- g. Menandatangani seluruh dokumen yang berkaitan dengan administrasi perusahaan.
- h. Membuat evaluasi kegiatan perusahaan bidang keuangan.
- i. Menetapkan metode-metode yang digunakan dalam pencatatan akuntansi.

6. *Manager*

Adapun tugas dan wewenang yang dapat dilakukan oleh *Manager* adalah sebagai berikut:

- a. Mengontrol dan mengevaluasi implementasi strategi agar memperoleh masukan strategi sebagai usulan untuk kebijakan tahun berikutnya.
- b. Memonitor pelaksanaannya kebijakan dan strategi perusahaan serta memastikan kelancaran pelaksanaannya agar dapat berjalan secara maksimal dan tepat.
- c. Mengarahkan fungsi setiap departemen dalam menjalankan strategi perusahaan.
- d. Melakukan evaluasi dan menganalisa hasil implementasi strategi perusahaan.

- e. Memberikan pengarahan terhadap fungsi setiap departemen.

7. *Marketing*

Adapun tugas dan wewenang yang dapat dilakukan oleh *Marketing* adalah sebagai berikut:

- a. Merencanakan dan merumuskan kebijakan strategis yang menyangkut pemasaran.
- b. Mengawasi dan mengarahkan proses-proses diseluruh divisi.
- c. Menetapkan pedoman harga barang dan jasa.
- d. Menetapkan dan mengevaluasi upaya strategis dan kebijakan pemasaran serta pengadaan barang dan jasa.
- e. Menetapkan sistem pengendalian hasil produksi serta bahan baku dan pelengkap.

8. *Administrasi+PPIC*

Adapun tugas dan wewenang yang dapat dilakukan oleh *Administrasi+PPIC* adalah sebagai berikut:

- a. Menandatangani segala urusan pembelian, PPC dan gudang.
- b. Bertanggung jawab atas segala urusan yang berhubungan dengan pembelian, PPC dan gudang.
- c. Memberikan kebijaksanaan mengenai hal-hal yang berkaitan dengan pembelian, PPC dan gudang.

9. *Production*

Adapun tugas dan wewenang yang dapat dilakukan oleh bagian *Production* adalah sebagai berikut:

- a. Mengatur waktu pekerjaan agar sesuai dengan waktu pengiriman.
- b. Mengatur posisi pengawas kualitas di masing-masing titik pekerjaan.
- c. Memberikan harga produksi untuk kepentingan *marketing* dalam membuat harga jual.
- d. Memproduksi produk sesuai dengan rencana jadwal produksi.
- e. Mengkoordinasikan para pekerja dalam proses produksi.

10. *Finishing & QA*

Adapun tugas dan wewenang yang dapat dilakukan oleh *Finishing & QA* adalah sebagai berikut:

- a. Mengevaluasi pelaksanaan pengendalian internal dan sistem manajemen resiko sesuai kebijakan perusahaan.
- b. Melakukan pemeriksaan secara mendetail pada produk.
- c. Membuat laporan hasil pemeriksaan.
- d. Memberikan instruksi untuk mengulang kembali pekerjaan karena belum sesuai dengan standar kualitas perusahaan.

11. HRD/GA

Adapun tugas dan wewenang yang dapat dilakukan oleh HRD/GA adalah sebagai berikut:

- a. Membuat perencanaan pegawai sesuai kebutuhan dari setiap departemen.
- b. Bertanggung jawab dalam memilih dan mendapatkan pegawai yang sesuai dengan kebutuhan perusahaan.
- c. Memberikan pelatihan kepada pegawai agar mempunyai motivasi kerja dan menemukan solusi untuk setiap persoalan yang dihadapi oleh pegawai perusahaan.
- d. Menilai dan mengukur kinerja pegawai.
- e. Memberikan sanksi kepada pegawai yang melanggar aturan perusahaan.
- f. Memberikan saran-saran dan pertimbangan kepada direktur.

12. Kasir Divisi

Adapun tugas dan wewenang yang dapat dilakukan oleh Kasir Divisi adalah sebagai berikut:

- a. Menerima daftar penerimaan kas, bukti kas masuk, dan kas/cek, kasir membuat bukti setor bank dan menyetorkan kas tersebut ke bank.
- b. Kasir mengarsipkan daftar penerimaan kas dan bukti kas masukurut tanggal.
- c. Memberikan tanggal dan cap lunas pada tiap bukti penerimaan dan pengeluaran kas.

- d. Berwenang dan bertanggung jawab dalam menerima dan mengeluarkan uang kas perusahaan.

13. *Purchasing*

Adapun tugas dan wewenang yang dapat dilakukan oleh bagian *Purchasing* adalah sebagai berikut:

- a. Menentukan pemasok yang dipilih dalam pengadaan barang.
- b. Mengeluarkan order pembelian kepada pemasok yang dipilih.
- c. Membuat pemesanan pembelian dan memesan barang kepada pemasok.
- d. Membuat pembelian yang berfungsi untuk menambah *stock* barang.

14. PPC

Adapun tugas bagian PPC adalah sebagai berikut:

- a. Memberikan penjelasan setiap pesanan barang ke bagian produksi.
- b. Menindak lanjuti perkembangan setiap minggu ke bagian produksi.

15. Gudang

Adapun tugas dan wewenang yang dapat dilakukan bagian Gudang adalah sebagai berikut:

- a. Mempersiapkan barang yang akan dikirim.
- b. Bertanggung jawab menyerahkan surat order penjualan dan barangnya ke bagian pengiriman.
- c. Mengajukan permintaan pembelian sesuai dengan posisi persediaan.
- d. Menginformasikan ke bagian *marketing* jika ada masalah diproses produksi.
- e. Menyusun jadwal proses produksi pada waktu *routing* dan *quantity* yang tepat.
- f. Mengontrol *stock* barang yang ada di gudang.

16. *Quality Control*

Adapun tugas bagian *Quality Control* adalah sebagai berikut:

- a. Memantau perkembangan semua produk yang diproduksi oleh perusahaan.
- b. Bertanggung jawab pada kualitas produk.

- c. Membuat laporan kualitas pada produk.

17. *Maintenance*

Adapun tugas dan wewenang yang dapat dilakukan oleh bagian *Maintenance* adalah sebagai berikut:

- a. Melakukan perawatan secara berkala pada mesin-mesin perusahaan.
- b. Melakukan perbaikan pada mesin-mesin yang rusak.

18. *Staff*

Adapun tugas *Staff* adalah sebagai berikut:

- a. Melaksanakan pengelolaan tugas administrasi kepegawaian.

4.6 Produk

PT Kurnia Manunggal Sejahtera memproduksi berbagai produk. Pada divisi *painting*, produk yang dihasilkan yaitu *accumulator*, *stay mirror* besar, *stay mirror* kecil blok mesin vario, blok mesin beat, rangka *air conditioner*. Pada laporan ini pengamatan tertuju pada *accumulator*. Produk *accumulator* tersebut diproduksi berdasarkan pesanan *customer*. Untuk proses produksi yang dilakukan melalui tahapan yang sama, perbedaan hanya ada di sablon dan warna. Berdasarkan sablon dan warna, *accumulator* diklasifikasi sebagai berikut:

1. *Accumulator white*, didistribusikan untuk PT Sanyo.
2. *Accumulator blue* lokal dan *blue export*, didistribusikan untuk PT Panasonic.

4.7 *Accumulator*

Accumulator merupakan alat penyimpan energi listrik sehingga dapat dipergunakan pada waktu dan tempat yang lain. *Accumulator* sebuah tabung hidrolis yang berfungsi untuk menyedot air. *Accumulator* dapat menyimpan [energi](#) umumnya energi [listrik](#) dalam bentuk energi kimia. Contoh-contoh akumulator adalah [baterai](#) dan [kapasitor](#).



Gambar IV.2 Accumulator White
(Sumber: PT Kurnia Manunggal Sejahtera, 2016)

4.8 Komponen Accumulator dan Fungsinya

Komponen-komponen *accumulator* terdiri atas beberapa bagian yang masing-masing memiliki fungsi tertentu, antara lain sebagai berikut:

Tabel IV.1 Komponen *Accumulator* dan Fungsinya

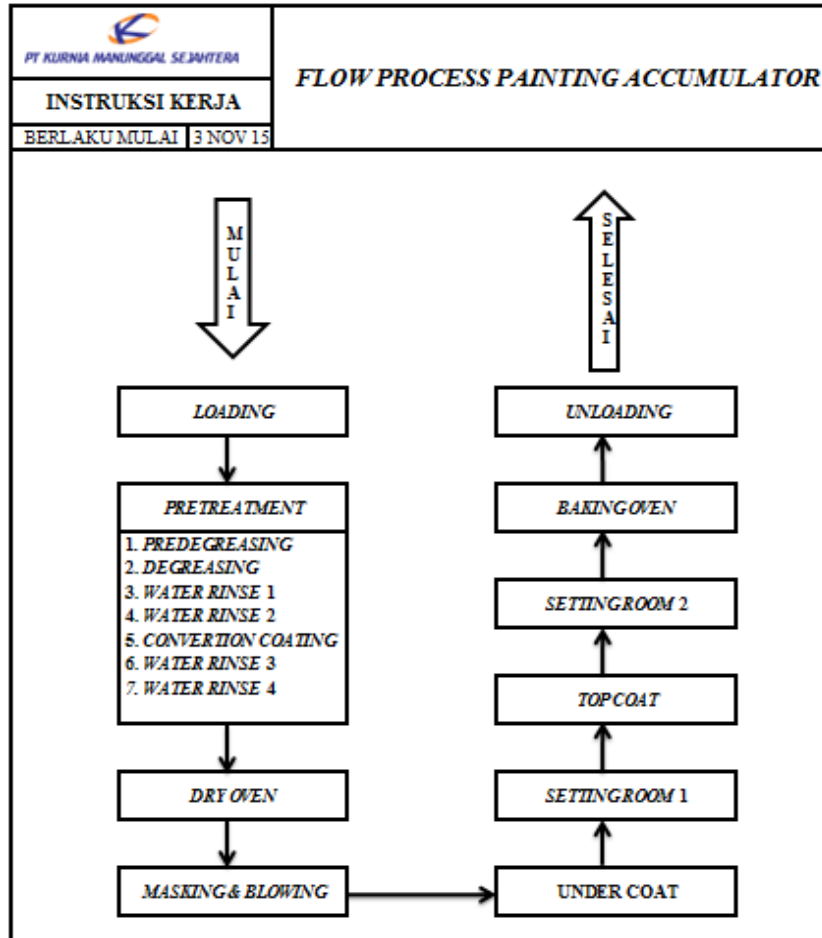
No	Komponen	Fungsi
1	1.1.1.1.1.1.1.1	<ul style="list-style-type: none"> • Mencegah ketidakstabilan <i>supply</i> udara ke <i>actuator</i>. • Menstabilkan kerja kompresor agar tidak terlalu sering mematikan dan menyalakannya.

2	1.1.1.1.1.1.1.2	<ul style="list-style-type: none"> • Pemisah yang terbuat dari bahan baku karet sintetis • Menampung nitrogen
3	1.1.1.1.1.1.1.3	<ul style="list-style-type: none"> • Menutup permukaan bawah pada <i>bladder</i>

(Sumber: PT Kurnia Manunggal Sejahtera, 2016)

4.9 Proses Produksi dan *Painting Accumulator*

Proses produksi *accumulator* pada PT Kurnia Manunggal Sejahtera terdiri dari *Check Raw Material, Blank, Draw 1, Draw 2, Draw 3, Trimming, Burring, Tapping, Treatment, Knurling, Painting, Sablon, Isi Nitrogen, dan Packaging*. Semua proses pembuatan *accumulator* dilakukan dengan penuh ketaatan terhadap standar mutu yang telah ditetapkan. Setiap lini produksi bertanggung jawab terhadap kualitas dari barang yang dihasilkan sehingga dapat meningkatkan kualitas dari *accumulator* itu sendiri. Pada divisi *painting* alur proses *accumulator*, yaitu: *Loading, Pretreatment, Dry Oven, Masking & Blowing, Under Coat, Setting Room 1, Top Coat, Setting Room 2, Baking Oven* dan *Unloading*. Berikut merupakan gambaran umum mengenai alur proses *painting accumulator*:

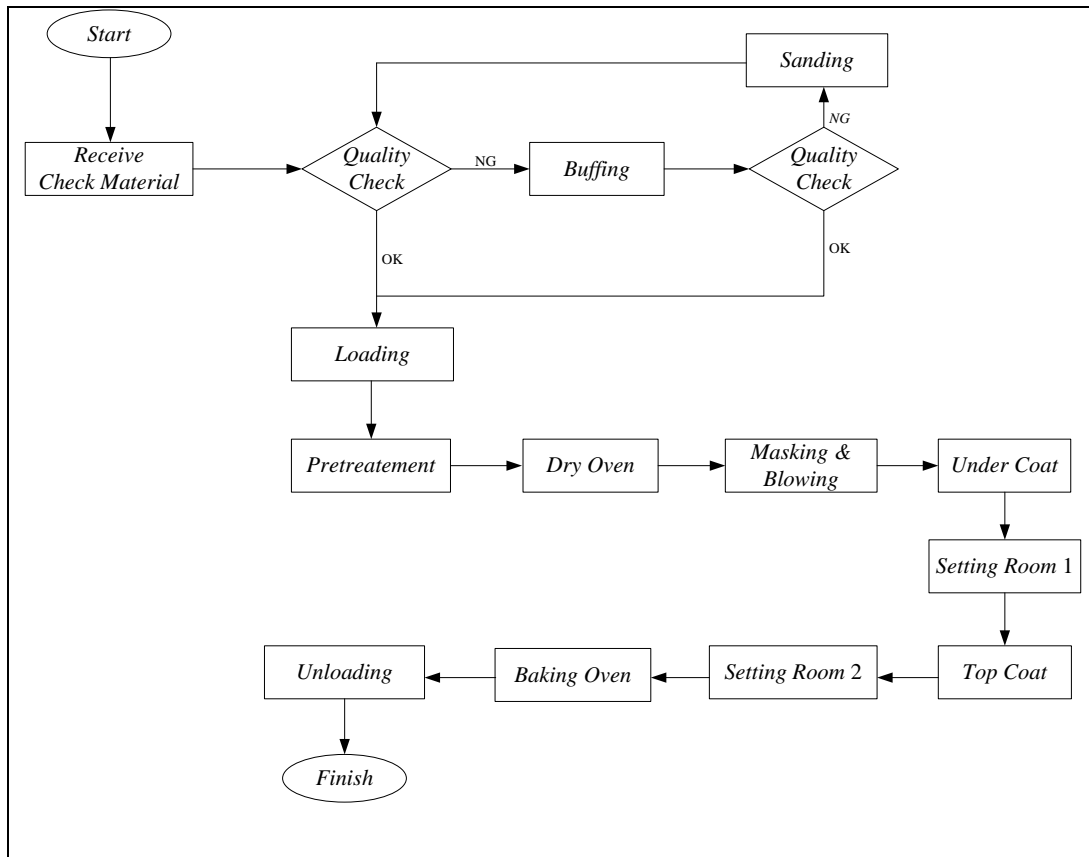


Gambar IV.3 *Flow Process Painting Accumulator*
 (Sumber: PT Kurnia Manunggal Sejahtera, 2016)

4.10 Proses Pengendalian Kualitas *Accumulator*

Proses pengendalian sangat diperlukan untuk proses produksi terutama pada proses pengawasan kualitas produk. Berikut ini merupakan gambaran umum mengenai alur proses pengecatan pada divisi *Painting* (lihat Gambar IV.4).





Gambar IV.4 Flowchart Proses Pengendalian Kualitas Accumulator
(Sumber: Hasil Analisis, 2016)

Proses tersebut dapat dijelaskan sebagai berikut:

1. *Receive Check Material*

Proses pengecekan *part* dari bagian produksi ke proses pengecatan.

2. *Quality check*

Proses pengecekan sebelum dilakukan proses *loading*, pengecekan sesuai standar yang sudah ditetapkan antara lain *part* tidak boleh berminyak, kotor, dekok dll.

3. *Buffing*

Proses *repair part not good* (NG) karena kotor dan cat tipis dengan cara *part* dipasang pada *polisher* dan *buffing* untuk memoles permukaan cat. Urutan prosesnya sebagai berikut:

- a. Ambil *part* kategori *buffing* yang dinyatakan NG oleh *checkman*.

- b. Amplas *part* pada area *reject* dengan amplas 2000.
- c. Oleskan pasta *buffing* pada area *reject*.
- d. Bersihkan *part* setelah *buffing* dengan kain lap.
- e. Periksa hasil *buffing*.

4. *Sanding*

Proses *repair part* NG berat yang tidak mampu *direpair* dengan cara *dibuffing* contohnya pada *reject* cat meler dan baret, pada proses ini menggunakan *sandpaper* dengan kekasaran 500-2000. Adapun prosesnya sebagai berikut:

- a. Ambil *part* kategori *sanding* yang dinyatakan NG oleh *checkman*.
- b. Amplas *part* dengan amplas sesuai urutan dari kecil ke besar (500, 1000, 1500) pada area *reject*.
- c. Bersihkan *part* setelah *sanding* dengan menggunakan kain lap.
- d. Periksa kembali hasil *sanding* dan pastikan bersih dan halus.
- e. Kirim kembali *part* setelah *disanding* ke proses *loading*.

5. *Loading*

Proses pemasangan *part* ke *subjig*, sebelum *part* dipasang ke *subjig part* tersebut harus sudah lolos dahulu dari *quality check material*. *Subjig* digantung pada *hanger* yang bergerak berdasarkan konveyor. Setiap *part* akan di-*loading* sesuai dengan *subjig* yang terpasang pada *hanger*. Pada proses ini yang harus diperhatikan adalah *part* harus terpasang pada posisi yang benar dan tidak jatuh pada proses pengecatan.

6. *Pretreatment*

Proses pendahuluan terhadap bahan sebelum dilakukan pengecatan. Proses *pretreatment* yang dilakukan sebelum pengecatan, sangat berpengaruh terhadap kualitas pengecatan, dan hal ini berkaitan dengan proses pengerjaan sebelum *part* di-*loading* untuk proses pengecatan.

a. *Predegreasing*

Merupakan proses persiapan permukaan aluminium sebelum *degreasing*.

Tujuan *predegreasing* sebagai berikut:

1. Menghilangkan sebagian besar kontaminasi yang berasal dari senyawa organik.
2. Menghilangkan pengotor zat organik lain untuk memudahkan dalam proses *degreasing*.

b. *Degreasing*

Adalah proses pencucian *part* dengan menggunakan larutan Alkali. Tujuannya sebagai berikut :

1. Membersihkan kotoran yang menempel pada *part* (senyawa organik atau anorganik).
2. Mengontrol permukaan aluminium untuk mendapatkan susunan kristal yang baik.

c. *Water Rinse 1*

Sebagai pembilas yang pertama agar permukaan *part* bersih dari bahan kimia yang menempel akibat dari proses sebelumnya, sehingga tidak terjadi kontaminasi antar larutan kimia dan bertujuan untuk membilas kelebihan pembersih yang menempel pada benda kerja.

d. *Water Rinse 2*

Sebagai pembilas yang kedua agar permukaan *part* bersih dari bahan kimia dan cara kerjanya sama dengan *water rinse 1*.

e. *Conversion Coating*

Merupakan proses pelapisan *part* secara kimiawi. Tahap *conversion coating* bertujuan untuk menambah daya rekat cat pada material.

f. *Water Rinse 3*

Sebagai pembilas yang ketiga agar permukaan *part* bersih dari bahan kimia dan cara kerjanya sama dengan *water rinse 1* dan 2.

g. *Water Rinse 4*

Sebagai pembilas yang keempat yang bertujuan untuk menyempurnakan pembersihan.

7. *Dry Oven*

Untuk menghilangkan sisa-sisa air yang masih melekat pada *part* setelah mengalami *pretreatment*. Proses *Dry Oven part steel* biasanya beroperasi minimal pada suhu 100°C (agar terjadi penguapan air di lapisan *phosphate*). Misalnya untuk *dry oven* yang menggunakan sistem *conveyor* beroperasi pada suhu 140°C-160°C. Apabila suhu lebih dari 200°C, maka kualitas lapisan *phosphating* sebagai hasil pengeringan terjadi pecah-pecah, sehingga mempengaruhi daya rekat cat pada *part*.

8. *Masking dan Blowing*

Proses penyemprotan anti *static* ke *part upper cover* untuk menghilangkan debu yang masih menempel.

9. *Under Coat*

Pengecatan awal sebagai cat dasar untuk cat tipe 2 komponen bertujuan untuk mendapatkan warna yang diharapkan.

10. *Setting Room 1*

Merupakan ruangan yang dibuat untuk proses penguapan cat secara alami sebelum dilakukan proses *spray top coat*.

11. *Top Coat*

Pengecatan akhir untuk mendapatkan hasil pengecatan yang diharapkan seperti *glossy, thickness*, maupun *color matching*.

12. *Setting Room 2*

Merupakan ruangan yang memiliki fungsi sama seperti setting room 1, yaitu untuk proses penguapan cat secara alami sebelum ke proses *baking oven*. Pemberian waktu jeda *part* yang telah diproses *top coat* agar *solvent* mengering secara alami sebelum proses *baking oven*.

13. *Baking Oven*

Merupakan proses pemberian panas pada *part upper cover* agar penguapan cat terjadi lebih cepat.

14. *Unloading*

Proses penurunan *part/komponen* dari *hanger*.

4.11 Jenis-Jenis Kerusakan *Accumulator*

Kerusakan *accumulator* merupakan penilaian akan mutu *accumulator* yang akan di produksi, salah satunya adalah kerusakan. Kerusakan menjadikan landasan atau pelajaran akan kesalahan yang mungkin terjadi pada proses produksi *accumulator* yang dapat menjadi alat untuk mengatasi permasalahan yang terjadi sehingga memudahkan untuk meminimalisasi kerugian. Kerusakan yang dialami *accumulator* pada PT Kurnia Manunggal Sejahtera dapat diklasifikasikan berdasarkan jenis kerusakannya, jenis-jenis tersebut yaitu:

Tabel IV.2 Jenis Kerusakan *Accumulator*

No.	Jenis <i>Reject</i>	Keterangan
1.	Tipis	Warna cat tidak sama pada suatu permukaan cat.
2.	Lecet	Material atau <i>part</i> yang di cat jelek.
3.	Meler	Permukaan cat yang tidak rapih yang membuat permukaan tidak bagus karena ada gumpalan-gumpalan cat.
4.	Kotor	Adanya benda asing yang terdapat di permukaan cat dikarenakan pada proses <i>dry oven</i> tidak semuanya kering.
5.	Serabut	Adanya bekas <i>hand glove</i> berbahan kain yang menempel dimaterial yang sudah di cat.
6.	Bintik	Material pada proses pencampuran kimia tidak melekat sempurna dipermukaan benda kerja.

(Sumber: PT Kurnia Manunggal Sejahtera, 2016)


4.12 Dokumen Pengendalian Kualitas

Sistem informasi pengendalian kualitas *accumulator* pada PT Kurnia Manunggal Sejahtera telah dilakukan secara terkomputerisasi dengan menggunakan *Microsoft Excel* untuk mengolah data kualitas tersebut. Dokumen

yang terlibat dalam sistem informasi pengendalian kualitas *accumulator* adalah sebagai berikut:

1. Laporan *Loading-Unloading*

Adalah laporan harian yang digunakan dalam proses *loading-unloading*. Laporan *loading-unloading* diisi oleh operator dan di periksa oleh *foreman*. Contoh laporan *loading-unloading* dapat dilihat pada gambar IV.5.


PT KURNIA MANUNGGAL SEJAHTERA

LAPORAN LOADING - UNLOADING

Tanggal : Kamis, 3-10-2016

No	JAM	No.Lot	NAMA PART	WARNA	PROSES			JUMLAH	JAM	UNLOADING							JUMLAH	
					F	R1	R2			NG							NG	OK
										Tipis	Lecet	Meler	Kotor	Serabut	Bintik			
1	8:15	12	Accumulator	White	✓			40	11:20			3					3	37
2		13	Accumulator	White	✓			40				16					16	24
3		14	Accumulator	White	✓			40				5	1				6	34
4		15	Accumulator	White	✓			40				5					5	35
5		16	Accumulator	White	✓			40	13:10	1		5					6	34
6		17	Accumulator	White	✓			40			1	8					9	31
7		18	Accumulator	White	✓			40										40
8		19	Accumulator	White	✓			40										40
9		20	Accumulator	White	✓			40										40
10		21	Accumulator	White	✓			40										40
11		22	Accumulator	White	✓			40				5					5	35
12	9:24	23	Accumulator	White	✓			40	13:55									40
13		24	Accumulator	White	✓			40				4					4	36
14		25	Accumulator	White	✓			40				3	1				4	36
15		26	Accumulator	White	✓			40										40
16		27	Accumulator	White	✓		✓	40				8					8	32
17		28	Accumulator	White	✓			40										40
18		29	Accumulator	White	✓			32				6					6	26
19		1	Accumulator	White	✓			40										40
20		2	Accumulator	White	✓			40										40
								792			1	1	68	2	0	0	72	720

No	KETERANGAN JENIS DOWNTIME	WAKTU (MENT)

Note: F = Material Fresh
R1 = Material Repair 1x
R2 = Material Repair 2x

OPR LOADING	OPR UNLOADING	DIPERIKSA

Gambar IV.5 Laporan *Loading-Unloading*
(Sumber: Arsip PT Kurnia Manunggal Sejahtera, 2016)

2. *Quality Monthly Report*

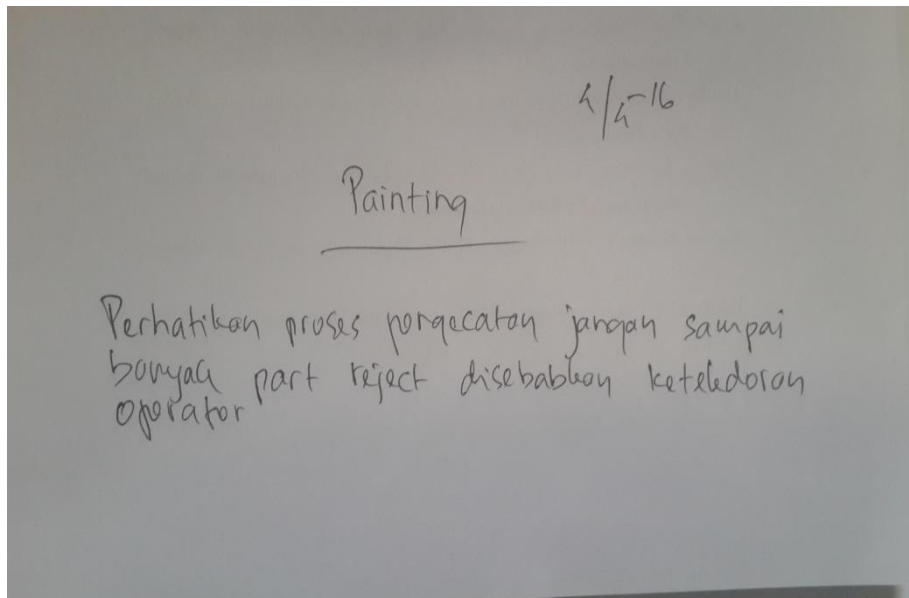
Adalah laporan bulanan yang berisikan dari laporan *loading-unloading* yang dipindahkan ke dalam file *Microsoft Office*. *Quality monthly report* dibuat oleh bagian *Quality Control Office* dan diterima oleh *Leader Produksi* dan *Manajer Produksi* dapat dilihat pada gambar IV.6.

PT. KURNIA MANUNGGAL SEJAHTERA		QUALITY MONTHLY REPORT Jan-16																														DIV. PAINTING				
TYPE	REJECTION	OK	PART NAME : ACCUMULATOR																														PROCESS : PAINTING			
			DATE																														TOTAL	TOTAL/HS		
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30			31	
ACCUMULATOR WHITE LOCAL	PRODUKSI	OK																																		
	TIRIS	HS																																		
	LEGAT	HS																																		
	WELER	HS																																		
	KOTOR	HS																																		
	BERAKUT	HS																																		
	BINTIK	HS																																		
																																	TOTAL	TOTAL/HS		
ACCUMULATOR BLUE LOCAL	PRODUKSI	OK																																		
	TIRIS	HS																																		
	LEGAT	HS																																		
	WELER	HS																																		
	KOTOR	HS																																		
	BERAKUT	HS																																		
	BINTIK	HS																																		
																																	TOTAL	TOTAL/HS		
ACCUMULATOR BLUE EKSPORT	PRODUKSI	OK																																		
	TIRIS	HS																																		
	LEGAT	HS																																		
	WELER	HS																																		
	KOTOR	HS																																		
	BERAKUT	HS																																		
	BINTIK	HS																																		
																																	TOTAL	TOTAL/HS		

Gambar IV.6 Quality Monthly Report
(Sumber: Arsip PT Kurnia Manunggal Sejahtera, 2016)

3. Note Saran

Adalah *note* saran yang dibuat dari melihat *quality monthly report*. *Note* saran dibuat jika target *rejection max* 3% dan dibuat oleh *Leader* Produksi dan diserahkan ke bagian Produksi dapat dilihat pada gambar IV.7.



Gambar IV.7 Note Saran
(Sumber: Arsip PT Kurnia Manunggal Sejahtera, 2016)

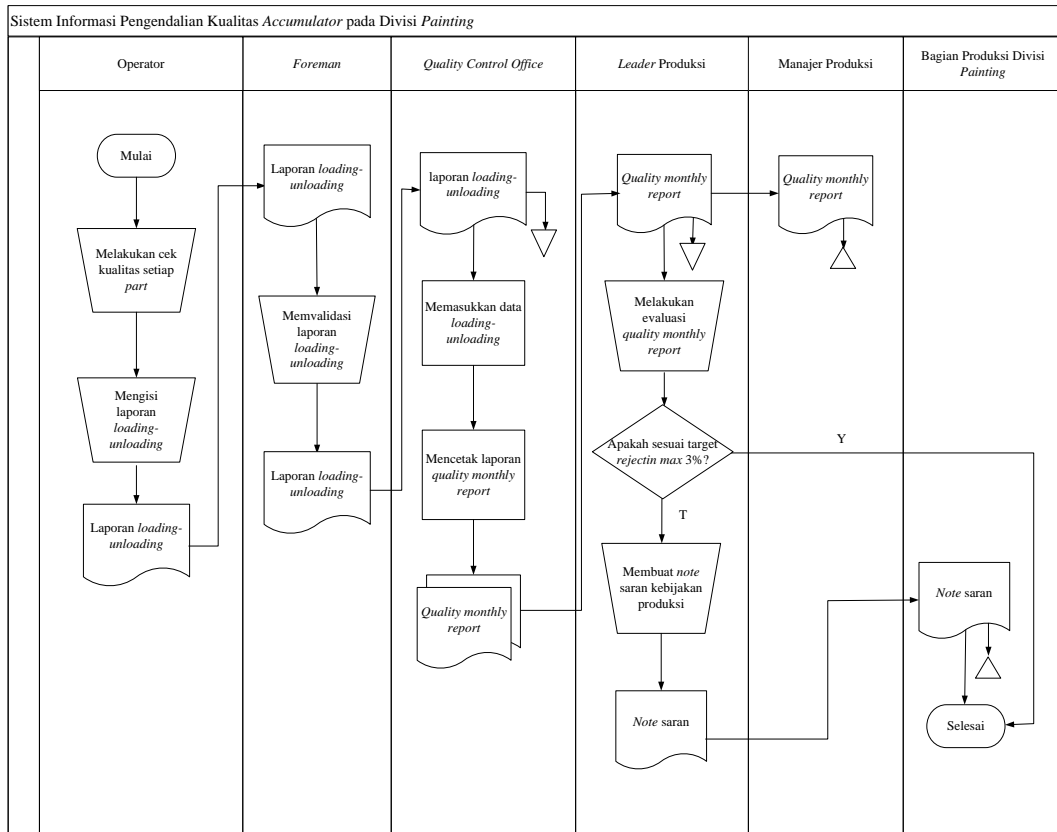
4.13 Prosedur Pengendalian Kualitas *Accumulator*

Pengendalian kualitas *accumulator* pada PT Kurnia Manunggal Sejahtera memiliki prosedur yang ditangani oleh beberapa *user* seperti Operator, *Foreman*, *Quality Control Office*, *Leader* Produksi, Manajer Produksi dan Bagian Produksi Divisi *Painting*. Adapun prosedurnya terdiri dari beberapa tahapan, yaitu:

1. Proses dimulai dari Operator yang bertugas mengecek kualitas setiap *part* hasil pengecatan dengan datang langsung ke proses *loading* dan *unloading*. Apabila ditemukan cacat atau tidak cacat pada *part* maka *part* tersebut dicatat oleh Operator di laporan *loading-unloading*. Berdasarkan sortiran Operator, *part* tersebut ditandai dengan pensil khusus guna mempermudah proses perbaikan *part* tersebut nantinya.
2. Kemudian laporan *loading-unloading* diserahkan ke *Foreman* untuk divalidasi.
3. Setelah laporan *loading-unloading* divalidasi, bagian *Quality Control Office* membuat *quality monthly report* menggunakan *Microsoft Excel* sebagai tempat penyimpanan data. Kemudian *quality monthly report* dicetak secara rangkap oleh bagian *Quality Control Office* dan diserahkan kepada *Leader* Produksi dan Manajer Produksi masing-masing satu laporan.
4. Setelah menerima *quality monthly report*, *Leader* Produksi melakukan evaluasi terhadap *quality monthly report*. Apakah hasil kualitas melebihi target *rejection max 3%*.
 - Jika berdasarkan hasil evaluasi didapat bahwa hasil kualitas dibawah target *rejection* yang ditentukan maka dapat dinyatakan penerapan proses produksi sudah berjalan baik.
 - Apabila berdasarkan evaluasi didapat fakta bahwa *part rejection* melebihi 3% dari produksi maka *Leader* Produksi akan membuat *note* untuk memperbaiki kualitas produksinya kepada bagian Produksi divisi *Painting*.

5. Dari saran *Leader* Produksi, bagian Produksi divisi *Painting* menerima *note* yang berisikan saran kebijakan produksi dan melaksanakan saran tersebut pada kegiatan produksi selanjutnya, agar kualitas produksinya menghasilkan *part* sesuai dengan standar yang telah ditetapkan.

Adapun aliran data sistem informasi pengendalian kualitas *accumulator* bagian *Quality Control* dapat dilihat pada *flowchart* pada Gambar IV.8.



Gambar IV.8 Aliran Dokumen Sistem yang Berjalan
(Sumber: Pengolahan Data, 2016)

4.14 Pemodelan Sistem Pengendalian Kualitas dengan *Use Case Diagram*

Use case diagram sistem informasi pengendalian kualitas *accumulator* yang sedang berjalan pada divisi *Painting* di PT Kurnia Manunggal Sejahtera adalah sebagai berikut:

- a. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi pengendalian kualitas *accumulator* yang sedang berjalan pada divisi *Painting* di PT Kurnia Manunggal Sejahtera dapat dilihat pada tabel IV.3 berikut:

Tabel IV.3 Definisi Aktor Sistem Informasi Pengendalian Kualitas

No.	Aktor	Definisi
1.	Operator	Operator bertugas melakukan pengecekan kualitas, mengisi laporan <i>loading-unloading</i> .

Tabel IV.3 Definisi Aktor Sistem Informasi Pengendalian Kualitas (lanjutan)

No.	Aktor	Definisi
2.	<i>Foreman</i>	<i>Foreman</i> bertugas memvalidasi laporan <i>loading-unloading</i> .
3.	<i>Quality Control Office</i>	<i>Quality Control Office</i> bertugas memasukkan data <i>loading-unloading</i> dan membuat dan mencetak <i>quality monthly report</i> .
4.	<i>Leader Produksi</i>	<i>Leader Produksi</i> bertugas untuk melakukan evaluasi terhadap <i>quality monthly report</i> , membuat <i>note</i> saran kebijakan produksi dan melakukan koordinasi dengan bagian Produksi divisi <i>Painting</i> .
5.	Manajer Produksi	Manajer Produksi bertugas sebagai penerima <i>quality monthly report accumulator</i> divisi <i>Painting</i> .

6.	Bagian Produksi Divisi <i>Painting</i>	Bagian Produksi Divisi <i>Painting</i> bertugas untuk menerima <i>note</i> saran.
----	---	---

(Sumber: Pengolahan Data, 2016)

b. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* sistem informasi pengendalian kualitas *accumulator* pada divisi *Painting* di PT Kurnia Manunggal Sejahtera dapat dilihat pada tabel IV.4 berikut:

Tabel IV.4 Definisi *Use Case* Sistem Informasi Pengendalian Kualitas

No.	<i>Use Case</i>	Definisi
1.	Melakukan cek kualitas setiap <i>part</i>	Proses pengecekan terhadap kualitas setiap <i>part</i> yang diperiksa oleh operator.
2.	Mengisi laporan <i>loading-unloading</i>	Proses mengisi laporan <i>loading-unloading</i> .

Tabel IV.4 Definisi *Use Case* Sistem Informasi Pengendalian Kualitas

(lanjutan)

No.	<i>Use Case</i>	Definisi
3.	Memvalidasi laporan <i>loading-unloading</i>	Proses memvalidasi laporan <i>loading-unloading</i> .
4.	Memasukkan data <i>loading-unloading</i>	Proses penginputan data dari laporan <i>loading-unloading</i> ke dalam file <i>Microsoft Excel</i> .
5.	Mencetak <i>quality monthly report</i>	Proses pembuatan dan pencetakan dari data <i>loading-unloading</i> yang di simpan di dalam file <i>Microsoft Excel</i> menjadi <i>quality monthly report</i> divisi <i>Painting</i> .
6.	Menerima <i>quality monthly report</i>	Proses penerimaan <i>quality monthly report</i> divisi <i>Painting</i> .

7.	Melakukan evaluasi terhadap <i>quality monthly report</i>	Proses mengevaluasi apakah kualitas sudah sesuai target yang ditetapkan.
8.	Membuat <i>note</i> saran	Proses membuat <i>note</i> untuk perbaikan kualitas hasil produksi selanjutnya.
9.	Menerima <i>note</i> saran	Proses menerima <i>note</i> saran.

(Sumber: Pengolahan Data, 2016)

c. *Use case diagram*

Use case diagram sistem informasi pengendalian kualitas *accumulator* yang sedang berjalan pada divisi *Painting* terdiri dari 5 aktor sebagaimana digambarkan pada Gambar IV.9 di bawah ini:



Gambar IV.9 Use Case Diagram Sistem Informasi Pengendalian Kualitas
(Sumber: Pengolahan Data, 2016)

Permasalahan dengan sistem yang sudah ada adalah proses pembuatan *quality monthly report accumulator* yang belum dilakukan secara otomatis dan belum adanya suatu aplikasi pengendalian kualitas yang membantu *user* lebih cepat melakukan proses penginputan laporan *loading-unloading* pada *part accumulator*. Untuk mengatasi masalah tersebut maka akan dibuatkan kebutuhan sistem untuk aplikasi sistem informasi pengendalian kualitas *accumulator* pada bagian *Quality Control* divisi *Painting*.

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, dibutuhkan perancangan aplikasi sistem informasi pengendalian kualitas *accumulator* sebagai sarana untuk memberikan informasi mengenai *quality monthly report accumulator* bagian *Quality Control* pada divisi *Painting*. Berikut adalah daftar kebutuhan sistem untuk aplikasi sistem informasi pengendalian kualitas *accumulator* pada bagian *Quality Control* divisi *Painting*.

Tabel V.1 Kebutuhan Sistem Informasi Pengendalian Kualitas *Accumulator*

Kebutuhan Sistem	
<i>Project Name</i>	Sistem Informasi Pengendalian Kualitas <i>Accumulator</i> .
<i>Project Sponsor</i>	Bagian <i>Quality Control</i> divisi <i>Painting</i>
<i>Business Need</i>	<p>Terwujudnya sebuah sistem informasi pengendalian kualitas <i>accumulator</i> divisi <i>Painting</i> yang mampu:</p> <ol style="list-style-type: none"> 1. Menginput dan mengolah laporan <i>loading-unloading part accumulator</i> secara otomatis. 2. Memiliki media penyimpanan berbasis <i>database</i> yang dapat memudahkan proses pencarian data <i>accumulator</i>. 3. Dalam Pembuatan <i>quality monthly report accumulator</i> menjadi lebih cepat, mudah dan akurat sehingga dapat digunakan oleh perusahaan dalam mengambil keputusan.
<i>Business Requirement</i>	Memberikan sistem informasi pengendalian kualitas <i>accumulator</i> yang dapat membantu bagian <i>Quality Control</i> divisi <i>Painting</i> dalam membuat laporan <i>loading-unloading</i> dan <i>quality monthly report</i> .

Tabel V.1 Kebutuhan Sistem Informasi Pengendalian Kualitas *Accumulator*
(lanjutan)

Kebutuhan Sistem	
<i>Bussiness Value</i>	<ol style="list-style-type: none"> 1. Memudahkan dalam proses penginputan dan pengolahan laporan <i>loading-unloading</i>. 2. Memudahkan dalam proses pencarian data bila diperlukan. 3. Memudahkan dalam pembuatan <i>quality monthly report</i> divisi <i>Painting</i> sehingga laporan menjadi lebih informatif dan mudah dimengerti.
<i>Special Issues or Constrains</i>	<p>Membangun sistem informasi pengendalian kualitas <i>accumulator</i> dengan rincian yang mampu:</p> <ol style="list-style-type: none"> 1. Menginput laporan <i>loading-unloading part accumulator</i> secara otomatis. 2. Memiliki media penyimpanan data berbasis <i>database</i> yang dapat memudahkan proses pencarian data <i>accumulator</i>. 3. Dalam pembuatan <i>quality monthly report accumulator</i> menjadi lebih cepat, mudah dan akurat sehingga dapat digunakan oleh perusahaan dalam mengambil keputusan.

(Sumber: Hasil Analisis Data, 2016)

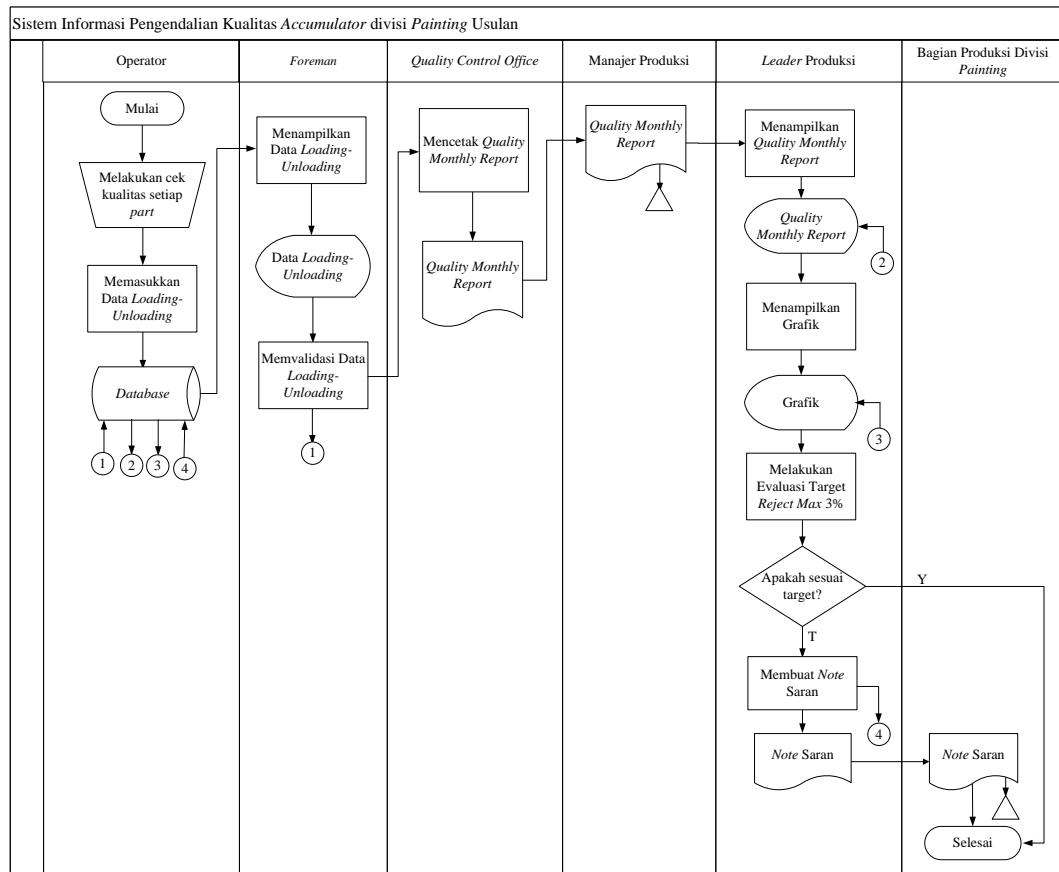
5.2 Prosedur Sistem Informasi Pengendalian Kualitas *Accumulator* Usulan

Prosedur sistem informasi pengendalian kualitas *accumulator* yang diusulkan melalui tahapan sebagai berikut:

1. Operator melakukan cek kualitas produk.
2. Operator memasukkan data *loading-unloading* pada aplikasi.
3. *Foreman* memvalidasi laporan *loading-unloading*.
4. Bagian *Quality Control Office* mencetak *quality monthly report* berdasarkan *input-an loading-unloading* dan laporan tersebut diserahkan kepada Manajer Produksi.

5. *Leader Produksi* melihat laporan *quality monthly report* dan melakukan evaluasi *quality monthly report* pada grafik, apabila banyak *part rejection* *max 3%* maka *Leader Produksi* akan membuat *note saran* untuk bagian *Produksi* divisi *Painting*.

Seperti yang dapat dilihat pada Gambar V.1 adalah *Flowmap* Sistem Informasi Pengendalian Kualitas *Accumulator* pada divisi *Painting* Usulan sebagai berikut:



Gambar V.1 *Flowmap* Sistem Informasi Pengendalian Kualitas *Accumulator* divisi *Painting* Usulan
(Sumber: Hasil Analisis Data, 2016)

5.3 Analisis dan Perancangan Sistem Usulan

Untuk mengatasi permasalahan dalam pengendalian produksi *accumulator* yang ada pada bagian *Quality Control* divisi *Painting* diajukan usulan sistem baru dengan menerapkan penggunaan aplikasi komputer untuk pengolahan data yang akan mendukung beberapa proses bisnis yang sedang berjalan pada perusahaan.

Dengan menggunakan aplikasi ini membuat sistem menjadi terintegrasi sehingga dapat menyajikan informasi secara cepat dan dapat meningkatkan fungsionalitas proses pengendalian kualitas *accumulator* yang ada pada bagian *Quality Control* divisi *Painting* di PT Kurnia Manunggal Sejahtera.

Sistem yang diusulkan yaitu mempercepat proses pembuatan *quality monthly report* yang kemudian digunakan untuk bahan evaluasi pengendalian kualitas *accumulator* selanjutnya. Analisis kebutuhan sistem informasi pengendalian kualitas *accumulator* divisi *Painting* menggunakan model pengembangan perangkat lunak *prototype evolutioner*. Perancangan model sistem menggunakan *tools UML (unified modeling language)* yang meliputi pembuatan *use case diagram, activity diagram* dan *sequence diagram*. Pemodelan data yang digunakan adalah *class diagram* dan kamus data. Perancangan aplikasi menggunakan *flowchart, HIPO* dan antarmuka.

5.4 Pemodelan Sistem Pengendalian Kualitas Usulan

Tahap ini merupakan tahap pertama dalam metodologi *prototype evolutioner*, yaitu tahap memodelkan sistem menggunakan *UML (unified modeling language)* yang meliputi pembuatan *use case diagram, activity diagram* dan *sequence diagram*.

5.4.1 Use Case Diagram

Perancangan *use case diagram* perancangan sistem informasi pengendalian kualitas *accumulator* dapat dijelaskan sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case* perancangan sistem informasi pengendalian kualitas *accumulator* usulan dapat dilihat pada Tabel V.2 berikut:

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

No.	Aktor	Deskripsi
1.	Operator	Orang yang bertugas memasukkan data <i>loading-unloading</i> .

Tabel V.2 Definisi Aktor *Use Case Diagram* Sistem Usulan (lanjutan)

No.	Aktor	Deskripsi
2.	<i>Quality Control Office</i>	Orang yang bertugas mengelola data master dan mencetak <i>quality monthly report</i> .
3.	<i>Foreman</i>	Orang yang bertugas melihat dan memvalidasi data <i>loading-unloading</i> .
4.	Leader Produksi	Orang yang bertugas melihat <i>quality monthly report</i> , grafik dan membuat <i>note</i> saran jika hasil kualitas dibawah target yang sudah ditetapkan.
5.	Manajer Produksi	Orang yang bertugas menerima dan melihat <i>quality monthly report</i> .

(Sumber: Hasil Analisis Data, 2016)

2. Definisi *Use Case*

Pendefinisian *use case* pada sistem informasi pengendalian kualitas *accumulator* usulan dapat dilihat pada Tabel V.3.

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

No.	Aktor	Deskripsi
1.	<i>Login</i>	Proses untuk melakukan <i>login</i> pada aplikasi.
2.	Mengelola data master	Proses generalisasi yang meliputi tiga buah proses pengolahan data master, yaitu data <i>user</i> dan produk.
3.	Memasukkan data <i>loading-unloading</i>	Proses memasukkan data <i>loading-unloading</i> untuk melakukan tambah, ubah, simpan, hapus dan cari.
4.	Menampilkan data <i>loading-unloading</i>	Proses menampilkan data <i>loading-unloading</i> .
5.	Memvalidasi data <i>loading-unloading</i>	Proses memvalidasi data <i>loading-unloading</i> .
6.	Mencetak <i>quality monthly</i>	Proses mencetak <i>quality monthly report</i> .

	<i>report</i>	
--	---------------	--

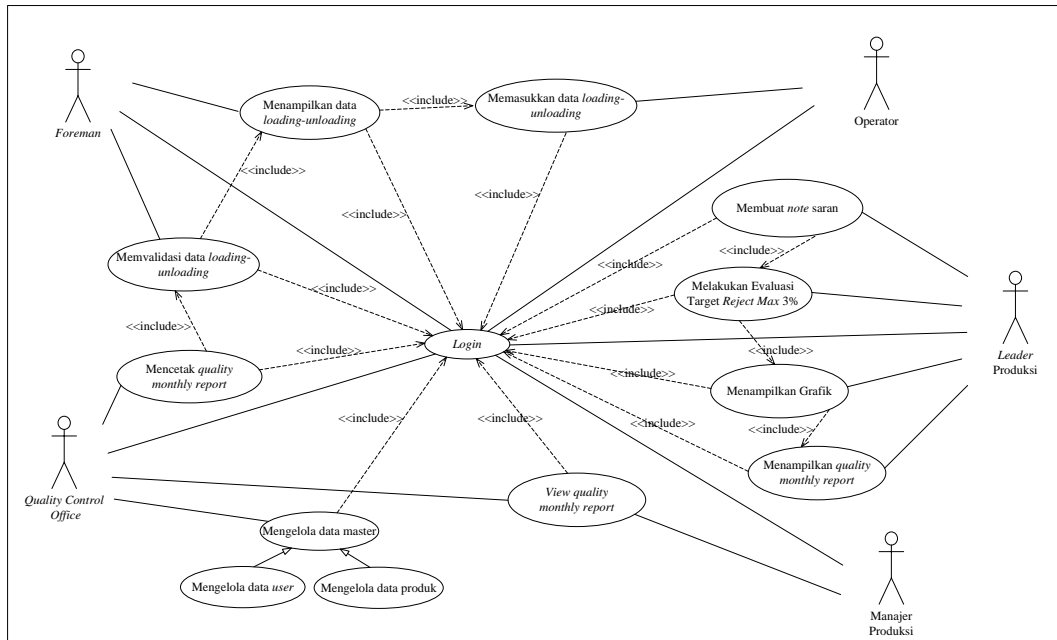
Tabel V.3 Definisi *Use Case Diagram* Sistem Usulan (lanjutan)

No.	Aktor	Deskripsi
7.	<i>View quality monthly report</i>	Proses untuk melihat <i>quality monthly report</i> .
8.	Menampilkan <i>quality monthly report</i>	Proses menampilkan <i>quality monthly report</i> .
9.	Menampilkan grafik	Proses menampilkan grafik.
10.	Melakukan Evaluasi Target <i>Reject Max 3 %</i>	Proses evaluasi apakah sudah sesuai target <i>reject max 3%</i> dari jumlah produksi.
11.	Membuat <i>note saran</i>	Proses pembuatan <i>note saran</i> .

(Sumber: Hasil Analisis Data, 2016)

3. Use Case Diagram

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Rancangan *use case diagram* sistem informasi pengendalian kualitas *accumulator* yang diusulkan dapat dilihat pada Gambar V.2.



Gambar V.2 *Use Case Diagram* Sistem Informasi Pengendalian Kualitas *Accumulator* Usulan

(Sumber: Hasil Analisis Data, 2016)

4. Skenario *Use Case*

Skenario jalannya masing-masing *use case* pada sistem informasi pengendalian kualitas *accumulator* usulan dapat dilihat pada poin berikut:

1. *Use Case Login*

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

Tabel V.4 Skenario *Use Case Login*

Nama Use Case	Login
<i>Primary Actor</i>	Operator, <i>Quality Control Office</i> , Foreman, Leader Produksi dan Manajer Produksi.
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan <i>login</i> sesuai dengan hak akses dari <i>user</i> sesuai jabatan.
<i>Normal Flow of Events</i>	<ol style="list-style-type: none">1. <i>User</i> membuka aplikasi.2. <i>Login Form</i>.3. <i>User</i> masukkan <i>username</i> dan <i>password</i> pada <i>login form</i>.4. Sistem mengecek ke basis data, apakah <i>user name</i> dan <i>password</i> benar.5. Jika <i>user name</i> dan <i>password</i> benar, maka muncul maka muncul pesan “<i>Login Berhasil!</i>”.6. Jika <i>user name</i> dan <i>password</i> salah (tidak valid), maka muncul pesan “<i>Username atau Password Yang Anda Masukan Salah!</i>” pada <i>login form</i>.

(Sumber: Hasil Analisis Data, 2016)

2. *Use Case Mengelola Data Master*

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

Tabel V.5 Skenario *Use Case* Mengelola Data Master

Nama Use Case	Mengelola Data Master
<i>Primary Actor</i>	<i>Quality Control Office</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan proses pengolahan data master</i>

Tabel V.5 Skenario *Use Case* Mengelola Data Master (lanjutan)

Nama Use Case	Mengelola Data Master
<i>Relationship</i>	<i>Association: Quality Control Office</i> <i>Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> melakukan <i>login</i>. 2. Sistem menampilkan menu utama. 3. <i>User</i> memilih menu data master. 4. Sistem menampilkan <i>form</i> data master. 5. <i>User</i> melakukan proses tambah, ubah dan hapus data. 6. Sistem menampilkan <i>form</i> isian data master. 7. <i>User</i> melakukan proses tambah, ubah, hapus data master.

(Sumber: Hasil Analisis Data, 2016)

3. *Use Case* Memasukkan Data *Loading-Unloading*

Berikut adalah skenario *use case* memasukkan data *loading-unloading* yang terdapat pada Tabel V.6:

Tabel V.6 Skenario *Use Case* Memasukkan Data *Loading-Unloading*

Nama Use Case	Memasukkan Data <i>Loading-Unloading</i>
<i>Primary Actor</i>	Operator
<i>Use Case</i>	<i>Use case ini menggambarkan proses memasukkan data</i>

<i>Description</i>	pada <i>form loading-unloading</i> .
<i>Relationship</i>	<i>Association: Operator</i> <i>Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> melakukan <i>login</i>. 2. Sistem menampilkan menu utama. 3. <i>User</i> memilih menu <i>loading-unloading</i>. 4. Sistem menampilkan <i>form data loading-unloading</i>. 5. <i>User</i> memasukkan data <i>loading-unloading</i> dan menyimpan ke basis data. 6. Sistem menampilkan data <i>loading-unloading</i>..

(Sumber: Hasil Analisis Data, 2016)

4. *Use Case* Menampilkan Data *Loading-Unloading*

Berikut adalah skenario *use case* menampilkan data *loading-unloading* yang terdapat pada Tabel V.7:

Tabel V.7 Skenario *Use Case* Menampilkan Data *Loading-Unloading*

Nama <i>Use Case</i>	Menampilkan Data <i>Loading-Unloading</i>
<i>Primary Actor</i>	Operator dan <i>Foreman</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses menampilkan data <i>loading-unloading</i> .
<i>Relationship</i>	<i>Association: Foreman</i> <i>Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> melakukan <i>login</i>. 2. Sistem menampilkan menu utama. 3. <i>User</i> memilih menu <i>loading-unloading</i>. 4. Sistem menampilkan <i>form data loading-unloading</i>. 5. <i>User</i> melihat data <i>loading-unloading</i>.

(Sumber: Hasil Analisis Data, 2016)

5. *Use Case Memvalidasi Data Loading-Unloading*

Berikut adalah skenario *use case* memvalidasi data *loading-unloading* yang terdapat pada Tabel V.8:

Tabel V.8 Skenario *Use Case Memvalidasi Data Loading-Unloading*

Nama Use Case	Memvalidasi Data Loading-Unloading
<i>Primary Actor</i>	<i>Foreman</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan proses memvalidasi data loading-unloading.</i>
<i>Relationship</i>	<i>Association: Foreman</i> <i>Include: - Login , Memasukan data loading-unloading.</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User melakukan Login</i> 2. <i>Sistem menampilkan menu utama.</i> 3. <i>User memilih menu loading-unloading.</i> 4. <i>Sistem menampilkan tabel data loading-unloading.</i> 5. <i>User memvalidasi data loading-unloading.</i>

(Sumber: Hasil Analisis Data, 2016)

6. *Use Case Mencetak Quality Monthly Report*

Berikut adalah skenario *use case* mencetak *quality monthly report* yang terdapat pada Tabel V.9:

Tabel V.9 Skenario *Use Case Mencetak Quality Monthly Report*

Nama Use Case	Mencetak Quality Monthly Report
<i>Primary Actor</i>	<i>Quality Control Office</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan mencetak quality monthly report.</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User melakukan login.</i> 2. <i>Sistem menampilkan menu utama.</i> 3. <i>User memilih menu laporan</i> 4. <i>Sistem menampilkan form quality monthly report.</i>

	<ol style="list-style-type: none"> 5. <i>User</i> memasukkan tanggal awal dan tanggal akhir lalu pilih cari. 6. Sistem melakukan pencarian dan menampilkan data <i>quality monthly report</i> berdasarkan tanggal awal dan tanggal akhir. 7. <i>User</i> melakukan cetak. 8. Sistem menampilkan halaman cetak <i>quality monthly report</i>.
--	--

(Sumber: Hasil Analisis Data, 2016)

7. *Use Case View Quality Monthly Report*

Berikut adalah skenario *use case view quality monthly report* yang terdapat pada Tabel V.10:

Tabel V.10 Skenario *Use Case View Quality Monthly Report*

Nama Use Case	<i>View Quality Monthly Report</i>
<i>Primary Actor</i>	<i>Quality Control, Leader Produksi dan Manajer Produksi.</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan proses melihat quality monthly report.</i>
<i>Relationship</i>	<i>Association: Quality Control, Leader Produksi dan Manajer Produksi.</i> <i>Include: - Login</i>

Tabel V.10 Skenario *Use Case View Quality Monthly Report* (lanjutan)

Nama Use Case	<i>View Quality Monthly Report</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> melakukan <i>login</i>. 2. Sistem menampilkan menu utama. 3. <i>User</i> memilih menu laporan. 4. Sistem menampilkan <i>form quality monthly report</i>. 5. <i>User</i> memasukkan tanggal awal dan tanggal akhir. 6. Sistem melakukan proses pencarian data <i>quality monthly report</i> berdasarkan tanggal awal dan tanggal

	akhir. 7. Sistem menampilkan data <i>quality monthly report</i> .
--	--

(Sumber: Hasil Analisis Data, 2016)

8. *Use Case* Menampilkan *Quality Monthly Report*

Berikut adalah skenario *use case* menampilkan *quality monthly report* yang terdapat pada Tabel V.11:

Tabel V.11 Skenario *Use Case* Menampilkan *Quality Monthly Report*

Nama <i>Use Case</i>	Menampilkan <i>Quality Monthly Report</i>
<i>Primary Actor</i>	<i>Leader</i> Produksi.
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses menampilkan <i>quality monthly report</i> .
<i>Relationship</i>	<i>Association: Leader</i> Produksi. <i>Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> melakukan <i>login</i>. 2. Sistem menampilkan menu utama. 3. <i>User</i> memilih menu laporan. 4. Sistem menampilkan <i>form quality monthly report</i>. 5. <i>User</i> memasukkan tanggal awal dan tanggal akhir. 6. Sistem melakukan proses pencarian data <i>quality monthly report</i> berdasarkan tanggal awal dan tanggal akhir. 7. Sistem menampilkan data <i>quality monthly report</i>.

(Sumber: Hasil Analisis Data, 2016)

9. *Use Case* Menampilkan Grafik

Berikut adalah skenario *use case* menampilkan grafik yang terdapat pada Tabel V.12:

Tabel V.12 Skenario *Use Case* Menampilkan Grafik

Nama <i>Use Case</i>	Menampilkan Grafik

<i>Primary Actor</i>	<i>Leader Produksi.</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan proses menampilkan grafik.</i>
<i>Relationship</i>	<i>Association: Leader Produksi.</i> <i>Include: - Login</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. <i>User</i> melakukan <i>login</i>. 2. Sistem menampilkan menu utama. 3. <i>User</i> memilih menu grafik. 4. Sistem menampilkan <i>form</i> grafik. 5. <i>User</i> memasukkan tanggal awal dan tanggal akhir. 6. Sistem melakukan proses pencarian data grafik berdasarkan tanggal awal dan tanggal akhir. 7. Sistem menampilkan data grafik.

(Sumber: Hasil Analisis Data, 2016)

10. *Use Case* Melakukan Evaluasi Target *Reject Max 3%*

Berikut adalah skenario *use case* melakukan evaluasi target *reject max 3%* yang terdapat pada Tabel V.13:

Tabel V.13 Skenario *Use Case* Melakukan Evaluasi Target *Reject Max 3%*

Nama <i>Use Case</i>	Melakukan Evaluasi Target <i>Reject Max 3%</i>
<i>Primary Actor</i>	<i>Leader Produksi</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan proses Melakukan Evaluasi Target <i>Reject Max 3%</i>.</i>
<i>Relationship</i>	<i>Association: Leader Produksi</i> <i>Include: - Login ,View Quality Monthly Report</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Sistem menampilkan menu utama. 2. <i>User</i> memilih menu grafik.

Tabel V.13 Skenario *Use Case* Melakukan Evaluasi Target *Reject Max* 3% (lanjutan)

Nama <i>Use Case</i>	Membuat <i>Note Saran</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 3. Sistem menampilkan grafik. 4. <i>User</i> mencari data grafik yang dibutuhkan. 5. Sistem menampilkan hasil pencarian data grafik. 6. <i>User</i> melihat apakah target hasil kualitas <i>reject max</i> 3%.

(Sumber: Hasil Analisis Data, 2016)

11. *Use Case* Membuat *Note Saran*

Berikut adalah skenario *use case* membuat *note saran* yang terdapat pada Tabel V.14:

Tabel V.14 Skenario *Use Case* Membuat *Note Saran*

Nama <i>Use Case</i>	Membuat <i>Note Saran</i>
<i>Primary Actor</i>	<i>Leader</i> Produksi
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses membuat <i>note saran</i> kebijakan produksi.
<i>Relationship</i>	<i>Association: Leader</i> Produksi <i>Include: - Login ,View Quality Monthly Report</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Sistem menampilkan menu utama. 2. <i>User</i> memilih menu <i>note saran</i>. 3. Sistem menampilkan <i>form note saran</i>. 4. <i>User</i> membuat dan mencetak <i>note saran</i>. 5. Sistem menampilkan hasil <i>note saran</i>

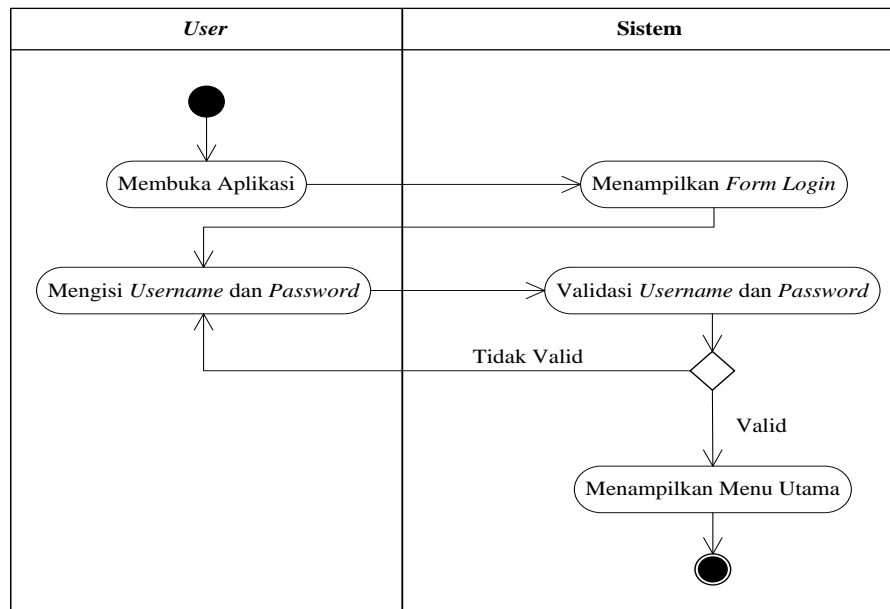
5.4.2 *Activity Diagram*

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar lebih memahami sistem yang akan dibuat, maka

perlu dibuatkan *activity diagram* tentang sistem informasi pengendalian kualitas ini.

1. *Activity Diagram Login*

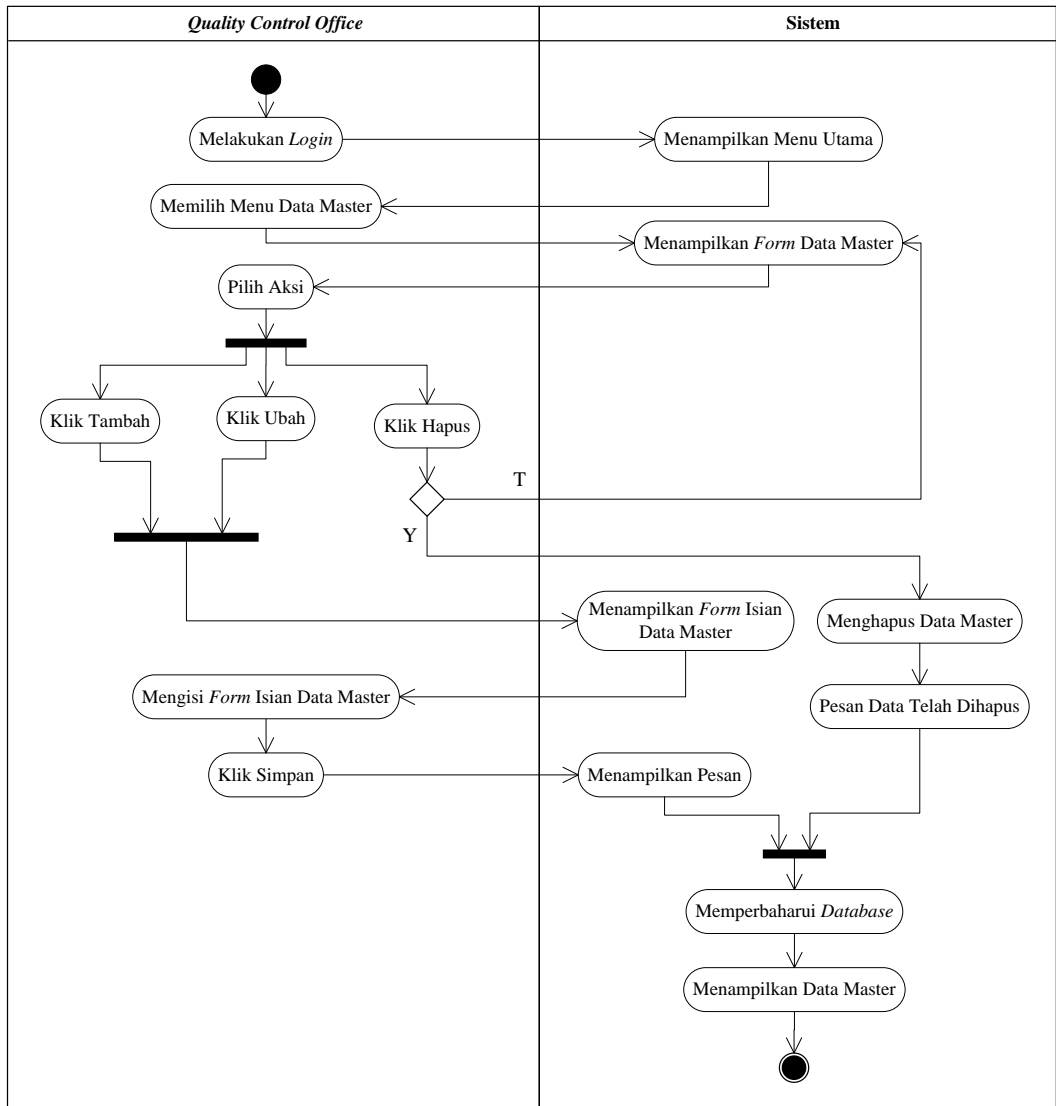
Activity diagram login ini menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu *Operator, Quality Control Office, Foreman, Leader Produksi* dan *Manajer Produksi* untuk dapat masuk ke dalam *Sistem Informasi Pengendalian Kualitas Accumulator*. *Activity diagram* dapat dilihat pada *Gambar V.3* sebagai berikut:



Gambar V.3 *Activity Diagram Login*
(Sumber: Hasil Analisis Data, 2016)

2. *Activity Diagram Mengelola Data Master*

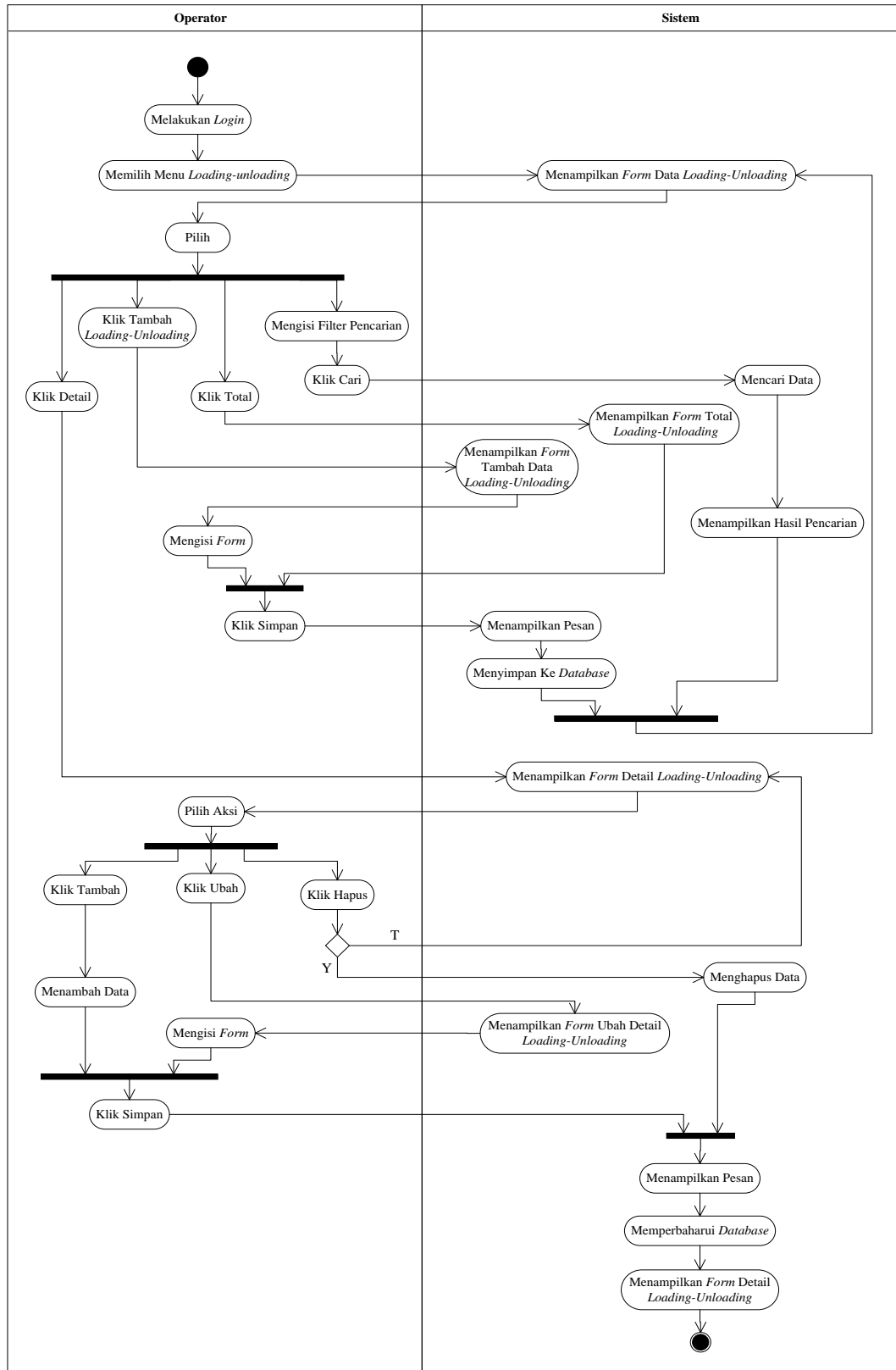
Activity diagram mengelola data master berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* mengelola data master yang diusulkan dapat dilihat pada *Gambar V.4* sebagai berikut:



Gambar V.4 Activity Diagram Mengelola Data Master
(Sumber: Hasil Analisis Data, 2016)

3. Activity Diagram Memasukkan Data Loading-Unloading

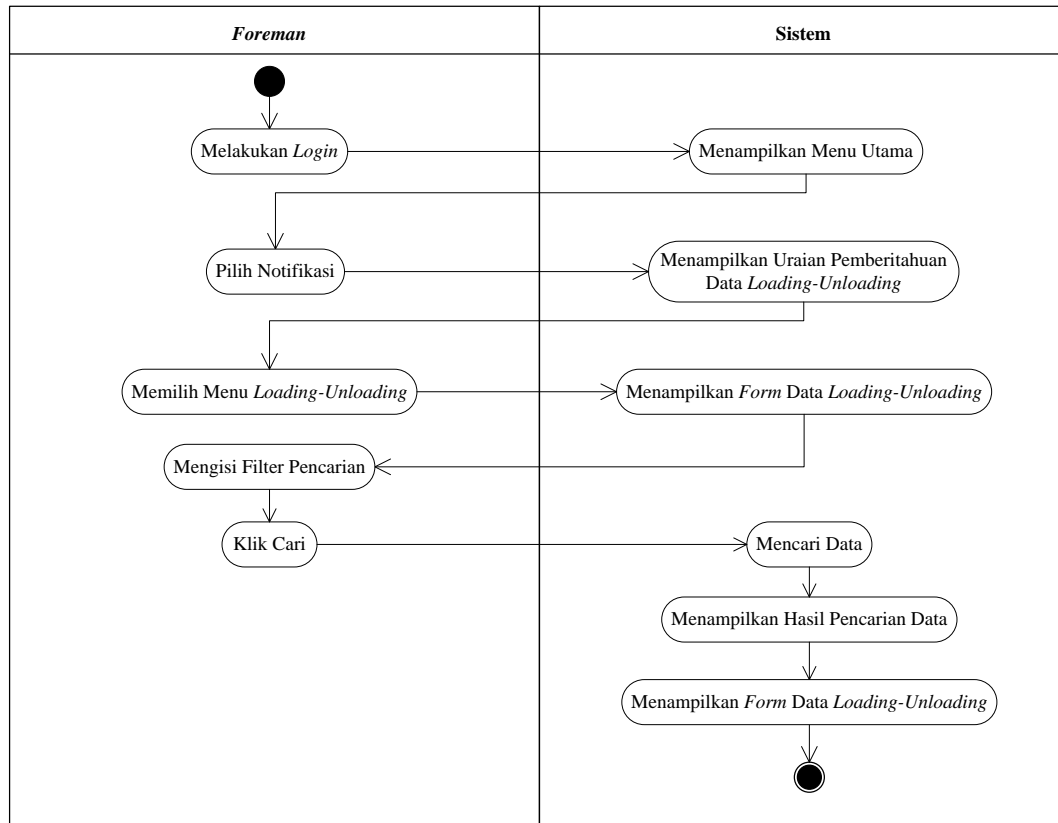
Activity diagram memasukkan data loading-unloading berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram memasukkan data loading-unloading yang diusulkan dapat dilihat pada Gambar V.5 sebagai berikut:



Gambar V.5 Activity Diagram Memasukkan Data Loading-Unloading
(Sumber: Hasil Analisis Data, 2016)

4. *Activity Diagram Menampilkan Data Loading-Unloading*

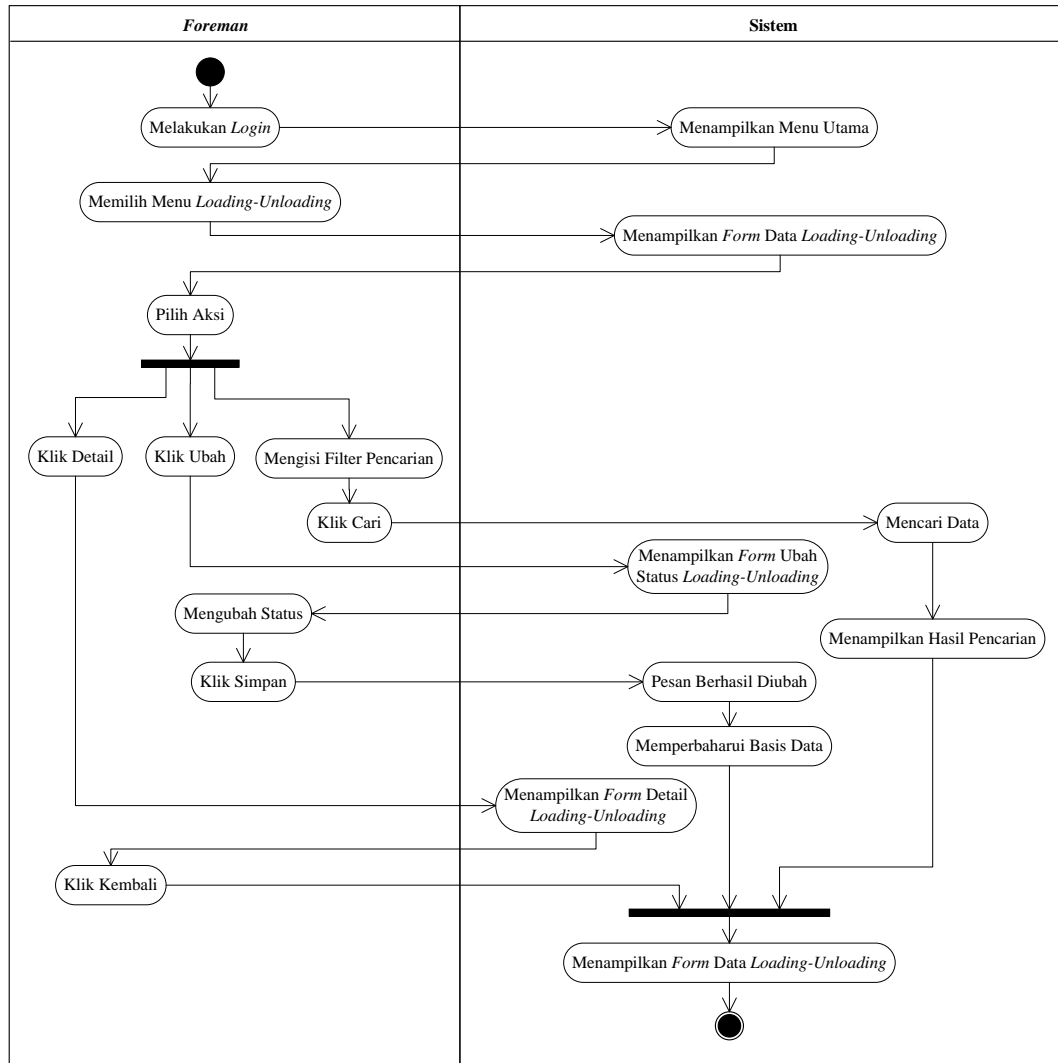
Activity diagram Menampilkan data *loading-unloading* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* menampilkan data *loading-unloading* yang diusulkan dapat dilihat pada Gambar V.6 sebagai berikut:



Gambar V.6 *Activity Diagram* Menampilkan Data *Loading-Unloading*
(Sumber: Hasil Analisis Data, 2016)

5. *Activity Diagram Memvalidasi Data Loading-Unloading*

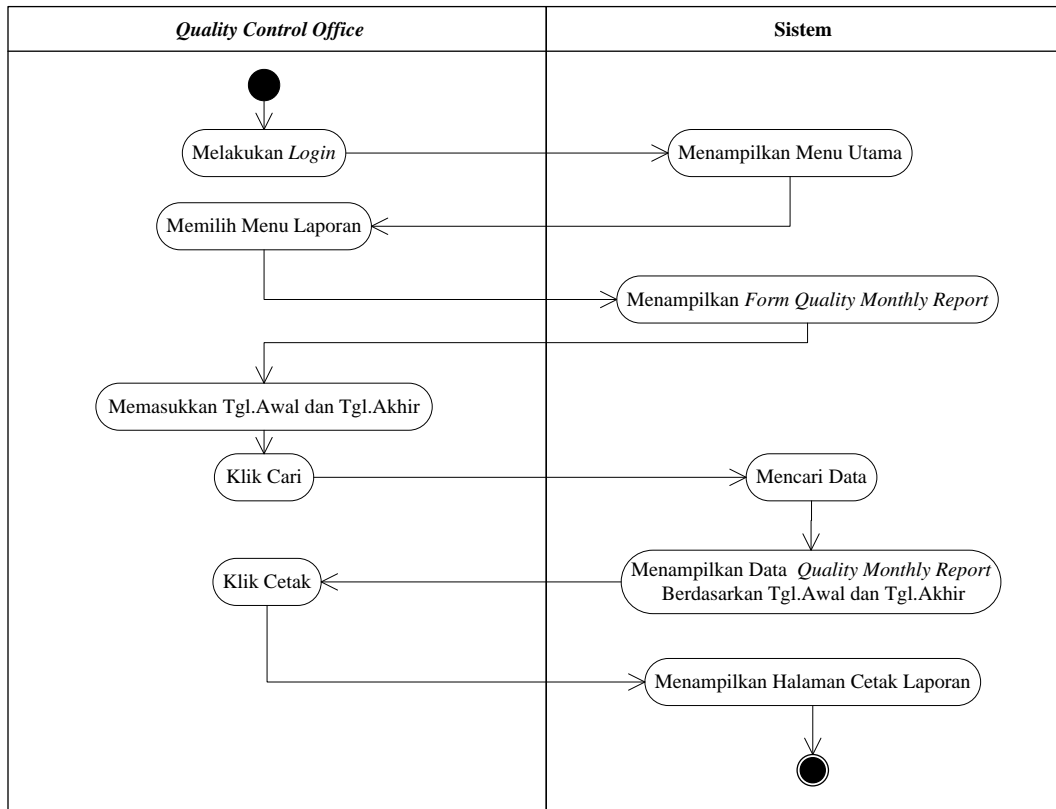
Activity diagram memvalidasi data *loading-unloading* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* memvalidasi data *loading-unloading* yang diusulkan dapat dilihat pada Gambar V.7 sebagai berikut:



Gambar V.7 Activity Diagram Memvalidasi Data Loading-Unloading
(Sumber: Hasil Analisis Data, 2016)

6. Activity Diagram Mencetak Quality Monthly Report

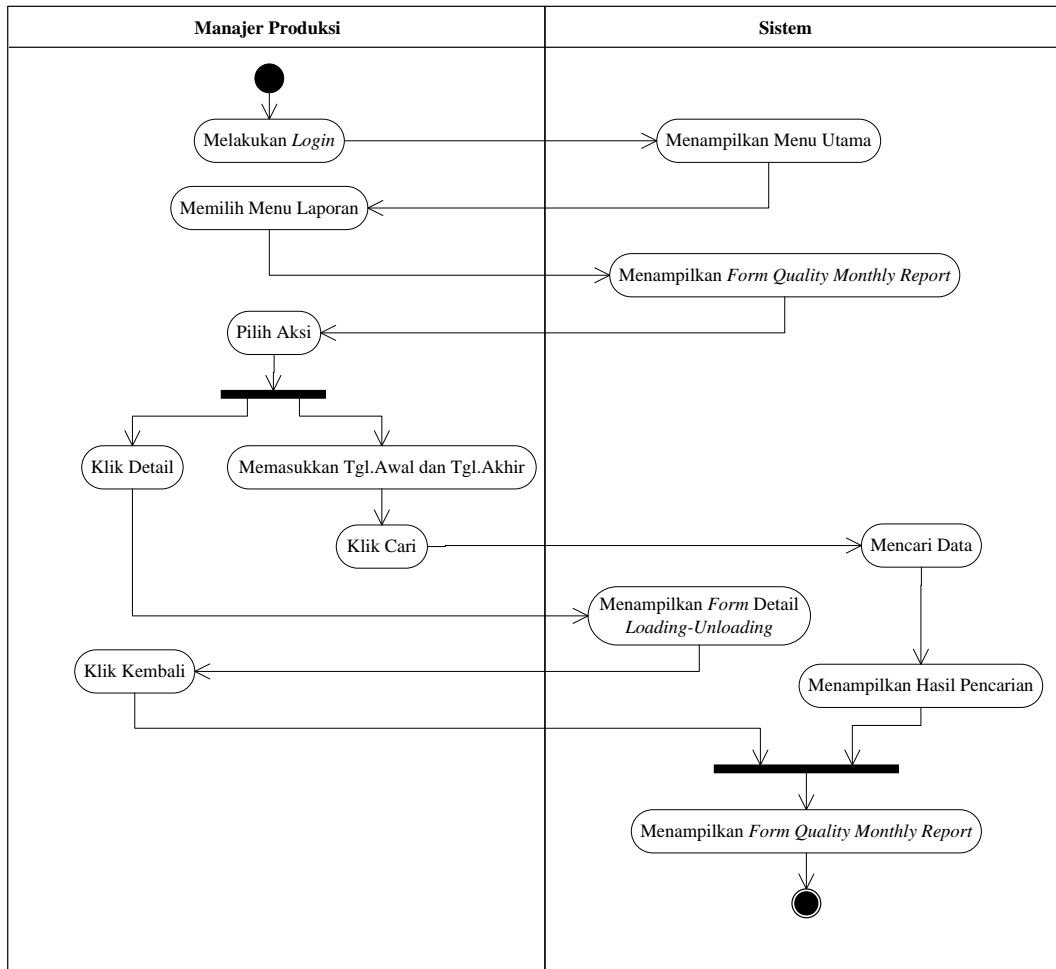
Activity diagram mencetak quality monthly report berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram mencetak quality monthly report yang diusulkan dapat dilihat pada Gambar V.8 sebagai berikut:



Gambar V.8 Activity Diagram Mencetak *Quality Monthly Report*
(Sumber: Hasil Analisis Data, 2016)

7. Activity Diagram View *Quality Monthly Report*

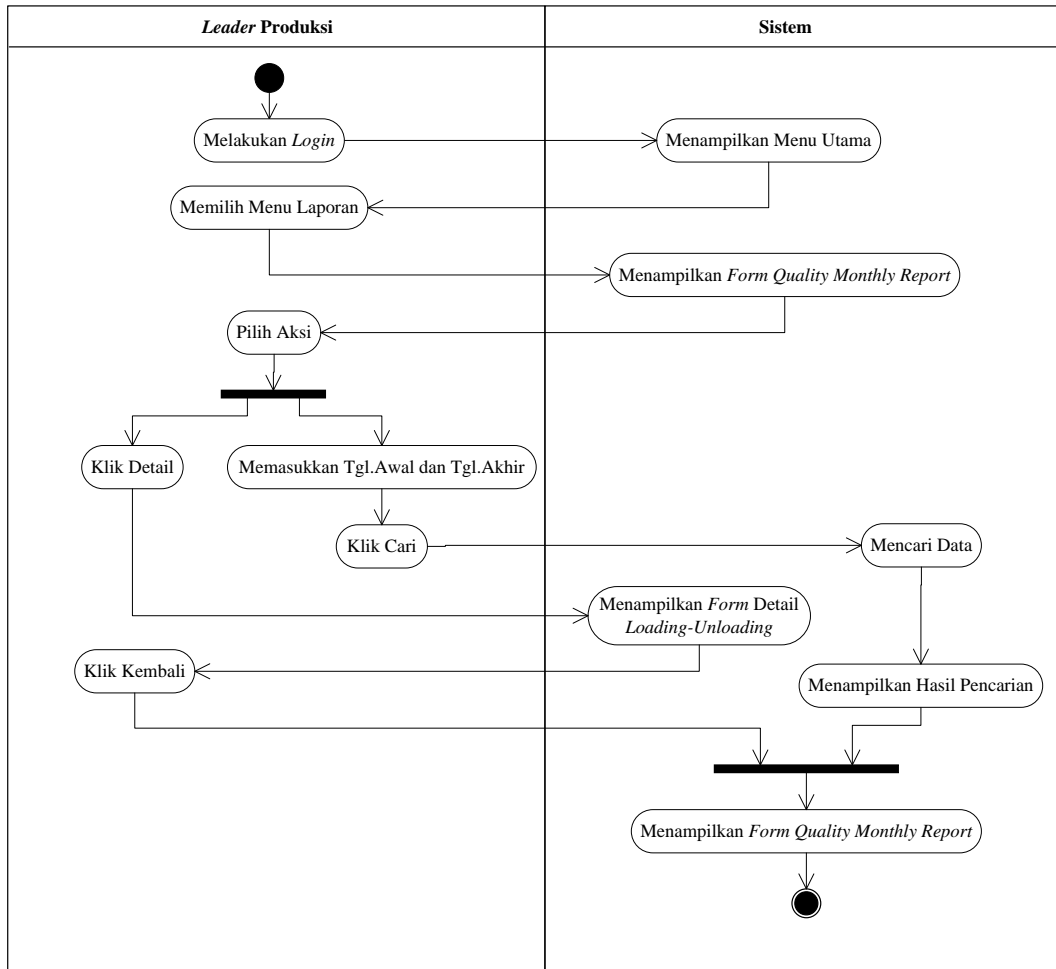
Activity diagram view quality monthly report berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram view quality monthly report yang diusulkan dapat dilihat pada Gambar V.9 sebagai berikut:



Gambar V.9 Activity Diagram View Quality Monthly Report
(Sumber: Hasil Analisis Data, 2016)

8. Activity Diagram Menampilkan Quality Monthly Report

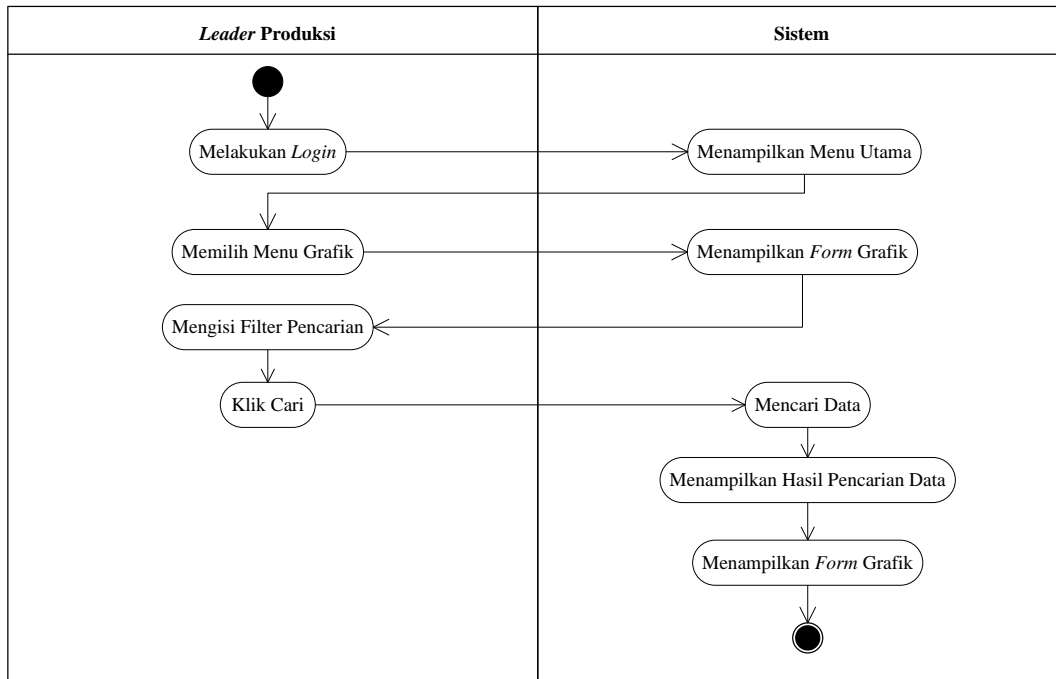
Activity diagram menampilkan quality monthly report berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram menampilkan quality monthly report yang diusulkan dapat dilihat pada Gambar V.10 sebagai berikut:



Gambar V.10 Activity Diagram Menampilkan *Quality Monthly Report*
(Sumber: Hasil Analisis Data, 2016)

9. Activity Diagram Menampilkan Grafik

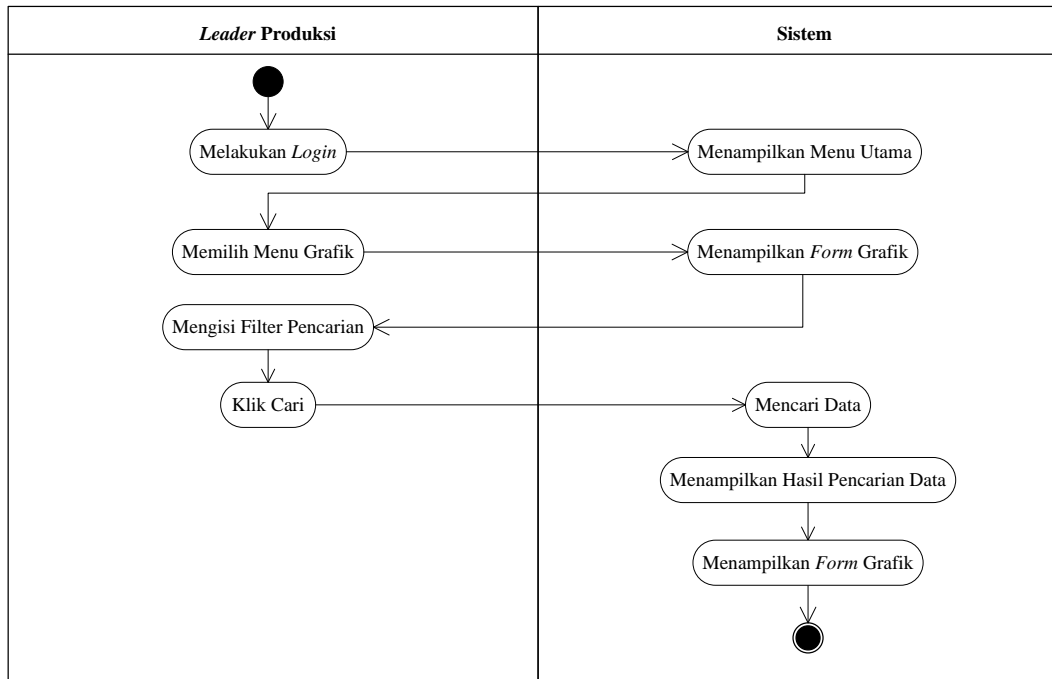
Activity diagram menampilkan grafik berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram menampilkan grafik yang diusulkan dapat dilihat pada Gambar V.11 sebagai berikut:



Gambar V.11 *Activity Diagram* Menampilkan Grafik
(Sumber: Hasil Analisis Data, 2016)

10. *Activity Diagram* Melakukan Evaluasi Target *Reject Max* 3%

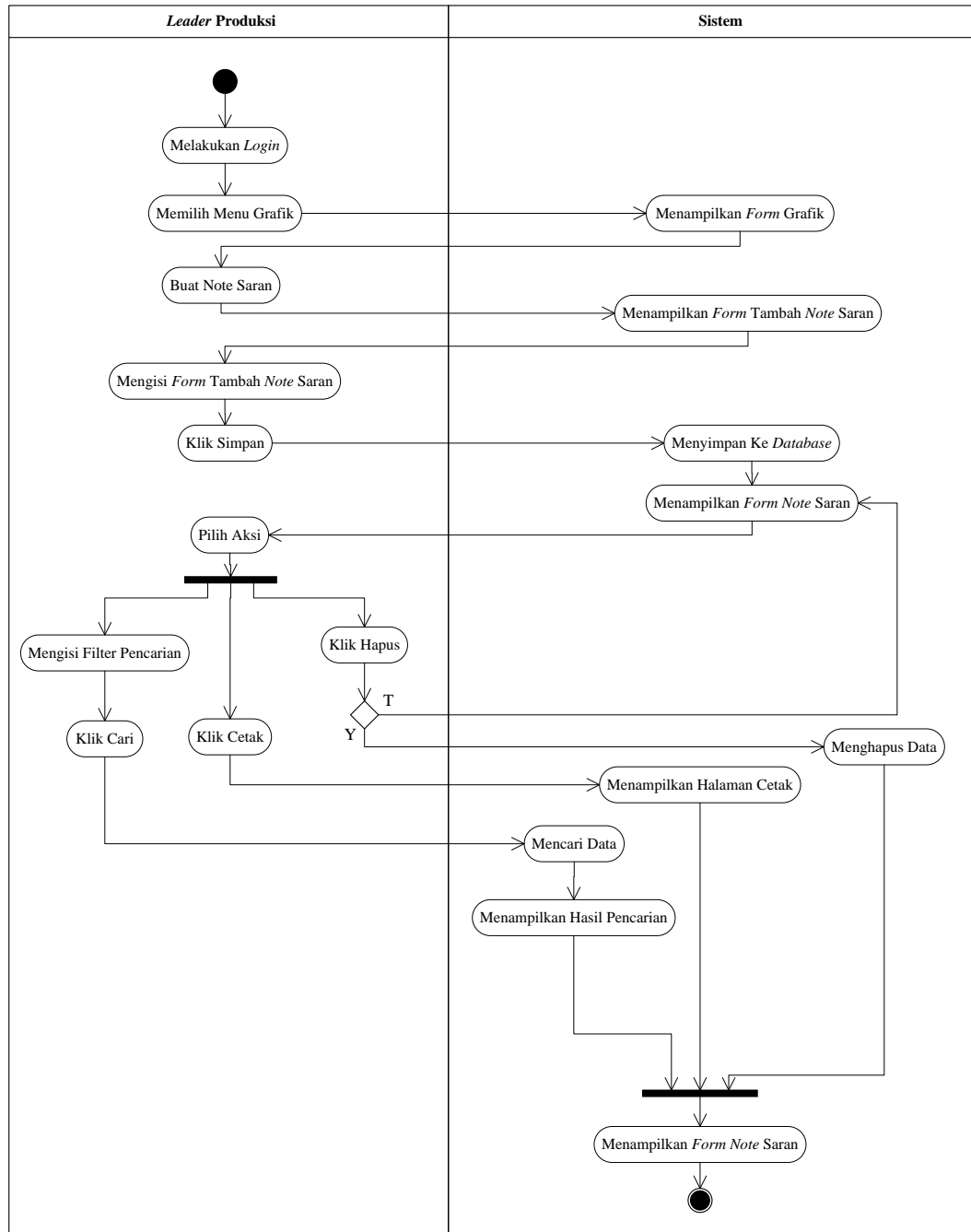
Activity diagram melakukan evaluasi target *reject max* 3% berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* melakukan evaluasi target *reject max* 3% yang diusulkan dapat dilihat pada Gambar V.12 sebagai berikut:



Gambar V.12 *Activity Diagram* Melakukan Evaluasi Target *Reject Max 3%*
(Sumber: Hasil Analisis Data, 2016)

11. *Activity Diagram* Membuat *Note* Saran

Activity diagram membuat *note* saran berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* membuat *note* saran yang diusulkan dapat dilihat pada Gambar V.13 sebagai berikut:



Gambar V.13 Activity Diagram Membuat Note Saran
(Sumber: Hasil Analisis Data, 2016)

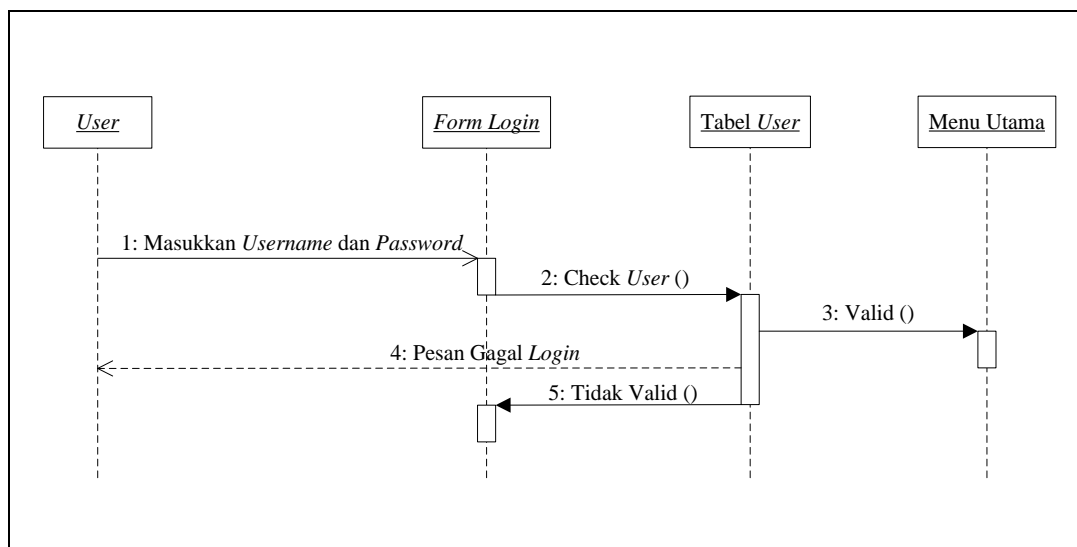
5.4.3 Sequence Diagram

Sequence Diagram menggambarkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu, dimana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat

gambaran pada diagram. Hubungan yang ada pada gambar di bawah ini adalah proses yang dilakukan oleh sistem ketika melakukan proses yang sesuai dengan suatu objek *use case diagram*, berikut adalah *sequence diagram* pada sistem informasi pengendalian kualitas *accumulator* divisi *Painting*:

1. *Sequence Diagram* Proses Login

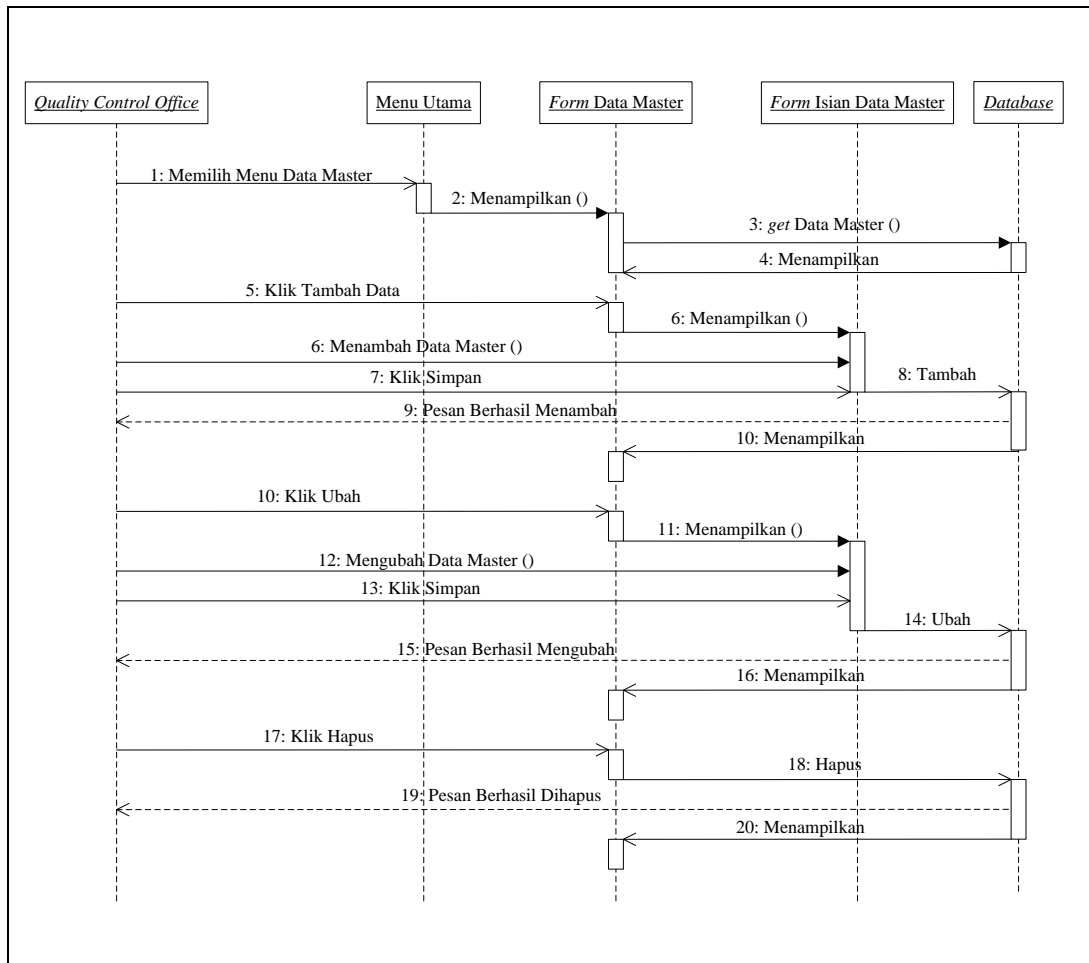
Sequence diagram login menjelaskan sebuah *sequence diagram* dalam proses *login*. Proses ini dilakukan oleh *user* sebelum masuk sistem. Adapun *sequence diagram* dari *use case login* dapat dilihat pada Gambar V.14.



Gambar V.14 *Sequence Diagram* Proses Login
(Sumber: Hasil Analisis Data, 2016)

2. *Sequence Diagram* Mengelola Data Master

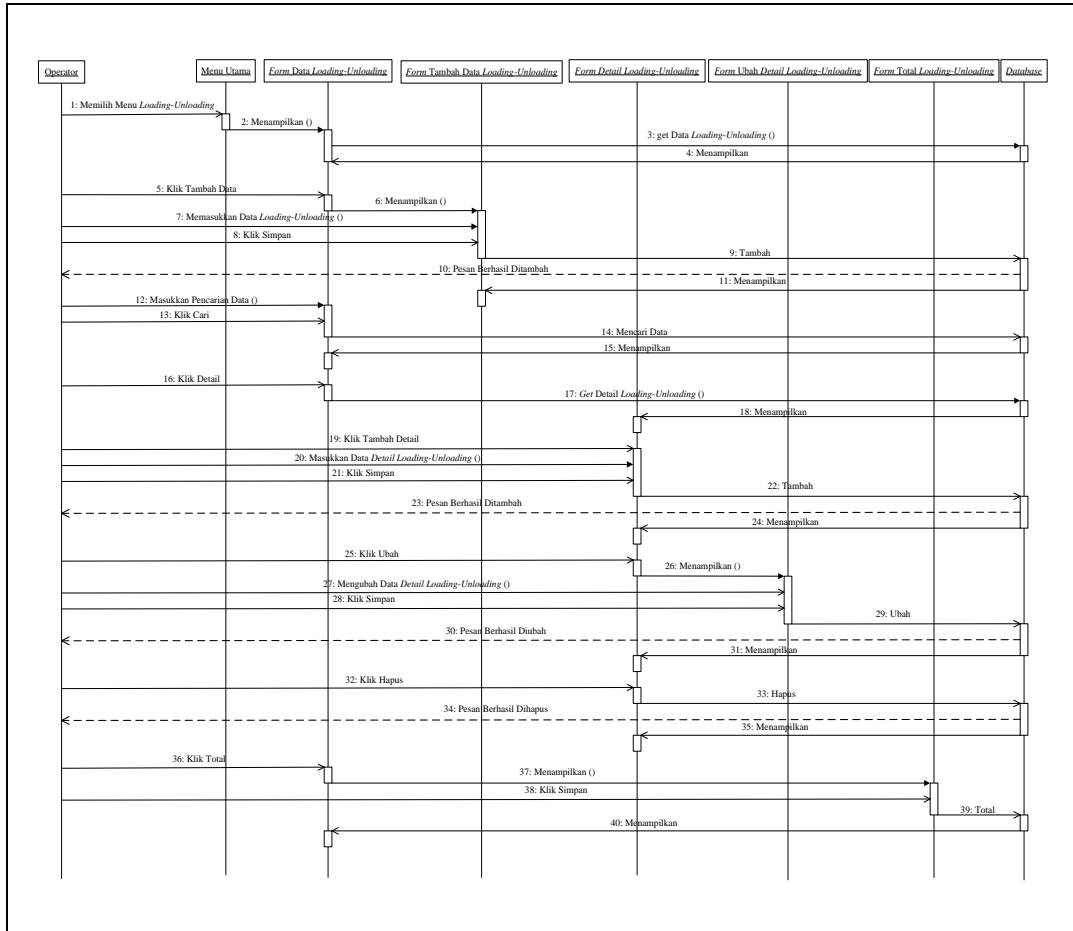
Sequence diagram mengelola data master menjelaskan sebuah *sequence diagram* dalam proses yang dilakukan menambah, menghapus dan mengubah data master. Adapun *sequence diagram* dari *use case* mengelola data master dapat dilihat pada Gambar V.15 sebagai berikut:



Gambar V.15 *Sequence Diagram* Mengelola Data Master
(Sumber: Hasil Analisis Data, 2016)

3. *Sequence Diagram* Memasukkan Data Loading-Unloading

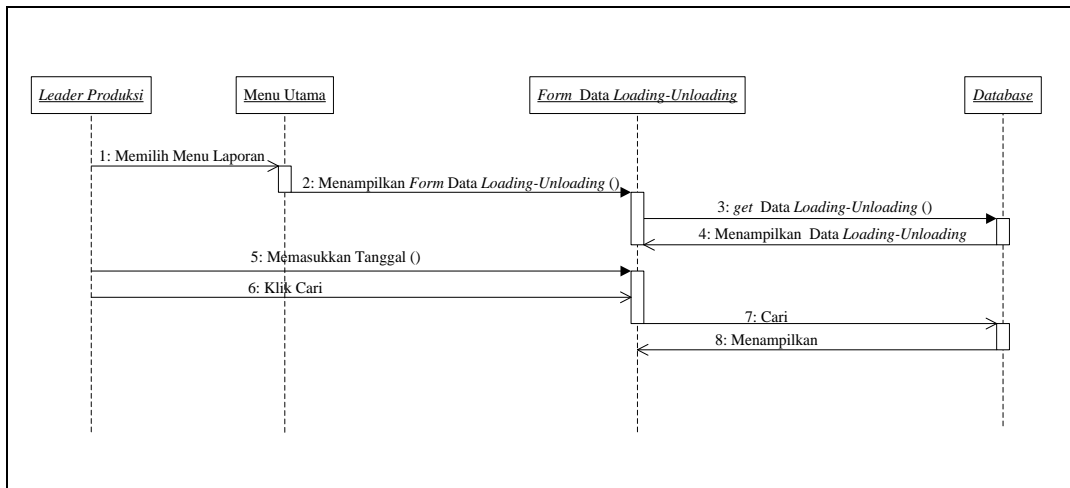
Sequence diagram memasukkan data *loading-unloading* menjelaskan sebuah *sequence diagram* dalam proses memasukkan data *loading-unloading*. Adapun *sequence diagram* dari *use case* memasukkan data *loading-unloading* dapat dilihat pada Gambar V.16 sebagai berikut:



Gambar V.16 Sequence Diagram Memasukkan Data Loading-Unloading (Sumber: Hasil Analisis Data, 2016)

4. Sequence Diagram Menampilkan Data Loading-Unloading

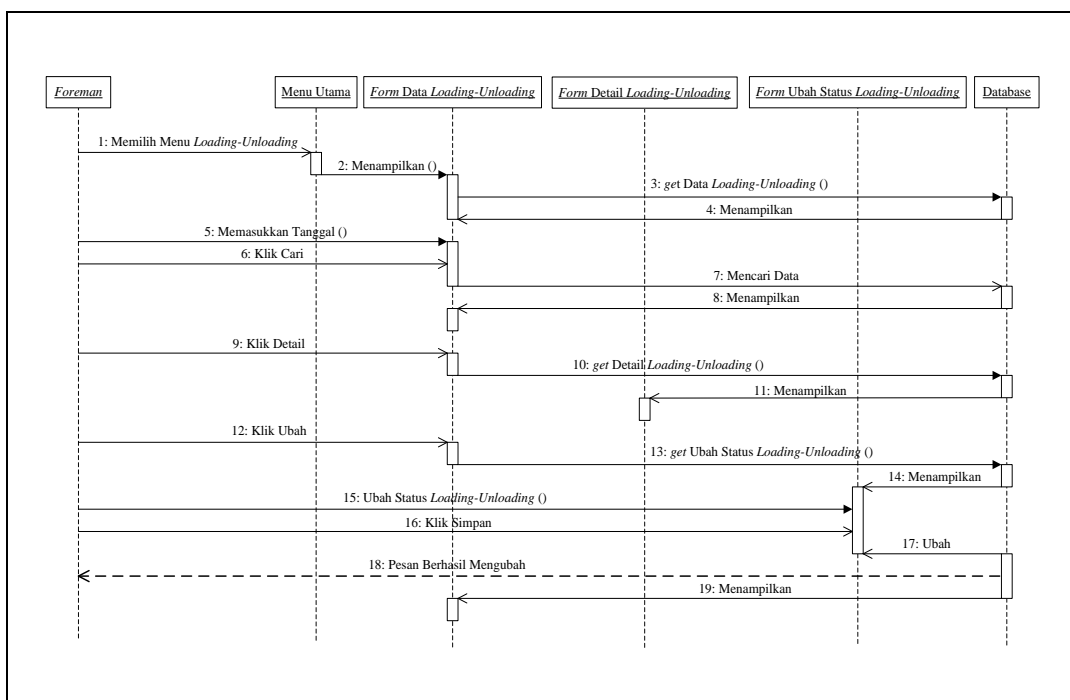
Sequence diagram menampilkan data loading-unloading menjelaskan sebuah sequence diagram dalam proses menampilkan data loading-unloading. Adapun sequence diagram dari use case menampilkan data loading-unloading dapat dilihat pada Gambar V.17 sebagai berikut:



Gambar V.17 *Sequence Diagram* Menampilkan Data Loading-Unloading
(Sumber: Hasil Analisis Data, 2016)

5. *Sequence Diagram* Memvalidasi Data Loading-Unloading

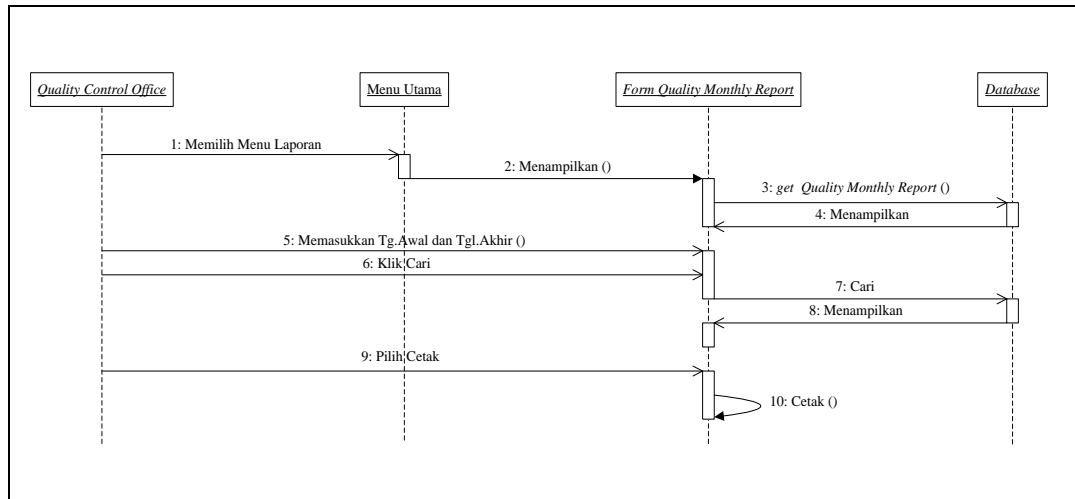
Sequence diagram memvalidasi data loading-unloading menjelaskan sebuah *sequence diagram* dalam proses memvalidasi data loading-unloading. Adapun *sequence diagram* dari use case memvalidasi data loading-unloading dapat dilihat pada Gambar V.18 sebagai berikut:



Gambar V.18 *Sequence Diagram* Memvalidasi Data Loading-Unloading
(Sumber: Hasil Analisis Data, 2016)

6. Sequence Diagram Mencetak Quality Monthly Report

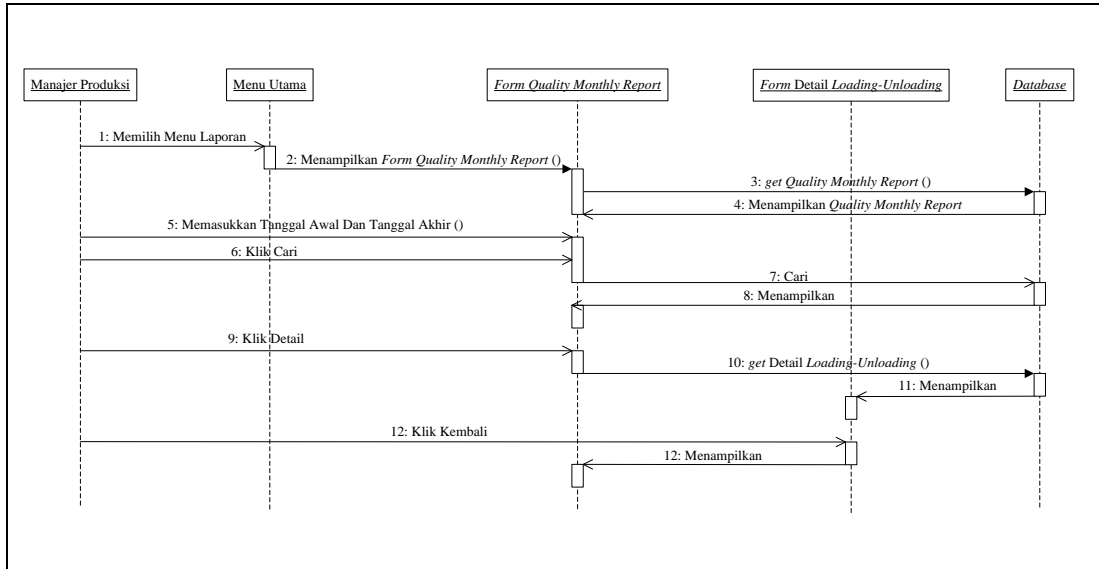
Sequence diagram mencetak quality monthly report menjelaskan sebuah sequence diagram dalam proses mencetak quality monthly report. Adapun sequence diagram dari use case mencetak quality monthly report dapat dilihat pada Gambar V.19 sebagai berikut:



Gambar V.19 Sequence Diagram Mencetak Quality Monthly Report
(Sumber: Hasil Analisis Data, 2016)

7. Sequence Diagram View Quality Monthly Report

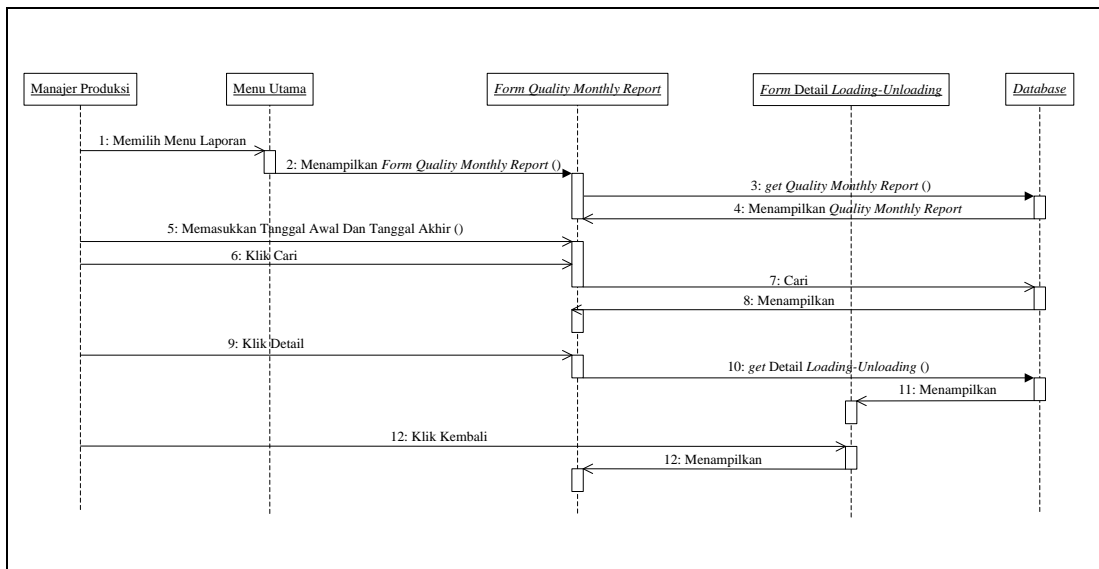
Sequence diagram view quality monthly report menjelaskan sebuah sequence diagram dalam proses view quality monthly report berdasarkan tanggal awal dan tanggal akhir yang akan dicari. Adapun sequence diagram dari use case view quality monthly report dapat dilihat pada Gambar V.20 sebagai berikut:



Gambar V.20 *Sequence Diagram View Quality Monthly Report*
(Sumber: Hasil Analisis Data, 2016)

8. *Sequence Diagram Menampilkan Quality Monthly Report*

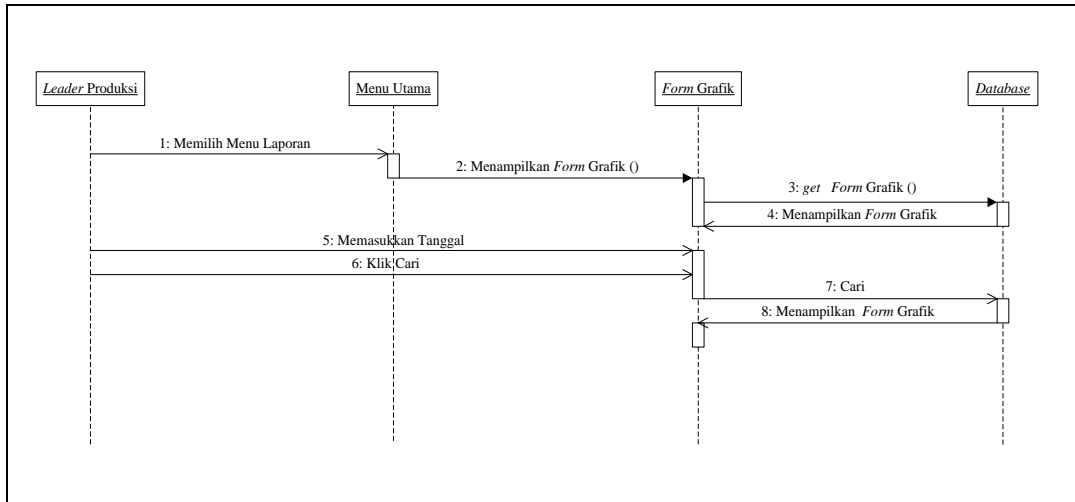
Sequence diagram menampilkan *quality monthly report* menjelaskan sebuah *sequence diagram* dalam proses menampilkan *quality monthly report* berdasarkan tanggal awal dan tanggal akhir yang akan dicari. Adapun *sequence diagram* dari *use case* menampilkan *quality monthly report* dapat dilihat pada Gambar V.21 sebagai berikut:



Gambar V.21 *Sequence Diagram Menampilkan Quality Monthly Report*
(Sumber: Hasil Analisis Data, 2016)

9. *Sequence Diagram* Menampilkan Grafik

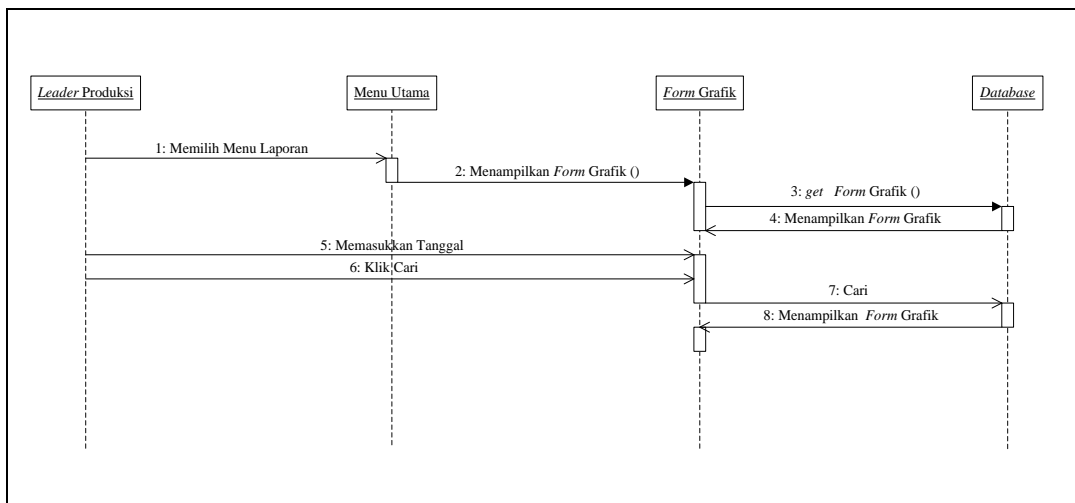
Sequence diagram menampilkan menjelaskan sebuah *sequence diagram* dalam proses menampilkan grafik. Adapun *sequence diagram* dari *use case* menampilkan grafik dapat dilihat pada Gambar V.22 sebagai berikut:



Gambar V.22 *Sequence Diagram* Menampilkan Grafik
(Sumber: Hasil Analisis Data, 2016)

10. *Sequence Diagram* Melakukan Evaluasi Target *Reject Max 3%*

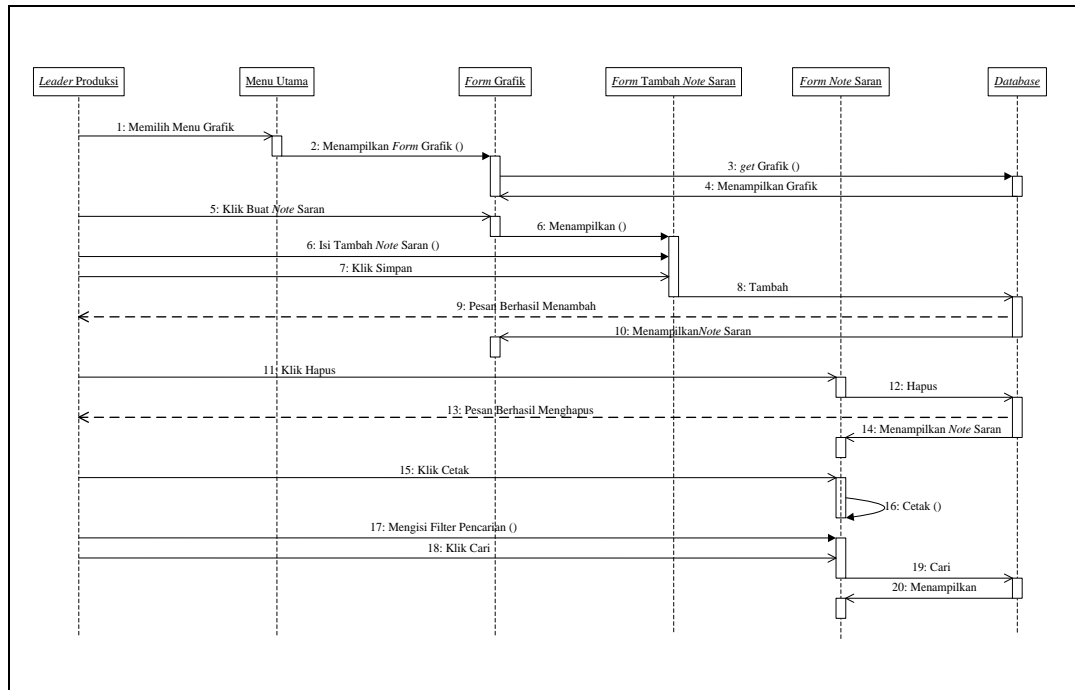
Sequence diagram melakukan evaluasi target *reject max 3%* menjelaskan sebuah *sequence diagram* dalam proses melakukan evaluasi target *reject max 3%*. Adapun *sequence diagram* dari *use case* melakukan evaluasi target *reject max 3%* dapat dilihat pada Gambar V.23 sebagai berikut:



Gambar V.23 *Sequence Diagram* Melakukan Evaluasi Target *Reject Max 3%*
(Sumber: Hasil Analisis Data, 2016)

11. Sequence Diagram Membuat Note Saran

Sequence diagram membuat *note* saran menjelaskan sebuah *sequence diagram* dalam proses membuat dan mencetak *note* saran. Adapun *sequence diagram* dari *use case* membuat *note* saran dapat dilihat pada Gambar V.24 sebagai berikut:



Gambar V.24 *Sequence Diagram* Membuat *Note* Saran
(Sumber: Hasil Analisis Data, 2016)

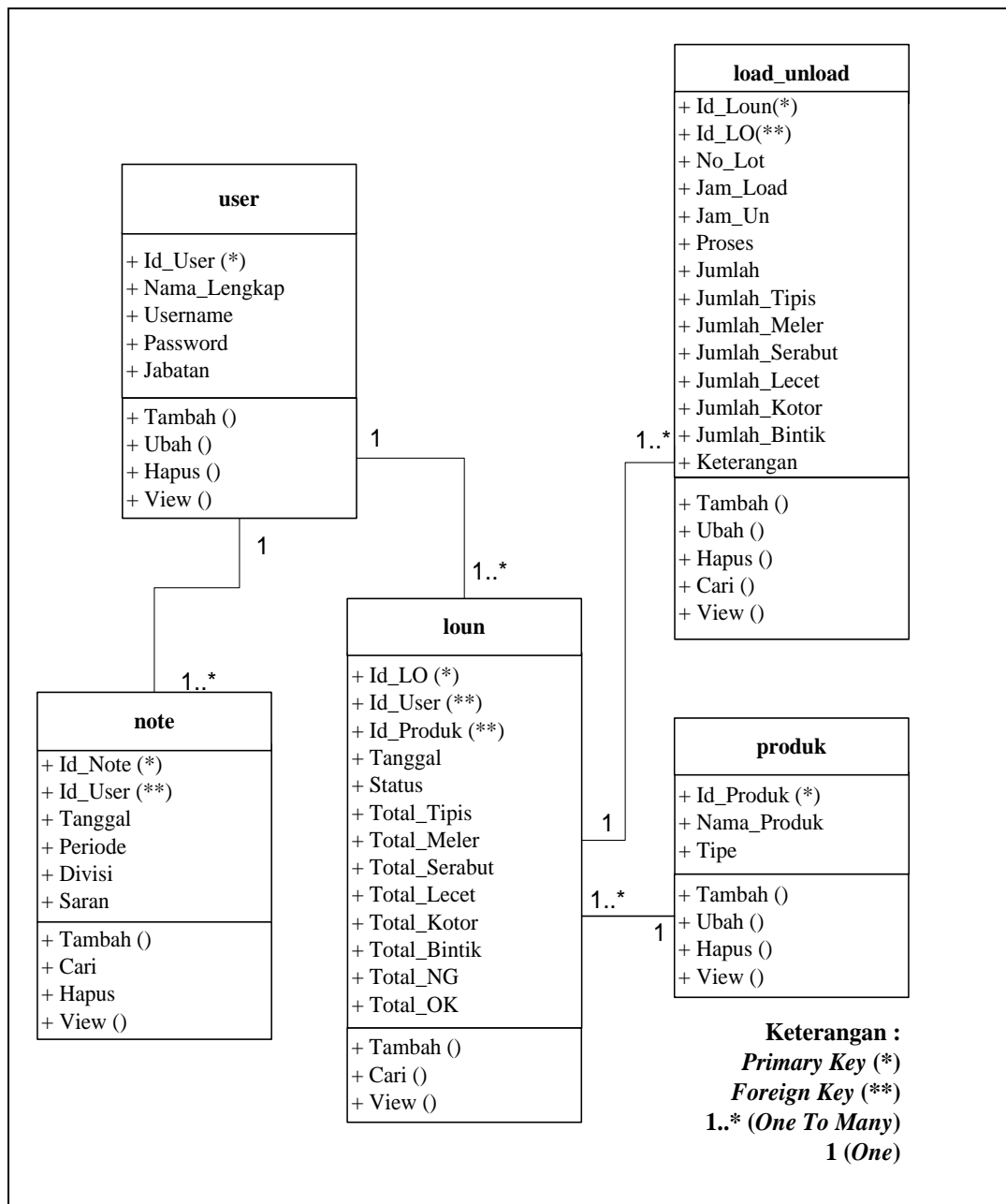
5.5 Pemodelan Data Pengendalian Kualitas Usulan

Tahap ini merupakan tahap kedua dalam metodologi *prototype evolutioner*, yaitu tahap memodelkan data menggunakan *class diagram* dan kamus data.

5.5.1 *Class Diagram*

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan *detail* tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi

pengendalian kualitas *accumulator* yang diusulkan dapat dilihat pada Gambar V.25 sebagai berikut:



Gambar V.25 *Class Diagram* Usulan
(Sumber: Hasil Analisis Data, 2016)

5.5.2 Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analisis sistem

mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan dan basis data. Berikut adalah kamus data sistem informasi pengendalian kualitas *accumulator*:

1. Spesifikasi Tabel *User*

Nama Tabel : *user*

Fungsi : Untuk menyimpan data *user*

Tipe : File data *master*

Tabel V.15 Tabel *User*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	<i>Id User</i>	Id_User	<i>Tinyint</i>	3	<i>Primary key</i>
2.	Nama Lengkap	Nama_Lengkap	<i>Varchar</i>	50	
3.	<i>Username</i>	Username	<i>Varchar</i>	20	
4.	<i>Password</i>	Password	<i>Varchar</i>	20	
5.	Jabatan	Jabatan	<i>Varchar</i>	20	

(Sumber: Hasil Analisis Data, 2016)

2. Spesifikasi Tabel Produk

Nama Tabel : produk

Fungsi : Untuk menyimpan data produk

Tipe : File data *master*

Tabel V.16 Tabel Produk

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id Produk	Id_Produk	<i>Char</i>	5	<i>Primary key</i>
2.	Nama Produk	Nama_Produk	<i>Varchar</i>	30	
3.	Tipe Produk	Tipe	<i>Varchar</i>	20	

(Sumber: Hasil Analisis Data, 2016)

3. Spesifikasi Tabel Detail *Loading-Unloading*

Nama Tabel : *load_unload*

Fungsi : Untuk menyimpan data detail *loading-unloading*

Tipe : File data transaksi

Tabel V.17 Tabel Detail *Loading-Unloading*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id>Loading	Id_Loun	<i>Int</i>	10	<i>Primary key</i>
2.	Id Lo	Id_LO	<i>Mediumint</i>	9	<i>Foreign key</i>
3.	Nomor Lot	No_Lot	<i>Tinyint</i>	3	
4.	Jam Loading	Jam_load	<i>Time</i>		
5.	Jam Unloading	Jam_Un	<i>Time</i>		
6.	Proses	Proses	<i>Char</i>	5	
7.	Jumlah	Jumlah	<i>Mediumint</i>	9	
8.	Jumlah Cat Tipis	Jumlah_Tipis	<i>Mediumint</i>	9	
9.	Jumlah Cat Meler	Jumlah_Meler	<i>Mediumint</i>	9	
10.	Jumlah Serabut	Jumlah_Serabut	<i>Mediumint</i>	9	
11.	Jumlah Lecet	Jumlah_Lecet	<i>Mediumint</i>	9	
12.	Jumlah Kotor	Jumlah_Kotor	<i>Mediumint</i>	9	
13.	Jumlah Bintik	Jumlah_Bintik	<i>Mediumint</i>	9	
14.	Keterangan	Keterangan	<i>Text</i>		

(Sumber: Hasil Analisis Data, 2016)

4. Spesifikasi Tabel *Loading-Unloading*

Nama Tabel : *loun*

Fungsi : Untuk menyimpan data *loading-unloading*

Tipe : File data transaksi

Tabel V.18 Tabel *Loading-Unloading*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id LO	Id_LO	<i>Mediumint</i>	5	<i>Primary key</i>
2.	Id <i>User</i>	Id_User	<i>Tinyint</i>	3	<i>Foreign key</i>
3.	Id_Produl	Id_Produk	<i>Char</i>	5	<i>Foreign key</i>
4.	Tanggal	Tanggal	<i>Date</i>		
5.	Status	Status	<i>Varchar</i>	25	
6.	Total Tipis	Total_Tipis	<i>Mediumint</i>	9	
7.	Total Meler	Total_Meler	<i>Mediumint</i>	9	
8.	Total Serabut	Total_Serabut	<i>Mediumint</i>	9	

Tabel V.18 Tabel *Loading-Unloading* (lanjutan)

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
9.	Total Lecet	Total_Lecet	<i>Mediumint</i>	9	
10.	Total Kotor	Total_Kotor	<i>Mediumint</i>	9	
12.	Total NG	Total_NG	<i>Mediumint</i>	9	
13	Total OK	Total_OK	<i>Mediumint</i>	9	

(Sumber: Hasil Analisis Data, 2016)

5. Spesifikasi *Note*

Nama Tabel : *note*

Fungsi : Untuk menyimpan data *note*

Tipe : File data laporan

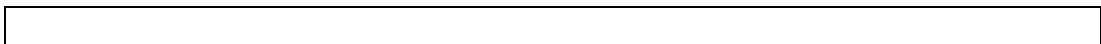
Tabel V.19 Tabel *Note*

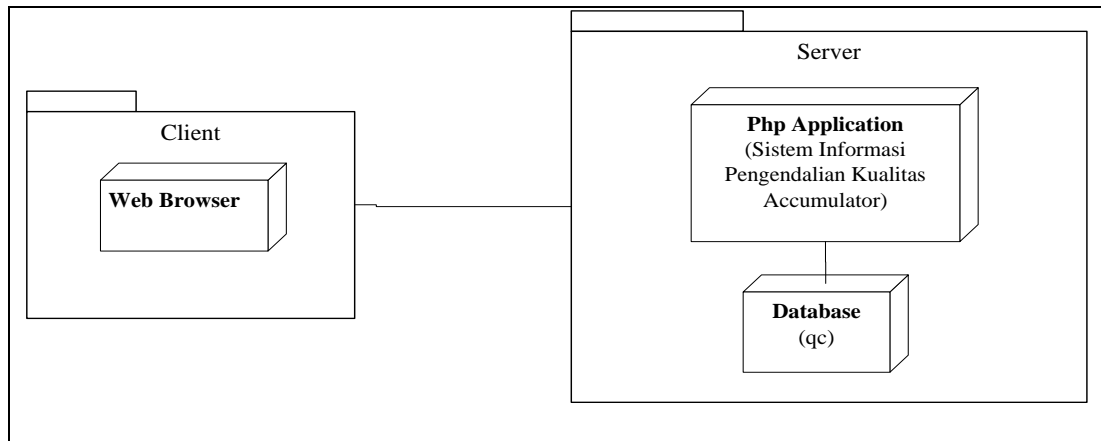
No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	<i>Id Note</i>	Id_Note	<i>Smallint</i>	5	<i>Primary key</i>
2.	<i>Id User</i>	Id_User	<i>Tinyint</i>	3	<i>Foreign key</i>
3.	Tanggal	Tanggal	<i>Date</i>		
4.	Periode	Periode	<i>Varchar</i>	30	
5.	Divisi	Divisi	<i>Varchar</i>	20	
6.	Saran	Saran	<i>Text</i>		

(Sumber: Hasil Analisis Data, 2016)

5.6 *Deployment Diagram*

Deployment diagram digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur fisik sebuah sistem. *Deployment diagram* sistem informasi pengendalian kualitas *accumulator* yang diusulkan dapat dilihat pada Gambar V.26.





Gambar V.26 *Deployment Diagram*
(Sumber: Hasil Analisis Data, 2016)

Berikut adalah penjelasan Gambar V.26 *deployment diagram* sistem informasi pengendalian kualitas *accumulator*:

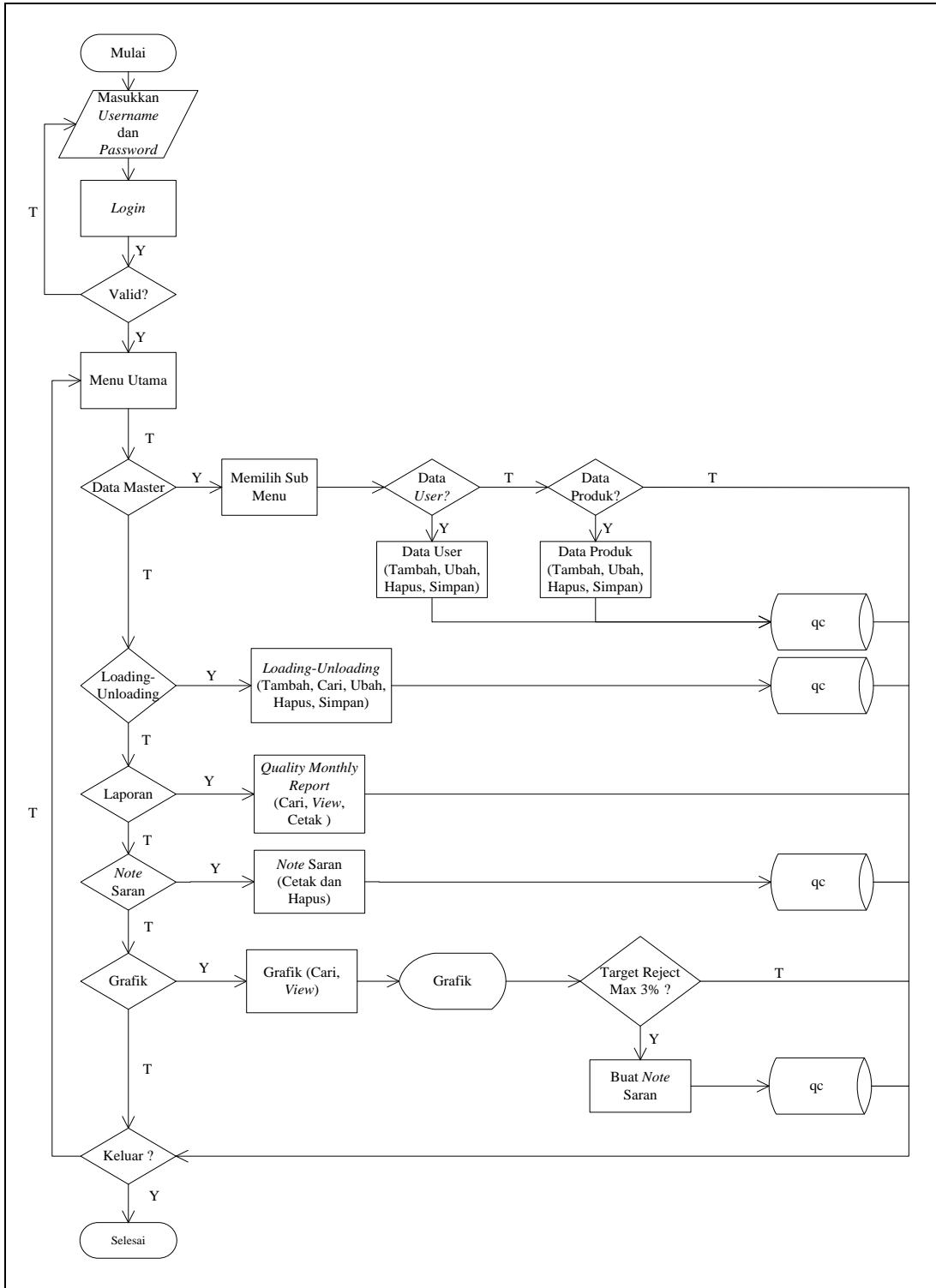
1. *Client* adalah komputer *client* yang terinstal. *Client* akan selalu terhubung dengan server.
2. *Server* aplikasi terdiri dari *php application* (sistem informasi pengendalian kualitas *accumulator*) dan *database (qc)*.

5.7 Perancangan Aplikasi Pengendalian Kualitas

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

5.7.1 *Flowchart* Program

Flowchart yang digunakan untuk mendokumentasikan aplikasi sistem informasi pengendalian kualitas produk ini menggunakan bagan alir logika program (program *logic flowchart*). Bagan alir ini digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika, bukan untuk menggambarkan instruksi-instruksi program komputer secara terinci. Gambar V.27 menunjukkan program *logic flowchart* aplikasi sistem informasi pengendalian kualitas *accumulator*.

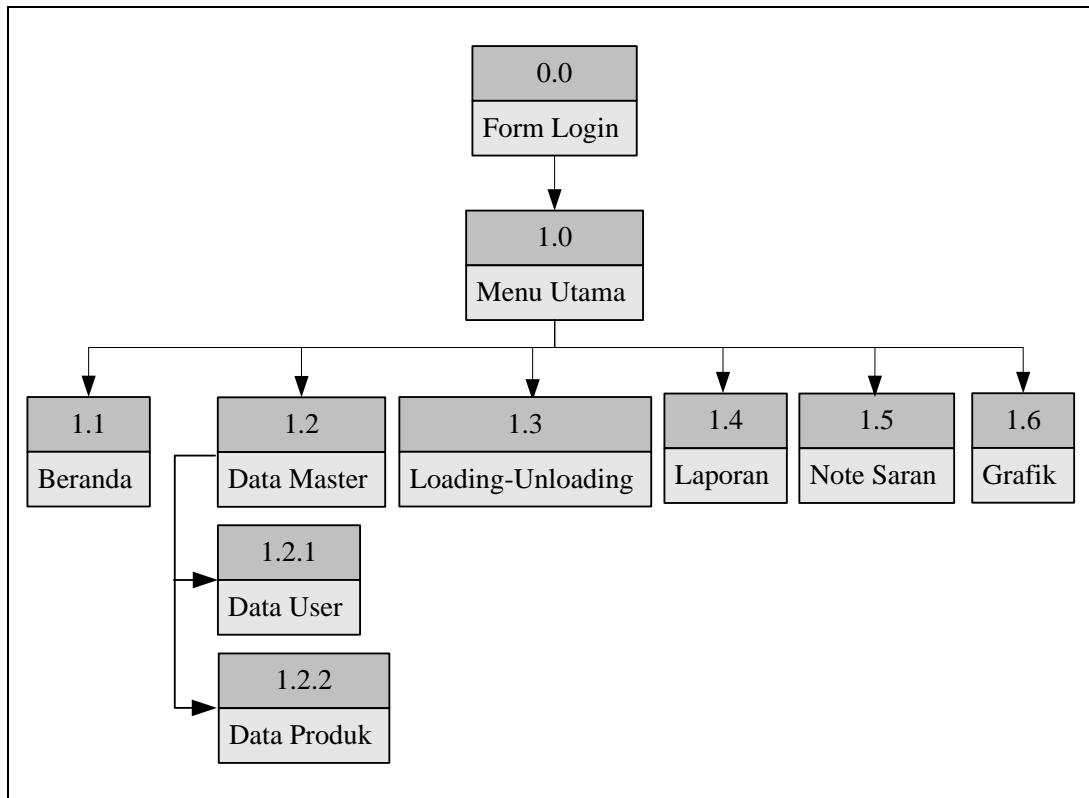


Gambar V.27 Program Logic Flowchart Aplikasi Sistem Informasi Pengendalian Kualitas Accumulator

(Sumber: Hasil Analisis Data, 2016)

5.7.2 Hierarchy plus Input-Process-Output (HIPO)

Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hierarkis. Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya. Perancangan HIPO aplikasi usulan sistem pengendalian kualitas *accumulator* dapat dilihat pada Gambar V.28 sebagai berikut:



Gambar V.28 HIPO Sistem Informasi Pengendalian Kualitas *Accumulator* Usulan
(Sumber: Hasil Analisis Data, 2016)

5.7.3 Perancangan *Interface* Program


Rancangan *interface* dari program pengendalian kualitas produk ini adalah sebagai berikut:

1. *Interface Form Login*

Interface form login adalah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan nama pengguna dan sandi yang benar. Rancangan *interface form login* dapat dilihat pada Gambar V.29.

Q

@



PT Kurnia Manunggal Sejahtera

Gambar V.29 *Interface Form Login*
(Sumber: Hasil Analisis Data, 2016)

2. *Interface Form Menu Utama*

Interface form menu utama adalah tampilan awal pada aplikasi yang berisi enam menu. Menu tersebut adalah Beranda, Data Master, *Loading-Unloading*, Laporan, *Note* Saran dan Grafik. Rancangan *interface form* menu utama dapat dilihat pada Gambar V.30 berikut:

 -KUMASE- 	<input type="button" value="Keluar"/>
Beranda	Sistem Informasi Pengendalian Kualitas Accumulator
Data Master	Selamat Datang
Loading-Unloading	
Laporan	Hallo
Note Saran	Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved. Version 1.0
Grafik	

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

3. Interface Form Data User

Interface form data user adalah form yang digunakan untuk mengelola data user. Rancangan interface form data user dapat dilihat pada Gambar V.31 berikut:

-KUMASE-							Keluar
Beranda	Sistem Informasi Pengendalian Kualitas Accumulator						
Data Master	Data User						
User	+ Tambah User						
Produk	No	Id User	Nama Lengkap	Username	Password	Jabatan	Aksi
Laporan							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved.						Version 1.0	

Gambar V.31 Interface Form Data User
(Sumber: Hasil Analisis Data, 2016)

4. Interface Form Tambah Data User

Interface form tambah data user adalah form yang digunakan untuk menambahkan data user. Rancangan interface dari form tambah data user dapat dilihat pada Gambar V.32 berikut:

-KUMASE-							Keluar
Beranda	Sistem Informasi Pengendalian Kualitas Accumulator						
Data Master	Data User						
User	+ Tambah User						
Produk	No	Id User	Nama Lengkap	Username	Password	Jabatan	Aksi
Laporan							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
							Ubah Hapus
Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved.						Version 1.0	

Gambar V.32 Interface Form Tambah Data User

(Sumber: Hasil Analisis Data, 2016)

5. *Interface Form* Data Produk

Interface form data produk adalah form yang digunakan untuk mengelola data produk. Rancangan *interface* dari *form* data produk dapat dilihat pada Gambar V.33 berikut:

The screenshot shows a web application interface titled "|-KUMASE-|". On the left is a navigation menu with options: Beranda, Data Master, User, Produk (highlighted), and Laporan. The main content area is titled "Sistem Informasi Pengendalian Kualitas Accumulator" and contains a "Data Produk" section. This section includes a "+Tambah Produk" button and a table with columns: No, Id Produk, Nama Produk, Tipe, and Aksi. The Aksi column contains "Ubah | Hapus" for each row. At the bottom, there is a copyright notice: "Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved." and "Version 1.0". A "Keluar" button is located in the top right corner.

No	Id Produk	Nama Produk	Tipe	Aksi
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus
				Ubah Hapus

Gambar V.33 *Interface Form* Data Produk
(Sumber: Hasil Analisis Data, 2016)

6. *Interface Form* Tambah Data Produk

Interface form tambah data produk adalah *form* yang digunakan untuk memasukkan data produk. Rancangan *interface* dari *form* tambah data produk dapat dilihat pada Gambar V.34 berikut:

|-KUMASE-| Keluar

Sistem Informasi Pengendalian Kualitas Accumulator

Tambah Produk

Id Produk

Nama Produk

Tipe

Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved. Version 1.0

Gambar V.34 Interface Tambah Data Produk
(Sumber: Hasil Analisis Data, 2016)

7. Interface Form Loading-Unloading

Interface form loading-unloading adalah *form* yang digunakan untuk melihat data *loading-unloading*. Rancangan *interface* dari *form loading-unloading* dapat dilihat pada Gambar V.35 berikut:

|-KUMASE-| Keluar

Sistem Informasi Pengendalian Kualitas Accumulator

Data Loading-Unloading

No	No. Lot	Nama Part	Tipe	Penginput	Tanggal	NG						Total		Status	Aksi
						Tipis	Meler	Kotor	Serabut	Lecet	Bintik	NG	OK		
														Disetujui	Detail
														Disetujui	Total
														Disetujui	Detail
														Disetujui	Total
														Menunggu Persetujuan	Detail
															Total

Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved. Version 1.0

Gambar V.35 Interface Form Loading-Unloading
(Sumber: Hasil Analisis Data, 2016)

8. Interface Form Tambah Loading-Unloading

Interface form tambah loading-unloading adalah *form* yang digunakan untuk menambah data *loading-unloading*. Rancangan *interface* dari *form tambah loading-unloading* dapat dilihat pada Gambar V.36 berikut:

-KUMASE-		Keluar
Beranda	Sistem Informasi Pengendalian Kualitas Accumulator	
Loading-Unloading	<h3>Tambah Data Loading-Unloading</h3> <p>Tanggal <input type="text"/></p> <p>Nama Produk <input type="text"/></p> <p style="text-align: center;"> <input type="button" value="Kembali"/> <input type="button" value="Simpan"/> </p>	
Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved.		Version 1.0

Gambar V.36 Interface Form Tambah Loading-Unloading
(Sumber: Hasil Analisis Data, 2016)

9. *Interface Form Detail Loading-Unloading*

Interface form detail loading-unloading adalah form yang digunakan untuk melihat data detail *loading-unloading*. Rancangan *interface* dari form detail *loading-unloading* dapat dilihat pada Gambar V.37 berikut:

-KUMASE-		Keluar																																																																																																																																																									
Beranda	Sistem Informasi Pengendalian Kualitas Accumulator																																																																																																																																																										
Loading-Unloading	<p>Detail Loading-Unloading</p> <p>Tambah Detail <input type="button" value="Kembali"/></p> <p>Note Proses: F: Material Fresh R1: Repair Material 1x R2: Repair Material 2x</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th rowspan="2">No</th> <th colspan="6">Loading</th> <th colspan="6">Unloading</th> <th rowspan="2">Keterangan</th> <th rowspan="2">Aksi</th> </tr> <tr> <th>Jam</th> <th>No.Lot</th> <th>Nama Part</th> <th>Tipe</th> <th>Proses</th> <th>Jumlah</th> <th>Jam</th> <th colspan="4">NG</th> <th>Jumlah</th> </tr> <tr> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th>Tipis</th> <th>Meler</th> <th>Kotor</th> <th>Serabut</th> <th>Lecet</th> <th>Bintik</th> <th>NG</th> <th>OK</th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Ubah</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Ubah</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Hapus</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Ubah</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Hapus</td> </tr> </tbody> </table>		No	Loading						Unloading						Keterangan	Aksi	Jam	No.Lot	Nama Part	Tipe	Proses	Jumlah	Jam	NG				Jumlah									Tipis	Meler	Kotor	Serabut	Lecet	Bintik	NG	OK																				Ubah																		Hapus																		Ubah																		Hapus																		Ubah																		Hapus
No	Loading						Unloading						Keterangan	Aksi																																																																																																																																													
	Jam	No.Lot	Nama Part	Tipe	Proses	Jumlah	Jam	NG				Jumlah																																																																																																																																															
								Tipis	Meler	Kotor	Serabut	Lecet	Bintik	NG	OK																																																																																																																																												
																	Ubah																																																																																																																																										
																	Hapus																																																																																																																																										
																	Ubah																																																																																																																																										
																	Hapus																																																																																																																																										
																	Ubah																																																																																																																																										
																	Hapus																																																																																																																																										
Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved.		Version 1.0																																																																																																																																																									

Gambar V.37 Interface Form Detail Loading-Unloading
(Sumber: Hasil Analisis Data, 2016)

10. *Interface Form Quality Monthly Report*

Interface form quality monthly report adalah form yang digunakan untuk mengelola *quality monthly report*. Rancangan *interface* dari *quality monthly report* dapat dilihat pada Gambar V.38 berikut:

-KUMASE-														Keluar			
Beranda		Sistem Informasi Pengendalian Kualitas Accumulator															
Data Master		Quality Monthly Report															
Laporan		Tgl. Awal <input type="text" value="mm/dd/yyyy"/> Tgl. Akhir <input type="text" value="mm/dd/yyyy"/> <input type="button" value="Cari"/> <input type="button" value="Cetak"/>															
No	No. Lot	Nama Part	Tipe	Penginput	Tanggal	NG					Jumlah		Status				
						Tipis	Meler	Kotor	Serabut	Lecet	Bintik	NG		OK			
Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved.														Version 1.0			

Gambar V.38 *Interface Form Quality Monthly Report*
(Sumber: Hasil Analisis Data, 2016)

11. *Interface Form Note Saran*

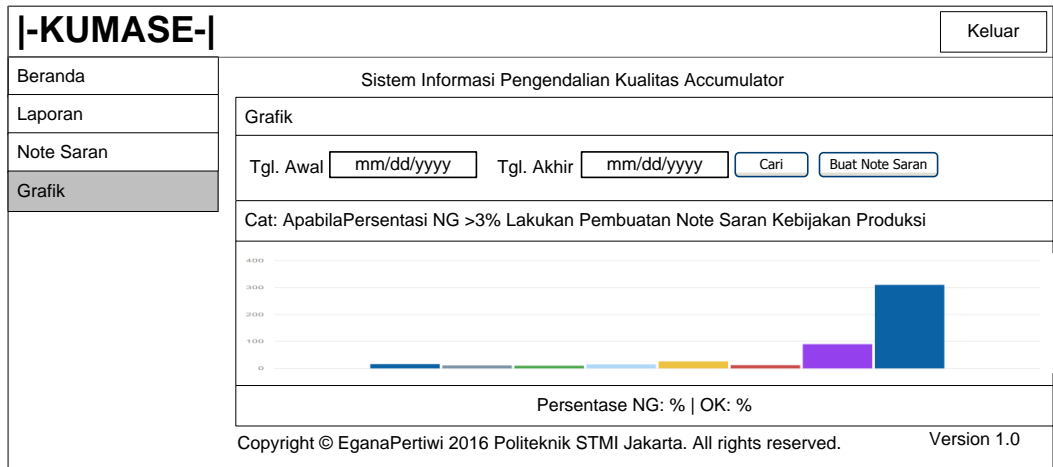
Interface form note saran adalah *form* yang digunakan untuk membuat *note saran* untuk kebijakan produksi selanjutnya. Rancangan *interface note saran* dapat dilihat pada Gambar V.39 berikut:

-KUMASE-														Keluar			
Beranda		Sistem Informasi Pengendalian Kualitas Accumulator															
Laporan		Note Saran															
Note Saran		<input type="text" value="mm/dd/yyyy"/> <input type="button" value="Cari"/>															
Grafik																	
No	Tanggal	Divisi	Saran					Aksi									
									<input type="button" value="Cetak"/> <input type="button" value="Hapus"/>								
Copyright © EganaPertiwi 2016 Politeknik STMI Jakarta. All rights reserved.														Version 1.0			

Gambar V.39 *Interface Form Note Saran*
(Sumber: Hasil Analisis Data, 2016)

12. *Form Grafik*

Form grafik adalah *form* yang digunakan untuk melihat grafik dapat dilihat pada Gambar V.40 berikut:



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

13. *Form* Buat Note Saran

Form grafik adalah *form* yang digunakan untuk membuat *note* saran dapat dilihat pada Gambar V.41 berikut:

Gambar V.41 *Interface Form Buat Note Saran*
(Sumber: Hasil Analisis Data, 2016)

5.8 Implementasi Sistem

Tahap ini adalah tahap pengkodean program menggunakan bahasa pemrograman PHP dengan menggunakan *Adobe Dreamweaver* dan *Notepad++* sebagai aplikasi editor. Setiap *interface* berisikan kode program agar program dapat dijalankan sesuai dengan fungsinya. Untuk mendukung kebutuhan

implementasi sistem diperlukan suatu spesifikasi perangkat lunak (*software*) maupun perangkat keras (*hardware*). Adapun spesifikasinya sebagai berikut:

1. Analisis Kebutuhan *Software*
 - a. Sistem Operasi : *Microsoft Windows 7*
 - b. *Database Server* : *MySQL versi 5.5.36*
 - c. Bahasa Pemrograman : *PHP 5.4.27*
 - d. *Web Browser* : *Mozilla Firefox, Google Chrome*
2. Analisis Kebutuhan *Hardware*
 - a. *Processor* : *Minimal Processor Pentium IV*
 - b. RAM : *Minimal RAM 512 MB*
 - c. *Harddisk* : *Minimal Harddisk 64 GB*
 - d. *Mouse, Keyboard, Monitor* sebagai peralatan antar muka.
 - e. *Printer* sebagai media pencetakan dokumen.

BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

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

1. Dengan sistem informasi pengendalian kualitas ini memudahkan bagian *Quality Control* divisi *Painting* dalam menginput dan mengolah laporan *loading-unloading* secara otomatis.
2. Dengan menggunakan *database* sebagai media penyimpanan, proses pencarian data yang dahulu memakan waktu lama sekarang bisa dilakukan dengan lebih cepat.
3. Aplikasi sistem informasi pengendalian kualitas *accumulator* ini memudahkan bagian *Quality Control Office* dalam mengelola data *quality monthly report* bagian *Quality Control* divisi *Painting*.

6.2. Saran

Saran yang diberikan untuk pengembangan sistem informasi pengendalian kualitas produk baterai ini selanjutnya adalah sebagai berikut:

1. Untuk penerapan sistem baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi penggunaan sistem ini kepada bagian yang terkait.
2. Diharapkan adanya pengembangan lebih lanjut pada sistem informasi pengendalian kualitas *accumulator* pada proses selain divisi *Painting* seperti divisi *Stamping* dan divisi Konstruksi Baja supaya proses pengolahan data dapat lebih informatif.

DAFTAR PUSTAKA

- Amsyah, Zulkifli, 2005, *Manajemen Sistem Informasi*, PT Gramedia Pustaka Utama, Jakarta.
- Assauri, Sofjan, 2004, *Manajemen Operasi dan Produksi*, LP FE UI Dwiwinarno, Jakarta.
- Badudu, J.S, 2003, *Kamus Kata-Kata Serapan Asing dalam Bahasa Indonesia*, Jakarta: PT Kompas Media Nusantara.
- Ginting, Rosnani, 2007, *Sistem Produksi*, Graha Ilmu, Yogyakarta.
- Heizer, Jay., Render, Barry, 2006, *Operations Management (Manajemen Operasi)*, Salemba Empat, Jakarta.
- Jogiyanto, HM. 2001, *Analisis Perancangan Sistem Informasi*, Yogyakarta: Penerbit Andi.
- Jogiyanto, 2005, *Analisis dan Desain Sistem Informasi*, Yogyakarta: Penerbit Andi.
- Kadir, Abdul, 2014, *Pengenalan Sistem Informasi*, Edisi Revisi, Yogyakarta: Andi Offset.
- Koontz, Harold., O'donell, Cyrill, 2007, *Pengertian Pengendalian*, <http://elib.unikom.ac.id/download.php?id=145068>. (Tanggal Akses: 20 Juli 2016)
- McLeod, Raymond., S, George. 2011. *Sistem Informasi Manajemen*. Jakarta: PT. Indeks.
- Render, Barry., Heizer, Jay, 2001 *Pengertian Kualitas*, <http://pengertian-pengertian-info.blogspot.co.id/2015/09/pengertian-dan-tujuan-pengendalian.html>. (Tanggal Akses: 20 Juli 2016)
- Rosa, A.S., Shalahuddin, M, 2014, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Satzinger, 2005, *OOAD*, <http://www.slideshare.net/safaiputrapulau/2011200041a-ksi-bab2001>. (Tanggal Akses: 8 Agustus 2016)
- Siagian, Sondang P., 2007, http://elib.unikom.ac.id/files/disk1/538/jbptunikompp-gdl-iraquraisy-26896-3-unikom_i-i.pdf. (Tanggal Akses: 20 Juli 2016)

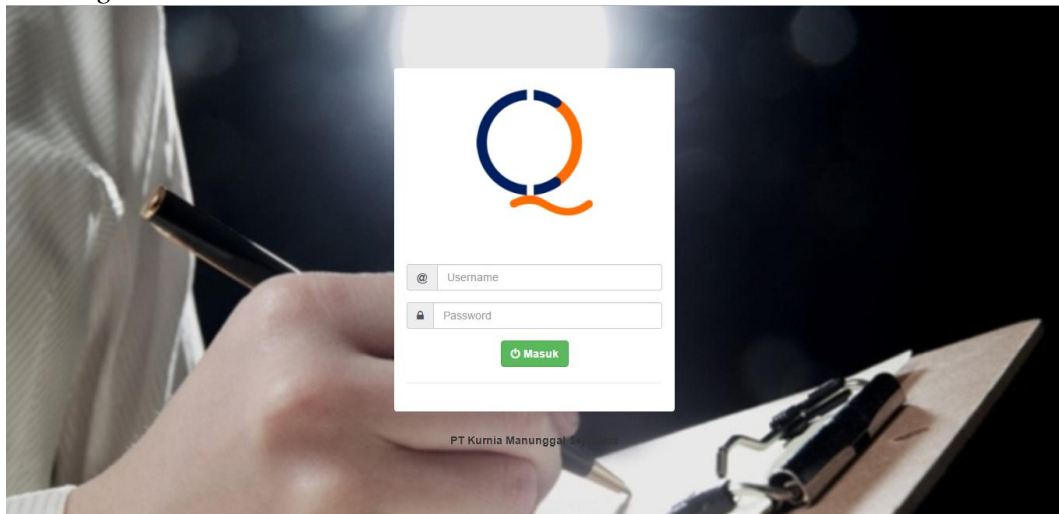
- Sitorus, Lamhot. 2015. *Algoritma dan Pemrograman*. Yogyakarta: Andi Offset.
- Sommerville, Ia,. 2003, *Software Engineering Rekayasa Perangkat Lunak*. Jakarta: Erlangga.
- Sutabri, Tata, 2012, *Analisis Sistem Informasi*, ANDI Yogyakarta, Yogyakarta.
- Sutaji, Deni, 2012, *Sistem Inventory Mini Market dengan PHP & Jquery*, Lokomedia, Yogyakarta.
- Tim Dosen, 2009, *Buku Ajar Teknik Pengendalian Kualitas, Universitas Wijaya Putra, Surabaya*.
- Welling, L., Thomson, L, 2003, *PHP and MySql Web Development, Second Edition*, Sams Publishing, Indianapolis.
- Wahana Komputer, 2010, *Panduan belajar MySQL Database Server*, Jakarta: Media Kita.
- Yakub. 2012. *Pengantar Sistem Informasi*, Yogyakarta: Graha Ilmu.
- Yanto, Robi. 2016. *Manajemen Basis Data Menggunakan MySQL*. Yogyakarta: Deepublish.
- _____, *Sejarah Baja Kurnia*, <http://bajakurnia.co.id/pages/detail/SEJARAH>. (Tanggal Akses; 23 Maret 2016)
- _____, *Pengertian Accumulator*, <https://id.wikipedia.org/wiki/Akumulator>. (Tanggal Akses: 6 April 2016)
- _____, *QC 7 Tools*, <http://thesis.binus.ac.id/Asli/Bab2/2007-1-00262-TI%20-Bab%202.pdf>. (Tanggal Akses: 6 April 2016)

LAMPIRAN

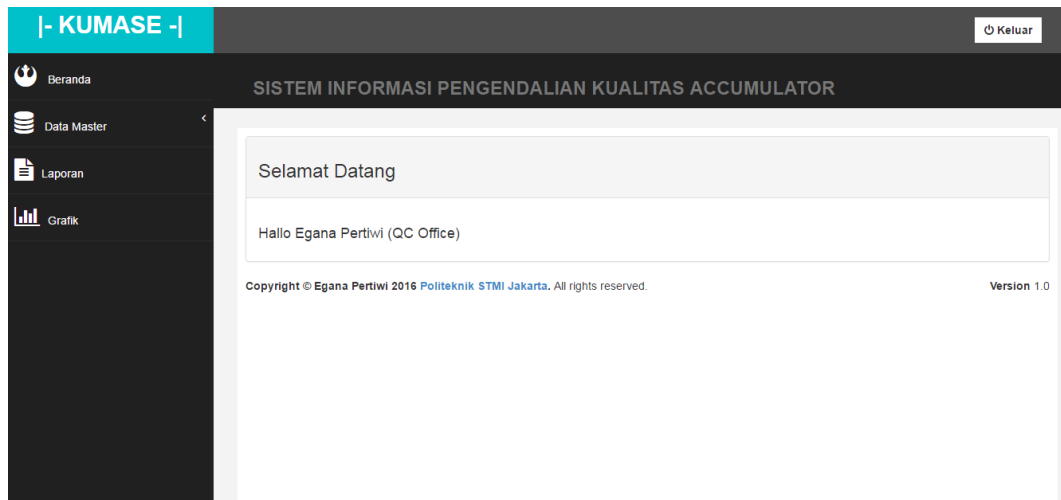
LAMPIRAN A

TAMPILAN PROGRAM

1. Login



2. Menu Utama



3. Data User

The screenshot shows the 'Data User' page in the KUMASE system. The page has a dark sidebar with navigation options: Beranda, Data Master, and Laporan. The main content area is titled 'SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR' and 'Data User'. There is a 'Tambah User' button and a table with the following data:

No	ID User	Nama Lengkap	Username	Password	Jabatan	Aksi
1	1	Egana Peritiwi	gena	12345	QC Office	<input type="checkbox"/> Ubah <input type="button" value="Hapus"/>
2	2	Annisa Setiawati	barbara	12345	Leader Produksi	<input type="checkbox"/> Ubah <input type="button" value="Hapus"/>
3	3	Riswandi	ris	12345	Operator	<input type="checkbox"/> Ubah <input type="button" value="Hapus"/>
4	4	Haura Artha	haura	12345	Manajer Produksi	<input type="checkbox"/> Ubah <input type="button" value="Hapus"/>
5	6	kaka	kaka	12345	Foreman	<input type="checkbox"/> Ubah <input type="button" value="Hapus"/>

4. Tambah Data User

The screenshot shows the 'Tambah User' form in the KUMASE system. The page has a dark sidebar with navigation options: Beranda, Data Master, and Laporan. The main content area is titled 'SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR' and 'Tambah User'. The form includes the following fields and buttons:

- Nama Lengkap:
- Username:
- Password:
- Jabatan:
-

Copyright © Egana Peritiwi 2016 Politeknik STMI Jakarta. All rights reserved. Version 1.0

5. Data Produk

|- KUMASE -| Keluar

SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR

Data Produk

[Tambah Produk](#)

No	ID Produk	Nama Produk	Tipe	Aksi
1	AC001	Accumulator	White	Ubah Hapus
2	AC002	Accumulator	Blue Ekspor	Ubah Hapus
3	AC003	Accumulator	Blue Lokal	Ubah Hapus

Copyright © Egana Pertiwi 2016 Politeknik STMI Jakarta. All rights reserved. Version 1.0

6. Tambah Data Produk

|- KUMASE -| Keluar

SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR

Tambah Produk

ID Produk

Nama Produk

Tipe

[Kembali](#) [Simpan](#)

Copyright © Egana Pertiwi 2016 Politeknik STMI Jakarta. All rights reserved. Version 1.0

7. Loading-Unloading

|- KUMASE -| Keluar

SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR

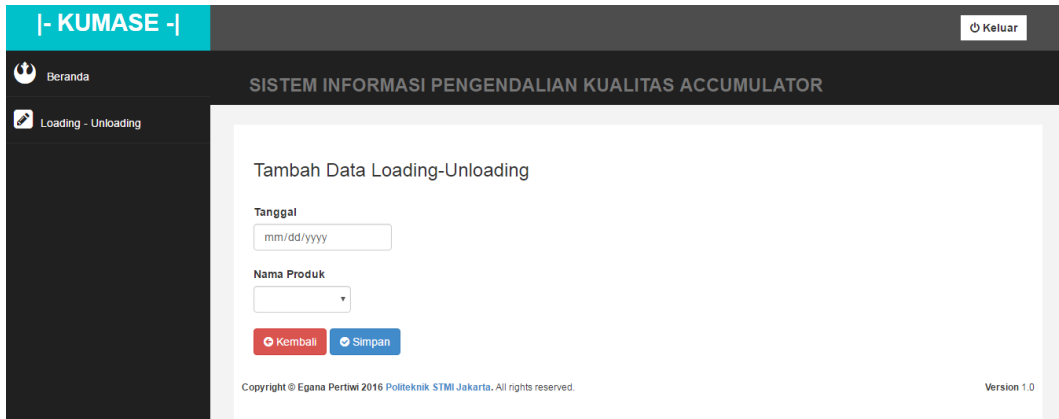
Data Loading-Unloading

[Tambah Loading-Unloading](#)

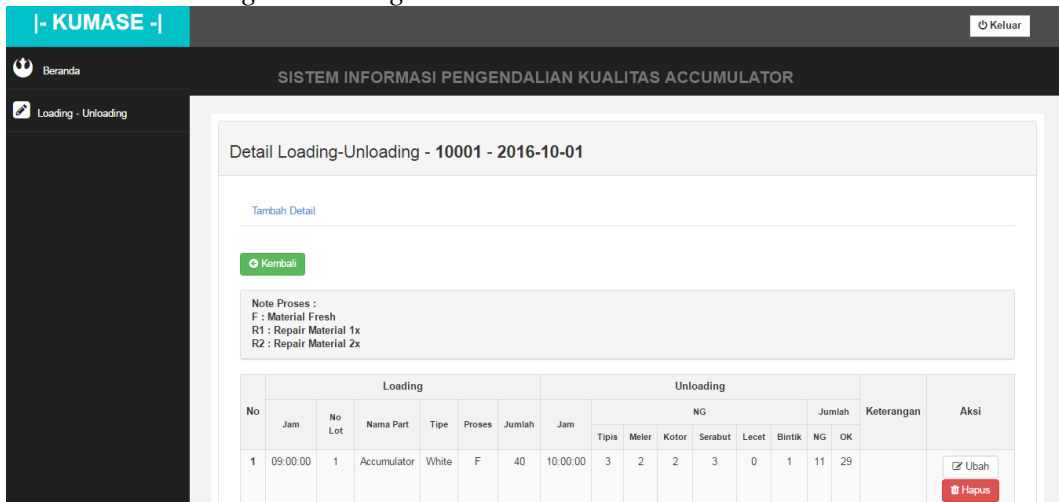
mm/dd/yyyy [CARI](#)

No	No LO	Nama Part	Tipe	Penginput	Tanggal	NG						Total		Status	Aksi
						Tipis	Meler	Kotor	Serabut	Lecet	Bintik	NG	OK		
1	10001	Accumulator	White	Riswandi	2016-10-01	20	15	14	12	21	8	90	150	Disetujui	Detail Total
2	10002	Accumulator	Blue Ekspor	Riswandi	2016-10-02	9	15	6	8	10	3	51	169	Disetujui	Detail Total

8. Tambah Loading-Unloading



9. *Detail Loading-Unloading*



10. *Quality Monthly Report*

- KUMASE - Keluar

Beranda

SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR

Data Master

Laporan

Data Loading-Unloading

Tgl. Awal Tgl. Akhir CARI CETAK

No	No LO	Nama Part	Tipe	Pengepuk	Tanggal	NG						Total		Status
						Tipis	Meler	Kotor	Serabut	Lecet	Bintik	NG	OK	
1	10001	Accumulator	White	Riswandi	2016-10-01	20	15	14	12	21	8	90	150	Disetujui
2	10004	Accumulator	White	Riswandi	2016-10-04	9	3	8	14	7	5	46	154	Disetujui
3	10007	Accumulator	White	Riswandi	2016-10-07	9	16	15	18	16	8	82	118	Disetujui
Sub Total						38	34	37	44	44	21	218	422	
4	10003	Accumulator	Blue Lokal	Riswandi	2016-10-03	14	1	7	8	8	3	41	159	Disetujui
5	10006	Accumulator	Blue Lokal	Riswandi	2016-10-06	5	4	16	7	5	0	37	163	Disetujui
6	10009	Accumulator	Blue Lokal	Riswandi	2016-10-09	8	2	4	1	2	5	22	98	Disetujui
Sub Total						27	7	27	16	15	8	100	420	
7	10002	Accumulator	Blue Ekspor	Riswandi	2016-10-02	9	15	6	8	10	3	51	189	Disetujui
8	10005	Accumulator	Blue Ekspor	Riswandi	2016-10-05	6	4	2	12	17	4	45	155	Disetujui

11. Note Saran

- KUMASE - Keluar

Beranda

SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR

Laporan

Note Saran

Grafik

Note Saran

CARI

No	No Note	Tanggal	Pembuat	Untuk Periode	Divisi	Saran	Aksi
1	10001	2016-10-02	Egana Peritwi		Painting	Perhatikan Proses pengecatan Jangan Sampai Banyak Produk Reject yang Disebabkan Keteledoran Operator	Cetak Hapus
2	10002	2016-10-31	Annisa Setiawati	2016-10-01 s.d. 20-10-30	Painting	Pehatkan Proses pengecatan pada unit produk karna banyak ditemukan produk reject yang diakibatkan cat yang meler	Cetak Hapus
3	10003	2016-03-31	Annisa Setiawati	2016-10-07 s.d. 2016-10-2	Painting	jslsl	Cetak Hapus

12. Grafik

- KUMASE - Keluar

Beranda

SISTEM INFORMASI PENGENDALIAN KUALITAS ACCUMULATOR

Laporan

Note Saran

Grafik

Grafik

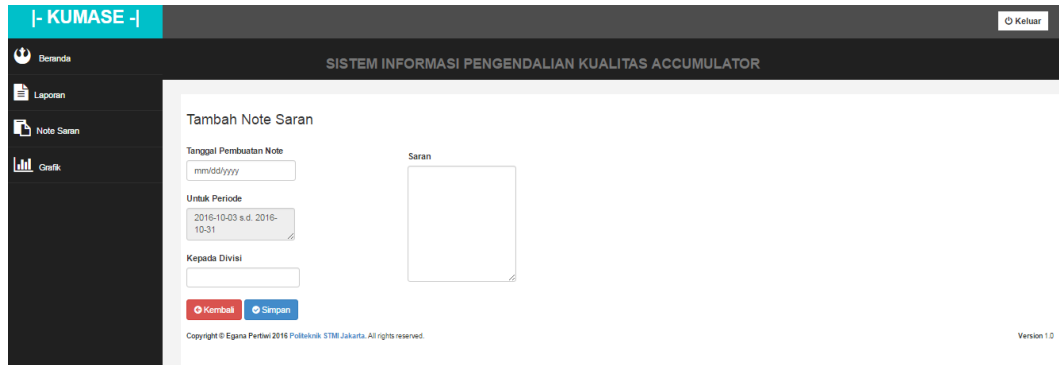
Tgl. Awal Tgl. Akhir CARI Buat Note Saran

Cat: Apabila Persentase NG > 3% Lakukan Pembuatan Note Saran Kebijakan Produksi

Persentase NG : 34.0625% | Persentase OK : 65.9375%

Legend: Tipis, Meler, Serabut, Lecet, Kotor, Bintik, NG, OK

13. Buat Note Saran



LAMPIRAN B

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' />
<style type="text/css">
body { background: url(images/1.JPG) !important; }
</style>
</head> <body>
<div class="container"> <div class="row text-center ">
<div class="col-md-12">
<br /><br /> <h2></h2> <br /> </div></div>
<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-transparent">
<div class="panel-heading" align="center">
<strong></strong>

```

```

</div>
<div class="panel-body">
<form role="form" method="POST">
<br />
<div class="form-group input-group">
<span class="input-group-addon"><b>@</b></span>
<input type="text" class="form-control" placeholder="Username" 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"
name="pass" />
</div>
<div align="center"><button type="submit" name="melogin" class="btn btn-
success"><i class="fa fa-power-off"></i><b> Masuk</b></button></div>
<hr /> </form>
<?php
if(isset($_POST['melogin']))
{
$scoballogin = $pengguna->login_pengguna($_POST['uid'],$_POST['pass']);
if($scoballogin)
{
echo"<script>alert('Yes! Login Berhasil');</script>";
echo"<script>>window.location='index.php';</script>";
}
Else
{
echo"<script>alert('Username atau Password Yang Anda Masukan
Salah!');</script>";
echo"<script>>window.location='login.php';</script>";
}}?>
</div> </div> </div> </div>
<div class="row text-center ">
<div class="col-md-12">
<h5><b>PT Kurnia Manunggal Sejahtera</b></h5>
<br /> </div> </div> </div>
<!-- SCRIPTS -AT THE BOTOM 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. Index

```
<?php
include 'class.php';
if(empty($_SESSION['Id_User']))
{
    echo "<script>alert('login dulu');</script>";
    echo "<script>>window.location='login.php';</script>";
}
if(isset($_GET['aksi']))
{
    if($_GET['aksi']=='logout')
    {
        $pengguna->logout_pengguna();
        echo "<script>alert('Anda Telah Keluar');</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 Quality Control Unit Accumulator</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"><b>|- KUMASE -|</b></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-default square-
btn-adjust"><i class="fa fa-power-off"></i><b> Keluar</b></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-ra fa-
2x"></i> Beranda </a>
</li>
<li>
<?php
if(($_SESSION['Jabatan']=="QC Office")){
?>
<li>
<a href="#"><i class="fa fa-database fa-2x"></i> Data
Master<span class="fa arrow"></span></a>
<ul class="nav nav-second-level">
<li>
<a href="index.php?halaman=datauser">User</a>
</li>
<li>
<a href="index.php?halaman=dataproduk">Produk</a>
</li>
</ul>
</li>
<li>
<?php
}
?>
<?php
if(($_SESSION['Jabatan']=="Operator")){
?>
<li>

```

```
        <a href="index.php?halaman=dataloun"><i class="fa fa-pencil-square fa-2x"></i>Loading - Unloading<span></span></a>
    </li>
```

```
<?php
}
?>
```

```
        <?php
        if(($_SESSION['Jabatan']=="QC Office")){
        ?>
        <li>
        <a href="index.php?halaman=lounqco"><i class="fa fa-file-text fa-2x"></i>Laporan<span></span></a>
        </li>
```

```
        <li>
        <a href="index.php?halaman=grafik"><i class="fa fa-bar-chart fa-2x"></i>Grafik<span></span></a>
```

```
    </li>
    <?php
    }
    ?>
```

```
        <?php
        if(($_SESSION['Jabatan']=="Leader Produksi")){
        ?>
        <li>
        <a href="index.php?halaman=lounview"><i class="fa fa-file-text fa-2x"></i>Laporan<span></span></a>
    </li>
```

```
<?php
}
?>
```

```
        <?php
        if(($_SESSION['Jabatan']=="Manajer Produksi")){
        ?>
        <li>
        <a href="index.php?halaman=lounview"><i class="fa fa-file-text fa-2x"></i>Laporan<span></span></a>
    </li>
```

```
<?php
```

```
}
```

```
?>
```

```
<?php
if(($_SESSION['Jabatan']=="Leader Produksi")){
?>
<li>
```

```

        <a href="index.php?halaman=notesaran"><i class="fa fa-clipboard
fa-2x"></i>Note Saran<span></span></a>
    </li>
    <?php
    }
    ?>

                                <?php
    if(($_SESSION['Jabatan']=="Foreman")){
    ?>
    <li>
        <a href="index.php?halaman=lounfor"><i class="fa fa-pencil-
square fa-2x"></i>Loading-Unloading<span></span></a>
    </li>
    <?php
    }
    ?>
</div>
</nav>
<!-- /. NAV SIDE -->
    <div class="content-header" align="center">
        <h1>
            <small><b>SISTEM INFORMASI PENGENDALIAN KUALITAS
ACCUMULATOR</b></small>
        </h1>
    </div>
    <div id="page-wrapper" >
        <div id="page-inner">
            <?php
            include "connect.php";
            if(isset($_GET['halaman']))
            {
                if($_GET['halaman']=='datauser')
                {
                    include 'datauser.php';
                }
                elseif ($_GET['halaman']=='tambahuser')
                {
                    include 'user/tambahuser.php';
                }
                elseif($_GET['halaman']=='saveuser')
                {
                    include 'user/save.php';
                }
                elseif ($_GET['halaman']=='ubahdatauser')
                {
                    include 'user/ubahuser.php';

```

```
}
elseif ($_GET['halaman']=='ubahuser')
{
    include 'user/ubah.php';
}
elseif($_GET['halaman']=='hapususer')
{
    include 'user/hapus.php';
}
elseif($_GET['halaman']=='dataproduk')
{
    include 'dataproduk.php';
}
elseif($_GET['halaman']=='tambahproduk')
{
    include 'produk/tambahproduk.php';
}
elseif($_GET['halaman']=='saveproduk')
{
    include 'produk/save.php';
}
elseif($_GET['halaman']=='ubahdataproduk')
{
    include 'produk/ubahproduk.php';
}
elseif($_GET['halaman']=='ubahproduk')
{
    include 'produk/ubah.php';
}
elseif($_GET['halaman']=='hapusproduk')
{
    include 'produk/hapus.php';
}
elseif($_GET['halaman']=='dataloun')
{
    include 'dataloun.php';
}
elseif ($_GET['halaman']=='tambahloun')
{
    include 'loun/tambahloun.php';
}
elseif($_GET['halaman']=='saveloun')
{
    include 'loun/save.php';
}
elseif ($_GET['halaman']=='ubahdataloun')
```

```
{
    include 'loun/ubahloun.php';
}
elseif ($_GET['halaman']=='ubahloun')
{
    include 'loun/ubah.php';
}
elseif($_GET['halaman']=='hapusloun')
{
    include 'loun/hapus.php';
    }
    elseif($_GET['halaman']=='inputtotal')
{
    include 'loun/inputtotal.php';
    }
    elseif($_GET['halaman']=='savetotal')
{
    include 'loun/savetotal.php';
    }
    elseif($_GET['halaman']=='dataloading')
{
    include 'dataloading.php';
}
elseif ($_GET['halaman']=='tambahloading')
{
    include 'loading/tambahloading.php';
}
elseif($_GET['halaman']=='saveloading')
{
    include 'loading/save.php';
}
elseif ($_GET['halaman']=='ubahdataloading')
{
    include 'loading/ubahloading.php';
}
elseif ($_GET['halaman']=='ubahloading')
{
    include 'loading/ubah.php';
}
elseif($_GET['halaman']=='hapusloading')
{
    include 'loading/hapus.php';
    }
    elseif($_GET['halaman']=='lounview')
{
    include 'datalounview.php';
}
```

```

    }
    elseif($_GET['halaman']=='lounqco')
{
    include 'datalounqco.php';
    }
    elseif($_GET['halaman']=='lounfor')
{
    include 'datalounfor.php';
    }
    elseif($_GET['halaman']=='ubahdataloun')
{
    include 'loun/ubahloun.php';
    }
    elseif($_GET['halaman']=='dataloadingfor')
{
    include 'dataloadingfor.php';
    }
    elseif($_GET['halaman']=='dataloadingqco')
{
    include 'dataloadingqco.php';
    }
    elseif($_GET['halaman']=='notesaran')
{
    include 'notesaran.php';
    }
    elseif($_GET['halaman']=='tambahnote')
{
    include 'note/tambahnote.php';
    }
    elseif($_GET['halaman']=='savenote')
{
    include 'note/save.php';
    }
    elseif($_GET['halaman']=='ubahdatanote')
{
    include 'note/ubahnote.php';
    }
    elseif($_GET['halaman']=='ubahnote')
{
    include 'note/ubah.php';
    }
    elseif($_GET['halaman']=='hapusnote')
{
    include 'note/hapus.php';
    }
    elseif($_GET['halaman']=='grafik')

```

```

    {
        include 'grafik.php';
    }
        elseif($_GET['halaman']=='hapusnote')
    {
        include 'note/hapus.php';
    }
        elseif($_GET['halaman']=='cetaksaran')
    {
        include 'cetak-saran.php';
    }
        elseif ($_GET['halaman']=='dataloadingacc')
    {
        include 'dataloadingacc.php';
    }
        elseif ($_GET['halaman']=='home')
    {
        include 'home.php';
    }
    }
else
{
    include 'home.php';
}
?>
        <footer class="main-footer">
<div class="pull-right hidden-xs">
    <b>Version</b> 1.0
</div>
    <strong>Copyright &copy; Egana Pertiwi 2016<a href="http://stmi.ac.id">
Politeknik STMI Jakarta</a>.</strong> All rights reserved.
</footer>
    </div>
        <!-- /. PAGE INNER -->

</div>
    <!-- /. PAGE WRAPPER -->
</div>
    <!-- /. WRAPPER -->
<!-- SCRIPTS -AT THE BOTOM 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>

```

3. Home

```

<div class="panel panel-default">
  <div class="panel-heading">
    <h3> Selamat Datang</h3>
  </div>
  <div class="panel-body">
    <ul class="nav nav-pills">
      <li class="active">
    </li> </ul>
  </div>
</div>
<div class="tab-content">
  <div class="tab-pane fade active in" id="home">
    <h4>Hallo <?php echo $_SESSION['Nama_Lengkap']; ?>
    (<?php echo $_SESSION['Jabatan']; ?>) </h4>
  </div> </div> </div> </div>

```

4. Data User

```

<?php
include('connect.php');
$no=0;
?>
<div class="row">
  <div class="col-md-12">
    <!-- Advanced Tables -->
    <div class="panel panel-default">
      <div class="panel-heading">
        <h3> Data User </h3>
      </div>
      <div class="panel-body">
        <a href="index.php?halaman=tambahuser" class="btn btn-
primary"><i class="fa fa-plus"></i> Tambah User </a>
      </div>
    </div>
  </div>
  <div class="table-responsive">
    <br>
    <table class="table table-striped table-bordered table-hover" id="dataTables-
example">
      <thead>
        <tr bgcolor="#f2f2f2">
          <th>No</th>
          <th>ID User</th>

```

```

        <th>Nama Lengkap</th>
            <th>Username</th>
        <th>Password</th>
            <th>Jabatan</th>
            <th>Aksi</th>
    </tr>
</thead>
<tbody>
    <?php
    if(isset($_POST['button'])){
        $tcari=$_POST['tcari'];
        $sus = mysql_query("SELECT * FROM user WHERE Nama_Lengkap
    LIKE '%$tcari%'");
    }else{
        $sus = mysql_query("SELECT * FROM user order by Id_User asc");
    }
    while($data=mysql_fetch_array($sus))
    {
        $sno++;
    >
    <tr>
        <td align="center"><b><?php echo $sno ; ?></b></td>
        <td><?php echo $data['Id_User'] ?></td>
        <td><?php echo $data['Nama_Lengkap'] ?></td>
        <td><?php echo $data['Username'] ?></td>
            <td><?php echo $data['Password'] ?></td>
            <td><?php echo $data['Jabatan'] ?></td>
        <td>
            <a href="index.php?halaman=ubahdatauser&Id_User=<?php echo
    $data['Id_User']; ?>" class="btn btn-info"><i class="fa fa-edit"></i> Ubah</a>
            <a href="index.php?halaman=hapususer&Id_User=<?php echo
    $data['Id_User']; ?>" onclick="return confirm('Apakah anda yakin akan
    menghapus data ini?')" class="btn btn-danger"><i class="fa fa-trash"></i>
    Hapus</a>
        </td>
    </tr>
    <?php
    }
    ?>
</tbody> </table> </form> </div> </div> </div>
    <!--End Advanced Tables -->
</div> </div>
<!-- DATA TABLE SCRIPTS -->
<script src="assets/js/dataTables/jquery.dataTables.js"></script>
<script src="assets/js/dataTables/dataTables.bootstrap.js"></script>
<script>

```

```

        $(document).ready(function () {
            $('#dataTables-example').dataTable();
        });
    </script>
    <!-- CUSTOM SCRIPTS -->
    <script src="assets/js/custom.js"></script>

```

5. Data Produk

```

<?php
include('connect.php');
$no=0;
?>
<div class="row">
    <div class="col-md-12">
        <!-- Advanced Tables -->
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3>Data Produk</h3>
            </div>
            <div class="panel-body">
                <a href="index.php?halaman=tambahproduk" class="btn btn-
primary"><i class="fa fa-plus"></i> Tambah Produk</a>
                <br></br>
                <div class="table-responsive">
</br>
<table class="table table-striped table-bordered table-hover" id="dataTables-
example">
    <thead>
        <tr bgcolor="#f2f2f2">
            <th>No</th>
            <th>ID Produk</th>
            <th>Nama Produk</th>
            <th>Tipe</th>
            <th>Aksi</th>
        </tr>
    </thead>
    <tbody>
        <?php
        if(isset($_POST['button'])){
            $tcari=$_POST['tcari'];
            $sus = mysql_query("SELECT * FROM produk WHERE Nama_Produk
LIKE '%$tcari%'");
        }else{
            $sus = mysql_query("SELECT * FROM produk order by Id_Produk asc");
        }
        while($data=mysql_fetch_array($sus))

```

```

{
    $sno++;
?>
<tr>
    <td align="center"><b><?php echo $sno ; ?></b></td>
    <td><?php echo $data['Id_Produk'] ?></td>
    <td><?php echo $data['Nama_Produk'] ?></td>
    <td><?php echo $data['Tipe'] ?></td>
    <td>
        <a href="index.php?halaman=ubahdataproduk&Id_Produk=<?php echo
$data['Id_Produk']; ?>" class="btn btn-info"><i class="fa fa-edit"></i> Ubah</a>
        <a href="index.php?halaman=hapusproduk&Id_Produk=<?php echo
$data['Id_Produk']; ?>" onclick="return confirm('Apakah anda yakin akan
menghapus data ini?')" class="btn btn-danger"><i class="fa fa-trash"></i>
Hapus</a>
    </td> </tr>
<?php
}
?>
</tbody> </table> </form> </div> </div> </div>
<!--End Advanced Tables -->
</div> </div>
<!-- DATA TABLE SCRIPTS -->
<script src="assets/js/dataTables/jquery.dataTables.js"></script>
<script src="assets/js/dataTables/dataTables.bootstrap.js"></script>
<script>
$(document).ready(function () {
    $('#dataTables-example').dataTable();
});
</script>
<!-- CUSTOM SCRIPTS -->
<script src="assets/js/custom.js"></script>

```

6. Dataloading

```

<?php
include('connect.php');
$id="";
$id2="";
if(ISSET($_GET['id']))
{
    $id=$_GET['id'];
}
if(ISSET($_GET['id2']))
{
    $id2=$_GET['id2'];
}

```

```

if(ISSET($_POST['hddID']))
{
    $id=$_POST['hddID'];
    $searching = $_POST['searching'];
}
if(ISSET($_GET['data']))
{
    $id=$_GET['id'];
    $searching = $_GET['data'];
}
if(ISSET($_POST['hddID']))
{
    $id2=$_POST['hddID'];
    $searching = $_POST['searching'];
}
if(ISSET($_GET['data']))
{
    $id2=$_GET['id2'];
    $searching = $_GET['data'];
}
$no=0;
$total_jml=0;
$total_OK=0;
$total_NG=0;
$total_tipis=0;
$total_meler=0;
$total_serabut=0;
$total_kotor=0;
$total_lecet=0;
$total_bintik=0;
$total_jml=0;
$blank="";
?>
<div class="row">
    <div class="col-md-12">
        <!-- Advanced Tables -->
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3>Detail Loading-Unloading - <b><?php echo $id; ?></b> -
<b><?php echo $id2; ?></b></h3>
            </div>
            <div class="panel-body">
                <br><a
href="index.php?halaman=lounfor" class="btn btn-success"><i class="fa fa-
arrow-circle-left"></i> Kembali</a><br>
            </div>
        </div>
    </div>
</div>

```

```

        <div class="panel panel-default"><div class="panel-
heading"><b>Note Proses : <br>F : Material Fresh <br> R1 : Repair Material 1x
<br> R2 : Repair Material 2x</b></div></div>
<table class="table table-striped table-bordered table-hover" id="dataTables-
example">
  <thead>
    <tr bgcolor="#f2f2f2">
      <th style='min-width:10px; vertical-align:middle'
rowspan="3">No</th>
      <th colspan='6' style='min-width:20px'>Loading</th>
      <th colspan='9' style='min-width:20px'>Unloading</th>
      <th style='min-width:10px; vertical-align:middle'
rowspan="3">Keterangan</th>
    </tr>
    <tr bgcolor="#f2f2f2">
      <th style='min-width:10px; vertical-align:middle'
rowspan="2"><small>Jam</small></th>
      <th style='min-width:10px; vertical-align:middle'
rowspan="2"><small>No Lot</small></th>
      <th style='min-width:10px; vertical-align:middle'
rowspan="2"><small>Nama Part</small></th>
      <th style='min-width:10px; vertical-align:middle'
rowspan="2"><small>Tipe</small></th>
      <th style='min-width:10px; vertical-align:middle'
rowspan="2"><small>Proses</small></th>
      <th style='min-width:10px; vertical-align:middle'
rowspan="2"><small>Jumlah</small></th>
      <th style='min-width:10px; vertical-align:middle'
rowspan="2"><small>Jam</small></th>
      <th colspan="6"><small>NG</small></th>
      <th colspan="2"><small>Jumlah</small></th></tr>
    <tr bgcolor="#f2f2f2"><th><small>Tipis</small></th>
    <th><small>Meler</small></th>
    <th><small>Kotor</small></th>
    <th><small>Serabut</small></th>
    <th><small>Lecet</small></th>
    <th><small>Bintik</small></th>
    <th><small>NG</small></th>
    <th><small>OK</small></th></tr>
  </thead>
  <tbody>
    <?php
      $conditions="";
      if(ISSET($_GET['id']))
    {
      $conditions = " and load_unload.Id_LO='".$_GET['id']."'";
    }

```

```

}
if(ISSET($_POST['hddID']))
{
    $conditions = " and load_unload.Id_LO='".$_POST['hddID']."'";
}
if(ISSET($_GET['data']))
{
    $conditions = " and load_unload.Id_LO='".$_GET['id']."' AND (Tanggal
LIKE '%".$_GET['data']."%')";
}
if(isset($_POST['button'])){
    $tcari=$_POST['tcari'];
    $sus = mysql_query("SELECT * FROM load_unload JOIN loun ON
load_unload.Id_LO = loun.Id_LO JOIN produk ON loun.Id_Produk =
produk.Id_Produk WHERE Id_Loun LIKE '%$tcari%'");
}else{
    $sus = mysql_query("SELECT * FROM load_unload JOIN loun ON
load_unload.Id_LO = loun.Id_LO JOIN produk ON loun.Id_Produk =
produk.Id_Produk $conditions");
}
while($data=mysql_fetch_array($sus))
{

    $jml_NG=$data['Jumlah_Tipis']+$data['Jumlah_Meler']+$data['Jumlah_S
erabut']+$data['Jumlah_Kotor']+$data['Jumlah_Lecet']+$data['Jumlah_Bintik'];
    $jml_OK=$data['Jumlah']-$jml_NG;
    $total_OK+=$jml_OK ;
    $total_NG+=$jml_NG ;
    $total_jml+=$data['Jumlah'];
    $total_tipis+=$data['Jumlah_Tipis'];
    $total_meler+=$data['Jumlah_Meler'];
    $total_serabut+=$data['Jumlah_Serabut'];
    $total_kotor+=$data['Jumlah_Kotor'];
    $total_lecet+=$data['Jumlah_Lecet'];
    $total_bintik+=$data['Jumlah_Bintik'];
    $no++;
?>
<tr align="center">
    <td><b><?php echo $no ; ?></b></td>
    <td><?php echo $data['Jam_Load'] ?></td>
    <td><?php echo $data['No_Lot'] ?></td>
    <td><?php echo $data['Nama_Produk'] ; ?></td>
    <td><?php echo $data['Tipe'] ; ?></td>
    <td><?php echo $data['Proses'] ; ?></td>
    <td><?php echo $data['Jumlah'] ; ?></td>
    <td><?php echo $data['Jam_Un'] ?></td>

```

```

        <td><?php echo $data['Jumlah_Tipis'] ?></td>
        <td><?php echo $data['Jumlah_Meler'] ?></td>
        <td><?php echo $data['Jumlah_Kotor'] ?></td>
        <td><?php echo $data['Jumlah_Serabut'] ?></td>
        <td><?php echo $data['Jumlah_Lecet'] ?></td>
        <td><?php echo $data['Jumlah_Bintik'] ?></td>
        <td><?php echo $jml_NG ; ?></td>
        <td><?php echo $jml_OK ; ?></td>
        <td><?php echo $data['Keterangan'] ?></td>
    </tr>
    <?php
    }
?>
    <tr>
        <td colspan='6' align='center' bgcolor="#f2f2f2"><b>Total</b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_jml ;
?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $blank ;
?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_tipis ; ?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_meler ; ?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_kotor ; ?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_serabut ;
?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_lecet ; ?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_bintik ; ?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_NG ; ?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $total_OK ; ?></b></td>
        <td align='center' bgcolor="#f2f2f2"><b><?php echo $blank ; ?></b></td>
    </tr>
</tbody>
</table>
</form>
</div>

    </div>
</div>
<!--End Advanced Tables -->
</div>
</div>

<!-- DATA TABLE SCRIPTS -->
<script src="assets/js/dataTables/jquery.dataTables.js"></script>
<script src="assets/js/dataTables/dataTables.bootstrap.js"></script>
<script>
    $(document).ready(function () {

```

```

        $('#dataTables-example').dataTable();
    });
</script>
<!-- CUSTOM SCRIPTS -->
<script src="assets/js/custom.js"></script>

```

7. Dataloun

```

<?php
include('connect.php');
$no=0;
?>
<div class="row">
    <div class="col-md-12">
        <!-- Advanced Tables -->
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3>Data Loading-Unloading</h3>
            </div>
            <div class="panel-body">
                <a href="index.php?halaman=tambahloun" class="btn btn-
primary"><i class="fa fa-share-square-o"></i> Tambah Loading-Unloading</a>
                <br></br>
            <form name="form1" method="post" action="">
                <input type="date" name="tcari" id="tcari">
                <input type="submit" name="button" id="button" value="CARI" class="btn
btn-success">
                <div class="table-responsive">
            <br>
            <table class="table table-striped table-bordered table-hover" id="dataTables-
example">
                <thead>
                    <tr bgcolor="#f2f2f2">
                        <th style='min-width:10px; vertical-align:middle'
rowspan="2">No</th>
                        <th style='min-width:10px; vertical-align:middle' rowspan="2">No
LO</th>
                        <th style='min-width:10px; vertical-align:middle'
rowspan="2">Nama Part</th>
                        <th style='min-width:10px; vertical-align:middle'
rowspan="2">Tipe</th>
                        <th style='min-width:10px; vertical-align:middle'
rowspan="2">Penginput</th>
                        <th style='min-width:10px; vertical-align:middle'
rowspan="2">Tanggal</th>
                        <th colspan='6' style='min-width:20px'>NG</th>
                        <th colspan='2' style='min-width:20px'>Total</th>

```

```

                <th style='min-width:10px; vertical-align:middle'
rowspan="2">Status</th>
                <th style='min-width:10px; vertical-align:middle' rowspan="2">Aksi</th>
            </tr>
            <tr bgcolor="#f2f2f2"><th><small>Tipis</small></th>
<th><small>Meler</small></th>
<th><small>Kotor</small></th>
<th><small>Serabut</small></th>
<th><small>Lecet</small></th>
<th><small>Bintik</small></th>
<th><small>NG</small></th>
<th><small>OK</small></th></tr>
        </thead>
        <tbody>
            <?php
            if(isset($_POST['button'])){
                $tcari=$_POST['tcari'];
                $sus = mysql_query("SELECT * FROM loun JOIN user ON loun.Id_User
= user.Id_User JOIN produk ON loun.Id_Produk = produk.Id_Produk WHERE
Tanggal LIKE '%" . $tcari . "%");
            }else{
                $sus = mysql_query("SELECT * FROM loun JOIN user ON loun.Id_User
= user.Id_User JOIN produk ON loun.Id_Produk = produk.Id_Produk order by
Id_LO");
            }
            while($data=mysql_fetch_array($sus))
            {
                $sno++;
            ?>
            <tr>
                <td><b><?php echo $sno ; ?></b></td>
                <td><?php echo $data['Id_LO'] ?></td>
                <td><?php echo $data['Nama_Produk'] ?></td>
                <td><?php echo $data['Tipe'] ?></td>
                <td><?php echo $data['Nama_Lengkap'] ?></td>
                <td><?php echo $data['Tanggal'] ?></td>
                <td><?php echo $data['Total_Tipis'] ?></td>
                <td><?php echo $data['Total_Meler'] ?></td>
                <td><?php echo $data['Total_Serabut'] ?></td>
                <td><?php echo $data['Total_Lecet'] ; ?></td>
                <td><?php echo $data['Total_Kotor'] ?></td>
                <td><?php echo $data['Total_Bintik'] ?></td>
                <td><?php echo $data['Total_NG'] ?></td>
                <td><?php echo $data['Total_OK'] ?></td>
            </tr>
        </tbody>
    </table>
    </div>
    </pre>

```

```

                <?php if($data['Status'] == "Menunggu Persetujuan") {echo
"<td align='center' bgcolor=#FFCC66><strong>Menunggu
Persetujuan</strong></td>";}

                else if($data['Status'] == "Disetujui") {echo "<td align='center'
bgcolor=#66CCCC><strong>Disetujui</strong></td>";} ?>
                <td>
                <a href="index.php?halaman=dataloading&id=<?php echo
$data['Id_LO'] ;?>&id2=<?php echo $data['Tanggal']; ?>" class="btn btn-
default"><i class="fa fa-play-circle"></i> Detail</a>
                <a
href="index.php?halaman=inputtotal&Id_LO=<?php echo $data['Id_LO'] ;?>"
class="btn btn-success"><i class="fa fa-edit"></i> Total</a>
                </td>
        </tr>
        <?php
        }
        ?>
    </tbody>
</table>
</form>
</div>

```

```

        </div>
    </div>
    <!--End Advanced Tables -->
</div>
</div>

```

```

<!-- DATA TABLE SCRIPTS -->
<script src="assets/js/dataTables/jquery.dataTables.js"></script>
<script src="assets/js/dataTables/dataTables.bootstrap.js"></script>
<script>
    $(document).ready(function () {
        $('#dataTables-example').dataTable();
    });
</script>
<!-- CUSTOM SCRIPTS -->
<script src="assets/js/custom.js"></script>

```

8. Grafik

```

<?php include('connect.php'); ?>
<div class="row">
    <div class="col-md-12">
        <!-- Advanced Tables -->
        <div class="panel panel-default">

```

```

    <div class="panel-heading">
        <h3>Grafik</h3>
    </div>
    <div class="panel-body" align="center">

    <form name="form1" method="post"
    action="index.php?halaman=grafikviewman">
        <div>
            <tr>
                <td>
                    <label>Tgl. Awal</label>
                    <input type="date" name="tgl_awal"> &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
                    <label>Tgl. Akhir</label>
                    <input type="date" name="tgl_akhir">
                </td>
            </tr>
            <input type="submit" name="button" id="button" value="CARI"
            class="btn btn-success">
        </div></div></div></div></div>

```

9. Note Saran

```

<?php
include('connect.php');
$no=0;
?>
<div class="row">
    <div class="col-md-12">
        <!-- Advanced Tables -->
        <div class="panel panel-default">
            <div class="panel-heading">
                <h3>Note Saran</h3>
            </div>
            <div class="panel-body">

    <form name="form1" method="post" action="">
        <input type="date" name="tcari" id="tcari">
        <input type="submit" name="button" id="button" value="CARI" class="btn
        btn-success">
            <div class="table-responsive">
    </div>
    <br>
    <table class="table table-striped table-bordered table-hover" id="dataTables-
    example">
        <thead>
            <tr bgcolor="#f2f2f2">
                <th>No</th>
                <th>No Note</th>

```

```

                <th>Tanggal</th>
                <th>Pembuat</th>
            <th>Untuk Periode</th>
                <th>Divisi</th>
                <th>Saran</th>
            <th>Aksi</th>
        </tr>
    </thead>
    <tbody>
        <?php
            if(isset($_POST['button'])){
                $tcari=$_POST['tcari'];
                $sus = mysql_query("SELECT * FROM note JOIN user ON note.Id_User
= user.Id_User WHERE Tanggal LIKE '%$tcari%'");
            }else{
                $sus = mysql_query("SELECT * FROM note JOIN user ON note.Id_User
= user.Id_User order by Id_Note asc");
            }
            while($data=mysql_fetch_array($sus))
            {
                $no++;
        <tr>
            <td align="center"><b><?php echo $no ; ?></b></td>
                <td><?php echo $data['Id_Note'] ?></td>
                <td><?php echo $data['Tanggal'] ?></td>
                <td><?php echo $data['Nama_Lengkap'] ?></td>
                <td><?php echo $data['Periode'] ?></td>
                <td><?php echo $data['Divisi'] ?></td>
                <td><?php echo $data['Saran'] ?></td>
            <td>
                <a
                    href="javascript:void(0);"
                    plain="true"
                    onclick="window.open('cetak-saran.php?Id_Note=<?php
                    echo
                    $data['Id_Note']?>', 'size=1200', 'height=1200', 'scrollbars=yes', 'resizeable=no')"
                    class="btn btn-default"><i class="fa fa-print"></i> Cetak</a>
                <a
                    href="index.php?halaman=hapusnote&Id_Note=<?php
                    echo
                    $data['Id_Note']; ?>"
                    onclick="return confirm('Apakah anda yakin akan
                    menghapus data ini?')"
                    class="btn btn-danger"><i class="fa fa-trash"></i>
                    Hapus</a>
            </td>
        </tr>
    </tbody>
    </table>

```

```
</form>
</div>
```

```
    </div>
  </div>
  <!--End Advanced Tables -->
</div>
</div>
```

```
<!-- DATA TABLE SCRIPTS -->
<script src="assets/js/dataTables/jquery.dataTables.js"></script>
<script src="assets/js/dataTables/dataTables.bootstrap.js"></script>
<script>
  $(document).ready(function () {
    $('#dataTables-example').dataTable();
  });
</script>
<!-- CUSTOM SCRIPTS -->
<script src="assets/js/custom.js"></script>
```