

**RANCANG BANGUN SISTEM INFORMASI PEMERIKSAAN
KOMPONEN *PART GUSSET CROSS MEMBER* DI BAGIAN
QUALITY ASSURANCE RECEIVING INSPECTION PADA
PT GEMALA KEMPA DAYA**

TUGAS AKHIR

Diajukan Untuk Memenuhi Sebagian Syarat Penyelesaian Jenjang Sarjana
Terapan Program Studi Sistem Informasi Industri Otomotif
Pada Politeknik STMI Jakarta

OLEH

**BHAGUS ARIEF PRAYOGO
1315050**



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2019**

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

JUDUL TUGAS AKHIR:

**“RANCANG BANGUN SISTEM INFORMASI PENGOLAHAN WORK
ORDER PADA PT GEMALA KEMPA DAYA”**

Disusun Oleh:

Nama : Bhagus Falih Prayogo
NIM : 1315103
Program Studi : Sistem Informasi Industri Otomotif

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia Pada Hari Senin Tanggal 23 September 2019.

Jakarta, 24 September 2019

Dosen Pembimbing



Abrian Ismono, S.Kom, MMSI
NIP : 197901072006041002

Ketua Penguji



Ulil Hamida, S.T., M.T
NIP : 198103272005022001

Dosen Penguji



Dedy Trisanto, S.Kom, MMSI
NIP : 197805052005021002

Dosen Penguji



Dr. Ridzky Kramanandita, S.Kom, M.T
NIP : 197403022002121001

**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA**

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

JUDUL TUGAS AKHIR :

**RANCANG BANGUN SISTEM INFORMASI PENGOLAHAN *WORK*
ORDER PADA PT GEMALA KEMPA DAYA**

Disusun Oleh:

Nama : Bhagus Falih Prayogo
NIM : 1315103
Program Studi : Sistem Informasi Industri Otomotif
Tanggal Seminar : 28 Agustus 2019
Tanggal Sidang : 23 September 2019
Tanggal Lulus : 23 September 2019

Jakarta, 24 September 2019

Dosen Pembimbing

Ahlan Ismono, S.Kom., MMSI

NIP. 197901072006041002



LEMBAR BIMBINGAN PERSETUJUAN DOSEN PEMBIMBING

Nama : Bhagus Falih Prayogo
 NIM : 1315103
 Judul Tugas Akhir : Rancang Bangun Sistem Informasi Proses Work Order Pada PT Gemala Kempa Daya
 Pembimbing : Ahlan Ismono. S.Kom. MMSI

Tanggal	Keterangan	Paraf
20/07/2019	Bimbingan Judul, Bab 1 & 2	
08/07/2019	Bimbingan Bab 3 & 4	
15/07/2019	Revisi Bab 3, 4 & Bimbingan Bab 5	
17/07/2019	Revisi Bab 4 & 5	
20/07/2019	Revisi Bab 4 & 5	
22/07/2019	Revisi Bab 5	
05/08/2019	Revisi Bab 5	
15/08/2019	Revisi Bab 5	
23/08/2019	Revisi Bab 5	
12/08/2019	Keseluruhan dan Acc BAB I,II,III,IV,V,IV	

Mengetahui,

Ketua Program Studi
Sistem Informasi Industri Otomotif

a.n 

Noveriza Yuliasari. S.Si. M.T.

NIP : 197811212009012003



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

Dosen Pembimbing



Ahlan Ismono. S.kom. MMSI

NIP : 197901072006041002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Bagus Falih Prayogo

Nim : 1315103

Program Studi : Sistem Informasi Industri Otomotif

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

“RANCANG BANGUN SISTEM INFORMASI PENGOLAHAN *WORK ORDER* PADA PT GEMALA KEMPA DAYA”. Merupakan dibuat dan diselesaikan sendiri, dengan menggunakan literatur hasil kuliah, survei lapangan, Dosen Pembimbing dan Asisten Dosen Pembimbing, melalui tanya jawab maupun asistensi serta buku-buku acuan yang tertera dalam referensi pada karya Tugas Akhir ini.

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

Jakarta, 09 Agustus 2019

Yang Membuat Pernyataan,



Bagus Falih Prayogo

ABSTRAK

Pemeriksaan barang atau komponen *part* merupakan hal utama dalam proses bisnis suatu perusahaan khususnya pada industri manufaktur. Meskipun diakui bahwa kualitas barang atau komponen *part* mempunyai pengaruh yang sangat besar terhadap kualitas produknya, akan tetapi pemeriksaan barang atau komponen *part* yang menentukan kualitas suatu produk. Dari hal tersebut, diperoleh permasalahan seperti belum adanya sistem yang dapat mengatur proses pemeriksaan barang atau komponen *part* yang berada di PT Gemala Kempa Daya. Proses pemeriksaan yang masih dilakukan dengan menggunakan media kertas serta belum ada nya sistem yang terintegrasi menyebabkan kesulitan dalam pencarian data yang dibutuhkan. Untuk memecahkan masalah tersebut, dibutuhkan sistem yang dapat terintegrasi satu sama lain secara otomatis. Sistem informasi ini dibuat menggunakan metode pengembangan sistem *waterfall*. Metode ini diawali dari identifikasi kebutuhan, analisis kebutuhan, merancang sistem dengan UML, merancang data dengan *ERD*, dan melakukan perancangan antar muka. Sistem ini dibangun menggunakan *MariaDB* 10.1.36 dan *CodeIgniter* 3.1.8. Sistem informasi pemeriksaan komponen *part* *Gusset Cross Member* ini nantinya dapat memudahkan pemeriksaannya terutama dilini *QA Receiving Inspection* dan departemen yang terlibat pada PT Gemala Kempa Daya.

Kata Kunci : Sistem Informasi, Pemeriksaan Komponen *Part*, *Waterfall*, UML

KATA PENGANTAR

Puji Syukur penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, yang senantiasa sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul **“Rancang Bangun Sistem Informasi Pemeriksaan Komponen *Part Gusset Cross Member* Di Bagian *Quality Assurance Receiving Inspection* Pada PT Gemala Kempa Daya”**.

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

Terselesaikannya 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 atas rahmat serta kemudahan yang diberikan.
2. Bapak Agus Imam Prayogo dan Ibu Eka Hajrahprayidianingsih serta Paman dan Tante yang selalu memberikan dukungan, kasih sayang serta doa untuk keberhasilan penulis.
3. Bapak Dr. Mustofa, ST, MT. Selaku Direktur Politeknik STMI Jakarta.
4. Ibu Noveriza Yuliasari, MT. Selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
5. Bapak Ahlan Ismono, S.Kom, MM SI. Selaku Dosen Pembimbing yang telah bersedia memberi arahan, bimbingan dan penjelasan dalam penyusunan Tugas Akhir.
6. Bapak Emmanuel selaku *General Manager* divisi K3 yang telah memberi saya kesempatan untuk magang di PT Gemala Kempa Daya
7. Bapak Yandi Fitra selaku pembimbing saya di Bagian Teknologi Informasi dan Bapak Agus di Bagian *Quality Assurance* yang telah memberikan pengetahuan baik secara lisan, tulisan, maupun dokumen terkait untuk penulis dalam menjalankan Tugas Akhir.

8. Bapak Gandhi, Bapak Erik dan Ibu Ienez selaku *HRD* di PT Gemala Kempa Daya yang telah memberikan informasi dan mengarahkan selama melakukan praktek kerja lapangan di PT Gemala Kempa Daya.
9. Seluruh pekerja di PT Gemala Kempa Daya yang tidak bisa saya sebutkan satu persatu terima kasih atas bantuan yang telah diberikan selama berlangsungnya kegiatan Praktik Kerja Lapangan 2.
10. Seluruh dosen Politeknik STMI Jakarta yang telah mengajarkan ilmunya sehingga penulis dapat menyelesaikan laporan ini.
11. Teman-teman Mahasiswa/i program studi Sistem Informasi Industri Otomotif di Politeknik STMI Jakarta. Khususnya untuk Yati Fams, SA02 serta senior dan alumni Politeknik STMI Jakarta.
12. Semua pihak yang telah membantu dalam penulisan dan penyusunan laporan ini baik secara langsung maupun tidak langsung, yang tentunya tidak dapat disebutkan satu per satu.

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

Jakarta, 10 Agustus 2019

Penulis
Bhagus Arif Prayogo

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
LEMBAR PERSETUJUAN DOSEN PEMBIMBING.....	ii
LEMBAR BIMBINGAN DENGAN DOSEN PEMBIMBING	iii
LEMBAR KEASLIAN	iv
ABSTRAK.....	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan.....	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah.....	4
1.5 Manfaat Tugas Akhir.....	4
1.6 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	6
2.1 Pengertian Rancang Bangun	6
2.2 Konsep Dasar Sistem.....	6
2.2.1 Karakteristik Sistem	7
2.2.2 Jenis Sistem	9
2.2.3 Klasifikasi Sistem.....	10
2.3 Konsep Dasar Sistem.....	11
2.2.1 Sumber Data Informasi.....	12
2.2.2 Siklus Informasi.....	12
2.4 Konsep Dasar Sistem Informasi	13

2.4.1	Komponen Sistem Informasi.....	15
2.5	Siklus Hidup Pengembangan Sistem.....	17
2.6	Metodologi Pengembangan Sistem	18
2.7	Konsep Dasar Pengendalian.....	20
2.8	Proses Pengendalian	20
2.9	Konsep Dasar Kualitas	21
2.10	Konsep Dasar Pengendalian Kualitas.....	22
2.10.1	Komponen Sistem Informasi.....	22
2.10.2	Tujuan Pengendalian Kualitas	23
2.10.2	Faktor Pengendalian Kualitas	24
2.11	Entity Relationship Diagram (ERD)	25
2.12	Pengertian Bahan Baku	27
2.13	Macam – Macam Bahan Baku	28
2.14	<i>Flowchart</i>	28
2.15	<i>Unified Modelling Language (UML)</i>	31
2.16	<i>Use Case Diagram</i>	33
2.17	<i>Activity Diagram</i>	34
2.17.1	<i>Sequence Diagram</i>	37
2.17.2	<i>Class Diagram</i>	38
2.17.3	<i>Entity Relationship Diagram (ERD)</i>	39
2.18	Kamus Data Dan <i>Database</i>	40
2.19	<i>Windows Navigation Diagram (WND)</i>	41
2.20	<i>CodeIgniter</i>	42
2.21	<i>MariaDB</i>	42
2.22	Tipe Data <i>MariaDB</i>	43
BAB III	METODOLOGI PENELITIAN	45
3.1	Metodologi Penelitian	45
3.2	Jenis Dan Sumber Data	45
3.3	Metode Pengumpulan Data	46
3.4	Metode Pengembangan Sistem	46
3.5	Kerangka Penelitian	48

BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA.....	52
4.1	Profil Dan Sejarah Perusahaan	52
4.1.1	Visi & Misi.....	54
4.1.2	Produk.....	54
4.1.3	Mesin Produksi.....	59
4.2	Struktur Organisasi & Deskripsi Tugas.....	59
4.2.1	Struktur Organisasi PT Gemala Kempa Daya.....	60
4.2.2	Deskripsi Tugas PT Gemala Kempa Daya.....	60
4.3	Jumlah Komposisi & Jumlah Karyawan	63
4.3.1	Sistem Kerja Karyawan.....	63
4.3.2	Sistem Penggajian Karyawan.....	65
4.4	Sarana Pendukung	65
4.5	Fasilitas Yang Dimiliki	66
4.6	Struktur Organisasi Divisi <i>Quality Assurance</i>	67
4.6.1	Struktur Organisasi Divisi <i>Quality Assurance</i>	68
4.6.2	Uraian Tugas Kerja & Tanggung Jawab.....	68
4.7	Proses Penerimaan Bahan Baku	70
4.8	Proses Pemeriksaan Bahan Baku	71
4.9	Analisis Dokumen Yang Berjalan.....	75
4.10	Arus Dokumen	75
4.11	Prosedur Pemeriksaan Bahan Baku Pada <i>Quality Assurance Receiving Inspection</i>	76
4.12	Penggambaran Sistem Pengendalian Kualitas Bahan Baku Dengan <i>UML (Unified Modeling Language)</i>	78
4.13	<i>Use Case Diagram</i>	79
BAB V	ANALISIS DAN PEMBAHASAN.....	81
5.1	Alur Proses Pemeriksaan Bahan Baku Usulan.....	81
5.2	Analisis Sistem Usulan Sistem Usulan	83
5.3	<i>Use Case Diagram</i>	83
5.4	Analisis Kebutuhan Sistem	94
5.4.1.	<i>Functional Requirement</i>	97

5.4.2. <i>Non Functional Requirement</i>	97
5.5 <i>Activity Diagram</i>	98
5.6 <i>Sequence Diagram</i>	109
5.7 <i>Class Diagram</i>	118
5.8 <i>Deployment Diagram</i>	119
5.9 <i>Windows Navigation Diagram (WND)</i>	120
5.10 <i>Pemodelan Data Sistem Usulan</i>	122
5.10.1 <i>Entity Relationship Diagram (ERD)</i>	122
5.10.2 <i>Kamus Data</i>	123
5.10.3 <i>Perancangan Interface Sistem Informasi</i> Pemeriksaan Bahan Baku	126
5.10.4 <i>Implementasi Sistem Software Dan Hardware</i>	133
BAB VI KESIMPULAN DAN SARAN	134
6.1 <i>Kesimpulan</i>	134
6.2 <i>Saran</i>	134
<i>Daftar Pustaka</i>	135
<i>Lampiran</i>	L

DAFTAR GAMBAR

	Halaman
Gambar II.1 Karakteristik Sistem	9
Gambar II.2 Sistem Terbuka.....	10
Gambar II.3 Sistem Tertutup	10
Gambar II.4 Siklus Informasi	13
Gambar II.5 Blok Sistem Informasi Yang Berinteraksi	15
Gambar II.6 Metode <i>Waterfall</i>	18
Gambar II.7 Siklus <i>PDCA</i>	25
Gambar II.8 Contoh <i>Activity Diagram</i>	36
Gambar II.9 Contoh <i>Windows Navigation Diagram</i>	42
Gambar III.1 Kerangka Penelitian	51
Gambar IV.1 Logo PT Gemala Kempa Daya	52
Gambar IV.2 Tata Letak Pabrik Produksi.....	53
Gambar IV.3 <i>Frame Chassis</i>	55
Gambar IV.4 Kategori Produk <i>Pressed Part Housing Rear Axle</i>	57
Gambar IV.2 Tata Letak Pabrik Produksi.....	53
Gambar IV.5 Struktur Organisasi PT Gemala Kempa Daya	60
Gambar IV.6 Pengelolaan Bahan Baku Divisi <i>Quality Assurance</i>	68
Gambar IV.7 Struktur Organisasi Divisi <i>Quality Assurance</i>	68
Gambar IV.8 Surat Jalan Pengiriman Bahan Baku	71
Gambar IV.9 Surat Retur Bahan Baku.....	71
Gambar IV.10 <i>Check Sheet</i> Bahan Baku <i>Gusset Cross Member</i>	72
Gambar IV.11 Bahan Baku Yang Ditandai Label <i>OK</i>	73
Gambar IV.12 Bahan Baku Yang Ditandai Label <i>NG</i>	73
Gambar IV.13 Laporan Sortir	74
Gambar IV.14 Laporan Penerimaan Barang.....	75
Gambar IV.15 Aliran Dokumen Sistem Yang Berjalan.....	78

Gambar V.1	<i>Flowmap Usulan Pemeriksaan Bahan Baku Usulan</i>	82
Gambar V.2	<i>Use Case Sistem Informasi Pemeriksaan Bahan</i>	84
Gambar V.3	<i>Activity Diagram Login</i>	98
Gambar V.4	<i>Activity Diagram Master Data Login</i>	99
Gambar V.5	<i>Activity Diagram Master Data Komponen Part</i>	100
Gambar V.6	<i>Activity Diagram Master Data Supplier</i>	101
Gambar V.7	<i>Activity Diagram Data Checksheet</i>	102
Gambar V.8	<i>Activity Diagram Memvalidasi Data Checksheet</i>	103
Gambar V.9	<i>Activity Diagram Memvalidasi Data Surat Jalan</i>	104
Gambar V.10	<i>Activity Diagram Data Surat Jalan</i>	105
Gambar V.11	<i>Activity Diagram Merekap Laporan Sortir</i>	106
Gambar V.12	<i>Activity Diagram Merekap Laporan Penerimaan Barang</i>	107
Gambar V.13	<i>Activity Diagram Merekap Laporan Retur</i>	108
Gambar V.14	<i>Sequence Diagram Login</i>	109
Gambar V.15	<i>Sequence Diagram Mengelola Master Data Login</i>	110
Gambar V.16	<i>Sequence Diagram Mengelola Master Data Komponen Part</i> ..	111
Gambar V.17	<i>Sequence Diagram Mengelola Master Data Supplier</i>	112
Gambar V.18	<i>Sequence Diagram Mengelola Data Surat Jalan</i>	113
Gambar V.19	<i>Sequence Diagram Mengelola Data Checksheet</i>	114
Gambar V.20	<i>Sequence Diagram Melakukan Validasi Data Surat Jalan</i>	115
Gambar V.21	<i>Sequence Diagram Melakukan Validasi Data Checksheet</i>	116
Gambar V.22	<i>Sequence Diagram Merekap Laporan Sortir</i>	116
Gambar V.23	<i>Sequence Diagram Merekap Laporan Penerimaan Barang</i>	117
Gambar V.24	<i>Sequence Diagram Merekap Laporan Retur</i>	118
Gambar V.25	<i>Class Diagram Sistem Informasi Pemeriksaan Komponen Part</i>	119
Gambar V.26	<i>Deployment Diagram Sistem Informasi Pemeriksaan Komponen Part</i>	120
Gambar V.27	<i>Windows Navigation Diagram Sistem Informasi Pemeriksaan Komponen Part</i>	121

Gambar V.28	<i>Entity Relationship Diagram</i> Sistem Informasi Pemeriksaan Komponen <i>Part</i>	122
Gambar V.29	Rancangan <i>Form Login</i>	127
Gambar V.30	Rancangan Menu Utama	128
Gambar V.31	Rancangan <i>Form</i> Data Komponen <i>Part</i>	128
Gambar V.32	Rancangan <i>Form</i> Data <i>Login</i>	129
Gambar V.33	Rancangan <i>Form</i> Data <i>Supplier</i>	129
Gambar V.34	Rancangan <i>Form</i> Data Surat jalan	130
Gambar V.35	Rancangan <i>Form</i> Data <i>Checksheet</i>	131
Gambar V.36	Rancangan <i>Form</i> Laporan Sortir.....	131
Gambar V.37	Rancangan <i>Form</i> Laporan Penerimaan Barang	132
Gambar V.38	Rancangan <i>Form</i> Laporan Retur.....	132

DAFTAR TABEL

	Halaman
Tabel II.1 Simbol <i>Flow Direction</i>	29
Tabel II.2 Simbol Proses	30
Tabel II.3 Simbol <i>Input Dan Output</i>	30
Tabel II.4 Jenis Diagram Pada <i>UML</i> Dan Fungsi-Fungsi nya	31
Tabel II.5 Simbol – Simbol <i>Use Case Diagram</i>	33
Tabel II.6 Simbol – Simbol <i>Activity Diagram</i>	35
Tabel II.7 Simbol – Simbol <i>Sequence Diagram</i>	37
Tabel II.8 Simbol – Simbol <i>Class Diagram</i>	38
Tabel II.9 Simbol – Simbol <i>Entity Relationship Diagram (ERD)</i>	39
Tabel II.10 Tipe Data Jenis Numerik Pada <i>MariaDB</i>	43
Tabel II.11 Tipe Data Jenis Karakter Pada <i>MariaDB</i>	44
Tabel II.12 Tipe Data Jenis Tanggal Atau Waktu Pada <i>MariaDB</i>	44
Tabel IV.1 Produk <i>Supporting Part</i>	56
Tabel IV.2 Produk <i>Supporting Part</i> (Lanjutan).....	57
Tabel IV.3 Kategori Produk <i>Pressed Part</i>	58
Tabel IV.4 Mesin Produksi Yang Dimiliki Oleh PT GKD.....	59
Tabel IV.5 Definisi Aktor Pemeriksaan Bahan Baku	79
Tabel IV.6 Definisi <i>Use Case</i> Pemeriksaan Bahan Baku	80
Tabel V.1 Definisi Aktor <i>Use Case Diagram</i> Sistem Usulan	86
Tabel V.2 Definisi <i>Use Case Login</i>	87
Tabel V.3 Definisi <i>Use Case</i> Mengelola Master Data <i>Supplier</i>	88
Tabel V.4 Definisi <i>Use Case</i> Mengelola Master Data Bahan Baku	89
Tabel V.5 Definisi <i>Use Case</i> Mengelola Master Data <i>Login</i>	90
Tabel V.6 Definisi <i>Use Case</i> Mengelola Data <i>Checksheet</i>	91
Tabel V.7 Definisi <i>Use Case</i> Mengelola Data Surat Jalan.....	92
Tabel V.8 Definisi <i>Use Case</i> Merekap Laporan Sortir	93

Tabel V.9	Definisi <i>Use Case</i> Melakukan Validasi Data Surat Jalan	93
Tabel V.10	Definisi <i>Use Case</i> Melakukan Validasi Data <i>Checksheets</i>	94
Tabel V.11	Definisi <i>Use Case</i> Merekap Laporan Penerimaan Barang	94
Tabel V.12	Definisi <i>Use Case</i> Merekap Laporan Retur.....	95
Tabel V.13	Definisi <i>Use Case</i> Kebutuhan Sistem.....	96
Tabel V.14	Definisi <i>Use Case</i> Kebutuhan Sistem (Lanjutan).....	97
Tabel V.15	Definisi <i>Use Case</i> Tabel Data <i>Login</i>	122
Tabel V.16	Definisi <i>Use Case</i> Tabel Data Bahan Baku.....	123
Tabel V.17	Definisi <i>Use Case</i> Tabel Data <i>Supplier</i>	123
Tabel V.18	Definisi <i>Use Case</i> Tabel Data Surat Jalan.....	124
Tabel V.19	Definisi <i>Use Case</i> Tabel Data <i>Checksheets</i> Bahan Baku	125

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem informasi sangat diperlukan bagi setiap perusahaan yang berada dalam lingkup industri untuk membuat suatu kemajuan dalam pelaksanaan proses bisnis di perusahaan tersebut. Sistem informasi menjadikan setiap perusahaan lebih tepat dalam informasi yang dibutuhkan. Sistem informasi dapat diterapkan atau digunakan dalam bagian-bagian yang terdapat dalam suatu perusahaan, seperti proses penerimaan bahan baku, proses pemeriksaan barang, administrasi, penjadwalan, produksi, dan pengiriman. Salah satu kegiatan yang menggunakan sistem informasi didalam perusahaan industri yaitu bagian proses pemeriksaan bahan baku. Proses pemeriksaan bahan baku yang terhambat atau tidak sesuai dengan yang telah ditargetkan, tentu saja akan mempengaruhi standar kualitas yang akan dibutuhkan.

Dalam perusahaan industri yang telah melakukan pemesanan barang atau bahan baku yang dipesan akan melakukan pemeriksaan terhadap kesesuaian data pengirim serta pemeriksaan kualitas terhadap barangnya. Dalam kata lain apakah surat pengantar barang yang dipesan baik produk maupun bahan baku yang dikeluarkan oleh vendor sesuai dengan jumlah, nama barang, jenis, ukuran, dan kondisi barang sebelum disimpan ke gudang untuk diolah oleh bagian gudang. Adapun kegiatan dalam proses pemeriksaan dari suatu barang atau bahan baku yang dipesan oleh pemasok akan lebih mudah dalam pemeriksaan data barang atau bahan baku untuk kesesuaian data yang dipesan dengan sistem yang terintegrasi yang menghasilkan informasi sesuai kebutuhan. Oleh karena itu proses pemeriksaan barang atau bahan baku harus didokumentasikan dengan penyimpanan data untuk barang atau bahan baku tersebut jika terjadi hal yang tak diinginkan, seperti pembuatan retur barang untuk dikembalikan kepada vendor atau *supplier*.

PT Gemala Kempa Daya merupakan industri yang bergerak di bidang industri otomotif kendaraan bermotor dengan jenis kendaraan roda empat, yang

merupakan salah satu anak perusahaan industri dari PT ASTRA Grup. Sebagai salah satu anak perusahaan industri PT ASTRA Grup, pemeriksaan bahan baku oleh lini *receiving inspection* harus teliti dan cermat dalam pemeriksaan untuk mengetahui standar kualitas pada PT Gemala Kempa Daya agar dapat bertahan dan memenangkan persaingan dengan industri lain yang sejenis. Oleh karena itu, untuk dapat memenuhi keinginan dan dapat memberikan kepuasan bagi konsumen, industri harus mampu melakukan perbaikan-perbaikan secara berkelanjutan atau *continuous improvement* dalam menjaga dan meningkatkan produknya.

PT Gemala Kempa Daya mempunyai perhatian besar terhadap peningkatan pemeriksaan barang atau bahan baku yang dipesan, khususnya dalam mempermudah pencarian data untuk informasi yang dibutuhkan yang datang dari pemasok di lini *Quality Assurance Receiving* hingga disimpan di gudang. PT Gemala Kempa Daya memiliki standar penjaminan kualitas untuk proses pemeriksaan barang atau bahan baku itu sendiri. Namun dalam kenyataannya, pada bagian proses pemeriksaan di PT Gemala Kempa Daya masih ditemukan beberapa permasalahan yang terjadi seperti sulitnya pencarian serta penyimpanan data pemeriksaan barang atau bahan baku, serta pelaporan data tersebut.

Selain permasalahan dalam pencarian data yang ditemukan pada proses pemeriksaan barang atau bahan baku Proses pemeriksaan yang masih menggunakan kertas surat jalan, *checksheet*, laporan sortir, laporan penerimaan barang, dan retur menyebabkan banyaknya dokumen yang menumpuk di dalam arsip penyimpanan. Selain itu, belum adanya sistem yang terintegrasi dengan *database* yang mengakibatkan kesulitan dalam melakukan pembaruan informasi mengenai pemeriksaan serta pembuatan laporan barang atau bahan baku yang dipesan, serta sering terjadinya kehilangan data, sehingga kurang maksimalnya proses pembuatan laporan untuk penjaminan kualitas bahan baku hingga penerimaan barang.

Berdasarkan latar belakang dan permasalahan tersebut, maka disusun tugas akhir ini dengan judul “SISTEM INFORMASI PEMERIKSAAN BAHAN BAKU DI BAGIAN *QUALITY ASSURANCE RECEIVING INSPECTION* PADA PT GEMALA KEMPA DAYA” untuk membantu perusahaan agar mempermudah

pembuatan pelaporan pemeriksaan bahan baku yang dipesan dalam proses pemeriksaan bahan baku.

1.2 Pokok Permasalahan

Berdasarkan latar belakang yang dipaparkan sebelumnya, maka dapat diidentifikasi permasalahan-permasalahan yang terjadi pada proses pemeriksaan bahan baku di Departemen QA dilini *QA Receiving Inspection* pada PT Gemala Kempa Daya adalah sebagai berikut:

1. Proses pelaporan pemeriksaan yang masih membutuhkan waktu yang lama karena berkas – berkas pemeriksaan komponen *part* harus diperiksa satu persatu terlebih dahulu.
2. Media penyimpanan masih menggunakan lemari, sehingga sulit mencari data pemeriksaan bahan baku bila diperlukan. Data-data di lemari menumpuk dan rawan rusak.
3. Belum adanya sistem yang terintegrasi dengan basis data yang mengakibatkan kesulitan dalam melakukan pembaruan informasi proses pemeriksaan bahan baku, serta sering terjadinya kehilangan data.

1.3 Tujuan Penelitian

Tujuan penelitian dari tugas akhir ini yang dilakukan terhadap departemen *Quality Assurance* terutama dilini *QA Receiving Inspection* di PT Gemala Kempa Daya dalam melakukan pemeriksaan kualitas bahan baku adalah untuk merancang dan membangun sistem yang terintegrasi khususnya untuk pemeriksaan bahan baku yang mampu:

1. Pembuatan sistem informasi pemeriksaan bahan baku yang *user friendly* untuk mempermudah pelaporan, pencarian data, serta pembaruan informasi pemeriksaan barang atau bahan baku dengan menggunakan *CodeIgniter 3.1.8*
2. Menyediakan fasilitas penyimpanan data pemeriksaan bahan baku dengan menggunakan *database* agar dapat disimpan di satu lokasi sehingga membantu proses pencarian data jika dibutuhkan dengan menggunakan *database MariaDB* versi 10.1.36.

1.4 Batasan Masalah

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

1. Tugas akhir ini hanya merancang bangun dan belum diimplementasikan.
2. Untuk memperoleh data dan tempat penelitian dilakukan pada Departemen *Quality Assurance* di lini *Quality Assurance Receiving Inspection* pada proses pemeriksaan bahan baku selama tiga bulan, mulai dari 12 Juli sampai dengan 12 Oktober 2018.
3. Ruang lingkup yang diamati sebatas pada Departemen *Quality Assurance* di lini *Quality Assurance Receiving Inspection* dalam proses pemeriksaan bahan baku *Gusset Cross Member* yang akan disimpan ke Gudang.

1.5 Manfaat Tugas Akhir

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

1. Mempermudah pencarian dan pembaruan data untuk informasi yang dibutuhkan dalam proses pemeriksaan komponen *part gusset cross member* pada PT Gemala Kempa Daya di Departemen *Quality Assurance* terutama di bagian *QA Receiving Inspection*.
2. Membantu *QA Receiving Inspection* di PT Gemala Kempa Daya dalam pembuatan pelaporan proses pemeriksaan bahan baku.

1.6 Sistematika Penulisan

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

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Pada bab ini membahas tentang berbagai teori yang diperoleh dari buku-buku literatur ataupun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan pada laporan ini adalah seputar sistem informasi, pemeriksaan kualitas bahan baku, *Quality Assurance*, *Unified Modelling Language* (UML), dan teori-teori lain yang berhubungan dengan perancangan sistem informasi.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan metode ilmiah dalam 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

Pada bab ini akan membahas tentang data yang telah diperoleh berdasarkan penelitian di PT Gemala Kempa Daya, meliputi profil perusahaan, struktur organisasi, proses bisnis sistem pemeriksaan bahan baku yang berjalan, dan dokumen yang terlibat pada proses bisnis tersebut.

BAB V ANALISIS DAN PEMBAHASAN

Pada bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis sistem yang meliputi diagram alir sistem usulan, pemodelan sistem dengan UML pemodelan data dengan kamus data, perancangan tampilan layar, perancangan hierarki menu, dan pembuatan spesifikasi sistem yang diperlukan.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran dalam penerapan sistem informasi pemeriksaan bahan baku untuk perusahaan dalam penelitian dan pengembangan selanjutnya.

BAB II

LANDASAN TEORI

2.1 Pengertian Rancang Bangun

Rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisis dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan. Rancangan sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru (McLeod, 2008).

Perancangan adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik. Sedangkan pengertian bangun atau pembangunan sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian. Bangun sistem adalah membangun sistem informasi dan komponen yang didasarkan pada spesifikasi desain.

Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil analisis ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut ataupun memperbaiki sistem yang sudah ada.

2.2 Konsep Dasar Sistem

Pengertian dan definisi sistem pada berbagai bidang berbeda-beda, tetapi meskipun istilah sistem yang digunakan bervariasi, semua sistem pada bidang-bidang tersebut mempunyai beberapa persyaratan umum, yaitu sistem harus mempunyai elemen, lingkungan, interaksi antar elemen, interaksi antara elemen dengan lingkungannya, dan yang terpenting adalah sistem harus mempunyai tujuan yang akan dicapai Jogiyanto (2005). Sistem yang diintisarikan dari beberapa sumber referensi, mendefinisikan bahwa sistem:

1. Menurut Jogiyanto (2005) Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan

suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

2. Menurut Sutarman (2009) Sistem adalah kumpulan elemen yang saling berhubungan dan saling berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama.
3. Menurut Mustakini (2009) Sistem dapat didefinisikan dengan pendekatan prosedur dan pendekatan komponen, sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu.

2.2.1 Karakteristik Sistem

Suatu sistem memiliki sifat atau karakteristik tertentu, diantaranya sebagai berikut (Jogiyanto, 2005):

1. Komponen (*component*)
 Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen sistem atau elemen sistem dapat berupa:
 - a. Elemen-elemen yang lebih kecil yang disebut subsistem, misalkan sistem komputer terdiri dari subsistem perangkat keras, perangkat lunak dan manusia.
 - b. Elemen-elemen yang lebih besar yang disebut supra sistem, misalkan bila perangkat keras adalah sistem yang memiliki subsistem CPU, perangkat I/O dan memori, maka supra sistem perangkat keras adalah sistem komputer.
2. Batas sistem (*boundary*)
 Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.
3. Lingkungan luar sistem (*environment*)

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

4. Penghubung sistem (*interface*)

Penghubung merupakan media perantara antar subsistem. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. Output dari satu subsistem akan menjadi input untuk subsistem yang lainnya dengan melalui penghubung tersebut. Dengan penghubung satu subsistem dapat berinteraksi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan sistem (*input*)

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

6. Keluaran sistem (*output*)

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

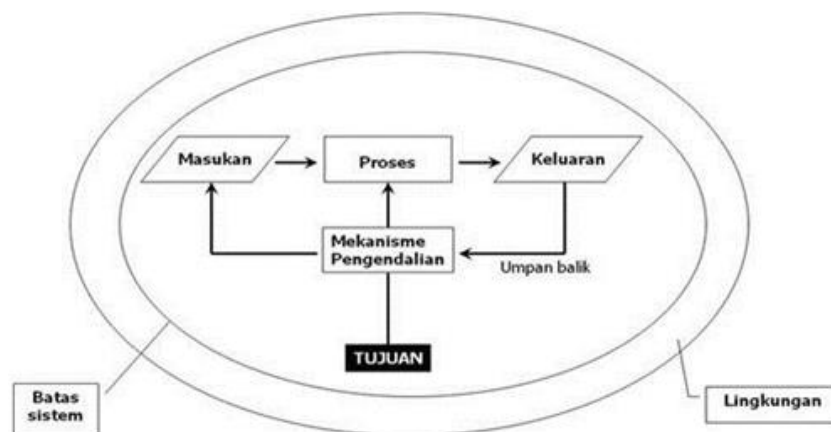
7. Pengolah sistem (*process*)

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

8. Sasaran sistem

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

Jogianto (2005: 3) dalam Rusdiana (2014), mengemukakan bahwa sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu *input*, *process* dan *output*. Hal ini merupakan konsep sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem memiliki karakteristik tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem (Sutabri, 2005) dalam Rusdiana (2014). Berikut ini yang merupakan karakteristik sistem yang dapat dilihat pada Gambar II.1.



Gambar II.1 Karakteristik Sistem
Sumber: Rusdiana dan Irfan (2014)

2.2.2 Jenis Sistem

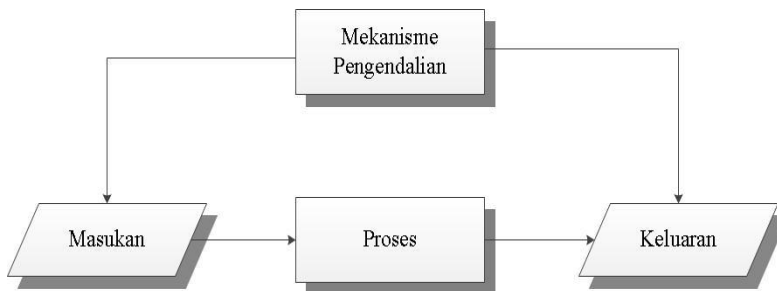
Jenis sistem secara umum terdiri atas sistem terbuka dan sistem tertutup. Sistem terbuka adalah sistem yang tidak memiliki sasaran, pengendalian mekanis dan umpan balik. Adapun sistem yang tertutup adalah sebuah sistem yang memiliki sasaran, pengendalian mekanis dan umpan balik (McLeod, Jr.,2001)

dalam Rusdiana (2014). Secara sederhana, sistem terbuka dapat digambarkan sebagai Gambar II.2 dibawah ini.



Gambar II.2 Sistem Terbuka
(Sumber: Rusdiana dan Irfan, 2014)

Sistem tertutup adalah sistem yang tidak berinteraksi secara langsung dengan lingkungannya melalui arus sumber daya tidak berhubungan dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luarnya skema sistem tertutup dapat dilihat pada Gambar II.3 di bawah ini



Gambar II.3 Sistem Tertutup
(Sumber: Rusdiana dan Irfan, 2014)

2.2.3 Klasifikasi Sistem

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

1. Sistem Abstrak dan Sistem Fisik

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

2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam,

pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

3. Sistem Deterministik dan Sistem Probabilistik

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

4. Sistem Terbuka dan Sistem Tertutup

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

2.3 Konsep Dasar Informasi

Informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi penerima dan bermanfaat bagi pengambilan keputusan saat ini atau mendatang Mcleod (2004). Sedangkan pendapat Bodnar (2000) Informasi adalah data yang diolah sehingga dapat dijadikan dasar untuk mengambil keputusan yang tepat. Sementara menurut Sutabri (2004) Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Informasi dikelompokkan menjadi 3 bagian, yaitu:

1. Informasi Strategis

Informasi ini digunakan untuk mengambil keputusan jangka panjang, mencakup informasi eksternal, rencana perluasan perusahaan, dan sebagainya.

2. Informasi Taktis

Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.

3. Informasi Teknis

Informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stock, retur penjualan dan laporan kas harian.

2.3.1 Sumber Data Informasi

Sumber data informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal atau *data-item*. Data adalah kenyataan yang menggambarkan suatu kejadian dan kesatuan nyata. Kejadian adalah sesuatu yang terjadi pada saat tertentu. Dalam dunia bisnis, kejadian-kejadian yang terjadi adalah perubahan dari suatu nilai yang disebut dengan transaksi. Misalnya, penjualan adalah transaksi perubahan nilai barang menjadi nilai uang atau nilai piutang dagang. Kesatuan nyata adalah berupa objek nyata, seperti tempat, benda dan orang yang ada dan terjadi. Informasi merupakan hasil pengolahan dari sebuah model, formasi, organisasi, ataupun suatu perubahan bentuk dari data yang memiliki nilai tertentu dan dapat digunakan untuk menambah pengetahuan bagi yang menerimanya. Dalam hal ini, data bisa dianggap sebagai objek dan informasi adalah suatu subjek yang bermanfaat bagi penerimanya (Rusdiana, 2014).

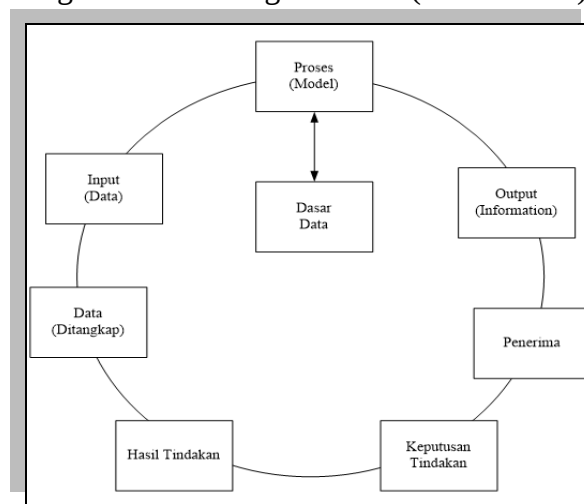
2.3.2 Siklus Informasi

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan. Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu. Di dalam dunia bisnis, kejadian-kejadian yang sering terjadi adalah transaksi perubahan dari suatu

nilai yang disebut transaksi. Kesatuan nyata adalah berupa suatu obyek nyata seperti tempat, benda dan orang yang betul-betul ada dan terjadi.

Data merupakan bentuk yang masih mentah, belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu metode untuk menghasilkan informasi. Data dapat berbentuk simbol-simbol semacam huruf, angka, bentuk suara, sinyal, gambar, dan sebagainya.

Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus informasi ini dapat digambarkan sebagai berikut (Gambar II.3):



Gambar II.4 Siklus Informasi
Sumber: Sutabri (2004)

2.4 Konsep Dasar Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari komponen *input*, komponen *model*, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran. Berikut ini adalah penjelasan dari masing-masing komponen yang terlibat didalamnya, yaitu Sutabri (2012):

1. Komponen *input*

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

2. Komponen model

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

3. Komponen *output*

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

4. Komponen teknologi

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

5. Komponen *hardware*

Hardware berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung *database* atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

6. Komponen *software*

Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari hardware untuk menciptakan suatu informasi.

7. Komponen basis data

Merupakan blok yang berisi definisi basis data yang disediakan untuk menyimpan data-data yang akan disimpan dalam media penyimpan. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya

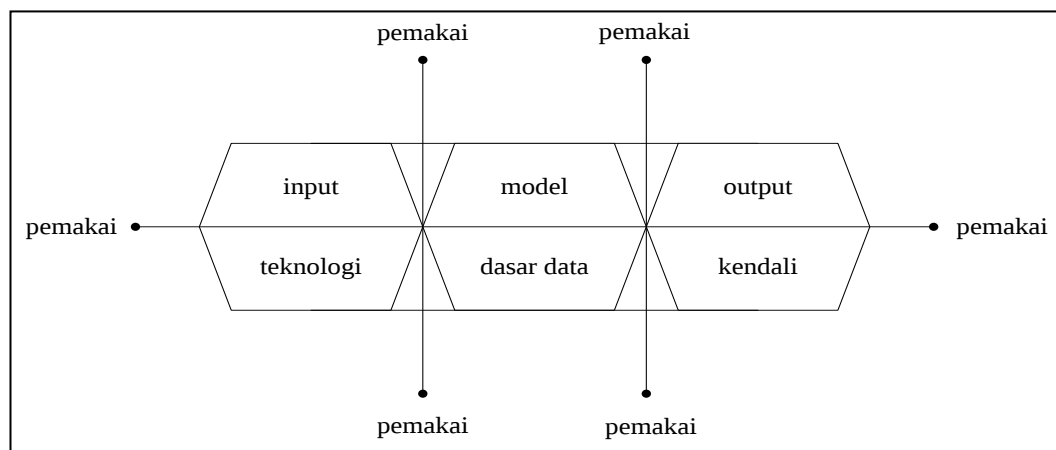
informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau di manipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

8. Komponen kontrol

Merupakan sebuah komponen yang bertugas mendefinisikan bagaimana kontrol terhadap sistem dilakukan sehingga sistem dapat berjalan dengan baik. Dalam blok kontrol ini misalnya didefinisikan bagaimana melindungi data yang ada di database agar selalu sama dengan kenyataan yang dicatat.

2.4.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah *building block* yaitu blok masukan, blok model, blok keluaran, blok teknologi, blok basis data, dan blok kendali. Sebagai suatu sistem, keenam blok tersebut membentuk suatu kesatuan untuk mencapai sasaran. Berikut ini merupakan blok sistem informasi yang berinteraksi (Gambar II.4):



Gambar II.5 Blok sistem informasi yang berinteraksi
(Sumber: Jogiyanto, 2005)

1. Blok Masukan

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

2. Blok Model

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

3. Blok Keluaran

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

4. Blok Teknologi

Teknologi merupakan *toolbox* dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*) perangkat keras (*hardware*). Teknisi dapat berupa orang-orang yang mengetahui teknologi dan membuatnya dapat beroperasi.

5. Blok Basis Data

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

6. Blok Kendali

Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperatur air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidakefisienan, sabotase dan lain sebagainya. Beberapa pengendalian

perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi. (Jogiyanto, 2005)

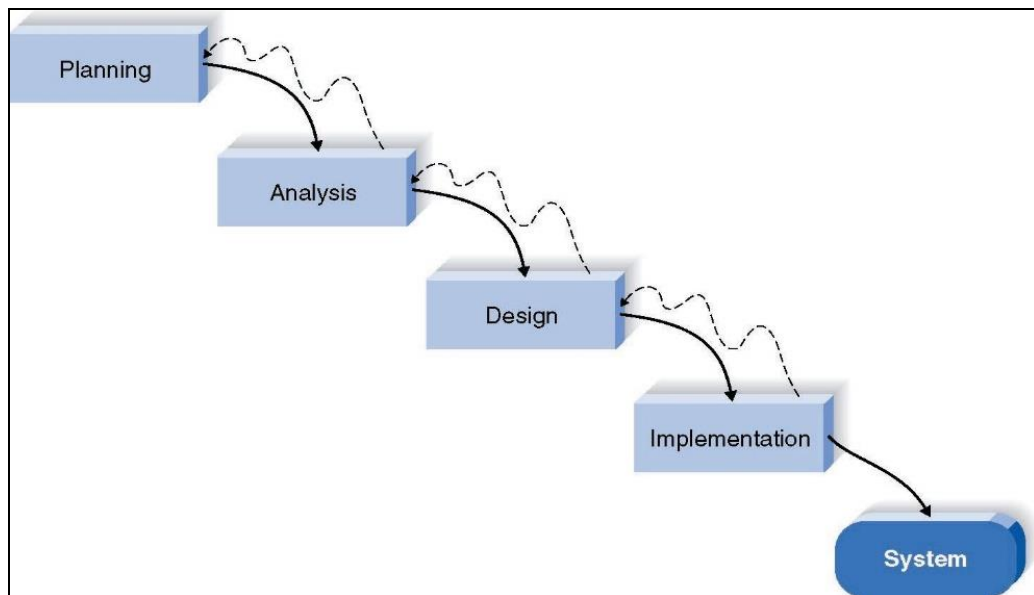
2.5 Siklus Hidup Pengembangan Sistem

Siklus hidup pengembangan sistem informasi merupakan sebuah model konseptual yang digunakan dalam manajemen proyek yang menggambarkan tahap-tahap yang terlibat dalam suatu proyek pengembangan sistem informasi, dari studi kelayakan awal melalui pemeliharaan aplikasi selesai. Secara konseptual siklus pengembangan sebuah sistem informasi adalah sebagai berikut:

1. Perancangan Sistem
Merancang output, input, struktur file, program, prosedur, perangkat keras dan perangkat lunak yang diperlukan untuk mendukung sistem informasi.
2. Analisis Sistem
Menganalisis dan mendefinisikan masalah dan kemungkinan solusinya untuk sistem informasi dan proses organisasi.
3. Pembangunan desain dan Testing Sistem
Membangun perangkat lunak yang diperlukan untuk mendukung sistem dan melakukan testing secara akurat. Melakukan instalasi dan testing terhadap perangkat keras dan mengoperasikan perangkat lunak
4. Implementasi Sistem
Beralih dari sistem lama ke sistem baru, melakukan pelatihan dan panduan seperlunya.
5. Evaluasi Sistem
Mengevaluasi sejauh mana sistem telah dibangun dan seberapa bagus sistem telah dioperasikan.
6. Operasi dan Perawatan
Mendukung operasi sistem informasi dan melakukan perubahan atau tambahan fasilitas.

2.6 Metodologi Pengembangan Sistem

Metode *Waterfall* adalah sebuah metode pengembangan sistem dimana antar satu fase ke fase yang lain dilakukan secara berurutan. Terdapat lima langkah tahapan dalam metode *waterfall* adalah sebagai berikut:



Gambar II.6 Metode *Waterfall*
(Sumber: Alan Dennis, 2010)

Adapun penjelasan dari tahapan-tahapan Metode *Waterfall* menurut Dennis et al (2010) tersebut sebagai berikut:

1. Perencanaan (*Planning*)

Dalam tahapan ini, menjelaskan dan mengargumentasikan untuk melanjutkan proyek yang telah dipilih, rencana kerja yang matang juga disusun untuk menjalankan tahapan-tahapan lainnya. Pada tahap ini ditentukan secara detail rencana kerja yang harus dikerjakan, durasi yang diperlukan masing-masing tahap, sumber daya manusia, perangkat lunak, dokumentasi, perangkat keras, maupun financial diestimasi. Pembuatan perencanaan ini bukan langkah mudah karena untuk mengestimasi beban kerja dan durasi dari masing-masing tahap dibutuhkan pengalaman cukup banyak. Kesalahan pada tahap ini akan mengakibatkan keuntungan yang

diperoleh tidak maksimal, bahkan bisa rugi. Pada tahapan ini peran manajemen sistem informasi berpengalaman yang dibutuhkan.

2. Analisis (*Analysis*)

Tahap kedua, adalah tahap analisis, yaitu tahap dimana kita berusaha mengenali segenap permasalahan yang muncul pada pengguna dengan mendekomposisi dan merealisasikan komponen-komponen sistem. Tujuan utama dari tahap analisis adalah untuk memahami dan mendokumentasikan kebutuhan bisnis dan persyaratan proses dari sistem baru. Menganalisa kebutuhan sebagai bahan dalam membuat spesifikasi di tahapan selanjutnya.

3. Perancangan (*Design*)

Tahap perancangan (*design*) dimana kita mencoba mencari solusi permasalahan yang didapat dari tahap analisis. Tahapan mengubah kebutuhan yang masih berupa konsep menjadi spesifikasi sistem yang nyata untuk diimplementasikan. Jika pada tahapan analisis (*form requirement to specification*), maka tahapan desain adalah (*form specification to implementation*). Jadi, bagaimana pembuatan spesifikasi yang detail untuk bisa diimplementasikan.

4. Implementasi (*Implementation*)

Tahap implementasi, dimana kita mengimplementasikan perancangan sistem ke situasi nyata atau desain harus diterjemahkan ke dalam bentuk mesin yang bisa dibaca. Disini kita mulai berurusan dengan pemilihan perangkat keras dan penyusunan perangkat lunak aplikasi (*pengkodean/coding*).

5. Sistem

Pada tahapan sistem dilakukan pengujian (*testing*) dan pemeliharaan, yang dapat digunakan untuk menentukan apakah sistem/perangkat lunak yang kita buat sudah sesuai dengan kebutuhan pengguna atau belum. Jika belum, proses selanjutnya adalah bersifat *iterative*, yaitu kembali ke tahap sebelumnya. Tahap pemeliharaan dan perawatan dimana kita mulai melakukan pengoperasian sistem dan jika diperlukan melakukan perbaikan-

perbaikan kecil. Kemudian jika waktu pengguna sistem habis, maka kita akan masuk lagi pada tahap perencanaan (*design*).

2.7 Konsep Dasar Pengendalian

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

Pengendalian menurut Welsch, Hilton, dan Gordon yang diterjemahkan oleh Purwatiningsih dan Warouw (2003) adalah suatu proses untuk menjamin terciptanya kinerja yang efisien yang memungkinkan tercapainya tujuan perusahaan. Menurut Syamsi (1983), pengendalian adalah fungsi manajemen yang mengusahakan agar pekerjaan atau kegiatan terlaksana sesuai dengan rencana, instruksi, pedoman, patokan, pengaturan, atau hasil yang telah ditetapkan sebelumnya.

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

2.8 Proses Pengendalian

Menurut Glenn, Hilton, dan Gordon (1988) untuk mengawasi kegiatan periodik dalam suatu usaha dari tiap pusat pertanggungjawaban mempunyai tahap sebagai berikut:

1. Bandingkan kinerja aktual dengan tujuan dan standar yang direncanakan secara periode.
2. Menyiapkan laporan kinerja yang menunjukkan hasil aktual, hasil yang direncanakan, dan perbedaan antara keduanya.

3. Menganalisis jenis dan tindakan lain untuk mengetahui penyebab terjadinya ketidaksesuaian atau penyimpangan.
4. Mengembangkan penyebab lain dari tindakan untuk memperbaiki kesalahan dan belajar dari keberhasilan.
5. Membuat pilihan, tindakan koreksi, perbaikan dari serangkaian alternatif dan menerapkannya.
6. menindaklanjuti penilaian atas efektivitas perbaikan, memprosesnya untuk melakukan perencanaan ulang.

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

2.9 Konsep Dasar Kualitas

Menurut *Gasperz* (1998), definisi dari kualitas adalah konsistensi peningkatan atau perbaikan dan penurunan variasi karakteristik dari suatu produk atau jasa yang dihasilkan, agar memenuhi kebutuhan yang telah dispesifikasikan, guna meningkatkan kepuasan pelanggan *internal* maupun *eksternal*. Pengertian kualitas dalam konteks pengendalian proses statistik adalah bagaimana baiknya suatu output (barang dan/atau jasa) itu memenuhi spesifikasi dan toleransi yang ditetapkan oleh bagian desain dari suatu perusahaan. Spesifikasi dan toleransi yang ditetapkan oleh bagian desain produk yang disebut sebagai kualitas desain (*quality of design*) harus berorientasi pada kebutuhan atau keinginan konsumen (orientasi pasar).

Definisi kualitas sebagai tingkat atau level bagaimana sebuah sistem, komponen, atau proses dalam memenuhi permintaan yang diminta pengguna, dan kebutuhan atau ekspektasi pengguna.

Menurut *Shikawa* (1993), kualitas berarti kepuasan pelanggan. Menurut *Crosby* (2001), kualitas adalah kesesuaian terhadap persyaratan (*to requirment*). Menurut Kamus Besar Bahasa Indonesia, kualitas diartikan sebagai (1) tingkat baik buruknya sesuatu atau (2) derajat atau taraf mutu. Berdasarkan teori-teori di atas yang dimaksud dengan kualitas adalah segala sesuatu yang berhubungan

dengan produk, jasa, manusia, proses dan lingkungan yang memenuhi atau melebihi harapan.

2.10 Konsep Dasar Pengendalian Kualitas

Pengendalian kualitas merupakan teknik yang sangat bermanfaat agar suatu perusahaan dapat mengetahui kualitas produknya sebelum dipasarkan kepada konsumen. Teknik pengendalian kualitas dapat membantu perusahaan dalam mengetahui kelayakan kualitas produk berdasarkan batas-batas kontrol yang telah ditentukan, (Assauri, 2013).

2.10.1 Pengertian Pengendalian Kualitas

Berikut ini merupakan pengertian pengendalian menurut para ahli:

1. Menurut Gasperz (2005)

Pengendalian adalah sebagai kegiatan yang dilakukan untuk memantau aktivitas dan memastikan kinerja sebenarnya yang dilakukan telah sesuai dengan yang direncanakan.

2. Menurut Assauri (2013)

Pengendalian dan pengawasan adalah kegiatan yang dilakukan untuk menjamin agar kegiatan produksi dan operasi yang dilaksanakan sesuai dengan apa yang direncanakan dan apabila terjadi penyimpangan, maka penyimpangan tersebut dapat dikoreksi sehingga apa yang diharapkan dapat tercapai.

Selanjutnya pengertian pengendalian kualitas dalam arti menyeluruh sebagai berikut:

1. Menurut Gasperz (2005)

Berdasarkan pengertian di atas, maka dapat disimpulkan bahwa pengendalian kualitas suatu teknik dan aktivitas atau tindakan yang terencana yang dilakukan untuk mencapai, mempertahankan dan meningkatkan kualitas suatu produk dan jasa agar sesuai dengan standar yang telah ditetapkan dan dapat memenuhi kepuasan konsumen.

2. Menurut Assauri (2013)

Pengawasan mutu merupakan usaha untuk mempertahankan mutu atau kualitas dari barang yang dihasilkan, agar sesuai dengan spesifikasi produk yang telah ditetapkan berdasarkan kebijaksanaan kepemimpinan perusahaan.

2.10.2 Tujuan Pengendalian Kualitas

Pengendalian kualitas memiliki tujuan tertentu. Adapun tujuan dari pengendalian kualitas menurut Assauri (1998) adalah:

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

Tujuan utama pengendalian kualitas adalah untuk mendapatkan jaminan bahwa kualitas produk atau jasa yang dihasilkan sesuai dengan standar kualitas yang telah ditetapkan dengan mengeluarkan biaya yang ekonomis atau serendah mungkin. Pengendalian kualitas tidak dapat dilepaskan dari pengendalian produksi karena pengendalian kualitas merupakan bagian dari pengendalian produksi. Pengendalian produksi baik secara kualitas maupun kuantitas merupakan kegiatan yang sangat penting dalam suatu perusahaan. Hal ini disebabkan karena semua kegiatan produksi yang dilaksanakan dan dikendalikan, supaya barang dan jasa yang dihasilkan sesuai dengan rencana yang telah ditetapkan, dimana penyimpangan-penyimpangan yang terjadi diusahakan serendah-rendahnya, (Assauri, 1998).

Pengendalian kualitas juga menjamin barang atau jasa yang dihasilkan dapat dipertanggungjawabkan seperti halnya pada pengendalian produksi. Dengan demikian antara pengendalian kualitas dan pengendalian produksi erat kaitannya dalam pembuatan barang, (Assauri, 1998).

2.10.3 Faktor Pengendalian Kualitas

Menurut Montgomery (2001), faktor yang mempengaruhi pengendalian kualitas adalah:

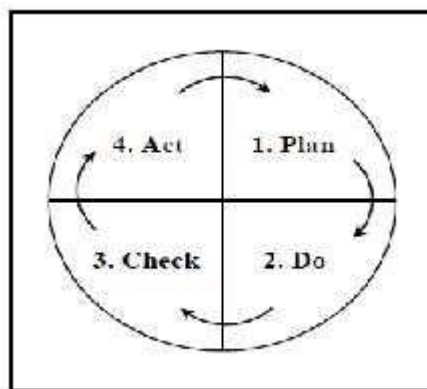
1. Kemampuan proses
Batas-batas yang ingin dicapai haruslah disesuaikan dengan kemampuan proses yang ada. Tidak ada gunanya mengendalikan suatu proses dalam batas-batas yang melebihi kemampuan atau kesanggupan proses yang ada.
2. Spesifikasi yang berlaku
Spesifikasi hasil produksi yang ingin dicapai harus dapat berlaku, bila ditinjau dari segi kemampuan proses dan keinginan atau kebutuhan konsumen yang ingin dicapai dari hasil produksi tersebut.
3. Tingkat ketidaksesuaian yang dapat diterima
Tujuan dilakukan pengendalian suatu proses adalah dapat mengurangi produk yang berada di bawah standar seminimal mungkin. Tingkat pengendalian yang diberlakukan tergantung pada banyaknya produk yang berada di bawah standar yang dapat diterima.
4. Biaya kualitas
Biaya kualitas sangat mempengaruhi tingkat pengendalian kualitas dalam menghasilkan produk dimana biaya kualitas mempunyai hubungan yang positif dengan terciptanya produk yang berkualitas.
 - a. Biaya pencegahan (*Prevention Cost*)
Biaya yang terjadi untuk mencegah terjadinya kerusakan produk.
 - b. Biaya Deteksi/Penilaian (*Detection/Appraisal Cost*)
Biaya yang timbul untuk menentukan apakah produk atau jasa yang dihasilkan telah sesuai dengan persyaratan-persyaratan kualitas sehingga dapat menghindari kesalahan dan kerusakan sepanjang proses produksi.
 - c. Biaya Kegagalan Internal (*Internal Failure Cost*)
Biaya yang terjadi karena adanya ketidaksesuaian dengan persyaratan dan terdeteksi sebelum barang atau jasa tersebut dikirim ke pihak luar (pelanggan).
 - d. Biaya Kegagalan Eksternal (*Eksternal Failure Cost*)

Biaya yang terjadi karena produk atau jasa tidak sesuai dengan persyaratan-persyaratan yang diketahui setelah produk tersebut dikirimkan kepada pelanggan.

2.11 Langkah Pengendalian Kualitas

Proses penjaminan kualitas suatu produk harus dilakukan secara terus - menerus dan berkelanjutan. Dalam proses penjaminan tersebut dilakukan suatu pengendalian kualitas yang dilakukan melalui penerapan *PDCA* (*Plan-Do-Check- Action*) yang diperkenalkan oleh Dr. W. Edward Deming, seorang pakar kualitas ternama, sehingga siklus ini disebut Siklus Deming (*Deming Wheel*). Siklus ini digunakan untuk mencoba dan mengimplementasikan perubahan-perubahan untuk memperbaiki kinerja produk, proses atau suatu sistem di masa yang akan datang.

Siklus *PDCA* dapat dilihat pada Gambar II.3 berikut:



Gambar II.7 Siklus *PDCA*
(Sumber: Nasution, 2005)

Tahapan dalam siklus *PDCA* adalah sebagai berikut (Nasution, 2005):

1. Menegembangkan rencana (*Plan*)
Merencanakan spesifikasi, menetapkan spesifikasi atau standar kualitas yang baik, memberi pengertian kepada bawahan akan pentingnya kualitas produk, pengendalian kualitas dilakukan secara terus-menerus dan berkesinambungan.
2. Melaksanakan rencana (*Do*)

Rencana yang telah disusun diimplementasikan secara bertahap, mulai dari skala kecil dan pembagian tugas secara merata sesuai dengan kapasitas dan kemampuan dari setiap personil. Selama dalam melaksanakan rencana harus dilakukan pengendalian, yaitu mengupayakan agar seluruh rencana dilaksanakan dengan sebaik mungkin agar sasaran dapat tercapai.

3. Memeriksa dan meneliti hasil yang dicapai (*Check*)

Memeriksa atau meneliti merujuk pada penetapan apakah pelaksanaannya berada dalam jalur, sesuai dengan rencana dan memantau kemajuan perbaikan yang direncanakan. Membandingkan kualitas hasil produksi dengan standar yang telah ditetapkan, berdasarkan penelitian diperoleh data kegagalan dan kemudian ditelaah penyebab kegagalannya.

4. Melakukan tindakan penyesuaian bila diperlukan (*Action*)

Penyesuaian dilakukan bila dianggap perlu, yang didasarkan hasil analisis di atas. Penyesuaian berkaitan dengan standarisasi prosedur baru guna untuk menghindari timbulnya kembali masalah yang sama atau menetapkan sasaran baru bagi perbaikan berikutnya.

Untuk melaksanakan pengendalian kualitas, terlebih dahulu perlu dipahami beberapa langkah dalam melaksanakan pengendalian kualitas. Menurut (Schroeder, 2003) dalam buku *Operations Management* (Heizer, 2006), untuk mengimplementasikan perencanaan, pengendalian dan pengembangan kualitas diperlukan langkah-langkah sebagai berikut:

1. Mendefinisikan karakteristik (atribut) kualitas.
2. Menentukan bagaimana cara mengukur setiap karakteristik.
3. Menetapkan standar kualitas.
4. Menetapkan program inspeksi.
5. Mencari dan memperbaiki penyebab kualitas yang rendah.

2.12 Pengertian Bahan Baku

Bahan baku adalah zat atau benda yang dari padanya sesuatu dapat dibuat, atau barang yang dibutuhkan untuk membuat sesuatu. Bahan baku menjadi sebuah masukan dalam produksi. Jadi bahan baku adalah bahan mentah yang belum diproses, tetapi kadang kala telah diproses sebelum digunakan untuk proses produksi lebih lanjut. Umumnya, dalam masyarakat teknologi maju, material adalah bahan konsumen yang belum selesai. Beberapa contohnya adalah besi, tembaga, aluminium, kertas dan sutra dan yang lainnya (Ashby, 2005).

Menurut (Masiyal Kholmi 2003:172) bahan baku memiliki beberapa faktor yang perlu diperhatikan, yaitu:

1. Perkiraan pemakaian

Merupakan perkiraan tentang jumlah bahan baku yang akan digunakan oleh perusahaan untuk proses produksi pada periode yang akan datang. Pemakaian bahan baku harus diperhitungkan dengan tepat agar proses produksi berjalan dengan lancar. Jika penggunaan bahan baku berlebihan maka persediaan bahan baku tidak akan mencukupi untuk memenuhi jumlah pemesanan yang diminta.

2. Harga bahan baku

Merupakan dasar penyusunan perhitungan dari perusahaan yang harus disediakan untuk investasi dalam bahan baku tersebut.

3. Biaya-biaya persediaan

Merupakan biaya-biaya yang dibutuhkan oleh perusahaan untuk pengadaan bahan baku.

4. Kebijakan pembelian

Merupakan faktor penentu dalam menentukan berapa besar persediaan bahan baku yang akan mendapatkan dana dari perusahaan.

5. Pemakaian sesungguhnya

Merupakan pemakaian bahan baku yang sesungguhnya dari periode lalu dan merupakan salah satu faktor yang perlu diperhatikan.

6. Waktu tunggu

Merupakan tenggang waktu yang tepat maka perusahaan dapat membeli bahan baku pada saat yang tepat pula, sehingga resiko penumpukan ataupun kekurangan persediaan dapat ditekan seminimal mungkin.

2.13 Macam – Macam Bahan Baku

Bahan baku yang diperlukan dalam proses produksi bermacam – macam. Menurut Gunawan Adisaputro dan Marwan Asri bahan baku terbagi dua macam, yaitu :

1. Bahan baku langsung

Bahan baku langsung adalah semua bahan baku yang merupakan bagian daripada barang jadi yang dihasilkan. Sebagai contoh ketika kita ingin membuat komponen otomotif, maka bahan baku yang digunakan biasanya adalah lempengan baja ringan.

2. Bahan baku tidak langsung

Bahan baku tak langsung atau disebut juga dengan *indirect material*, adalah bahan baku yang ikut berperan dalam proses produksi tetapi tidak secara langsung tampak pada barang jadi yang dihasilkan seperti baut.

2.14 Flowchart

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

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

1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.

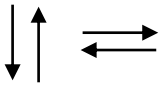
3. Harus ditunjukkan darimana kegiatan akan dimulai dan dimana akan berakhirnya.
4. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol-simbol bagan alir yang standar.

Simbol-simbol *flowchart* yang digunakan merupakan simbol-simbol *flowchart* standar yang dikeluarkan oleh ANSI dan ISO. Berikut simbol-simbol standar yang digunakan untuk menggambarkan diagram alir sesuai kegunaan symbol (Febriani, 2015):

1. *Flow Direction Symbols*

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

Tabel II.1 Simbol *Flow Direction*

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

(Sumber: Febriani, 2015)

2. Simbol Proses

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

Tabel II.2 Simbol Proses

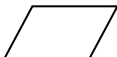
Simbol	Nama	Keterangan
	<i>Process</i>	Simbol yang menunjukkan pengolahan yang dilakukan komputer.
	<i>Decision</i>	Simbol untuk kondisi yang akan menghasilkan beberapa kemungkinan jawaban/aksi.
	<i>Predefined Process</i>	Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan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

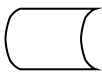
(Sumber: Febriani, 2015)

3. Simbol *Input* dan *Output*

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

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

Simbol	Nama	Keterangan
	<i>Input-Output</i>	Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.

Simbol	Nama	Keterangan
	<i>Document</i>	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak di kertas.
	<i>Disk and On-line Storage</i>	Simbol untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .

(Sumber: Febriani, 2015)

2.15 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Tujuan dari UML adalah untuk menyediakan kosa kata yang umum dari istilah-istilah berbasis objek dan teknik yang cukup banyak untuk memodelkan proyek pengembangan sistem dari analisis ke desain. Menurut Dennis (2011), diagram-diagram yang ada dalam UML terbagi menjadi dua bagian utama yaitu, *Struktur diagram* dan *Behavior diagram*. *Structure diagram* biasanya digunakan untuk merepresentasikan data dan hubungan statik yang ada di dalam sebuah sistem informasi. Sedangkan *Behavior diagram* menyediakan para analis dengan sebuah gambaran hubungan yang dinamis antara instansi atau objek yang merepresentasikan sistem informasi bisnis. Versi 2.0 ini sudah diterima oleh *Object Management Group* (OMG) pada tahun 2003. Versi UML ini mendefinisikan suatu set dari 14 teknik untuk pemodelan sistem. Berikut adalah tabel dari jenis diagram yang ada dan fungsi-fungsinya.

Tabel II.4 Jenis Diagram Pada UML dan Fungsi-fungsinya

Nama Diagram	Fungsi
<i>Structure Diagrams</i>	
<i>Class Diagram</i>	Menggambarkan hubungan antara pemodelan-pemodelan <i>class</i> di dalam sistem

Nama Diagram	Fungsi
Structure Diagrams	
<i>Object Diagram</i>	Menggambarkan hubungan antara pemodelan-pemodelan objek di dalam sistem
<i>Package Diagram</i>	Mengelompokkan elemen UML lainnya menjadi satu untuk membentuk level konstruksi yang lebih tinggi
<i>Deployment Diagram</i>	Menampilkan arsitektur fisik dari suatu sistem. Bisa juga digunakan untuk menunjukkan komponen <i>software</i> sedang dijadikan arsitektur fisik suatu sistem
<i>Component Diagram</i>	Menggambarkan hubungan fisik diantara komponen-komponen <i>software</i>
<i>Composite Diagram</i>	Menggambarkan struktur internal dari suatu <i>class</i> dan hubungan diantara bagian-bagian dari suatu <i>class</i>
<i>Activity Diagram</i>	Menggambarkan proses bisnis masing-masing <i>class</i>
<i>Sequence Diagram</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam aktifitas berdasarkan urutan waktu
<i>Communication Diagram</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam komunikasi antara satu set dari kolaborasi objek dari suatu objek
<i>Interaction Overview Diagram</i>	Menggambarkan suatu ikhtisar alur dari kontrol suatu proses
<i>Timing Diagram</i>	Menggambarkan interaksi yang terjadi diantara suatu set dari objek-objek dan perubahan keadaan selama perjalanan waktu
<i>Behavioral State Machine Diagram</i>	Memeriksa kebiasaan dari suatu <i>class</i>
<i>Protocol State Machine Diagram</i>	Menggambarkan ketergantungan diantara perbedaan-perbedaan <i>interface</i> dari suatu <i>class</i>
<i>Use Case Diagram</i>	Menangkap kebutuhan bisnis untuk sistem dan untuk menggambarkan interaksi diantara sistem dan lingkungannya

Sumber: Dennis (2011)

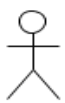
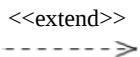
Berikut ini penjelasan singkat dari pembagian kategori UML tersebut:

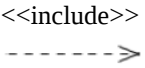
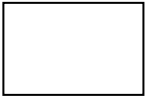
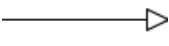
1. *Structure diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan proses sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

2.16 Use Case Diagram

Diagram *use case* dibuat untuk menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Suatu landasan pola pikir yang ditekankan dalam diagram *use case* ini adalah “apa” yang dapat diperbuat oleh sistem, dan bukan “bagaimana” sistem melakukannya. Diagram *use case* menggambarkan kebutuhan sistem dari sudut pandang *user*, yaitu proses yang dilakukan oleh sistem dalam melayani *user* yang berinteraksi dengan sistem tersebut. Simbol-simbol yang terdapat didalam *use case diagram* dijelaskan pada tabel dibawah sebagai berikut:

Tabel II.5 Simbol-simbol *Use Case Diagram*

No.	Simbol	Nama	Deskripsi
1		<i>Use case</i>	Merepresentasikan bagian utama dari sistem secara fungsional.
2		<i>Actor</i>	Seseorang atau sistem yang mendapatkkn keuntungan dari sistem
3		<i>Association</i>	Menghubungkan suatu <i>actor</i> dengan <i>use case</i> .
4		<i>Extend</i>	Merepresentasikan eksistensi dari <i>use case</i> untuk menyertakan perilaku

No.	Simbol	Nama	Deskripsi
			<i>optional.</i>
5		<i>Include</i>	Mewakili dimasukkannya satu <i>use case</i> ke <i>use case</i> yang lain.
6		<i>Subject</i>	Menyertakan nama subjek didalam maupun diatas
7		<i>Generalization</i>	Merepresentasikan <i>use case</i> khusus ke yang lebih umum.

(Sumber: Dennis, 2015)

2.17 Activity Diagram

Activity diagram digunakan untuk memodelkan perilaku dalam proses bisnis yang independen terhadap objek. Dalam banyak hal, *activity diagram* dapat dipandang sebagai data flow diagram canggih yang digunakan pada konjungsi dengan analisis terstruktur. Walaupun begitu, *activity diagram* tidak seperti data flow diagram, *activity diagram* menggunakan notasi yang mengalamatkan pemodelan paralel. (Dennis, 2010)

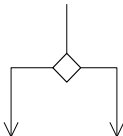
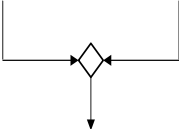
Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan actor, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
2. Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan

3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
4. Rancangan menu yang akan ditampilkan pada perangkat lunak

Berikut adalah simbol-simbol yang ada pada Activity Diagram:

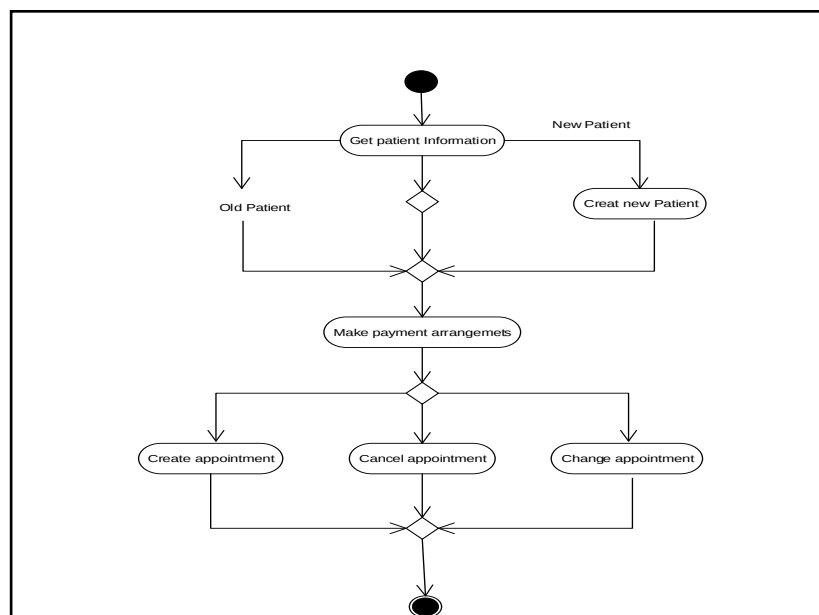
Tabel II.6 Simbol-simbol *Activity Diagram*

No.	Simbol	Deskripsi
1.	<i>Initial Node</i> 	Merupakan tanda awal dari sebuah aktifitas.
2.	<i>Activity</i> 	Merupakan sebuah gambaran aktifitas yang terjadi.
3.	<i>Decision Node</i> 	Pilihan untuk pengambilan keputusan.
4.	<i>Final Node</i> 	Merupakan notasi yang digunakan sebagai penanda diakhirinya sekumpulan aktivitas yang terdapat pada suatu proses bisnis.
5.	<i>Merge Node</i> 	Pilihan untuk menyatukan / menutup alur logika sebelumnya.

No.	Simbol	Deskripsi
6.	<i>Control flow</i> 	Menunjukkan urutan eksekusi
7.	<i>Object Flow</i> 	Menunjukkan aliran objek dari satu kegiatan (atau tindakan) untuk kegiatan lain (atau tindakan).

(Sumber: Dennis, 2015)

Berikut ini merupakan contoh penggambaran dari *activity diagram* beserta dengan penjelasannya pada masing-masing simbol dapat dilihat pada Gambar II.9 berikut ini.

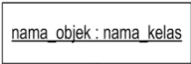


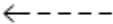
Gambar II.8 Contoh *Activity Diagram*
Sumber: Dennis (2011)

2.18 Sequence Diagram

Sequence diagram adalah salah satu dari dua jenis diagram interaksi. Mereka menggambarkan benda-benda yang berpartisipasi dalam kasus penggunaan dan pesan yang melewati antara mereka dari waktu ke waktu untuk satu use case. Sebuah diagram *sequence* adalah model dinamis yang menunjukkan urutan eksplisit pesan yang lewat di antara objek dalam interaksi didefinisikan. Karena urutan diagram menekankan pemesanan berbasis waktu kegiatan yang terjadi di antara set benda, mereka sangat membantu untuk memahami spesifikasi *real-time* dan kompleks menggunakan kasus (Alan Dennis, 2011). Berikut simbol yang ada pada Sequence Diagram:

Tabel II.7 Simbol-simbol *Sequence Diagram*

No	Simbol	Nama	Deskripsi
1.		Aktor	Orang atau sistem yang berasal dari manfaat dan eksternal ke sistem yang berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan
2.		Garis Hidup/ <i>lifeline</i>	Menyatakan kehidupan suatu objek.
3.		Objek	Berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan yang ditempatkan diatas diagram.
4.		<i>Execution Occurence</i>	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.
5.		<i>Message</i>	Pesan yang menggambarkan komunikasi yang terjadi antar objek.

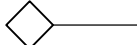
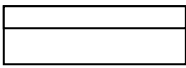
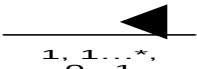
No	Simbol	Nama	Deskripsi
6.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri secara langsung.
7.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri.

(Sumber: Alan Dennis, 2011)

2.19 Class Diagram

Sebuah diagram kelas adalah model statis yang menunjukkan kelas dan hubungan antar kelas yang tetap konstan dalam sistem dari waktu ke waktu. Sebuah *class* merupakan deskripsi dari sekumpulan objek yang memiliki properti (*attribute*), operasi (*method*), relasi (*association*), dan tingkah laku (*behavior*) yang sama. Sebuah class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda /fungsi). (Alan Dennis, 2011). Berikut simbol-simbol yang ada pada diagram kelas:

Tabel II.8 Simbol-simbol *Class Diagram*

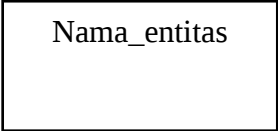
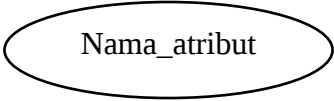
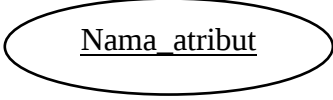
No	Simbol	Nama	Deskripsi
1		<i>Generalization</i>	Merupakan sebuah <i>taxonomic relationship</i> antara <i>class</i> yang lebih umum dengan <i>class</i> yang lebih khusus
2		<i>Aggregation</i>	Menggambarkan suatu <i>class</i> terdiri dari <i>class</i> lain atau suatu <i>class</i> adalah bagian dari <i>class</i> lain.
3		<i>Class</i>	Kelas pada struktur sistem
4		<i>Association</i>	Asosiasi yang menghubungkan <i>class</i> dengan <i>class</i> <i>Multiplicity</i> .

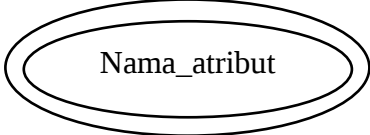
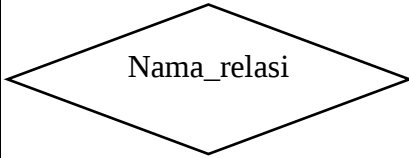
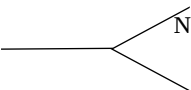
(Sumber: Alan Dennis, 2011)

2.20 Entity Relationship Diagram (ERD)

Menurut Rosa dan Shalahuddin (2014), ERD adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Simbol-simbol yang digunakan dalam ERD dapat dilihat pada Tabel II.9

Tabel II.9 Simbol-simbol *Entity Relationship Diagram (ERD)*

No.	Simbol	Deskripsi
1.	Entitas 	Entitas merupakan data inti yang akan disimpan; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi computer; penamaan entitas biasanya lebih ke benda.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan.

No.	Simbol	Deskripsi
4.	Atribut multivalai 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja.
6.	Asosiasi 	Penghubung antara relasi dan entitas dimana kedua ujungnya memiliki kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan lainnya yang disebut kardinalitas.

(Sumber: Rosa dan Shalahuddin, 2014)

2.21 Kamus Data Dan Database

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.

Database (basis data) merupakan kumpulan dari *file-file* yang saling berelasi, dimana relasi tersebut ditunjang dengan kunci dari setiap *file* yang ada. (Kristanto, 2004). *Database* dapat didefinisikan dalam sejumlah sudut pandang seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Himpunan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.

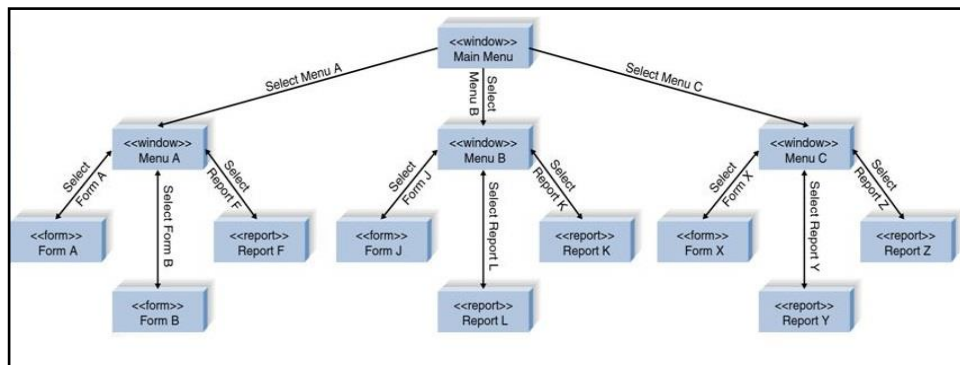
Kumpulan *file* atau tabel atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.22 *Windows Navigation Diagram (WND)*

Windows Navigation Diagram (WND) menurut Dennis (2011):

1. Menunjukkan semua layar, formulir dan laporan terkait.
2. Menunjukkan bagaimana pengguna bergerak dari satu *form* ke *form* lainnya
3. Seperti diagram keadaan untuk *user interface*
 - Kotak mewakili komponen
 - Panah mewakili transisi
 - Stereotip menunjukkan tipe antarmuka

Contoh WND menurut Dennis (2011) dapat dilihat pada Gambar II.10.



Gambar II.9 Contoh Windows Navigation Diagram

Sumber: Dennis (2011)

2.23 CodeIgniter

CodeIgniter adalah sebuah *web application network* yang bersifat *open source* yang digunakan untuk membangun aplikasi php dinamis. *CodeIgniter* menjadi sebuah *framework* PHP dengan model MVC (*Model, View, Controller*) untuk membangun website dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web. Selain ringan dan cepat, *CodeIgniter* juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodenya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih *CodeIgniter* sebagai *framework* pilihannya. Karena kelebihan-kelebihan yang dimiliki oleh *CodeIgniter*, pembuat PHP Rasmus Lerdorf memuji *CodeIgniter* di frOSCon (Agustus 2008). *CodeIgniter* pertama kali dikembangkan pada tahun 2006 oleh Rick Ellie.

2.24 MariaDB

MariaDB adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan *MariaDB*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. *MariaDB* merupakan versi pengembangan terbuka dan mandiri dari *MySQL*. Sejak diakuisisinya *MySQL* oleh *Oracle* pada September 2010, Monty Program

sebagai penulis awal kode sumber *MySQL* memisahkan diri dari pengembangan dan membuat versi yang lebih mandiri yakni *MariaDB*.

2.25 Tipe Data MariaDB

MariaDB mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Tipe data yang ada pada MariaDB adalah:

1. Jenis *numerik*

Jenis data *numerik* dapat dilihat pada tabel II.10 berikut ini:

Tabel II.10 Tipe Data jenis *numerik* pada MariaDB

Data Type	Description
<i>TINYINT</i>	<i>A very small integer. The signed range is -128 to 127.</i>
<i>SMALLINT</i>	<i>A small integer. The signed range is -32768 to 32767.</i>
<i>MEDIUMINT</i>	<i>A medium integer. The signed range is -8388608 to 8388607.</i>
<i>INT</i>	<i>A normal size integer. The signed range is -2147483648 to 2147483647.</i>
<i>DEC, NUMERIC, FIXED</i>	<i>A packed "exact" fixed-point number.</i>
<i>DOUBLE</i>	<i>A Normal-size (double-precision) floating-point number.</i>
<i>BIT</i>	<i>A Bit field type.</i>

(Sumber: mariadb.com, 2019)

2 Jenis karakter

Jenis data karakter dapat dilihat pada table II.11 berikut ini :

Tabel II.11 Tipe Data jenis karakter pada MariaDB

Data Type	Description
<i>CHAR</i>	<i>A fixed-length string that is always right-padded with space to specified length when store. M represent to column length in characters. The range of M is 0 to 255.</i>
<i>VARCHAR</i>	<i>A variable-length string.</i>
<i>BINARY</i>	<i>A fixed-length binary byte string.</i>
<i>VARBINARY</i>	<i>A variable-length binary byte string.</i>
<i>TINYBLOB</i>	<i>A tiny binary large object up to 255 byte.</i>
<i>TEXT</i>	<i>A text column with a maximum length of 255 characters.</i>
<i>ENUM</i>	<i>An enumeration, or string object that can have one value chosen from list of values.</i>

(Sumber: mariadb.com, 2019)

3. Jenis tanggal atau waktu

Jenis data tanggal atau waktu dapat dilihat pada tabel II.12 berikut ini:

Tabel II.12 Tipe Data jenis tanggal atau waktu pada MariaDB

Data Type	Description
<i>DATE</i>	<i>The date time type YYYY-MM-DD.</i>
<i>TIME</i>	<i>Time format HH:MM:SS.ssssss.</i>
<i>DATETIME</i>	<i>Date and time combination displayed as YYYY-MM-DD HH:MM:SS..</i>
<i>TIMESTAMP</i>	<i>A timestamp in the format YYYY-MM-DD HH:MM:SS..</i>
<i>YEAR DATA</i> <i>DATE</i>	<i>A four-digit year..</i>

(Sumber: mariadb.com, 2019)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, dan memilih langkah-langkah sistematis. Metodologi penelitian mempelajari cara-cara melakukan pengamatan dengan pemikiran yang tepat secara terpadu melalui tahapan-tahapan yang disusun secara ilmiah untuk mencari, menyusun, serta menganalisis dan menyimpulkan data-data berdasarkan fakta-fakta secara ilmiah.

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

3.2 Jenis dan Sumber Data

Data yang diperlukan dalam penelitian ini ada dua jenis, yaitu data primer dan data skunder. Sumber dari data-data ini berasal dari tempat yang diamati pada praktek kerja lapangan pada PT Gemala Kempa Daya.

1. Data Primer

Data Primer adalah data yang diperoleh secara langsung dari perusahaan, dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data-data tersebut adalah data yang digunakan dalam proses rekrutmen karyawan diantaranya struktur organisasi, analisis sistem yang telah berjalan, proses bisnis sistem saat ini dan yang akan diusulkan, dan kebutuhan pengguna sistem.

2. Data Sekunder

Data sekunder adalah data yang diperoleh dari data yang tersedia dan telah terlebih dahulu dikumpulkan dan dilaporkan oleh orang lain, buku-buku dan kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian. Data

tersebut adalah data umum perusahaan, profil perusahaan, dan struktur organisasi perusahaan.

3.3 Metode Pengumpulan Data

Menurut Sugiyono (2013) teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Penelitian ini menggunakan beberapa teknik pengumpulan data, yaitu:

1. Studi Lapangan

Studi lapangan adalah usaha melakukan pengumpulan data secara langsung pada objek yang dilakukan dengan cara sebagai berikut:

- a. Pengamatan, tahap ini dilakukan secara langsung di departemen *Quality Assurance* dilini *Quality Assurance Receiving Inspection* pada proses pemeriksaan bahan baku di PT Gemala Kempa Daya dengan mengamati proses pemeriksaan bahan baku untuk produk *frame chassis* yang berjalan, pengamatan dilakukan mulai dari proses penerimaan bahan baku dari *supplier*, hingga proses pemeriksaan dan penyimpanan bahan baku ke gudang. Melalui teknik ini, data yang dibutuhkan diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.
- b. Wawancara, Pengambilan data dengan cara berdialog dan bertanya dengan karyawan departemen terkait tentang proses bisnis yang berjalan pada sistem pengendalian kualitas bahan baku. Sebagai data yang diperlukan untuk penyusunan tugas akhir.

2. Studi Kepustakaan

Mengumpulkan data dan menambah referensi dengan membaca buku-buku, literatur, artikel di internet atau sumber tertulis lain yang berhubungan dengan judul dan permasalahan guna melengkapi data yang diperlukan dalam penulisan laporan tugas akhir ini, agar dalam praktik dan teori tidak jauh berbeda.

3.4 Metode Pengembangan Sistem

Menurut Dennis (2015) *System Development Life Cycle* (SDLC) memiliki empat perangkat fase dasar yaitu *planning*, *analysis*, *design*, dan *implementation*. Beberapa cara dalam membangun sebuah sistem mungkin dapat mempunyai

perbedaan, tetapi hampir semua proyek memiliki elemen dari empat fase tersebut. Masing-masing fase tersebut tersusun dari beberapa langkah-langkah yang menghasilkan *deliverable* atau hasil kegiatan seperti beberapa dokumen spesifik dan file yang menjelaskan pemahaman tentang proyek. Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *waterfall*.

Metode *waterfall* sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*). Metode *waterfall* menyediakan pendekatan alur hidup perangkat lunak dengan terurut dari fase awal (*planning*) hingga fase akhir (*implementation*), dan pada setiap fase pihak pengembang dan sponsor akan selalu ada kontak untuk mendapatkan persetujuan sebelum masuk pada tahap selanjutnya (Dennis, 2010).

Tahapan-tahapan dalam *waterfall* adalah sebagai berikut:

- a. Identifikasi kebutuhan pengguna (*planning*)
Pengembang melakukan diskusi dengan *head section* divisi *Quality Assurance* tentang kebutuhan sistem yang digunakan.
- b. Analisis kebutuhan perangkat lunak (*Analysis*)
Pengembang melakukan proses pengumpulan kebutuhan yang dilakukan secara intensif dengan pegawai divisi *Quality Assurance* tentang kebutuhan sistem yang diinginkan.
- c. Desain (*Design*)
Pengembang membuat sebuah desain program perangkat lunak seperti struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean serta dokumentasinya dari sistem yang telah dijelaskan oleh *Staff Control Improvement* dan pegawai divisi *Quality Assurance*
- d. Pembuatan Kode Program (*Implementaion*)
Pengembang mengimplementasikan perancangan sistem ke situasi nyata atau desain harus diterjemahkan ke dalam suatu bahasa pemrograman yang dapat dibaca oleh mesin.

3.5 Kerangka Penelitian

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

1. Studi Pendahuluan

Pada tahap awal penelitian melakukan sebuah studi pendahuluan yaitu dengan membaca buku literatur, *browsing internet*, membaca artikel serta sumber-sumber lain dalam lingkup perkuliahan maupun di luar lingkup perkuliahan. semua yang berhubungan dengan judul dan permasalahan tugas akhir yang di ambil dan dibaca. Selain itu juga melakukan proses pemagangan yang dilakukan di divisi *Quality Assurance* di lini *Quality Assurance Receiving Inspection* dalam jangka waktu tiga bulan.

2. Identifikasi Masalah

Pada tahap ini dilakukan cara dalam mengidentifikasi suatu masalah yang terjadi pada divisi *Quality Assurance* yang berfokus pada proses pemeriksaan bahan baku dalam mengidentifikasi suatu masalah yaitu melakukan beberapa cara seperti:

- a. Melakukan wawancara dan observasi yang melibatkan PT Gemala Kempa Daya dan beberapa karyawan yang bekerja pada bagian QA sebagai sumbernya
- b. Mengumpulkan data-data proses bisnis yang berjalan,
- c. Menganalisis identifikasi permasalahan

3. Identifikasi Pemecahan Masalah

Setelah mengidentifikasi suatu masalah, maka akan melakukan pemecahan suatu masalah berdasarkan data-data yang telah di dapat pada tahap sebelumnya. Terdapat beberapa pemecahan masalah yang akan lakukan seperti:

- a. Menentukan tujuan penelitian serta batasan-batasan pada tugas akhir
 - b. Mengembangkan sistem dengan menggunakan metode *waterfall*, dengan alasan dokumen yang akan diolah akan terorganisir dengan baik. Hal ini dikarenakan setiap fase yang dilalui harus selesai semua terlebih dahulu sebelum masuk ke fase berikutnya.
4. Identifikasi Kebutuhan Pengguna (*Analysis*)
- Pada tahap ini akan melakukan identifikasi kebutuhan sistem dengan beberapa cara seperti:
- a. Mengumpulkan data-data pada proses bisnis yang berjalan, pengumpulan data-data permasalahan serta menganalisis dokumen masuk dan keluar pada proses pengendalian kualitas bahan baku.
 - b. Identifikasi proses bisnis yang berjalan lalu menganalisis proses bisnis tersebut.
 - c. Identifikasi kebutuhan sistem informasi pemeriksaan kualitas bahan baku.
5. Desain
- Pada tahap ini melakukan perancangan dalam pembuatan sistem dengan menggunakan UML diagram yang bertujuan untuk memodelkan sistem. Dalam pembuatan UML terdapat beberapa diagram seperti:
- a. *Use case Diagram* yang bertujuan untuk mendeskripsikan sebuah interaksi antara *user* (aktor) dengan sistem yang akan di rancang.
 - b. *Use case Description* yang bertujuan untuk menjelaskan setiap *use case* yang telah dibuat secara lebih spesifik
 - c. *Activity Diagram* yang bertujuan untuk menggambarkan *workflow* atau aliran kerja pada proses bisnis.
 - d. *Class Diagram* yang bertujuan untuk mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat di antara objek-objek tersebut.
 - e. *Sequence Diagram* bertujuan untuk menggambarkan kelakuan atau perilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang diterima dan dikeluarkan antar objek.

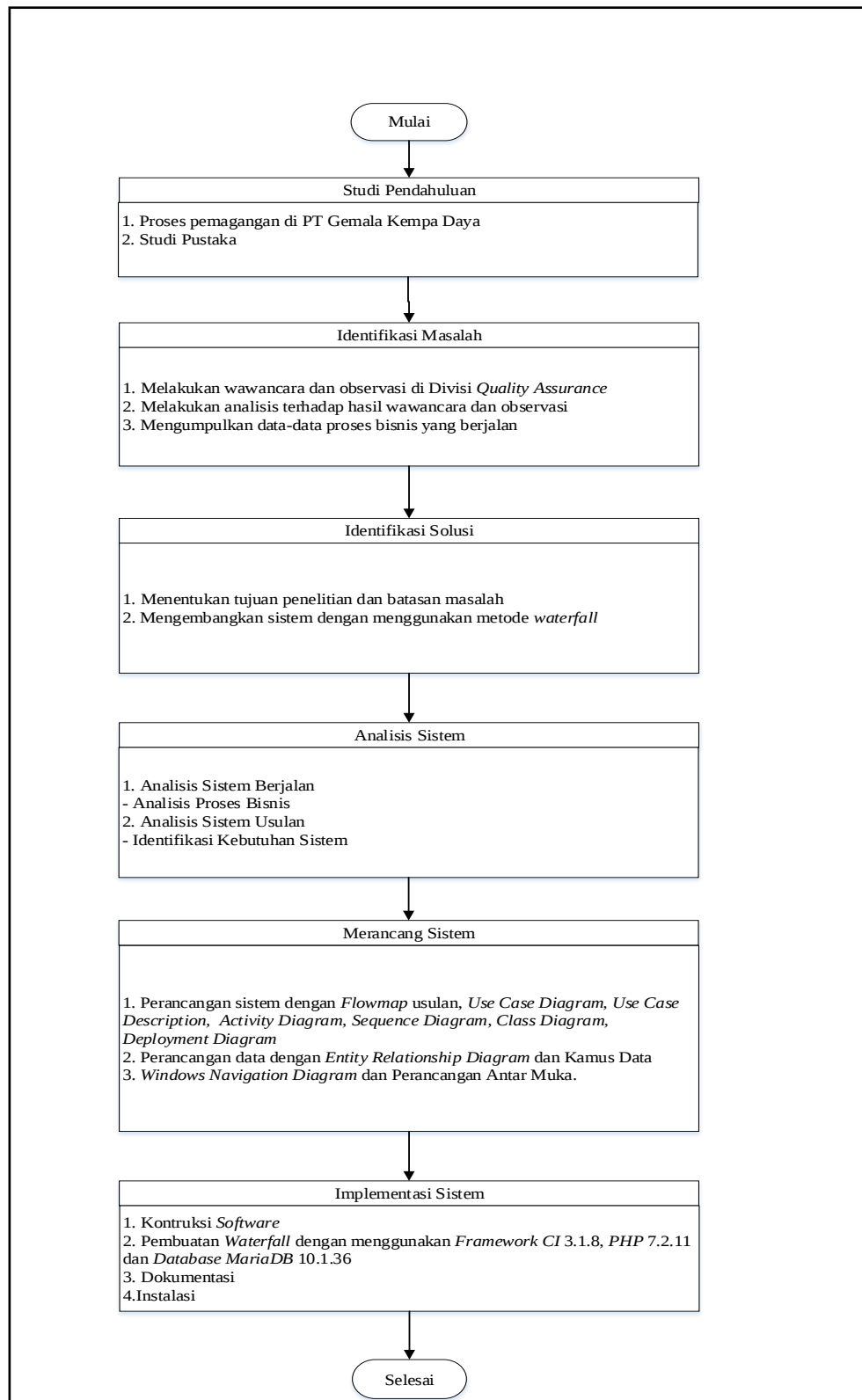
f. *Deployment Diagram* yang bertujuan untuk memvisualisasikan, menspesifikasikan dan mendokumentasi proses yang terjadi pada suatu sistem perangkat lunak yang akan dibangun

Selain itu dilakukan pembuatan *User Interface* dengan metode *Windows Navigation Diagram* (WND) yang bertujuan dalam memberikan suatu gambaran navigasi yang terdapat dalam sistem yang akan dibuat, pembuatan *Entity Relationship Diagram* (ERD) yang bertujuan untuk memodelkan suatu data dan hubungan data serta pembuatan kamus data dengan tujuan untuk memvalidasi diagram alir data dalam hal kelengkapan dan keakuratan data.

6. Implementasi Sistem

Pada tahap terakhir ini melakukan penulisan kode program dan pembuatan *database* menggunakan *CodeIgniter* versi 3.1.8 dan *MariaDB* versi 10.1.36 yang semuanya dijalankan melalui XAMPP v3.2.2. sesuai dengan analisis yang telah dilakukan pada tahap sebelumnya, selain itu dilakukan sebuah pengujian yang bertujuan untuk menguji coba apakah setiap fungsi atau tombol-tombol yang terdapat pada sistem dapat berjalan sesuai yang direncanakan atau diharapkan oleh penulis dan setelah itu penulis melakukan pembuatan laporan pengembangan sistem informasi penyimpanan bahan baku.

Kerangka penelitian yang telah dibuat untuk laporan Tugas Akhir ini dapat dilihat pada Gambar III.1



Gambar III.1 Kerangka Penelitian
(Sumber: Pengolahan Data, 2019)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Profil dan Sejarah Perusahaan

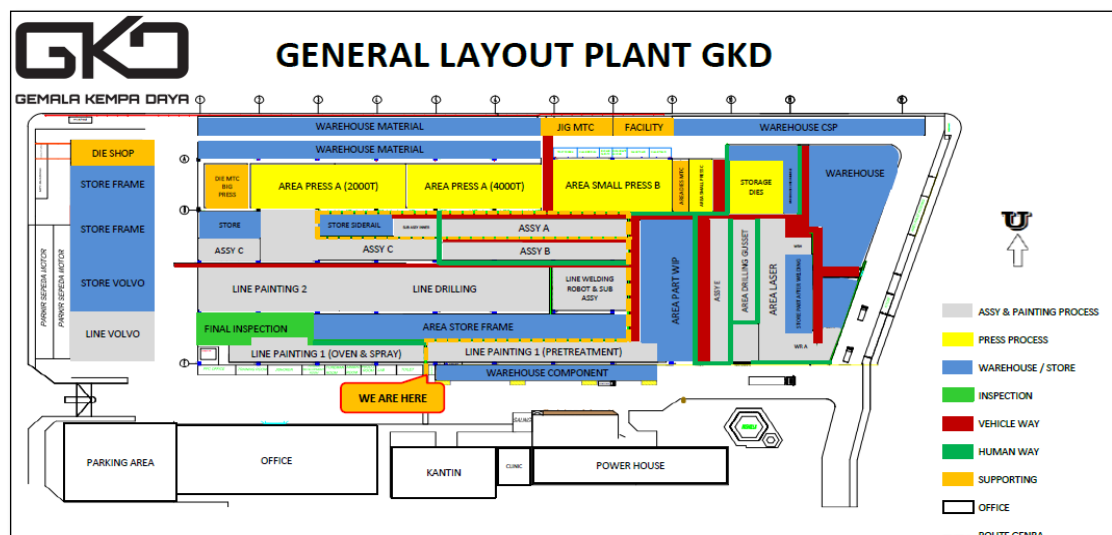


Gambar IV.1 Logo PT Gemala Kempa Daya
(Sumber : PT Gemala Kempa Daya, 2018)

IGP *Group* dimulai dengan berdirinya PT GKD pada tahun 1980 dengan *Frame Chasis* dan *Press Part* sebagai bisnis utamanya menjawab tantangan pasar PT Gemala Kempa Daya melengkapi sarana produksinya dengan mesin *press* 2000 ton dan 4000 ton. Data umum PT GKD adalah sebagai berikut:

Nama	: PT Gemala Kempa Daya
Alamat Pabrik	: Jalan Pegangsaan Dua Blok A1 KM 1.6, Kelapa Gading, Jakarta Utara 14250, Jakarta – Indonesia
Luas Area	: 45.353 m ²
Telepon	: (021) 4602755
Fax	: (021) 4602765
Tahun Berdiri	: 1980
Hasil Produksi	: 1. <i>Frame Chassis medium sized truck</i> 2. <i>Press Part</i>

PT Gemala Kempa Daya memiliki tata letak pabrik produksi yang tersusun dalam satu gedung produksi. Tata letak pabrik produksi dibuat guna mencapai pemanfaatan peralatan pabrik yang optimal, penggunaan jumlah tenaga kerja yang efektif, kebutuhan persediaan yang rendah, biaya produksi minimum, dan investasi modal yang rendah. Tata letak pabrik produksi terdiri dari beberapa stasiun kerja (*shop*) dan memiliki dua Gudang serta di bagi menjadi beberapa blok. Stasiun kerja pada pabrik produksi antara lain *Raw Material*, *Stamping*, *Assembling Proseses*, *Painting*, *Drilling*, *Welding Robot*, *Cutting Laser*, *Warehouse* dan *finish good*. Sedangkan pembagian blok Gudang seperti untuk bahan baku, bahan jadi dan komponen bahan baku.



Gambar IV.2 Tata Letak Pabrik Produksi
(Sumber : PT Gemala Kempa Daya, 2018)

Industri otomotif Indonesia mulai berkembang sejalan dengan kebijakan Penanaman Modal Dalam Negeri (PMDN) dan Penanaman Modal Asing (PMA), dalam rangka realisasi Program Pembangunan Lima Tahun I Republik Indonesia. PT Astra Internasional sebagai induk kelompok besar perusahaan Astra adalah salah satu pelopor industri otomotif Indonesia yang menangani berbagai merek dunia. Sebagai anggota kelompok perusahaan tersebut IGP *Group* telah berperan sebagai industri pendukung yang signifikan.

Sejarah PT GKD sejak berdiri hingga tahun 2018 adalah sebagai berikut:

1. 1980: PT Gemala Kempa Daya didirikan pada tanggal 19 Juni 1980.

2. 1983: PT Gemala Kempa Daya bergabung dengan IGP Group dengan tiga perusahaan lainnya yaitu: PT Inti Ganda Perdana (IGP), PT Tri Dharma Wisesa (TDW), PT Wahana Eka Paramitra (WEP)
3. 1990: PT Gemala Kempa Daya menambah fokus bisnisnya ke bisnis industri otomotif *press part* ditandai dengan dibuatnya area khusus untuk *press part*.
4. 2000: Seiring kebutuhan pelanggan yang terus meningkat PT Gemala Kempa Daya membeli mesin *press* terbesar di Asia Tenggara yaitu mesin *press* 4000 ton.
5. 2014: Kebijakan pemerintah DKI Jakarta yang mengharuskan semua pabrik di luar kawasan industri harus di relokasi maka PT GKD membuat pabrik kedua yaitu di daerah Cikampek.
6. 2016: *Plant* GKD Cikampek mulai di operasikan sebagai *plant press part*.

IGP Group terus melakukan pembenahan terutama dalam hal *Quality, Cost, Delivery* dan *Development*. Sebagai bagian dari proses adaptasi pada kondisi pasar global, khususnya dalam memenuhi kepuasan pelanggan. Sebagai manufaktur komponen otomotif, keberhasilan IGP Group ditandai dengan begitu banyaknya penghargaan yang telah diraih dari dalam maupun luar negeri, serta keberhasilan dalam meraih sertifikat standar mutu internasional seperti (*International Organization for Standardization*) ISO 14001, (*International Organization for Standardization*)/(*Technical Specification*) ISO/TS 16949 serta (*Occupational Health & Safety Assessment Series*) OHSAS 18001 untuk kesehatan dan keselamatan kerja, yang merupakan syarat mutlak untuk tetap bersaing di dunia internasional.

IGP Group juga menaruh perhatian besar pada kelestarian lingkungan di sekitar perusahaan, bukan karena mengikuti isu internasional, melainkan bentuk komitmen total IGP Group terhadap pembangunan industri yang mengutamakan kesehatan dan keselamatan kerja serta berwawasan lingkungan.

4.1.1 Visi & Misi

PT Gemala Kempa Daya dengan bisnis utamanya yaitu memproduksi *frame chassis & press part*, telah menetapkan misi untuk menjadi produsen *frame chassis*

dan *press part* yang dapat diandalkan, dengan visi untuk menjadi perusahaan dengan daya saing terbaik di wilayah global.

PT Gemala Kempa Daya adalah perusahaan industri komponen otomotif yang selalu berkomitmen yaitu sebagai berikut :

- a) Mengembangkan sumber daya
- b) Mencapai efektifitas dan efisiensi serta peduli lingkungan
- c) Memenuhi kepuasan pelanggan
- d) Mereduksi biaya dan mencegah polusi, kecelakaan kerja dan penyakit akibat kerja serta memenuhi peraturan pemerintah dan peraturan lainnya.

4.1.2 Produk

Produk unggulan PT Gemala Kempa Daya adalah *frame chassis* Kategori II (*medium sized truck*) . Untuk pemeliharaan kualitas produk, PT GKD di dukung secara teknis oleh pelanggan utama. Berikut mesin dan hasil produksi *frame chassis*, *supporting part* dan *press part* pada PT Gemala Kempa Daya. Dengan *supporting part* yang terdiri dari *inner front*, *bracket mounting*, *bracket bolt*, *bracket plate*, *side rail*, *cross member* dan *gusset*. Sedangkan *pressed part* terdiri dari *cover housing*, *housing*, *brake housing*, *backing plate*, *ring plate*. Mesin dan Produk *frame chassis* dapat dilihat pada Gambar IV.3, Tabel IV.1 dan Tabel IV.2

1. *Frame Chassis*



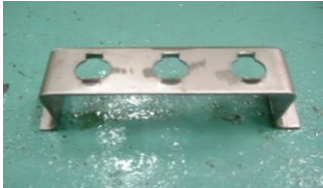
Gambar IV.3 *Frame Chassis*
(Sumber : PT Gemala Kempa Daya, 2018)

Tabel IV.1 Produk *supporting part*

Gambar	Nama Komponen	Fungsi
	<i>Inner Front</i>	Sebagai penerus putaran ke roda juga sebagai tempat <i>knuckle</i> agar roda bisa dibelok-belokan.
	<i>Bracket mounting</i>	Mencegah pergeseran horisontal yang sangat penting ketika truk membawa muatan pada jalan menanjak
	<i>Bracket Bolt</i>	Memberikan kontribusi untuk menjaga kendaraan tetap selaras dan karakteristik tumpuan untuk titik-titik vital kendaraan.
	<i>Bracket Plate</i>	Menahan konsentrasi tekanan yang akan menyebabkan sasis retak jika longgar.
	<i>Said Rial</i>	Penopang untuk badan mobil

(Sumber : PT Gemala Kempa Daya, 2018)

Tabel IV.2 Produk *supporting part* (Lanjutan)

Gambar	Nama Komponen	Fungsi
	<i>Said Rial</i>	Penopang untuk badan mobil
	<i>Cross Member</i>	Penyambung antar <i>Side rail</i>
	<i>Gusset</i>	sebagai penahan dan penyeimbang beban

(Sumber : PT Gemala Kempa Daya, 2018)

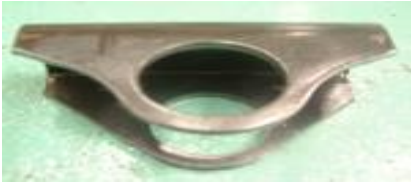
2. Kategori Produk *Pressed Part*

Kategori produk *pressed part* adalah produk buatan PT Gemala Kempa Daya yang terbuat dari mesin *press*. Produk yang dihasilkan dapat dilihat pada Gambar IV.4 dan Tabel IV.3



Gambar IV.4 Kategori Produk *Pressed Part Housing Rear Axle*
(Sumber : PT Gemala Kempa Daya, 2018)

Tabel IV.3 Kategori Produk *Pressed Part* (Lanjutan)

Gambar	Nama Komponen	Fungsi
	<i>Housing</i>	Merupakan komponen dari <i>Rear Axle</i> penggerak kendaraan yang digunakan sebagai rumah gardan
	<i>Brake Housing</i>	Salah satu komponen dari sistem pengereman
	<i>Dust Cover</i>	Sebagai penutup atau tempat untuk melindungi <i>Backing Plate</i>
	<i>Backing Plate</i>	Merupakan komponen dari sistem pengereman yang berfungsi sebagai rangka komponen rem tromol lainnya.

(Sumber : PT Gemala Kempa Daya, 2018)

Berikut beberapa *Customers* dari PT Gemala Kempa Daya yaitu :

1. PT Krama Yudha Berlian Motor – Mitsubishi
2. PT Toyota Motor Manufacturing Indonesia – Toyota
3. PT Astra Nissan Diesel Indonesia – Nissan UD
4. PT Hino Motor Manufacturing Indonesia – Hino
5. PT Inti Ganda Perdana – *Astra Group*
6. PT Tri Dharma Wisesa

4.1.3 Mesin Produksi

PT Gemala Kempa Daya memiliki dua mesin *press* yang unggulan sehingga menghasilkan produk yang unggul dan berkualitas. Berikut dua gambar mesin *press* yang dimiliki oleh PT GKD dapat dilihat pada Tabel IV.4

Tabel IV.4 Mesin produksi yang dimiliki oleh PT GKD

Gambar	Jenis Mesin	Fungsi
	Mesin Kawasaki 4000 Ton	Untuk memproduksi <i>frame chassis</i>
	Mesin Hitachi Zosen 2000 Ton	Untuk memproduksi <i>pressed part</i>

(Sumber : PT Gemala Kempa Daya, 2018)

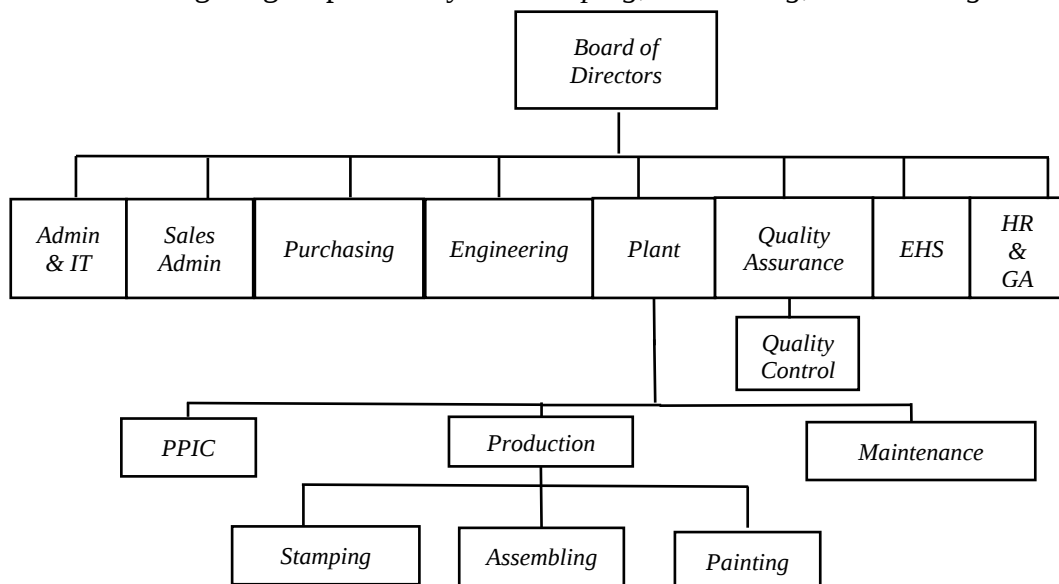
4.2 Struktur Organisasi dan Deskripsi Tugas

Dalam suatu organisasi terdapat pembagian kerja untuk masing-masing bagian sehingga adanya pertanggungjawaban dari bagian tersebut agar perusahaan dapat berjalan secara terstruktur dan memiliki susunan birokrasi yang jelas. Hal ini dilakukan supaya tidak adanya kesalahpahaman antara bagian satu dengan bagian lainnya dalam menjalankan suatu tugas sehingga perusahaan dapat terus berjalan

dengan baik. Maka secara ringkas dapat disebutkan tugas dan wewenang dari setiap susunan struktur organisasi PT Gemala Kempa Daya secara umum sebagai berikut.

4.2.1 Struktur Organisasi PT Gemala Kempa Daya

PT Gemala Kempa Daya di pimpin oleh *BOD (Board of Directors)* yang membawahi delapan divisi yaitu *Admin & IT*, *Sales Admin*, *Purchasing*, *Engineering*, *plant*, *Quality Assurance*, *EHS (Environment, Health and Safety)*, *HR (Human Resources) & GA (General Affair) Division*. Dimana *Plant* membawahi tiga divisi yaitu *PPIC*, *Production*, dan *Maintenance*. Sedangkan *Production* membawahi tiga bagian produksi yaitu *Stamping*, *Assembling*, dan *Painting*.



Gambar IV.5 Struktur Organisasi PT Gemala Kempa Daya
(Sumber : PT Gemala Kempa Daya, 2018)

4.2.2 Deskripsi Tugas

1. *BOD (Board of Director)*
 - a. Menetapkan arah, sasaran dan tujuan jangka panjang perusahaan.
 - b. Menentukan dan menetapkan strategi serta kebijaksanaan dan pengembangan usaha.
 - c. Mengawasi kegiatan perusahaan secara keseluruhan.
 - d. Mengatur organisasi dengan menetapkan kebijakan dan tujuan yang luas.
 - e. Pemilihan, pengangkatan, mendukung dan meninjau kinerja kepala eksekutif.

- f. Menjamin ketersediaan sumber daya keuangan yang memadai.
- g. Menyetujui anggaran tahunan.
- h. Akuntansi kepada para pemangku kepentingan untuk kinerja organisasi.

2. *Admin & IT*

- a. Bertanggung jawab dalam hal *Technical Support* baik pada *hardware* maupun *software* yang digunakan oleh perusahaan.
- b. Bertugas mengatur dan mengontrol jaringan intranet dan internet yang ada di perusahaan.
- c. Keuangan untuk dipergunakan sebagai dasar pengambilan keputusan oleh manajer.
- d. Bertanggung jawab atas pengeluaran keuangan perusahaan.
- e. Mengatur dan mengendalikan semua bentuk laporan keuangan di perusahaan.
- f. Mengatur, mengendalikan dan menganalisa semua bentuk informasi keuangan untuk dipergunakan sebagai dasar pengambilan keputusan oleh manajer.
- g. Merencanakan sumber-sumber keuangan dan mengatur pengalokasian penggunaan dana.
- h. Bertanggung jawab untuk memberikan informasi keuangan.

3. *Quality Control*

- a. Bertanggung jawab dalam mengontrol kuantitas dan kualitas produk.

4. *Purchasing*

- a. Bertanggung jawab terhadap pengadaan barang sesuai dengan permintaan pelanggan.
- b. Bertanggung jawab dalam hal pembelian barang yang dibutuhkan untuk proses produksi.

5. *Engineering*

- a. Menyiapkan rekomendasi secara terinci atas usulan desain
- b. Menjamin bahwa semua isi dari kerangka acuan pekerjaan ini akan di penuhi dengan baik
- c. Membantu tim di lapangan dalam mengendalikan kegiatan kerja

6. *HR (Human Resources) & GA (General Affair) Division*
 - a. Mengembangkan program sumber daya manusia seperti *recruitment*, *training* dan pendidikan.
 - b. Merencanakan dan mengawasi sumber daya manusia untuk jangka pendek maupun jangka panjang.
 - c. Mengelola sumber daya manusia sesuai dengan kebijaksanaan perusahaan.
 - d. Bertanggung jawab terhadap fasilitas karyawan.
7. *Plant*
 - a. *Production* bertanggung jawab dalam aktivitas produksi yang berlangsung di dalam perusahaan.
 - b. *PPIC (Production Planning Inventory Control)* Memimpin dan bertanggung jawab untuk kegiatan pekerjaan di bagian *PPIC* dan bagian gudang. Pekerjaan aktivitas di *PPIC* termasuk pengendalian produksi perencanaan, dan kontrol pengiriman. Pekerjaan aktivitas di gudang, termasuk bahan baku yang masuk, penyimpanan, penyediaan, dan pengiriman.
 - c. *Maintenance* bertanggung jawab terhadap menjaga, merawat, dan memperbaiki peralatan-peralatan yang ada di pabrik.
8. *Quality Assurance*
 - a. Memiliki tugas pokok dalam perencanaan prosedur jaminan kualitas suatu produk
 - b. Menafsirkan dan menerapkan standar jaminan kualitas produk ataupun bahan baku dari *supplier*
 - c. Mengevaluasi kecukupan standar jaminan kualitas
 - d. Merancang sampel prosedur dan petunjuk untuk mencatat dan melaporkan data berkualitas
 - e. Meninjau pelaksanaan dan efisiensi kualitas dan inspeksi agar berjalan sesuai rencana, melaksanakan dan memantau pengujian bahan baku serta produk
9. *EHS (Environment, Health and Safety)*

- a. Bertanggung jawab mengenai lingkungan sekitar pabrik, kesehatan pegawai dan prosedur keselamatan dalam bekerja.
- b. Mencegah pencemaran lingkungan, kecelakaan kerja, dan penyakit akibat kerja.
- c. Melindungi kelestarian sumber daya dan asset perusahaan
- d. Melakukan perbaikan berkesinambungan di bidang lingkungan keselamatan dan kesehatan kerja.
- e. Memenuhi peraturan pemerintah dan persyaratan lain-lain yang terkait lingkungan, keselamatan dan kesehatan kerja.

4.3 Komposisi & Jumlah Karyawan

Berdasarkan data rekapitulasi tenaga kerja yang diperoleh dari *HRD* PT Gemala Kempa Daya per bulan Januari 2013 maka jumlah tenaga kerja sebanyak 1465 orang.

4.3.1 Sistem Kerja Karyawan

Sistem kerja yang ada di PT Gemala Kempa Daya dapat dibedakan menjadi karyawan regular (*non shift*) dan *shift*.

1. Jam Kerja Regular

Hari kerja kantor atau regular terdiri dari lima hari kerja dalam setiap minggu, yaitu Senin sampai Jumat, dengan pengaturan jam kerja dan jam kerja istirahat sebagai berikut :

Hari Senin – Kamis : 07.30-16.15 WIB

Istirahat : 11.45-12.30 WIB

Hari Jumat : 07.30-16.30 WIB

Istirahat : 11.45-12.45 WIB

Hari Sabtu, Minggu dan hari besar libur

2. Jam Kerja *shift*

PT Gemala Kempa Daya terdapat tiga kelompok *shift* yaitu *shift* pagi, *shift* sore dan *shift* malam. Pergantian *shift* yang ada di PT Gemala Kempa Daya setiap satu minggu sekali, *shift* ini terus menerus bergilir. Apabila karyawan mendapat

shift pagi, untuk satu minggu ke depan maka karyawan tersebut akan mendapat *shift* sore, begitupun dan seterusnya. Hari kerja untuk karyawan *shift* terdiri dari lima hari kerja dalam setiap minggu, yaitu hari Senin sampai dengan Jumat, dengan pengaturan jam kerja sebagai berikut :

Shift pagi : 07.30-16.15 WIB

Shift sore : 16.15-24.00 WIB

Shift malam : 24.00-07.30 WIB

1. Jam Kerja Lembur (*Overtime*)

Untuk waktu lembur, perusahaan memperkerjakan pekerja maksimal tujuh dalam sehari dengan perintah tertulis dari *Section Head* dengan persetujuan dari *Departemen Head*. Kehadiran setiap pekerja diatur dengan seksama oleh manajemen perusahaan, adapun peraturan tersebut antara lain :

- a) Pekerja melaksanakan sendiri pencatatan kehadirannya pada mesin pencatat pada waktu datang dan pulang, sebagai bukti kehadiran di tempat kerja.
- b) Pekerja yang tidak dapat masuk kerja, terlambat datang, pulang lebih awal, meninggalkan pekerjaan dalam jam kerja tanpa seizin atasannya akan dikenakan sanksi berupa potongan upah gaji pokok.
- c) Ketidakhadiran karena alasan sakit yang melebihi satu hari harus dibuktikan dengan surat keterangan sakit.
- d) Pekerja tidak dapat masuk kerja diluar jam kerja, kecuali telah mendapat surat tertulis dari atasannya.
- e) Pelanggaran oleh pekerja mengenai kehadiran atau meninggalkan tempat kerja lebih cepat akan dikenakan sanksi administrasi sesuai dengan aturan kedisiplinan.
- f) Apabila pekerja tidak masuk kerja selama lima hari kerja berturut-berturut tanpa keterangan secara tertulis dengan bukti yang sah dan telah dipanggil oleh perusahaan, maka ia dianggap mengundurkan diri dari pekerjaannya sesuai dengan Peraturan Menteri Tenaga Kerja (Permenaker No.03/MEN/1996)
- g) Pekerja yang meninggalkan pekerjaan dalam jam kerja, wajib meminta izin dari atasannya langsung dengan memberikan alasan.

4.3.2 Sistem Penggajian Karyawan

Sistem penggajian di PT Gemala Kempa Daya diberikan dua kali, yaitu :

- a) Pada akhir bulan
Setiap tanggal 25 karyawan PT Gemala Kempa Daya menerima gaji pokok. Gaji pokok yang ada di PT Gemala Kempa Daya sesuai UMR DKI Jakarta. Gaji yang diterima karyawan memiliki nominal tersendiri sesuai dengan jabatan yang dipegangnya.
- b) Pada tanggal 10
Setiap tanggal 10 para karyawan PT Gemala Kempa Daya menerima gaji, gaji ini sebagai uang transpor untuk karyawan. Selain uang transpor, uang lembur dan uang bonus juga diberikan pada tanggal 10. Uang lembur biasanya diterima oleh karyawan operator, sedangkan untuk uang bonus biasanya diterima oleh karyawan yang telah memiliki jabatan tertentu.

4.4 Sarana Pendukung

Selain penggunaan bahan baku guna untuk memperlancar jalannya proses produksi maka dibutuhkan juga sarana pendukung. Sarana pendukung tersebut antara lain berupa :

- a) Penyediaan Air
Kebutuhan air bersih di PT Gemala Kempa Daya dipenuhi dari PDAM (Perusahaan Daerah Air Minum). Penggunaan untuk pemakaian produksi 1.307 M³/bulan, sedangkan untuk pemakaian domestik 1.902 M³/bulan.
- b) Listrik
Kebutuhan listrik di PT Gemala Kempa Daya dipenuhi PLN (Perusahaan Listrik Negara) dengan daya 4.330 kVA (satuan yang digunakan untuk mengukur kapasitas *genset*) PT Gemala Kempa Daya mempunyai cadangan listrik dari *genset* sebesar 1250 kVA. Tenaga listrik dimanfaatkan untuk semua kegiatan di PT Gemala Kempa Daya, seperti proses produksi, penerangan, pemompaan air, perkantoran, dan lain-lain.
- c) Bahan Bakar

Untuk mendukung proses produksi, terutama dalam hal kelancaran kegiatan angkat-angkat produk maupun bahan baku dan transportasi. PT Gemala Kempa Daya memanfaatkan alat-alat seperti : *forklift*, *crane*, *pick-up*, *trailer*, truk dan mobil perusahaan yang menggunakan bahan bakar jenis cair berupa solar. Khusus untuk bahan bakar solar PT Gemala Kempa Daya menggunakan 25.760 liter/bulan.

4.5 Fasilitas Yang Dimiliki

Fasilitas dalam sebuah perusahaan sangat diperlukan dimana fasilitas-fasilitas ini akan mendukung jalannya kegiatan operasional perusahaan. PT Gemala Kempa Daya mempunyai fasilitas yang cukup lengkap yang dapat dimanfaatkan oleh karyawan-karyawannya dan tamu yang datang ke perusahaan. Fasilitas yang diberikan perusahaan adalah:

1. Kantor
Area kantor mempunyai fasilitas yang cukup lengkap yaitu adanya meja kerja, kursi, peralatan tulis, telepon untuk masing-masing *line*, *Air Conditioner*, *dispenser*, komputer untuk masing-masing karyawan, ruang rapat yang dilengkapi dengan meja dan papan tulis, papan pengumuman, mesin *fotocopy*, mesin *fax*, *printer*, loker dokumen, area serbaguna yang dilengkapi dengan meja dan kursi.
2. *Lobby*
Merupakan tempat untuk tamu menunggu karyawan atau orang kantor yang berkepentingan. Terdapat *receptionist* untuk menghubungi orang kantor yang ingin ditemui, serta kursi dan meja untuk menunggu.
3. Kantin
Merupakan tempat makan yang diberikan oleh perusahaan karena perusahaan memberikan makan siang kepada karyawan-karyawannya.
4. Area ibadah
Area ibadah disini berupa musholla yang ditempatkan dekat area kantin, masjid yang terletak di samping area atau dekat dengan gerbang masuk perusahaan.
5. Alat Pelindung Diri (APD)

Setiap karyawan area produksi maupun tamu yang akan ke area produksi selalu diberikan alat pelindung diri berupa *helmet*, topi, *safety shoes* untuk area tertentu, seragam yang tertutup (celana panjang dan baju seragam), masker, dan *air plank* untuk area bising.

6. *Toilet*

Terdapat *Toilet* pria dan wanita, dimana jumlah *Toilet* wanita lebih sedikit daripada *Toilet* pria. Hal ini dikarenakan 99% karyawan PT Gemala Kempa Daya adalah pria.

7. Parkir Mobil dan Motor

Terdapat tiga jenis parkir untuk parkir, yaitu parkir khusus kendaraan mobil, kendaraan karyawan dan kendaraan magang atau pkl. Untuk parkir khusus mobil terdapat di dalam perusahaan yang masuknya melalui gerbang utama perusahaan, sedangkan untuk kendaraan karyawan dan kendaraan magang berada di dalam perusahaan yang masuknya melalui gerbang samping perusahaan.

8. Tempat Istirahat Area Produksi

9. Tempat Pembuangan Sampah Sementara (TPSS)

10. Mesin Absensi

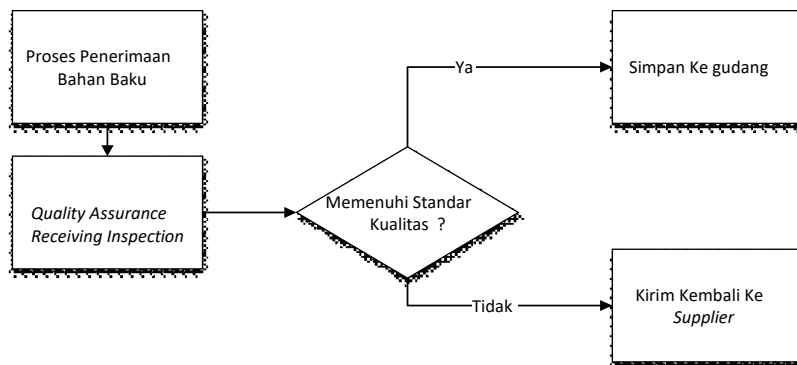
11. Ruang Loker

4.6 Divisi *Quality Assurance*

QA (Quality Assurance) pada PT Gemala Kempa Daya berperan dalam memeriksa kualitas bahan baku yang telah dipesan dari vendor, Selain itu *QA* juga melakukan penjaminan kualitas terhadap bahan baku serta menyatakan *OK/NG* terhadap bahan baku tersebut. Pada bab ini akan membahas tentang bagaimana proses penerimaan hingga pemeriksaan bahan baku di PT Gemala Kempa Daya berjalan. *Quality Assurance* merupakan suatu pendekatan yang berbasis proses dengan tujuan utamanya adalah mencegah bahan baku cacat mulai dari tahap penerimaan bahan baku (*Raw Material*) hingga sebelum tahap penyimpanan bahan baku ke Gudang sehingga menghindari terjadi pengerjaan ulang dan keluhan pada saat di produksi yang akan merugikan reputasi perusahaan serta pengeluaran biaya-biaya akibat

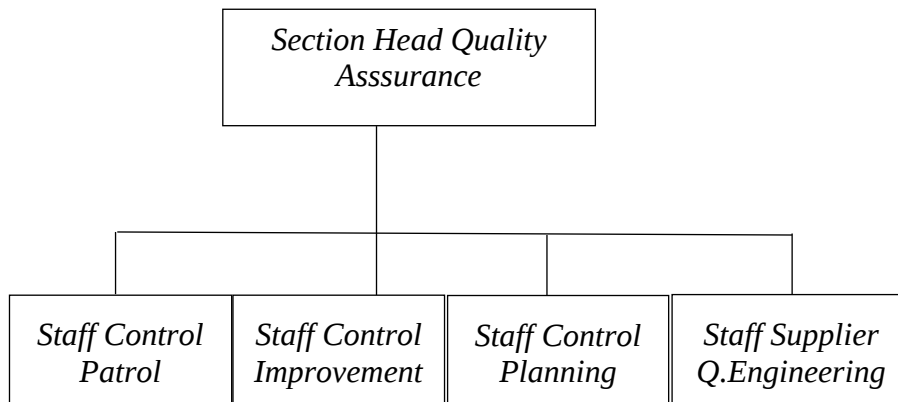
kualitas yang buruk. Berikut adalah tugas utama yang harus dilakukan divisi QA pada PT Gemala Kempa Daya :

- Menafsirkan, menerapkan serta perencanaan prosedur standar jaminan kualitas bahan baku
- Mengevaluasi kecukupan standar jaminan kualitas bahan baku
- Meninjau, melaksanakan dan memantau pemeriksaan dan inspeksi kualitas bahan baku agar menghasilkan produk yang unggulan



Gambar IV.6 Pengelolaan Bahan Baku Pada Divisi *Quality Assurance*
(Sumber : PT Gemala Kempa Daya, 2018)

4.6.1 Struktur Organisasi Divisi *Quality Assurance*



Gambar IV.7 Struktur Organisasi Divisi *Quality Assurance*
(Sumber : PT Gemala Kempa Daya, 2018)

4.6.2 Uraian Tugas Kerja dan Tanggungjawab

Berikut adalah uraian tugas dan tanggungjawab kerja dari masing-masing jabatan yang tercantum dari struktur organisasi diatas:

1. *Section Head Quality Assurance*

- a. Membantu dalam upaya pemberdayaan sumber daya bahan baku, kualitas produk serta perusahaan untuk mencapai tingkat kualitas dan efesiensi yang maksimal.
- b. Merencanakan dan memimpin pelaksanaan pemeriksaan kualitas bagi bahan baku.
- c. Menjembatani komunikasi antara *supplier* dengan perusahaan dalam hal kebutuhan bahan baku sesuai dengan yang diminta oleh perusahaan.
- d. Menyusun kebijakan kualitas bahan baku, prosedur kerja, instruksi kerja dan kelengkapan dokumen yang berhubungan dengan proses penerimaan dan pemeriksaan kualitas bahan baku.
- e. Mengevaluasi kinerja vendor dalam hal pengiriman bahan baku berdasarkan data lingkup perusahaan.

2. *Staff Control Patrol Melakukan patrol atas Problem Identification Corrective Action Claim & Particular Audit untuk line stamping.*

- a. *Follow up abnormality* yang terjadi di *line* produksi.
- b. *Backup* untuk *Shift* malam.
- c. *Judgement part Not Good/Non-Conforming Product* untuk *Press Part*

3. *Staff Control Improvement*

- a. Melaksanakan *monitoring & improvement part accuracy stamping*.
- b. Menganalisis masalah bahan baku dan pembuatan *Problem Identification Corrective Action*.
- c. Melakukan tindakan atas masalah *part stamping*.
- d. Melakukan *Pitch Diameter Circle part stamping*.
- e. Melakukan pengukuran *part component*.

4. *Staff Control Planning*

- a. Penetapan dan pengembangan tujuan dan kebutuhan untuk sebuah kualitas serta penerapan sistem kualitas yang telah ditetapkan sebelumnya
- b. Melakukan *Employment Cost Index (ECI)* dari pelanggan

- c. Melakukan dan menyusun data kualitas statistic (*Measurement Systems Analysis*).

4. *Staff Supplier Control Engineering*

- a. *Supplier Development*
- b. *Kontrol Quality dokumen Supplier*
- c. *Pengecekan Checksheet dari Supplier*
- d. *Judgement part NG untuk part dari Supplier*

4.7 **Proses Penerimaan Bahan Baku Pada PT Gemala Kempa Daya**

Setelah *purchasing* melakukan pemesanan maka proses selanjutnya adalah proses penerimaan bahan baku. Proses ini bertujuan untuk menerima dan memastikan bahan baku yang datang sesuai dengan laporan jadwal kedatangan. baik dalam jenis bahan baku, waktu kedatangan, jumlah bahan baku serta dokumen-dokumen penunjang lainnya. Bahan baku yang datang juga harus diperiksa apakah bahan baku tersebut sudah sesuai dengan spesifikasi yang diminta atau tidak. Urutan prosesnya adalah sebagai berikut:

1. *Supplier* akan mengirimkan material bahan baku serta surat jalan nya kepada *PPIC* untuk pemeriksaan jumlah atau *Quantity* yang di kirim oleh *Supplier* .
2. *Staff PPIC* memeriksa *Quantity* pada bahan baku dan menerima surat jalan
3. Jika sesuai maka *Staff PPIC* akan menandatangani dan memberikan stempel pada surat jalan kemudian surat jalan langsung diserahkan ke Departemen *Warehouse*.
4. Jika tidak sesuai maka membuat surat untuk retur barang serta menghubungi *Supplier* agar mengambil barang untuk dikembalikan dan ditandai dengan cat merah.
5. Setelah bahan baku di nyatakan sesuai dengan spesifikasi dan jumlah nya akan di serahkan kepada *Quality Assurance Receiving Inspection*.

DHJ PT. DUTA HITA JAYA
STEEL STRUCTURES & POLES FABRICATOR

SURAT JALAN

Kepada Yth.
PT. GEMALA KEMPA DAYA

No. SJ : 05081
Mobil :
Sopir : 33778
SPK No.:
PO. No.: 14997/PO-GKD/0918

Kami kirimkan barang-barang sebagai berikut :

No.	Nama Barang	Jumlah
1	JASA JASA PRESS BUSHING BRKT SPRING FRONT AH 31 LPR1	240 PCS

Barang-barang tersebut telah diterima dalam keadaan baik dan lengkap sesuai dengan jumlahnya.

Pengetahuan: Pengirim, Quality Control, Bag Gudang, Keamanan, Sopir

Nama Tgl. Nama Tgl. Nama Tgl. Nama Tgl. Nama Tgl. Nama Tgl.

PJ. Marketing, Man. Marketing, Bk. Adm. Keuangan, Hk. P. P. Kuning Customer

TAMBUK

Gambar IV.8 Surat Jalan Pengiriman Bahan Baku
(Sumber : PT Gemala Kempa Daya, 2018)

SURAT JALAN

TGL S.J : 10/10/2018
NO S.J : TJ18B10P0108
NO PO :
NO KEND : B.9557 LKCC

KEPADA YTH
PT. GEMALA KEMPA DAYA
DI PULOGADUNG

NO	NAMA BARANG	PACKING	JUML. BARANG	KETERANGAN
1	NIUT WELDING M10 H8 (9004A-17208)	6 DUS	10.000 PCS	6 DUS x 1500 Pcs + 2 KTG x 500
2	RIVET VOLVO D17 x 26.5 MM	50 BAG	20.000 PCS	50 BAG x 400 Pcs
3				
4				
5				
6				
7				
8				
9				
10				
11				

RECEIVED 10 OCT 2018

DIEMERKSA
GDG SUP SEC

HORMAT KAMI
BAG ORDER

RECEIVED 10 OCT 2018

RECEIVED 10 OCT 2018

Gambar IV.9 Surat Retur Bahan Baku
(Sumber : PT Gemala Kempa Daya, 2018)

4.8 Proses Pemeriksaan Bahan Baku Pada PT Gemala Kempa Daya

Setelah bahan baku diterima, bahan baku harus diperiksa apakah bahan baku tersebut sudah memenuhi standar spesifikasi yang ditetapkan. Jika spesifikasi bahan baku tidak memenuhi standar yang ada, maka bahan baku akan dikembalikan

ke *Supplier* untuk mendapatkan bahan baku pengganti. Berikut adalah langkah-langkah dalam memeriksa bahan baku.:

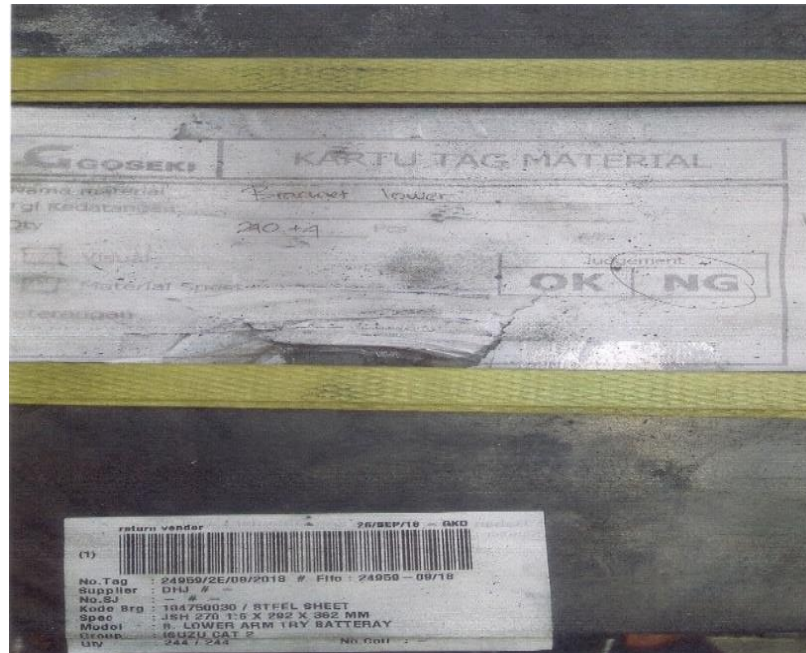
1. *Staff Control Patrol (QA)* melakukan pemeriksaan material atau *part*, apabila sudah sesuai dengan standar dan gambar, maka *Staff Control Patrol* memberi label *OK/NG* sesuai dengan kondisi *part* atau material.
2. *Staff QA* mempersiapkan sarana pemeriksaan
3. *Staff QA* melakukan pemeriksaan *part/material* sesuai *Check Sheet*
4. *Staff QA* melakukan pemeriksaan dengan acak minimal lima *pieces* atau dengan perbandingan 1 : 2 (10 *pieces* : 5 *pieces*)
5. Jika barang memiliki kualitas yang buruk maka akan diberikan label *NG (Not Good)* dengan ditandai dengan cat merah.
6. Jika barang memiliki kualitas yang baik maka akan diberikan label *OK*.
7. Hasil pemeriksaan dicatat di laporan sortir oleh *Staff Control Patrol*.
8. Hasil dari laporan sortir, selanjutnya bahan baku yang dilabel *NG* akan dipisahkan dan dikumpulkan untuk di simpan sementara untuk dibuatkan surat retur dikembalikan ke *Supplier*.
9. Setelah semua bahan baku yang telah diperiksa kualitas nya akan disimpan ke Gudang, maka *Staff Gudang* akan membuat laporan penerimaan barang.

PT Gemala Kempa Daya		CHECK SHEET		QCO / PRO		Date																																																																																	
Quality Inspection Department		Inspection		Inspection		1-May-08																																																																																	
Doc No	QI / QMS Form 1 / 01	Date	1-May-08	Shift	01	1-May-08																																																																																	
Rev	01 / QMS Form 1 / 01	Part Name	Part Gusset Cross Member no. 1, 1, 1, 1, 1	Part No	MS/01/01/01	Rev																																																																																	
Brand / Type	MS/01/01/01	Part No	MS/01/01/01	Part No	MS/01/01/01	Rev																																																																																	
Supplier	PT 100	Part No	MS/01/01/01	Part No	MS/01/01/01	Rev																																																																																	
		<table border="1"> <thead> <tr> <th>No.</th> <th>Particulars</th> <th>Inspection Method</th> <th>Inspection Result</th> <th>Remarks</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Appearance</td> <td>Visual</td> <td>OK</td> <td></td> </tr> <tr> <td>2</td> <td>Width</td> <td>V-Caliper</td> <td>40.0</td> <td></td> </tr> <tr> <td>3</td> <td>Pinch Hole</td> <td>Height Gauge</td> <td>10.0</td> <td></td> </tr> <tr> <td>4</td> <td>Length</td> <td>V-Caliper</td> <td>10.0</td> <td></td> </tr> <tr> <td>5</td> <td>Pinch Hole</td> <td>Height Gauge</td> <td>10.0</td> <td></td> </tr> <tr> <td>6</td> <td>Pinch Hole</td> <td>Height Gauge</td> <td>10.0</td> <td></td> </tr> <tr> <td>7</td> <td>Distance</td> <td>V-Caliper</td> <td>10.0</td> <td></td> </tr> <tr> <td>8</td> <td>Distance</td> <td>V-Caliper</td> <td>10.0</td> <td></td> </tr> <tr> <td>9</td> <td>Hole dia (1 per)</td> <td>V-Caliper</td> <td>10.0</td> <td></td> </tr> <tr> <td>10</td> <td>Distance</td> <td>V-Caliper</td> <td>10.0</td> <td></td> </tr> <tr> <td>11</td> <td>Thread M12 x 1.25 (1 per)</td> <td>Thread plug</td> <td>OK</td> <td></td> </tr> <tr> <td>12</td> <td>Thread M12 x 1.25 (1 per)</td> <td>Thread plug</td> <td>OK</td> <td></td> </tr> <tr> <td>13</td> <td>Hole dia</td> <td>V-Caliper</td> <td>10.0</td> <td></td> </tr> <tr> <td>14</td> <td>Distance</td> <td>Height Gauge</td> <td>10.0</td> <td></td> </tr> <tr> <td>15</td> <td>Hole dia (1 per)</td> <td>V-Caliper</td> <td>10.0</td> <td></td> </tr> </tbody> </table>						No.	Particulars	Inspection Method	Inspection Result	Remarks	1	Appearance	Visual	OK		2	Width	V-Caliper	40.0		3	Pinch Hole	Height Gauge	10.0		4	Length	V-Caliper	10.0		5	Pinch Hole	Height Gauge	10.0		6	Pinch Hole	Height Gauge	10.0		7	Distance	V-Caliper	10.0		8	Distance	V-Caliper	10.0		9	Hole dia (1 per)	V-Caliper	10.0		10	Distance	V-Caliper	10.0		11	Thread M12 x 1.25 (1 per)	Thread plug	OK		12	Thread M12 x 1.25 (1 per)	Thread plug	OK		13	Hole dia	V-Caliper	10.0		14	Distance	Height Gauge	10.0		15	Hole dia (1 per)	V-Caliper	10.0	
No.	Particulars	Inspection Method	Inspection Result	Remarks																																																																																			
1	Appearance	Visual	OK																																																																																				
2	Width	V-Caliper	40.0																																																																																				
3	Pinch Hole	Height Gauge	10.0																																																																																				
4	Length	V-Caliper	10.0																																																																																				
5	Pinch Hole	Height Gauge	10.0																																																																																				
6	Pinch Hole	Height Gauge	10.0																																																																																				
7	Distance	V-Caliper	10.0																																																																																				
8	Distance	V-Caliper	10.0																																																																																				
9	Hole dia (1 per)	V-Caliper	10.0																																																																																				
10	Distance	V-Caliper	10.0																																																																																				
11	Thread M12 x 1.25 (1 per)	Thread plug	OK																																																																																				
12	Thread M12 x 1.25 (1 per)	Thread plug	OK																																																																																				
13	Hole dia	V-Caliper	10.0																																																																																				
14	Distance	Height Gauge	10.0																																																																																				
15	Hole dia (1 per)	V-Caliper	10.0																																																																																				
Note: Mark "OK" Shift Date after checked: White color by PHO inspector Year Month Yellow color by QCO inspector		Judgement (✓ = OK, X = NG)																																																																																					
REVISION	EFFECTIVE DATE	DESCRIPTION																																																																																					
1	01-Jun-2009	Part Gusset Cross Member no. 1, 1, 1, 1, 1																																																																																					
Inspector	Checked	Remarks																																																																																					

Gambar IV.10 *Check Sheet* Komponen *Part Gusset Cross Member*
(Sumber : PT Gemala Kempa Daya, 2018)



Gambar IV.11 Bahan Baku Yang Ditandai Label OK
(Sumber : PT Gemala Kempa Daya, 2018)



Gambar IV.12 Bahan Baku Cacat Yang Ditandai Label NG
(Sumber : PT Gemala Kempa Daya, 2018)

 GEMALA KEMPA DAYA	LAPORAN SORTIR NO :/QA/GKD/...../.....		Doc. no	: FM/QAS/01-05																				
			Revision	: 02																				
			Eff. date	: 06-10-2017																				
NAMA SUPPLIER : PART NAME : MODEL : PROBLEM : TANGGAL SORTIR : WAKTU SORTIR : JUMLAH SORTIR :																								
		<table border="1"> <thead> <tr> <th>Area Sortir</th> <th>OK</th> <th>NG</th> <th>Total</th> </tr> </thead> <tbody> <tr> <td>Finish Good</td> <td>pcs</td> <td>pcs</td> <td>pcs</td> </tr> <tr> <td>In Proses/ Line</td> <td>pcs</td> <td>pcs</td> <td>pcs</td> </tr> <tr> <td>Ware House</td> <td>pcs</td> <td>pcs</td> <td>pcs</td> </tr> <tr> <td>Sub Total</td> <td>pcs</td> <td>pcs</td> <td>pcs</td> </tr> </tbody> </table>			Area Sortir	OK	NG	Total	Finish Good	pcs	pcs	pcs	In Proses/ Line	pcs	pcs	pcs	Ware House	pcs	pcs	pcs	Sub Total	pcs	pcs	pcs
Area Sortir	OK	NG	Total																					
Finish Good	pcs	pcs	pcs																					
In Proses/ Line	pcs	pcs	pcs																					
Ware House	pcs	pcs	pcs																					
Sub Total	pcs	pcs	pcs																					
Consumeable : 1 2 3 4		Note: Supplier																						
Mengetahui PT. GKD																								

Gambar IV.13 Laporan Sortir
(Sumber : PT Gemala Kempa Daya, 2018)

 PT. GEMALA KEMPA DAYA		LAPORAN PENERIMAAN BARANG / JASA		No. : 33776/000581/BA/10/18 Tgl. : 04-10-2018
DITERIMA DARI / SUPPLIER : DUTA HITA JAYA PT. No. SJ : 05081		PEMAKAI : PPIC GKD KPP/PP No. : JE507GPPC918 PO No. : J4997/PO-GKD/0918 LOKASI : OUT SOURCING		
NO.	BARANG / JASA	KODE BARANG	JUMLAH SATUAN	KETERANGAN PEMERIKSAAN
1	JASA PRESS BUSHING BRKT SPRING FRONT 31 U P/H: 15F- P652192090	AH 806390010	13 240 PCS	240 / 2000
PENERIMA/PEMERIKSA : PENERIMA PEMERIKSA		MENGETAHUI :		CATATAN : Gudang : Plant gkd jakarta Tgl Delivery : 04-10-2018 Jam Cetak : 14:40:00 FM/PPC/020 Revisi : 01
LEMBAR 1 ACCOUNTING		LEMBAR 2 PPC		LEMBAR 3 GUDANG

Gambar IV.14 Laporan Penerimaan Barang
(Sumber : PT Gemala Kempa Daya, 2018)

4.9 Analisis Sistem Yang Berjalan

Analisis sistem adalah penguraian dari suatu sistem informasi yang utuh ke dalam komponen bagian-bagian dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan. Tahap analisis dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem. Maka dari itu dilakukan analisis sistem informasi pemeriksaan bahan baku pada Divisi *Quality Assurance* di PT Gemala Kempa Daya. Teknik pengumpulan fakta dan data yang digunakan adalah wawancara, diskusi dan observasi terhadap beberapa karyawan yang berada di PT Gemala Kempa Daya.

Dalam suatu industri, pemeriksaan bahan baku sangatlah penting, oleh karena itu diperlukan sistem pada tiap proses pemeriksaan bahan baku itu sendiri. PT Gemala Kempa Daya mempunyai bagian yang memeriksa bahan baku dari pemasok yang dikirim ke perusahaan, yaitu dilini *Quality Assurance Receiving Inspection*.

Pada *Quality Assurance Receiving Inspection*, dalam melaksanakan pemeriksaan bahan baku masih dilakukan dengan bantuan *Microsoft Excel* untuk membuat *Checksheet* yang kemudian di cetak sehingga untuk melakukan pengecekan pada bahan baku tiap harinya memerlukan banyak kertas dan waktu untuk proses pemeriksaan dan informasi masalah ke bagian terkait.

4.10 Arus Dokumen

Dokumen yang terlibat dalam sistem informasi pemeriksaan bahan baku pada Divisi *Quality Assurance* dilini *Quality Assurance Receiving Inspection* pada PT GKD adalah sebagai berikut:

1. *Form Surat Jalan*

Form Surat Jalan adalah lembar dokumen pengiriman yang mencantumkan data bahan baku yang dikirim oleh *Supplier* di sertai kuantitas dan bahan baku tersebut.

2. *Tag Not Conforming Product Reject*

Tag Not Conforming Product Reject adalah lembaran sebuah tanda bahwa bahan baku tersebut tidak sesuai standar atau tidak layak untuk di simpan dan tidak terjamin kualitasnya.

3. *Tag OK*

Tag OK adalah lembaran sebuah tanda bahwa bahan baku tersebut sudah sesuai standar atau layak untuk simpan dan sudah terjamin kualitasnya.

4. *Form Check sheet*

Form Check sheet adalah lembar pemeriksaan yang terstruktur atau lembaran standarisasi untuk kualitas bahan baku untuk pengambilan data yang kemudian di olah menjadi sebuah informasi yang bermanfaat dalam pengambilan keputusan.

5. Laporan Sortir

Laporan Sortir adalah *form* yang digunakan untuk menghitung hasil pemeriksaan bahan baku yang telah dilakukan. Dan digunakan agar di olah lagi lebih lanjut.

6. Laporan Penerimaan Barang

Laporan Penerimaan Barang merupakan laporan bahwa bahan baku yang telah di periksa oleh Divisi *Quality Assurance* yang akan di simpan ke gudang.

7. Surat Retur

Surat Retur merupakan laporan bahan baku yang telah di periksa baik jumlah atau pun kualitasnya yang tidak sesuai untuk pengembalian ke *Supplier*.

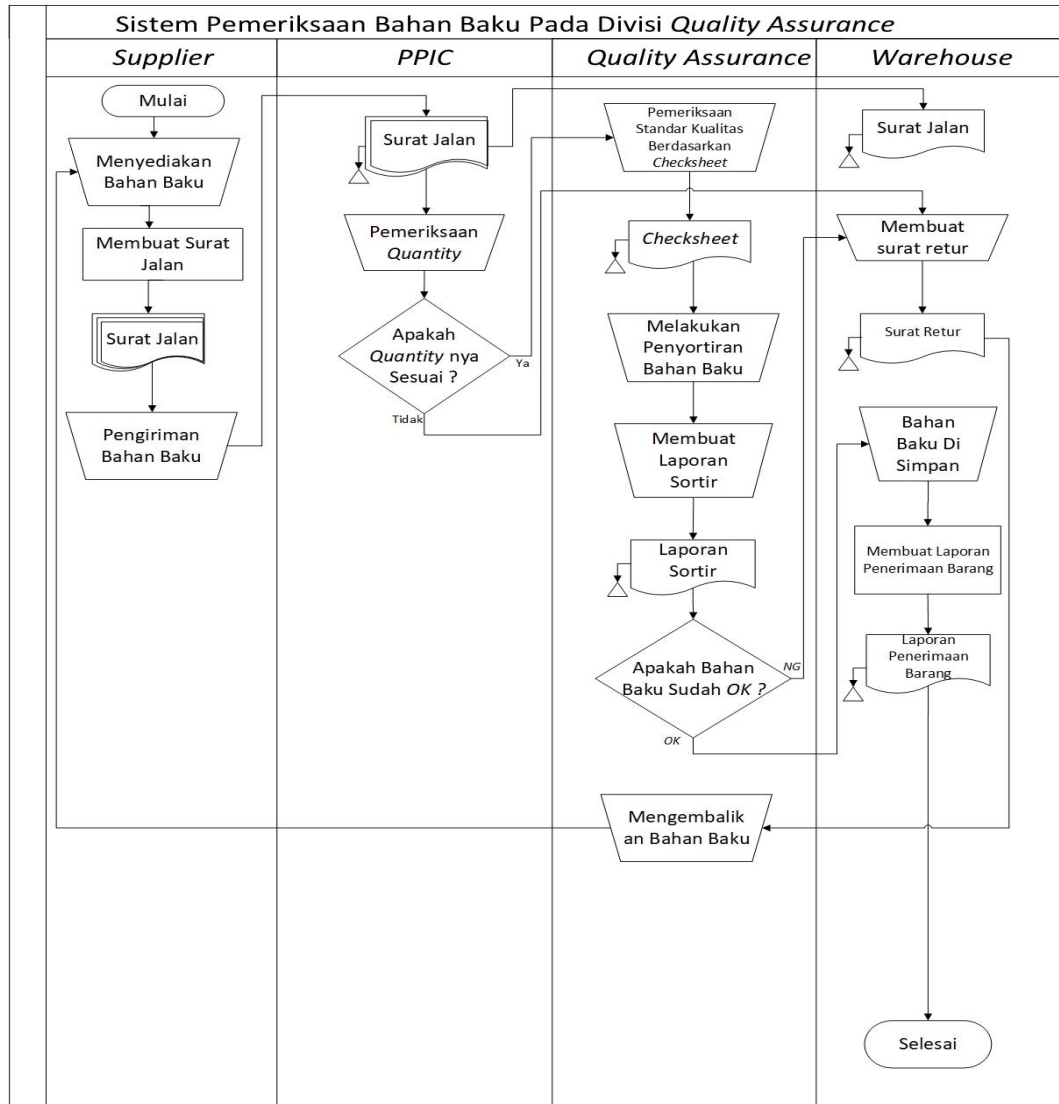
4.11 Prosedur Pemeriksaan Bahan Baku Pada *Quality Assurance Receiving Inspection*

Adapun prosedur sistem informasi dalam pemeriksaan bahan baku pada *Quality Assurance Receiving Inspection* di PT Gemala Kempa Daya:

1. *Supplier* akan mengirimkan material bahan baku serta surat jalan nya kepada *PPIC* untuk pemeriksaan jumlah atau *Quantity* yang di pesan oleh *Purchasing*

2. *Staff PPIC* memeriksa *Quantity* pada bahan baku dan menerima surat jalan
3. Setelah bahan baku diperiksa, Maka *Staff QA* melakukan pemeriksaan serta mencatat seluruh hasil pemeriksaannya dengan menggunakan *form Check Sheet*.
4. *Staff QA* melakukan pemeriksaan material atau *part* yang telah dinyatakan *OK/NG*, maka *Staff QA* memberi label *OK/NG* sesuai dengan kondisi *part* atau material tersebut.
5. Berdasarkan hasil pengecekan yang telah dilakukan, *Staff QA* melakukan penyortiran bahan baku yang di beri label *OK/NG* dengan membuat laporan sortir.
6. Hasil dari laporan sortir, selanjutnya bahan baku yang dilabel kan *NG* akan dipisahkan.
7. Setelah semua bahan baku yang telah diperiksa kualitas nya akan disimpan ke Gudang, maka *Staff Gudang* akan membuat laporan penerimaan barang.

Berikut merupakan *flowmap* sistem informasi pemeriksaan bahan baku pada *Quality Assurance Receiving Inspection* di PT Gemala Kempa Daya:



Gambar IV.15 Aliran Dokumen Sistem yang Berjalan
(Sumber: Hasil Analisis, 2019)

4.12 Penggambaran Sistem Pengendalian Kualitas Bahan Baku Dengan UML (Unified Modeling Language)

Kegiatan analisis sistem yang berjalan dengan menggunakan analisis sistem yang berorientasi pada objek-objek yang sangat diperlukan oleh sistem yang akan dirancang. Dengan maksud untuk menitikberatkan kepada fungsionalitas sistem yang berjalan dengan tidak terlalu menitikberatkan pada alur proses dari sistem. Kemudian dari hasil analisis ini direpresentasikan dengan *UML* melalui diagram

use case. Pertimbangan dari diagram ini dapat mewakili secara keseluruhan sistem yang berjalan yang dapat dimengerti oleh *user*.

4.13 Use Case Diagram

Use case diagram adalah deskripsi fungsi sebuah sistem dari perspektif pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem dipakai (Munawar, 2005).

Sistem informasi pengendalian kualitas bahan baku memiliki keterkaitan antar aktor, diantaranya *PPC*, *Quality Assurance* dan *Warehouse*. Setiap aktor memiliki kepentingan-kepentingan yang berkaitan dengan sistem tersebut. Adapun untuk detail dari analisis adalah sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* pengendalian kualitas bahan baku yang sedang berjalan pada divisi *Quality Assurance* di PT Gemala Kempa Daya dapat dilihat pada tabel IV.5 berikut:

Tabel IV.5 Definisi Aktor Pemeriksaan Bahan Baku

No.	Aktor	Definisi
1.	<i>PPIC</i>	Pada Divisi <i>PPIC (Product & Planning Inventory Control)</i> akan dilakukan pemeriksaan jumlah dan nama bahan baku atau material yang dipesan berdasarkan <i>Purchase Order</i> .
2.	<i>Quality Assurance</i>	Petugas <i>Quality Assurance</i> bertugas melakukan pemeriksaan berdasarkan form yang telah disediakan. Serta melakukan penyortiran bahan baku yang telah dinyatakan OK atau NG.
3.	<i>Warehouse</i>	Menyimpan seluruh bahan baku yang telah dijamin kualitas maupun kuantitas nya serta membuat laporan penerimaan barang dan retur.

(Sumber: Hasil Analisis, 2019)

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* pemeriksaan bahan baku yang sedang berjalan pada *Quality Assurance Receiving Inspection* Pada PT Gemala Kempa Daya dapat dilihat pada tabel IV.6 berikut:

Tabel IV.6 Definisi *Use Case* Pemeriksaan Bahan Baku

No	<i>Use Case</i>	Definisi
1.	Pemeriksaan jumlah dan nama barang	Proses pemeriksaan jumlah dan bahan baku yang dipesan dan menerima surat jalan dari <i>Supplier</i>
2.	Pemeriksaan pada QA <i>Receiving Inspection</i>	Proses pengecekan bahan baku pada pengendalian kualitas berdasarkan <i>form Check Sheet</i> yang ada.
3.	Melakukan penyotiran	Proses memilah-milah bahan baku yang berlabel <i>OK/NG</i> .
4.	Membuat Surat Retur	Proses pembuatan laporan pada masalah yang terjadi berdasarkan bahan baku berlabel <i>NG</i> atau tidak sesuai.
5.	Menyimpan bahan baku	Proses penyimpanan bahan baku yang sudah dijamin kualitas dan kuantitas nya.
6.	Membuat Laporan Penerimaan Barang	Proses pembuatan Laporan Penerimaan Barang pada bahan baku yang disimpan di gudang.

(Sumber: Hasil Analisis, 2019)

BAB V

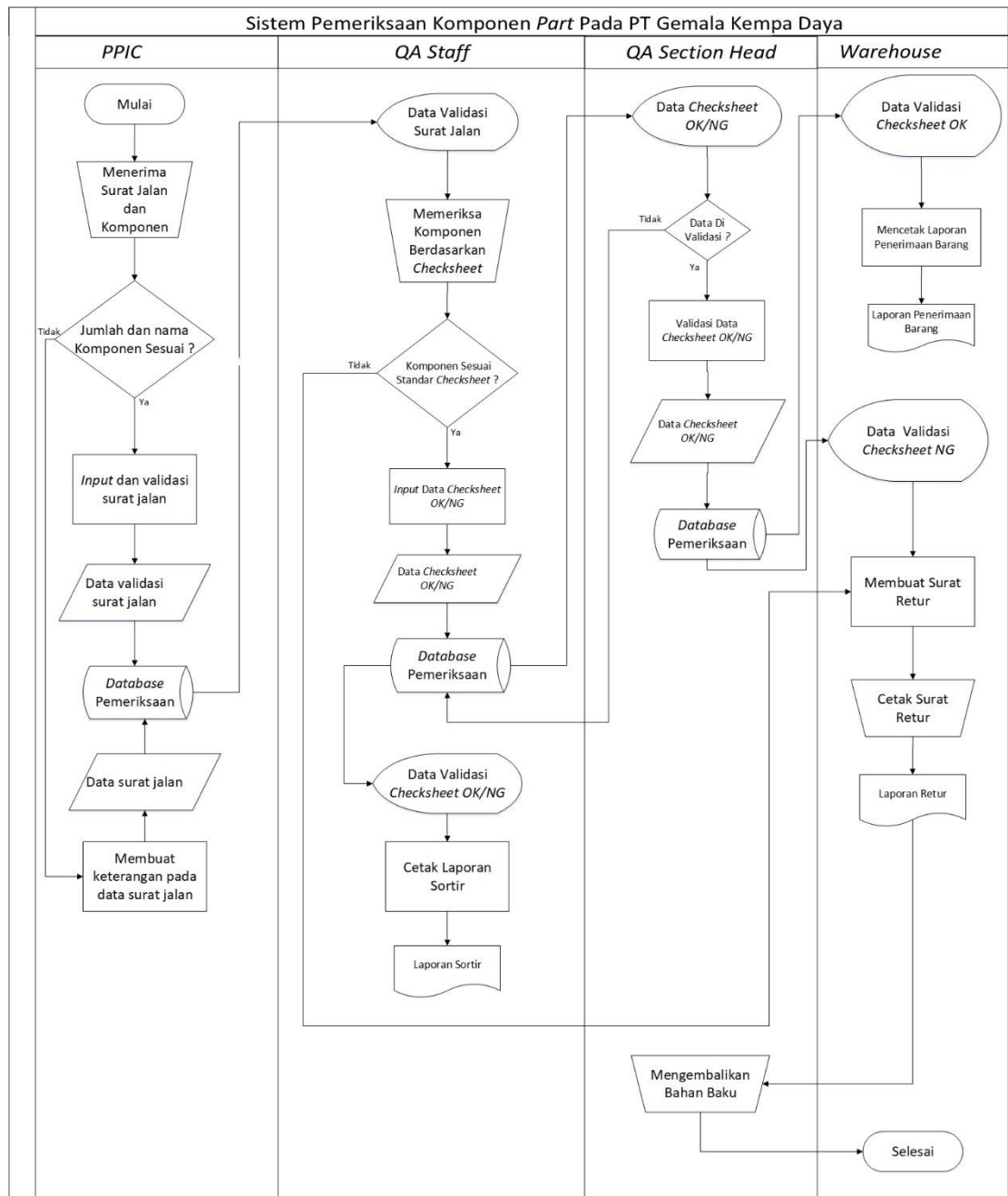
ANALISIS DAN PEMBAHASAN

5.1 Alur Proses Pemeriksaan Komponen Part Usulan

Proses sistem informasi pemeriksaan Komponen Part dilini *Quality Assurance Receiving Inspection* yang diusulkan sebagai berikut :

1. Proses pemeriksaan Komponen Part diawali dengan *PPIC* yang menerima surat jalan dan Komponen Part nya akan di periksa sesuai jumlah dan nama Komponen Part yang di pesan,jika sesuai maka *Staff PPIC* menginputkan nomer surat jalan serta nama dan jumlahnya yang tertera di surat jalan dan divalidasi.jika Komponen Part tidak sesuai akan dibuat keterangan.
2. *Staff QA* melihat data validasi surat jalan Komponen Partnya dan memeriksa Komponen Part berdasarkan *Checksheets* serta merekap laporan sortir.
3. Setelah memeriksa Komponen Part, *Staff QA* menginput data *Checksheets OK/NG* dan merekap laporan sortir ke *database* untuk di validasi oleh *QA Head Section*. Jika belum di validasi oleh *QA Head Section* maka data *Checksheets* masih didalam *database*.
4. Kemudian *Staff Warehouse* akan melihat data laporan sortir Komponen Part, untuk merekap laporan penerimaan barang untuk barang *OK* serta mencetak surat retur untuk barang *NG* atau yang tidak sesuai dengan surat jalan.

Flowmap pada gambar V.1 akan menjelaskan tentang proses pemeriksaan Komponen Part dilini *Quality Assurance Receiving Inspection* tersebut:



Gambar V.1 *Flowmap* Usulan Pemeriksaan Komponen Part Usulan

Sumber: Hasil Analisis (2019)

5.2 Analisis Sistem Usulan

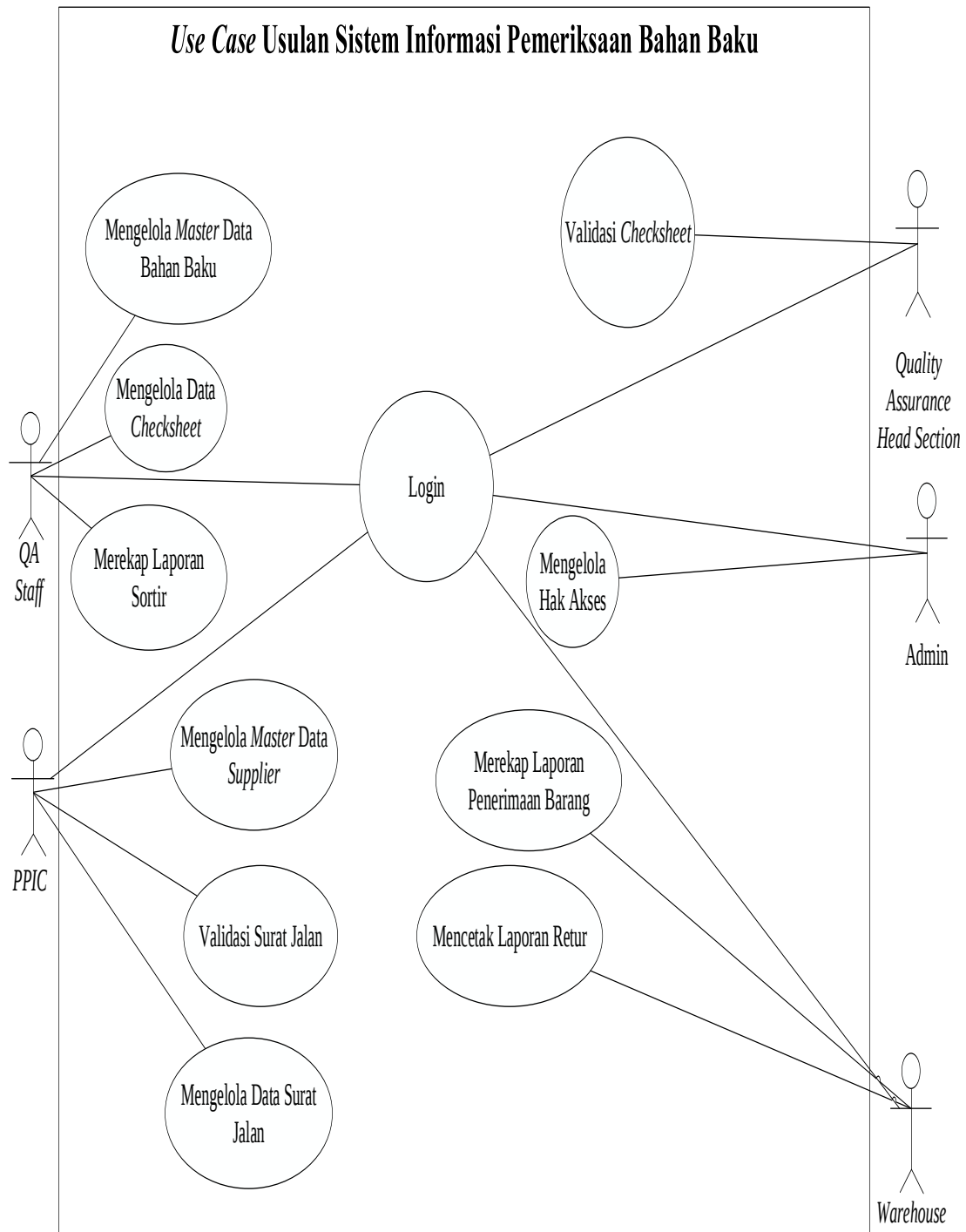
Perancangan aplikasi yang diusulkan menggunakan model pengembangan perangkat lunak *waterfall*. Metode *Waterfall* adalah sebuah metode pengembangan sistem dimana antar satu fase ke fase yang lain dilakukan secara berurutan. sistem informasi pemeriksaan Komponen Part, ada beberapa tahapan yang digunakan yaitu:

1. Pembuatan model sistem berbasis objek dengan *Unified Modeling Language* (UML) menggunakan *use case diagram*, *activity diagram*, *sequence diagram* dan *class diagram*
2. Pembuatan model data dengan *Entity Relationship Diagram* (ERD) dan kamus data.
3. Perancangan sistem dengan menggunakan WND (*Windows Navigation Diagram*) dan perancangan antar muka.
4. Pembuatan aplikasi menggunakan *Codeigniter* 3.1.8 dan *MariaDB* 10.1.36 sebagai *database*.

Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun. Selanjutnya akan dilakukan perancangan sistem informasi pemeriksaan Komponen Part berdasarkan hasil analisis.

5.3 Use Case Diagram

Use case diagram menggambarkan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* sistem informasi pemeriksaan Komponen Part setiap aktor harus *login* terlebih dahulu sebelum menggunakan sistem. Dan harus melakukan *logout* jika ingin keluar dari sistem. *Use case diagram* sistem informasi pemeriksaan Komponen Part yang diusulkan dapat dilihat pada Gambar V.2 sebagai berikut:



Gambar V.2 Use Case Sistem Informasi Pemeriksaan Komponen Part Usulan
(Sumber: Hasil Analisis (2019))

1. Definisi Aktor

Pendefinisian aktor pada *use case* sistem informasi pengendalian kualitas usulan dapat dilihat pada Tabel V.1 sebagai berikut:

Tabel V.1 Definisi Aktor *Use Case Diagram* Sistem Usulan

No.	Aktor	Deskripsi
1	Admin	Admin bertugas yang mengelola data hak akses.
2	<i>Staff PPIC</i>	<i>Staff PPIC</i> bertugas untuk mengelola <i>master data Supplier</i> dan validasi surat jalan Komponen Part yang dikirim oleh <i>Supplier</i>
3	<i>Staff QA</i>	<i>Staff QA</i> bertugas untuk mengelola <i>master data</i> Komponen Part, data hasil <i>Checksheet</i> dan merekap Laporan Sortir pemeriksaan Komponen Part
4	<i>QA Head Section</i>	<i>QA Head Section</i> bertugas untuk memvalidasi Data <i>Checksheet</i> .
5	<i>Staff Warehouse</i>	<i>Staff Warehouse</i> bertugas untuk mencetak surat retur dan merekap Laporan Penerimaan Barang.

(Sumber: Hasil Analisis Data, 2019)

2. Skenario *Use Case*

Berdasarkan Gambar V.2 *use case diagram* usulan di atas maka dibuatlah skenario jalannya masing-masing *use case* pada sistem informasi pemeriksaan Komponen Part, dapat dilihat pada poin berikut:

a. *Use Case Login*

Berikut adalah definisi *use case login* yang dapat dilihat pada Tabel V.2:

Tabel V.2 Definisi *Use Case Login*

Nama <i>Use Case</i>	<i>Login</i>
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses <i>user</i> masuk ke sistem pemeriksaan Komponen Part.
Aktor	Admin, <i>Staff PPIC</i> , <i>Staff QA</i> , <i>QA Head Section</i> , <i>Staff Warehouse</i>
<i>Normal Flow</i>	1. <i>Actor</i> membuka aplikasi. 2. Sistem akan menampilkan halaman <i>login</i>. 3. <i>Actor</i> memasukkan <i>username</i> dan <i>password</i>. 4. <i>Actor</i> mengklik tombol <i>login</i>. 5. Sistem akan memvalidasi <i>username</i> dan <i>password</i>. 6. Jika <i>username</i> dan <i>password</i> valid, sistem akan menampilkan halaman utama. 7. Jika <i>username</i> dan <i>password</i> tidak valid, sistem akan meminta <i>Actor</i> memasukkan <i>username</i> dan <i>password</i> sekali lagi

(Sumber: Hasil Analisis Data, 2019)

b. *Use Case Mengelola Master Data Supplier*

Berikut adalah deskripsi *use case* mengelola *master data supplier* yang dapat dilihat pada Tabel V.3:

Tabel V.3 Definisi *Use Case* Mengelola *Master Data Supplier*

Nama <i>Use Case</i>	Mengelola <i>Master Data Supplier</i>
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses pengelolaan <i>Master Data Supplier</i> .
Aktor	<i>Staff PPIC</i>
<i>Normal Flow</i>	1. <i>Staff PPIC Login</i> dan masuk ke tampilan menu utama 2. <i>Staff PPIC</i> memilih menu <i>Data Master</i>, lalu memilih Sub Menu <i>Data Supplier</i>. 3. Sistem akan menampilkan halaman <i>Data Supplier</i>. 4. <i>Staff PPIC</i> dapat memilih aksi tambah data, ubah data, dan hapus data. 5. Jika memilih aksi tambah data maka sistem akan menampilkan <i>form</i> tambah data. 6. <i>Staff PPIC</i> akan memasukkan data, lalu mengklik tombol simpan. Sistem akan menyimpan data ke dalam <i>database</i>. 7. Jika memilih aksi ubah data maka sistem akan menampilkan <i>form</i> ubah data. 8. <i>Staff PPIC</i> mengubah data, lalu mengklik tombol ubah. Sistem akan mengubah data didalam <i>database</i>. 9. Jika memilih aksi hapus data maka <i>Staff PPIC</i> mengklik tombol hapus. Sistem akan menghapus data didalam <i>database</i>.

(Sumber: Hasil Analisis Data, 2019)

c. *Use Case* Mengelola *Master Data* Komponen Part

Berikut adalah deskripsi *use case* mengelola *master data* Komponen Part yang dapat dilihat pada Tabel V.4:

Nama <i>Use Case</i>	Mengelola <i>Master Data</i> Komponen Part
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses pengelolaan <i>Master Data</i> Komponen Part.
Aktor	<i>Staff QA</i>
<i>Normal Flow</i>	1. <i>Staff QA Login</i> dan masuk ke tampilan menu utama 2. <i>Staff QA</i> memilih menu <i>Data Master</i>, lalu memilih Sub Menu <i>Data Komponen Part</i>. 3. Sistem akan menampilkan halaman <i>Data Komponen Part</i>. 4. <i>Staff QA</i> dapat memilih aksi tambah data, ubah data, dan hapus data. 5. Jika memilih aksi tambah data maka sistem akan menampilkan <i>form</i> tambah data. 6. <i>Staff QA</i> akan memasukkan data, lalu mengklik tombol simpan. Sistem akan menyimpan data ke dalam <i>database</i>. 7. Jika memilih aksi ubah data maka sistem akan menampilkan <i>form</i> ubah data. 8. Admin mengubah data, lalu mengklik tombol ubah. Sistem akan mengubah data didalam <i>database</i>. 9. Jika memilih aksi hapus data maka Admin mengklik tombol hapus. Sistem akan menghapus data didalam <i>database</i>.

(Sumber: Hasil Analisis Data, 2019)

d. *Use Case Mengelola Master Data Login*Tabel V.5 Definisi *Use Case Mengelola Master Data Login*

Nama <i>Use Case</i>	Mengelola <i>Master Data Login</i>
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses pengelolaan <i>Master Data Login</i> .
Aktor	Admin
<i>Normal Flow</i>	1. Admin <i>Login</i> dan masuk ke tampilan menu utama 2. Admin memilih menu <i>Data Master</i>, lalu memilih Sub Menu <i>Data Login</i>. 3. Sistem akan menampilkan halaman <i>Data Login</i>. 4. Admin dapat memilih aksi tambah data, ubah data, dan hapus data. 5. Jika memilih aksi tambah data maka sistem akan menampilkan <i>form</i> tambah data. 6. Admin akan memasukkan data, lalu mengklik tombol simpan. Sistem akan menyimpan data ke dalam <i>database</i>. 7. Jika memilih aksi ubah data maka sistem akan menampilkan <i>form</i> ubah data. 8. Admin mengubah data, lalu mengklik tombol ubah. Sistem akan mengubah data didalam <i>database</i>. 9. Jika memilihn aksi hapus data maka Admin mengklik tombol hapus. Sistem akan menghapus data didalam <i>database</i>.

(Sumber: Hasil Analisis Data, 2019)

e. *Use Case Mengelola Data Checksheet.*Tabel V.6 Definisi *Use Case* Mengelola Data *Checksheet*.

Nama <i>Use Case</i>	Mengelola Data <i>Checksheet</i> .
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses pengelolaan Data <i>Checksheet</i> .
Aktor	<i>Staff QA</i>
<i>Normal Flow</i>	1. <i>Staff QA Login</i> dan masuk ke tampilan menu utama 2. <i>Staff QA</i> memilih menu Data Transaksi, lalu memilih Sub Menu Data <i>Checksheet</i>. 3. Sistem akan menampilkan halaman Data <i>Checksheet</i>. 4. <i>Staff QA</i> dapat memilih aksi tambah data, ubah data, dan hapus data. 5. Jika memilih aksi tambah data maka sistem akan menampilkan <i>form</i> tambah data. 6. <i>Staff QA</i> akan memasukkan data, lalu mengklik tombol simpan. Sistem akan menyimpan data ke dalam <i>database</i>. 7. Jika memilih aksi ubah data maka sistem akan menampilkan <i>form</i> ubah data. 8. <i>Staff QA</i> mengubah data, lalu mengklik tombol ubah. Sistem akan mengubah data didalam <i>database</i>. 9. Jika memilih aksi hapus data maka <i>Staff QA</i> mengklik tombol hapus. Sistem akan menghapus data didalam <i>database</i>.

(Sumber: Hasil Analisis Data, 2019)

f. *Use Case* Mengelola Data Surat Jalan.Tabel V.7 Definisi *Use Case* Mengelola Data Surat Jalan.

Nama <i>Use Case</i>	Mengelola Data Surat Jalan.
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses pengelolaan Data Surat Jalan.
Aktor	<i>Staff PPIC</i>
<i>Normal Flow</i>	1. <i>Staff PPIC Login</i> dan masuk ke tampilan menu utama 2. <i>Staff PPIC</i> memilih menu Data Transaksi, lalu memilih Sub Menu Data Transaksi. 3. Sistem akan menampilkan halaman Data Transaksi. 4. <i>Staff PPIC</i> dapat memilih aksi tambah data, ubah data, dan hapus data. 5. Jika memilih aksi tambah data maka sistem akan menampilkan <i>form</i> tambah data. 6. <i>Staff PPIC</i> akan memasukkan data, lalu mengklik tombol simpan. Sistem akan menyimpan data ke dalam <i>database</i>. 7. Jika memilih aksi ubah data maka sistem akan menampilkan <i>form</i> ubah data. 8. <i>Staff PPIC</i> mengubah data, lalu mengklik tombol ubah. Sistem akan mengubah data didalam <i>database</i>. 9. Jika memilih aksi hapus data maka <i>Staff PPIC</i> mengklik tombol hapus. Sistem akan menghapus data didalam <i>database</i>.

g. *Use Case* Merekap Laporan Sortir.Tabel V.8 Definisi *Use Case* Merekap Laporan Sortir.

Nama <i>Use Case</i>	Merekap Laporan Sortir
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses perekapan Laporan Sortir.
Aktor	<i>Staff QA</i>
<i>Normal Flow</i>	1. <i>Staff QA Login</i> dan masuk ke tampilan menu utama 2. <i>Staff QA</i> memilih menu Laporan, lalu memilih Sub Menu Laporan Sortir. 3. Sistem akan menampilkan halaman data hasil pengendalian kualitas <i>Gusset Cross Member</i> Laporan Sortir.

(Sumber: Hasil Analisis Data, 2019)

h. *Use Case* Melakukan Validasi Data Surat Jalan.Tabel V.9 Definisi *Use Case* Melakukan Validasi Data Surat Jalan.

Nama <i>Use Case</i>	Melakukan Validasi Data Surat Jalan
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan <i>PPIC</i> melakukan validasi Data Surat Jalan yang telah di <i>input</i> .
Aktor	<i>Staff PPIC</i>
<i>Normal Flow of Events</i>	1. <i>Staff PPIC</i> memilih menu <i>Master Data</i>, lalu memilih Sub Menu Surat jalan. 2. Sistem akan menampilkan data surat jalan yang tersimpan di dalam <i>database</i>. 3. <i>Staff PPIC</i> akan memeriksa data surat jalan yang telah di <i>input</i>, lalu <i>Staff PPIC</i> menekan tombol validasi.

(Sumber: Hasil Analisis Data, 2019)

i. *Use Case Melakukan Validasi Data Checksheet.*

Tabel V.10 Definisi *Use Case* Melakukan Validasi Data *Checksheet*.

Nama <i>Use Case</i>	Melakukan Validasi Data <i>Checksheet</i>
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan <i>QA Head Section</i> melakukan validasi Data <i>Checksheet</i> yang telah di <i>input</i> .
Aktor	<i>QA Head Section</i>
<i>Normal Flow of Events</i>	1. <i>QA Head Section Login</i> dan masuk ke tampilan menu utama 2. <i>QA Head Section</i> memilih menu Data Transaksi, lalu memilih Sub Menu <i>Chcecksheet</i>. 3. Sistem akan menampilkan data <i>checksheet</i> yang tersimpan di dalam <i>database</i>. 4. <i>QA Head Section</i> akan memeriksa data <i>checksheet</i> yang telah di <i>input</i>, lalu <i>Head Section QA</i> menekan tombol validasi.

(Sumber: Hasil Analisis Data, 2019)

j. *Use Case Merekap Laporan Penerimaan Barang.*

Tabel V.11 Definisi *Use Case* Merekap Laporan Penerimaan Barang.

Nama <i>Use Case</i>	Merekap Data Laporan Penerimaan
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses perekapan Laporan Penerimaan Barang.
Aktor	<i>Staff Warehouse</i>
<i>Normal Flow</i>	1. <i>Staff Warehouse Login</i> dan masuk ke tampilan menu utama. 2. <i>Staff Warehouse</i> memilih menu Laporan, lalu memilih Sub Menu Laporan Penerimaan Barang. 3. Sistem akan menampilkan halaman Data Laporan Penerimaan Barang.

(Sumber: Hasil Analisis Data, 2019)

k. *Use Case* Mencetak Laporan Retur.

Tabel V.12 Definisi *Use Case* Mencetak Data Laporan Retur.

Nama <i>Use Case</i>	Mencetak Laporan Retur
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses mencetak Laporan Retur.
Aktor	<i>Staff Warehouse</i> .
<i>Normal Flow</i>	1. <i>Staff Warehouse Login</i> dan masuk ke tampilan menu utama. 2. <i>Staff Warehouse</i> memilih menu Laporan,lalu memilih Submenu Laporan Retur 3. Sistem akan menampilkan halaman data Laporan Retur. 4. <i>Staff Warehouse</i> melakukan cetak Laporan Retur.

(Sumber: Hasil Analisis Data, 2019)

5.4 Analisis Kebutuhan Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, maka dibutuhkan perancangan aplikasi sistem informasi pemeriksaan Komponen Part *Gusset Cross Member* di lini *QA Receiving Inspection*. Berikut adalah daftar kebutuhan *Functional* sistem untuk aplikasi sistem informasi pemeriksaan Komponen Part *Gusset Cross Member* di lini *QA Receiving Inspection*.

Tabel V.13 Kebutuhan Sistem.

No	Masalah	Kebutuhan User	Solusi	Kebutuhan Sistem	
				Functional Requirement	Non Functional Requirement
1	Proses pemeriksaan Komponen Part <i>Gusset Cross Member</i> menggunakan media kertas serta tidak terorganisir dengan baik contoh nya redundansi dan kehilangan data	Sistem yang dapat melakukan validasi terhadap surat jalan dan <i>checksheet</i> yang hanya dilakukan oleh <i>staff PPIC</i> dan <i>head section QA</i> .	Membuat Sistem dengan <i>database</i> seperti <i>MariaDB</i> sebagai media penyimpanan	Sistem dapat merekap laporan sortir dan laporan penerimaan barang.	Laporan sortir, dan laporan penerimaan barang dikelola oleh <i>staff Warehouse</i> .
2	Belum ada nya sistem yang terintegrasi dalam proses pemeriksaan Komponen Part <i>Gusset Cross Member</i> .		Membuat sistem informasi berbasis komputer dengan <i>database</i> yang saling terhubung.	Mengelola dan menyimpan data <i>master</i> Komponen Part, dan data <i>supplier</i> .	- Data Komponen Part dikelola oleh <i>staff QA</i> - Data <i>supplier</i> dikelola oleh <i>staff PPIC</i> .

(Sumber: Hasil Analisis Data, 2019)

Tabel V.14 Kebutuhan Sistem (Lanjutan)

No	Masalah	Kebutuhan User	Solusi	Kebutuhan Sistem	
				Functional Requirement	Non Functional Requirement
3	Sulitnya pencarian data surat jalan untuk pembuatan laporan retur		Membuat Sistem yang dapat mengeluarkan <i>output</i> berupa laporan retur yang sudah terkomputerisasi	Sistem dapat merekap laporan retur serta dapat mencetak laporan retur	laporan retur dikelola oleh <i>Staff Warehouse</i> .
4	Keamanan data lemah karena pengelolaan data pemeriksaan Komponen Part dapat diakses oleh semua orang	Adanya keamanan data untuk pengelolaan data pemeriksaan Komponen Part.	Membuat sistem informasi yang mampu mengelola hak akses	- Dapat melakukan <i>login</i> - Mengelola data sesuai hak aksesnya.	- <i>Login</i> dengan NIP dan <i>password</i> - NIP dan <i>password</i> dari gabungan huruf dan angka yang dimana NIP nomor induk pegawai dan <i>password</i> berisi maksimal delapan karakter.

(Sumber: Hasil Analisis Data, 2019)

5.4.1 *Functional Requirement*

Functional requirement dari sistem informasi pemeriksaan Komponen Part yang diusulkan antara lain:

1. *Staff PPIC* mengelola data *master Supplier*.
2. *Staff QA* mengelola data *master* Komponen Part
3. Karyawan PT GKD dilini *QA Receiving Inspection* mengelola data *Checksheet* Komponen Part.
4. *Head Section QA* dan *Staff PPIC* melakukan validasi *Checksheet* dan surat jalan.
5. *Staff QA* mengelola data *Checksheet*.
6. *Staff Warehouse* merekap laporan penerimaan barang dan mencetak laporan retur.
7. *Staff QA* merekap laporan sortir.
8. *Staff PPIC*, *Staff QA*, *Head Section QA* dan *Staff Warehouse* di Gemala Kempa Daya melakukan *Login*.

5.4.2 *Non Functional Requirement*

Non Functional requirement dari sistem informasi pemeriksaan Komponen Part yang diusulkan antara lain:

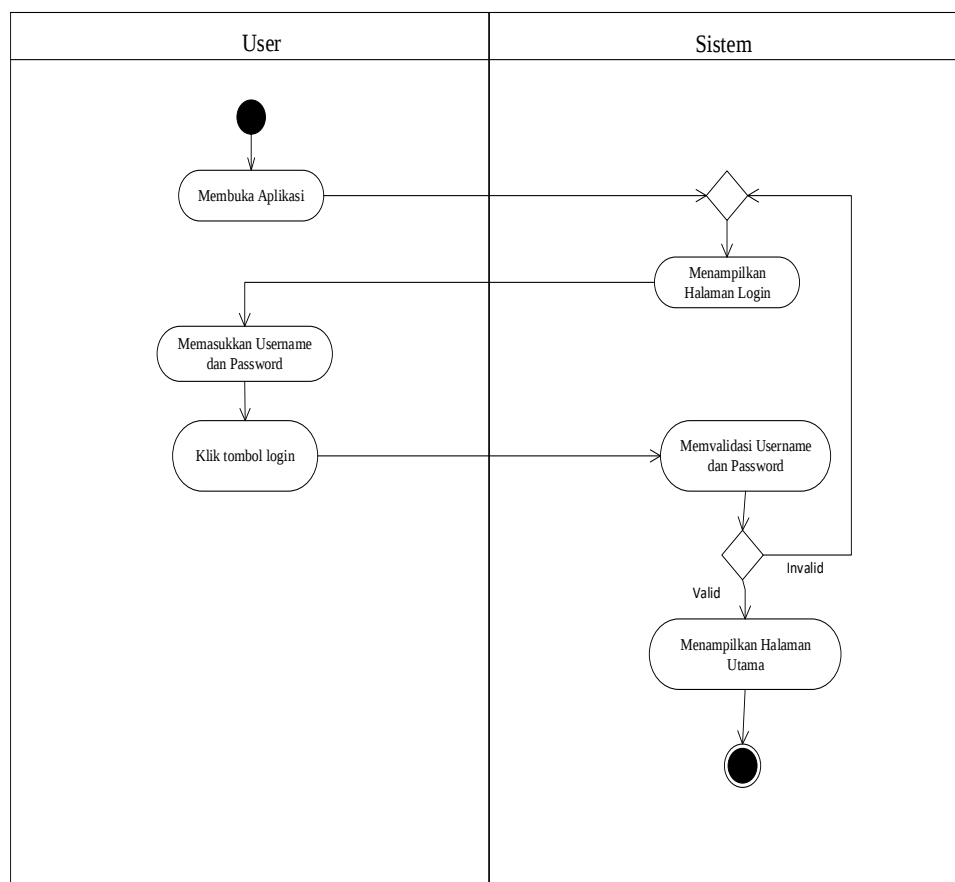
1. Laporan sortir, dan laporan penerimaan barang dikelola oleh *staff Warehouse*.
2. Data Komponen Part dikelola oleh *staff QA*
3. Data *supplier* dikelola oleh *staff PPIC*.
4. laporan retur dikelola oleh *Staff Warehouse*.
5. *Login* dengan NIP dan *password*
6. NIP dan *password* dari gabungan huruf dan angka yang dimana NIP nomor induk pegawai dan *password* berisi maksimal delapan karakter

5.5 Activity Diagram

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar memahami sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem informasi pemeriksaan Komponen Part ini.

1. Activity Diagram Login

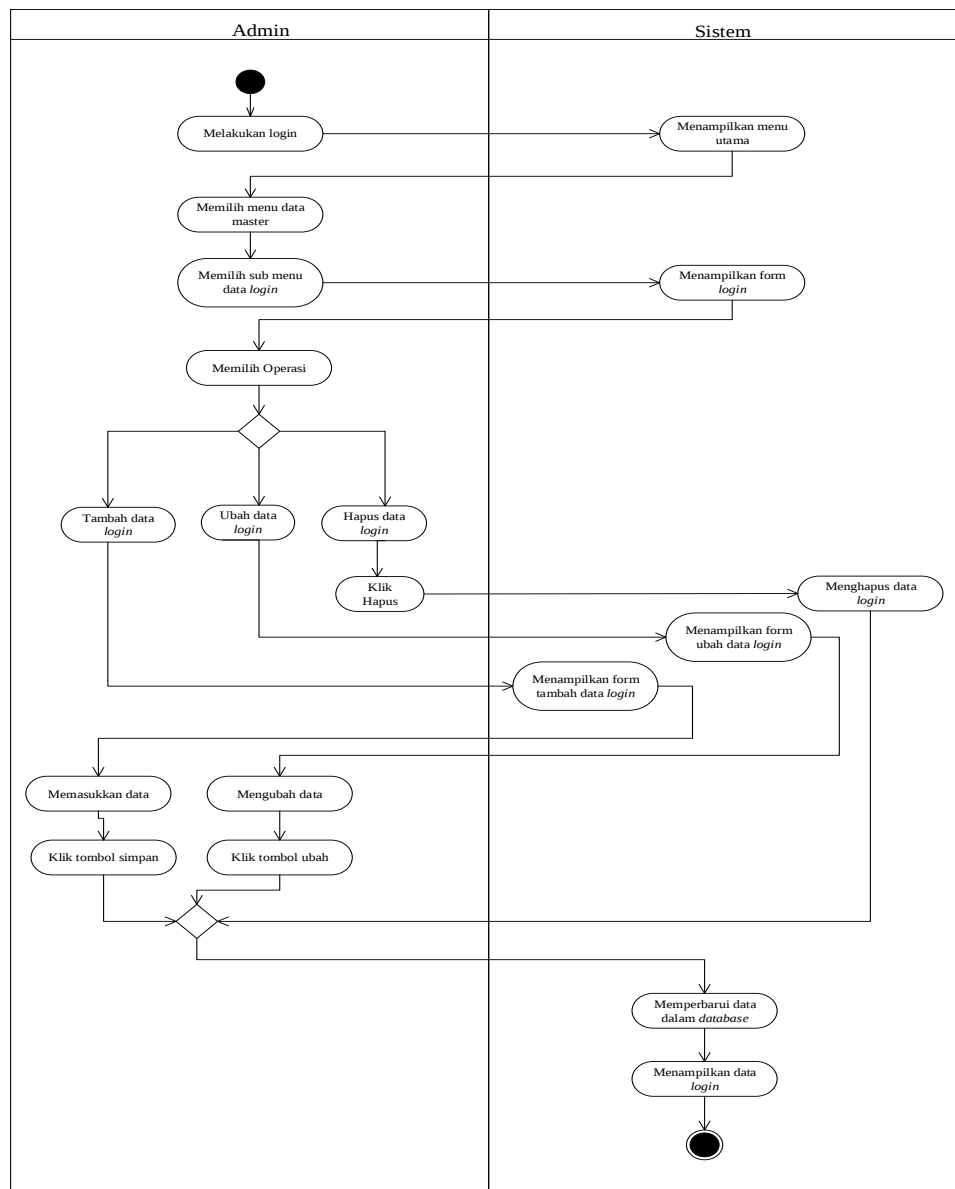
Activity diagram login ini menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu *admin*, *Staff PPIC*, *Staff QA*, *Head Section QA*, serta *Staff Warehouse* untuk dapat masuk ke dalam sistem informasi pemeriksaan Komponen Part *Gusset Cross Member* dilini *QA Receiving Inspection*. *Activity diagram* dapat dilihat pada Gambar V.3:



Gambar V.3 Activity Diagram Login
(Sumber: Hasil Analisis Data 2019)

2. Activity Diagram Mengelola Master Data Login

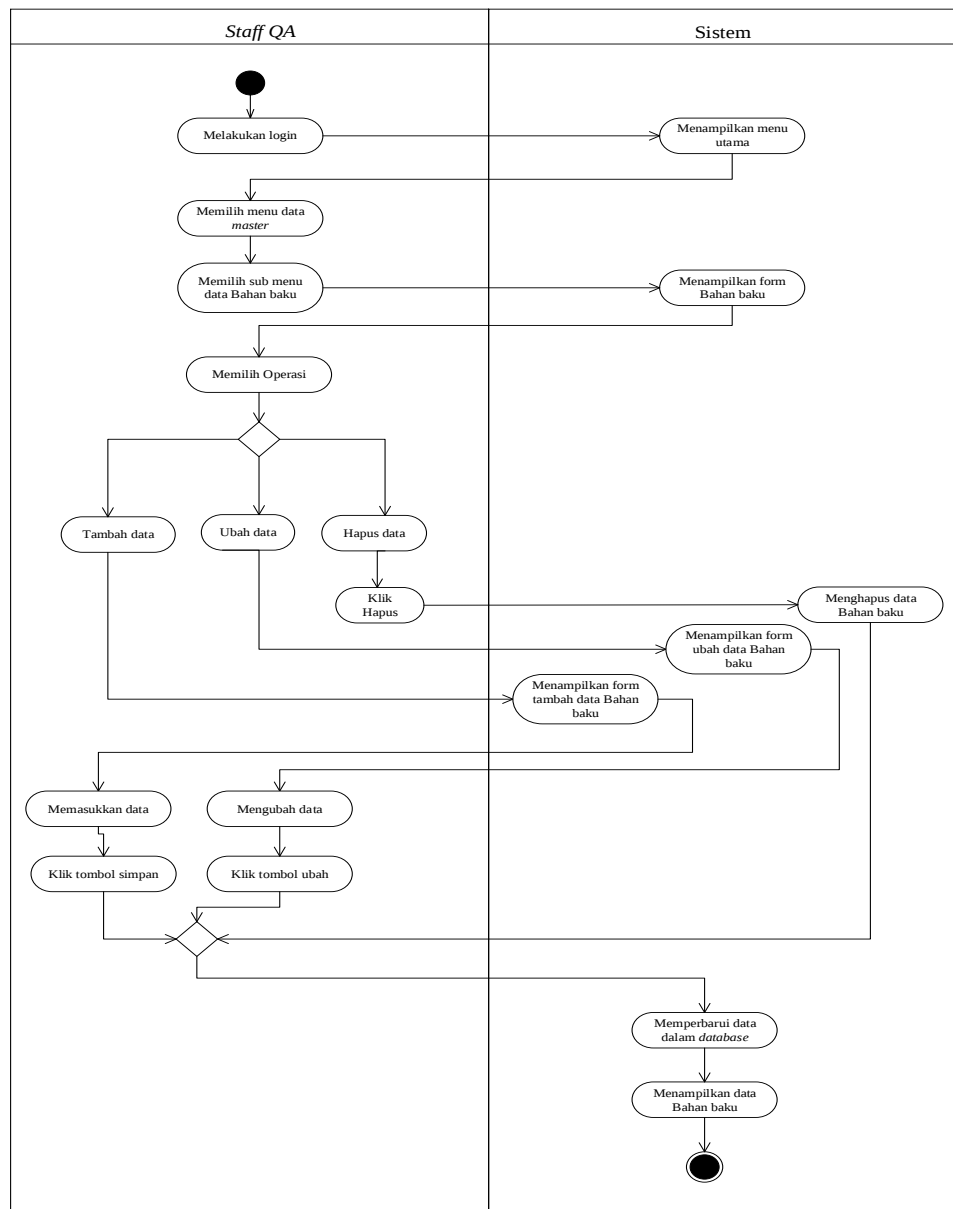
Activity diagram mengelola master data login berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam pengelolaan master data login. Activity diagram data master data login yang diusulkan dapat dilihat pada Gambar V.4:



Gambar V.4 Activity Diagram Master Data Login
(Sumber: Hasil Analisis Data 2019)

3. Activity Diagram Mengelola Master Data Komponen Part.

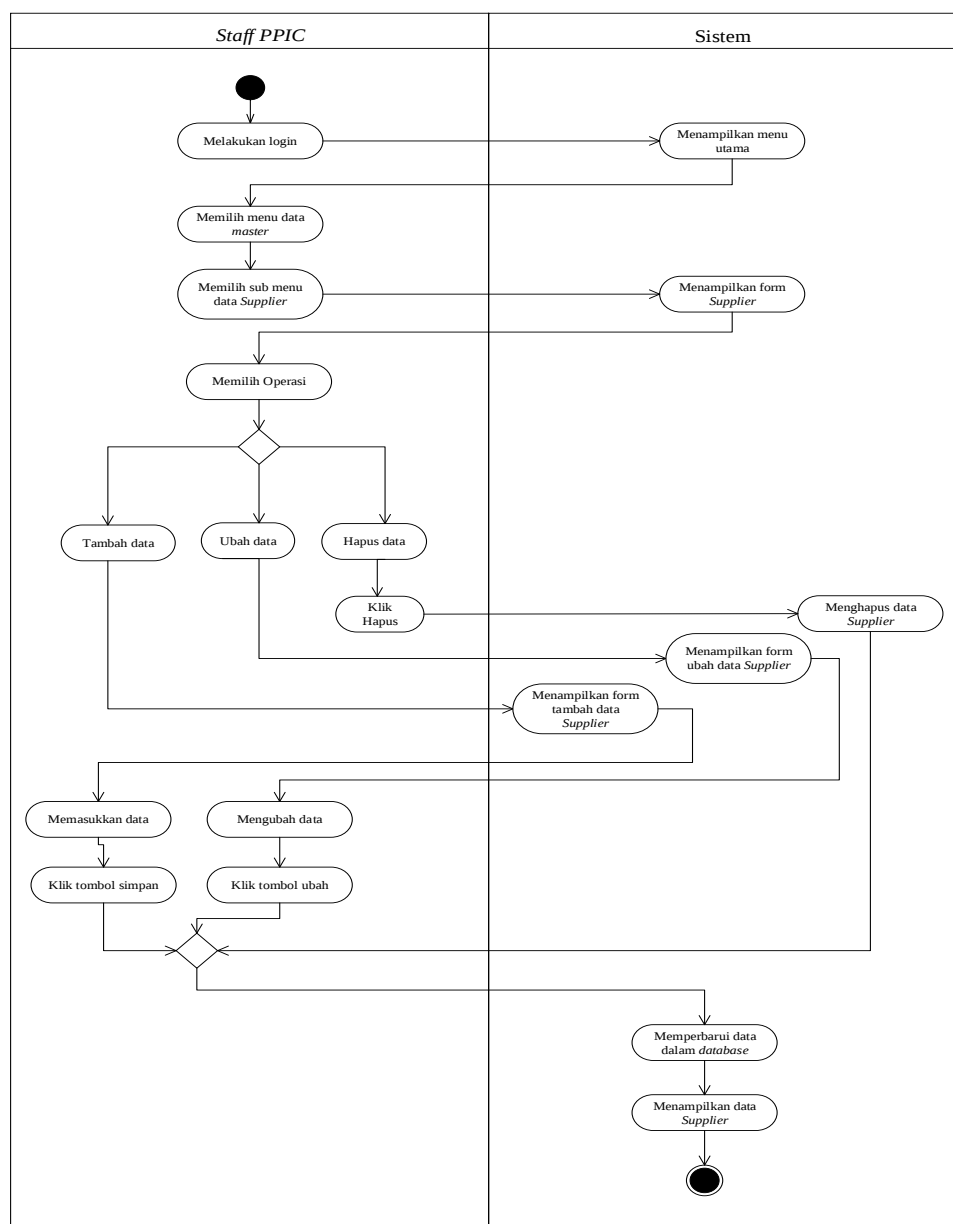
Activity diagram mengelola master data Komponen Part berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam pengelolaan master data Komponen Part yang dihasilkan. Activity diagram master data Komponen Part yang diusulkan dapat dilihat pada Gambar V.5:



Gambar V.5 Activity Diagram Master Data Komponen Part
(Sumber: Hasil Analisis Data 2019)

4. Activity Diagram Mengelola Master Data Supplier.

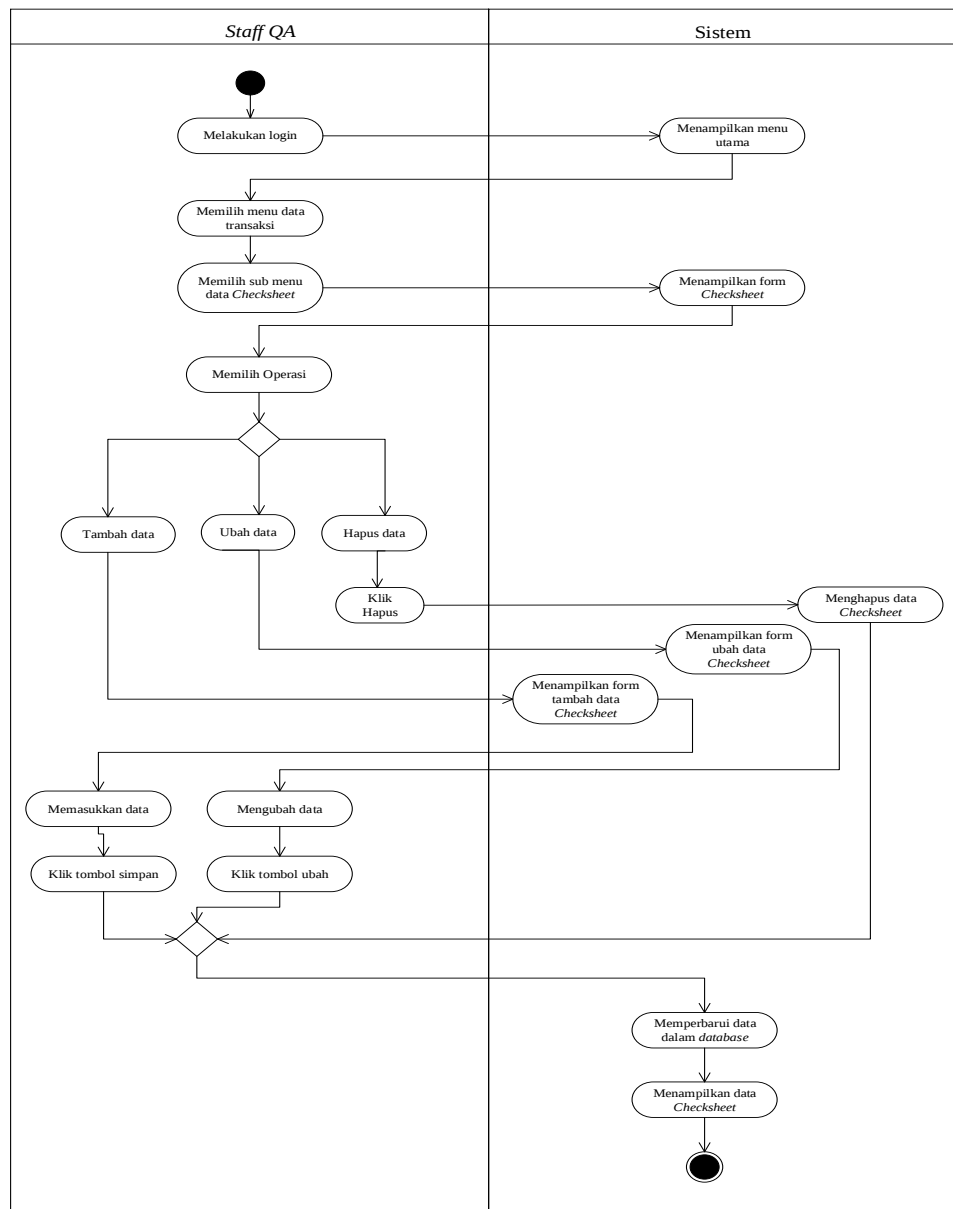
Activity diagram mengelola master data supplier berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam pengelolaan master data supplier yang dihasilkan. Activity diagram master data supplier yang diusulkan dapat dilihat pada Gambar V.6:



Gambar V.6 Activity Diagram Master Data Supplier
(Sumber: Hasil Analisis Data 2019)

5. Activity Diagram Mengelola Data *Checksheet*.

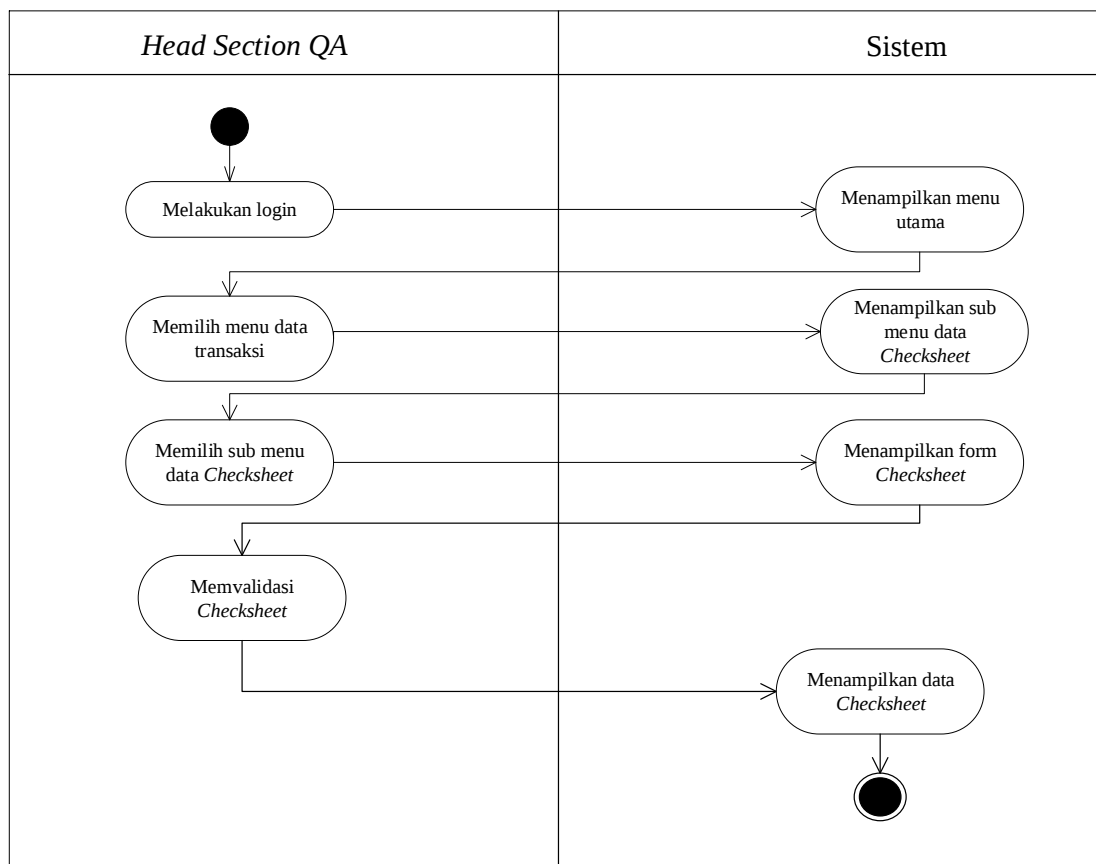
Activity diagram mengelola data *checksheet* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam pengelolaan data *checksheet* yang dihasilkan. Activity diagram data surat jalan yang diusulkan dapat dilihat pada Gambar V.7:



Gambar V.7 Activity Diagram Data *Checksheet*
Sumber: Hasil Analisis Data 2019)

6. *Activity Diagram Memvalidasi Data Checksheet.*

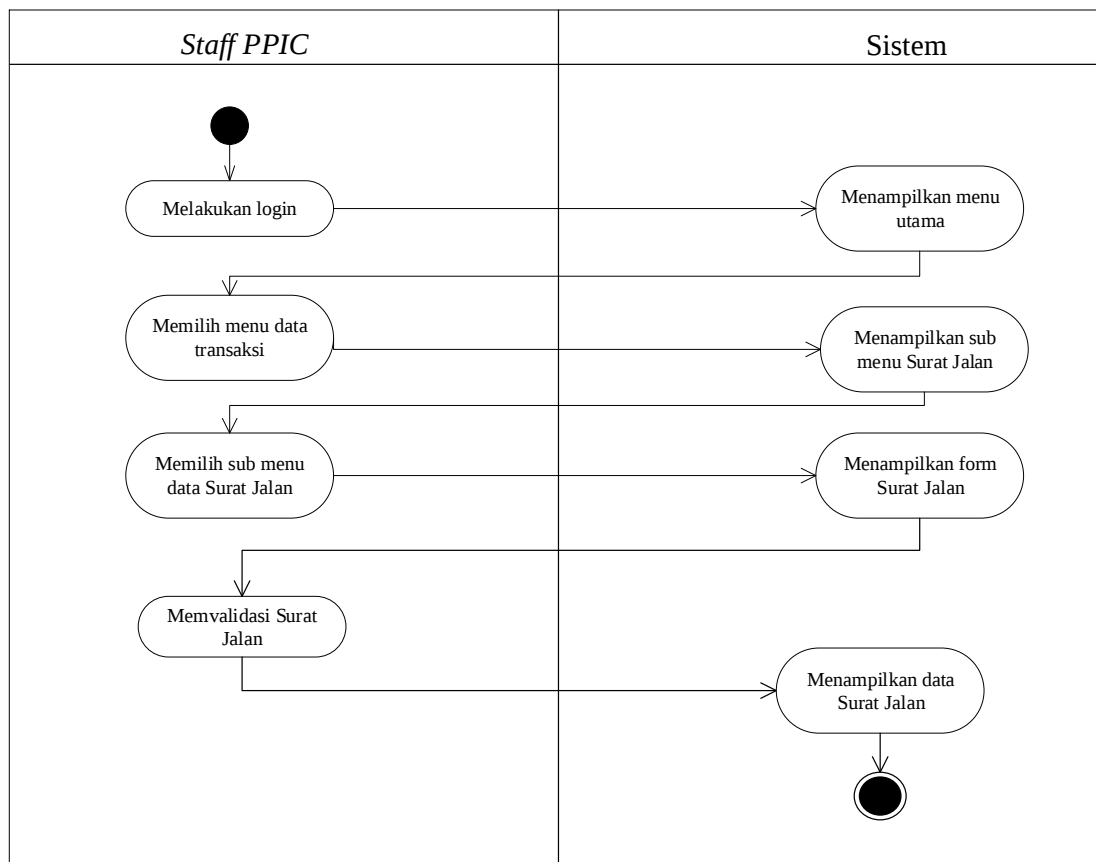
Activity diagram memvalidasi data checksheet berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam memvalidasi data *checksheet* yang dihasilkan. *Activity diagram* validasi data *checksheet* yang diusulkan dapat dilihat pada Gambar V.8:



Gambar V.8 *Activity Diagram Memvalidasi Data Checksheet*
(Sumber: Hasil Analisis Data 2019)

7. *Activity Diagram Memvalidasi Data Surat Jalan.*

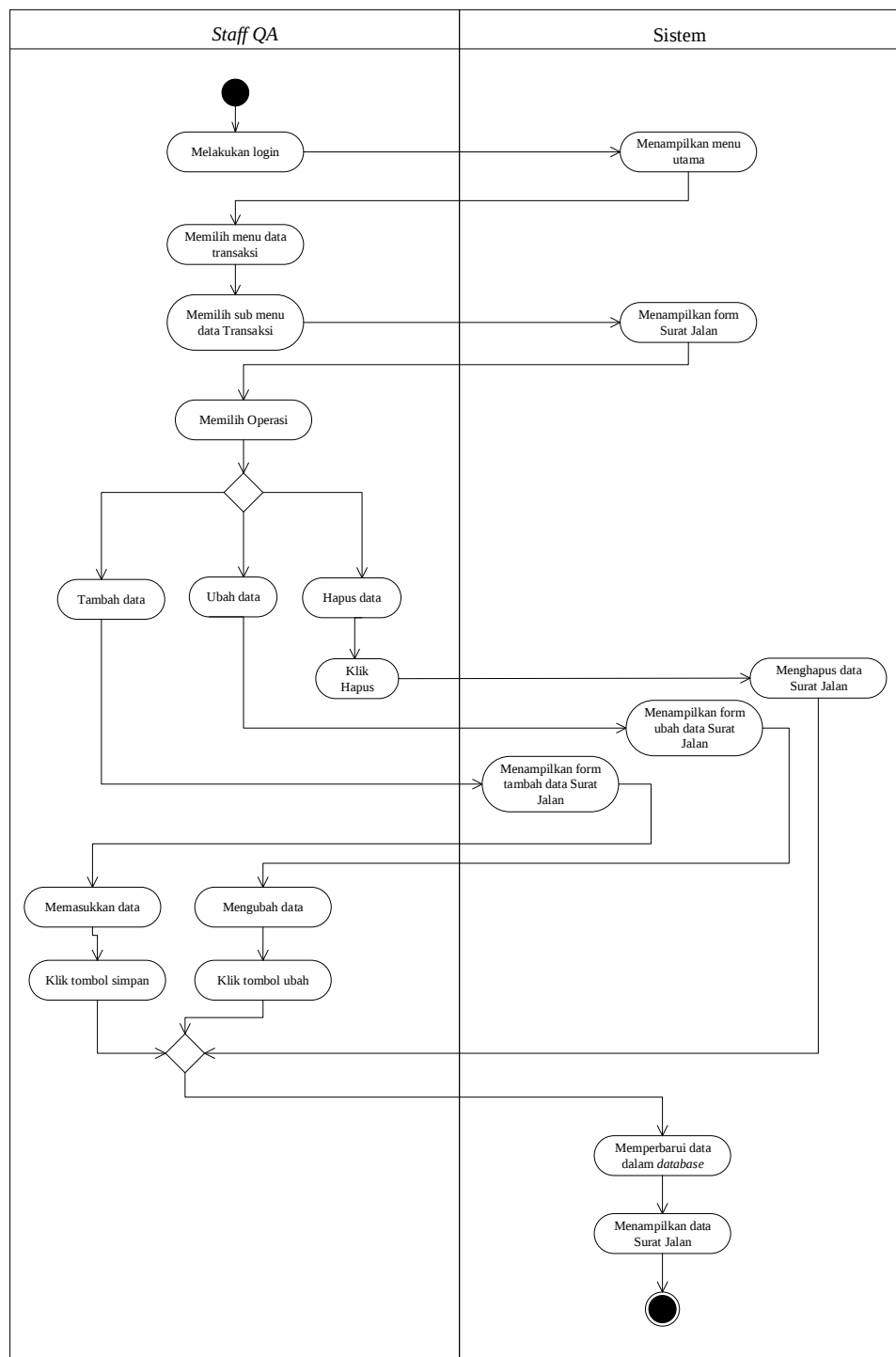
Activity diagram validasi data surat jalan berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam validasi data surat jalan yang dihasilkan. *Activity diagram* validasi data surat jalan yang diusulkan dapat dilihat pada Gambar V.9:



Gambar V.9 Activity Diagram Memvalidasi Data Surat Jalan
(Sumber: Hasil Analisis Data 2019)

8. Activity Diagram Data Surat Jalan.

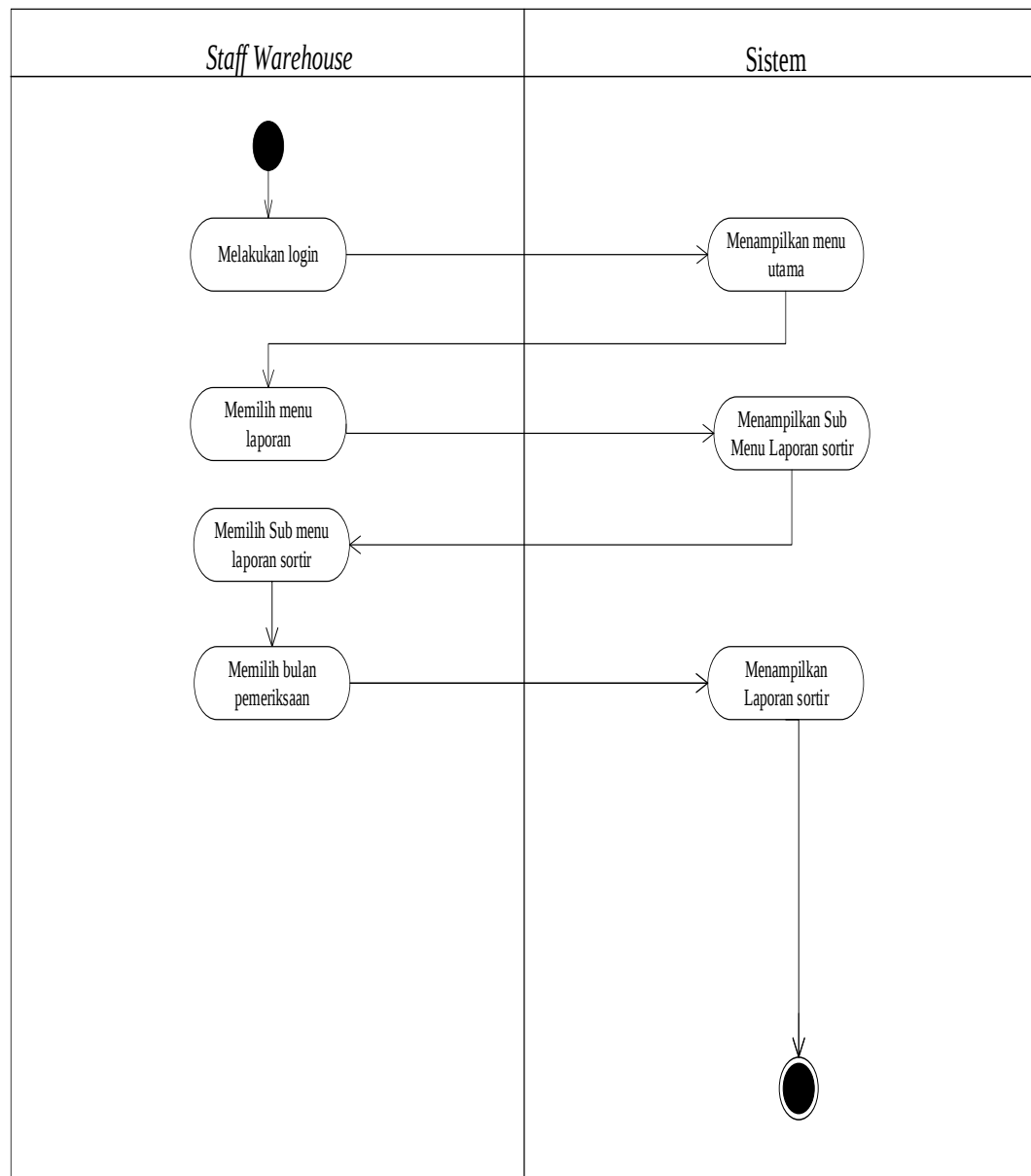
Activity diagram data surat jalan berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam data surat jalan yang dihasilkan. Activity diagram data surat jalan yang diusulkan dapat dilihat pada Gambar V.10:



Gambar V.10 Activity Diagram Data Surat Jalan
(Sumber: Hasil Analisis Data 2019)

9. Activity Diagram Merekap Laporan Laporan Sortir

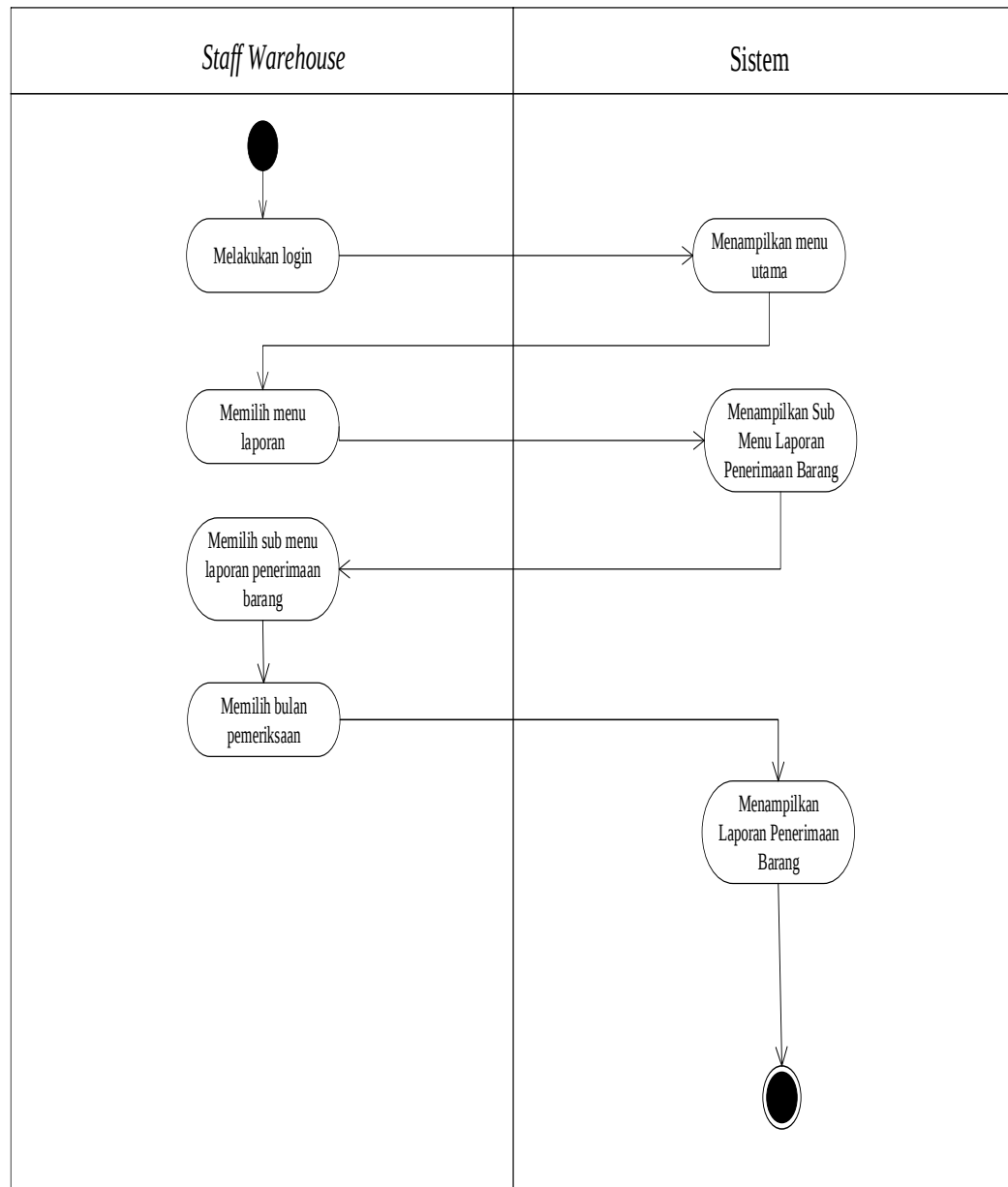
Activity diagram merekap laporan sortir berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam merekap laporan. Activity diagram merekap laporan sortir yang diusulkan dapat dilihat pada Gambar V.11:



Gambar V.11 Activity Diagram Merekap Laporan Sortir
(Sumber: Hasil Analisis Data 2019)

10. Activity Diagram Merekap Laporan Penerimaan Barang

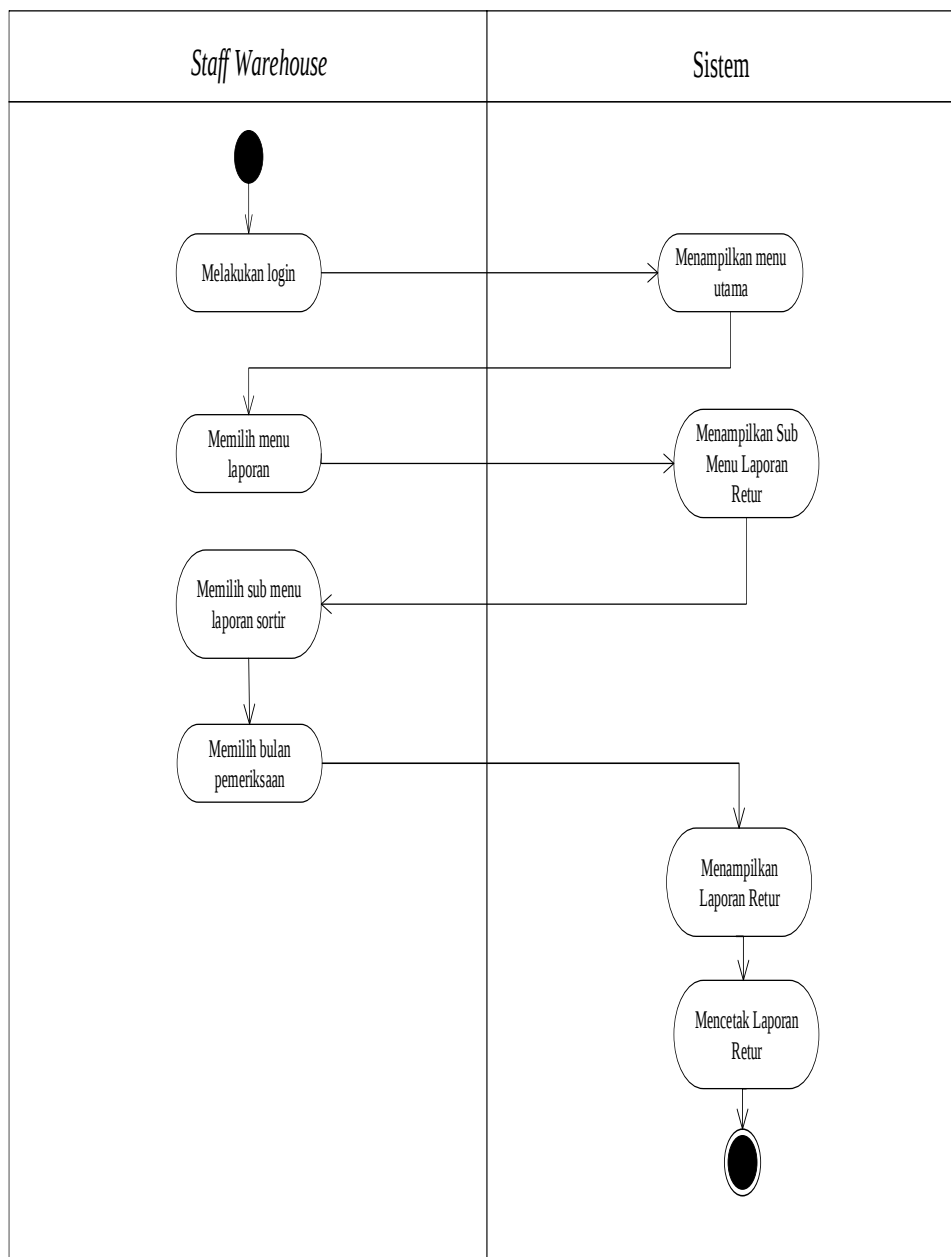
Activity diagram merekap laporan sortir berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam merekap laporan. Activity diagram merekap laporan sortir yang diusulkan dapat dilihat pada Gambar V.12:



Gambar V.12 Activity Diagram Merekap Laporan Penerimaan Barang
(Sumber: Hasil Analisis Data 2019)

11. Activity Diagram Mencetak Laporan Retur

Activity diagram mencetak laporan retur berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi dalam mencetak laporan. Activity diagram mencetak laporan retur yang diusulkan dapat dilihat pada Gambar V.13:



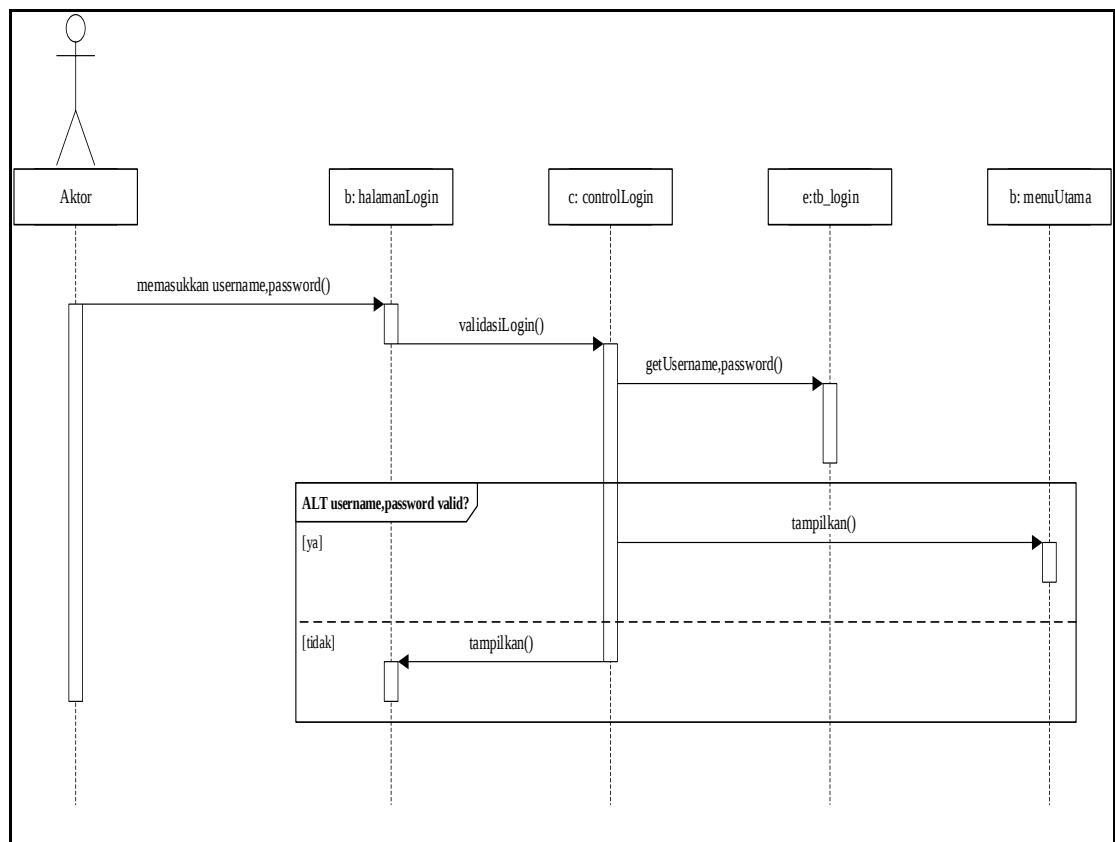
Gambar V.12 Activity Diagram Mencetak Laporan Retur
(Sumber: Hasil Analisis Data 2019)

5.6 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* pada sistem informasi pemeriksaan Komponen Part.

1. Sequence Diagram Proses Login

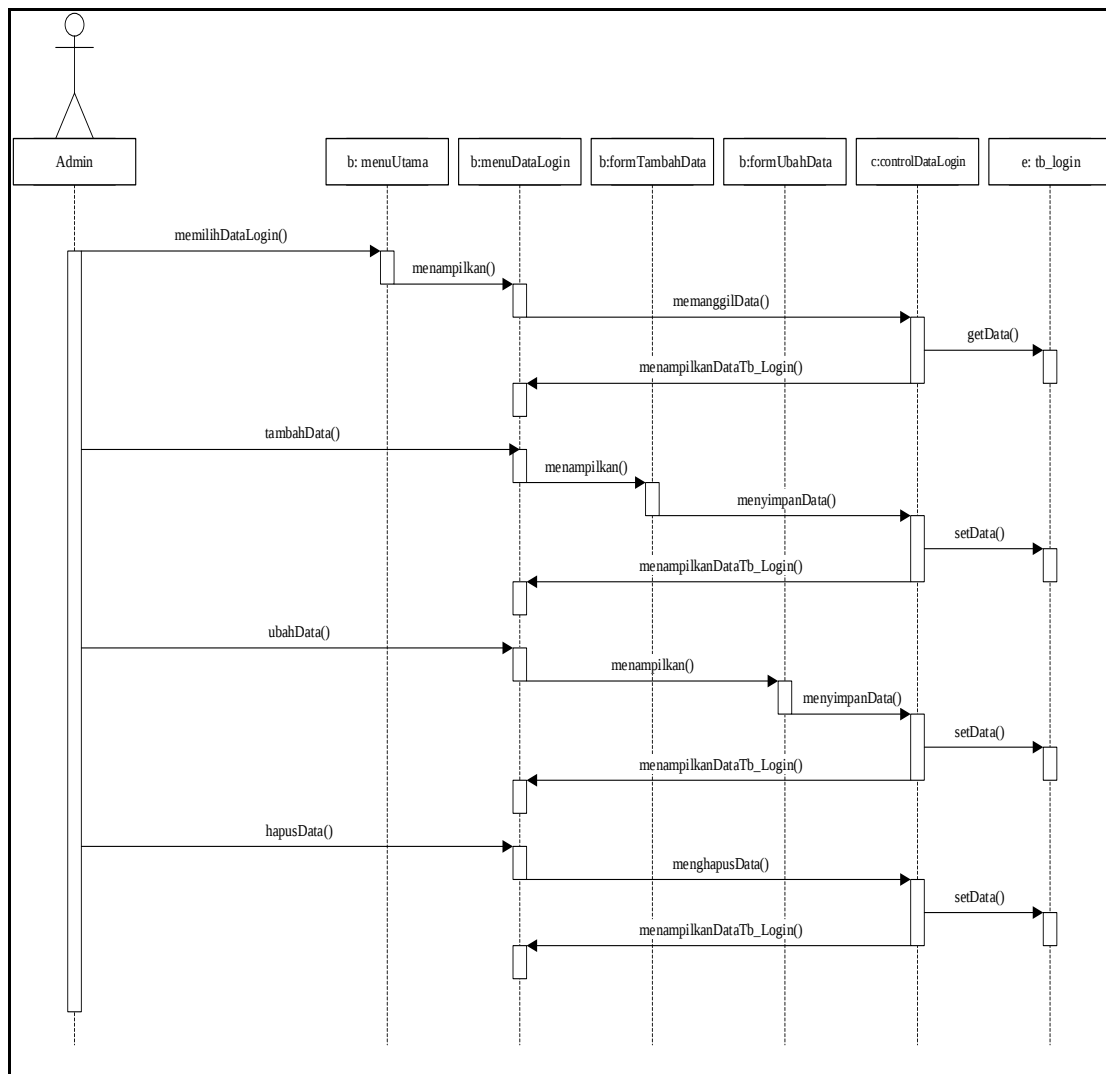
Sequence diagram login menjelaskan sebuah *sequence diagram* dalam proses *login*. Proses ini dilakukan oleh pengguna yang mempunyai hak akses sebelum masuk ke sistem. Adapun *sequence diagram* dari *use case login* dapat dilihat pada Gambar V.14:



Gambar V.14 Sequence Diagram Login
(Sumber: Hasil Analisis Data 2019)

2. Sequence Diagram Mengelola Master Data Login

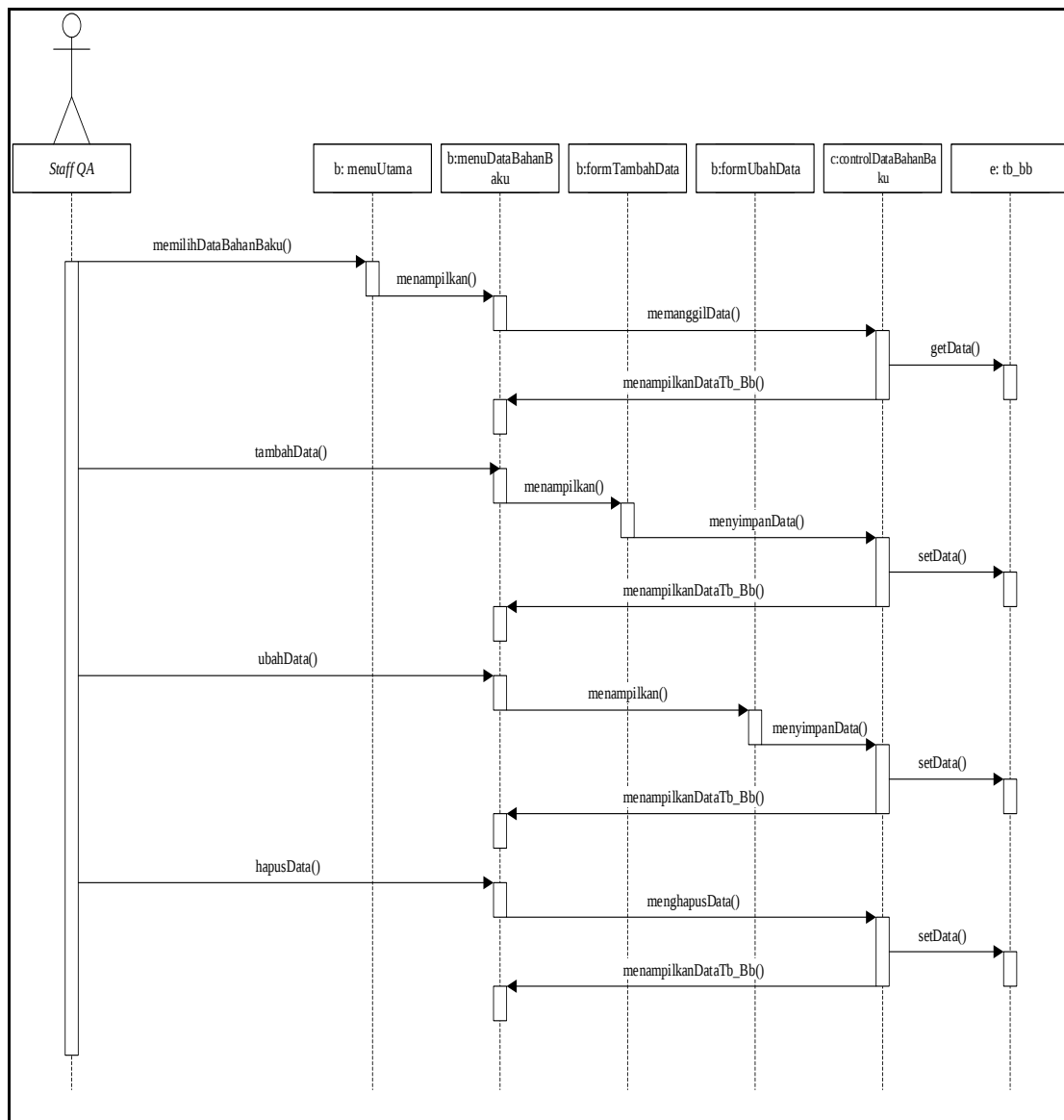
Sequence diagram mengelola master data login menjelaskan sebuah sequence diagram dalam proses pengelolaan master data login. Proses ini dilakukan oleh user, yaitu admin agar dapat menyimpan, menambah, mengubah, maupun menghapus data. Berikut merupakan sequence diagram dari use case mengelola master data login dapat dilihat pada Gambar V.15:



Gambar V.15 Sequence Diagram Mengelola Master Data Login
(Sumber: Hasil Analisis Data 2019)

3. Sequence Diagram Mengelola Master Data Komponen Part

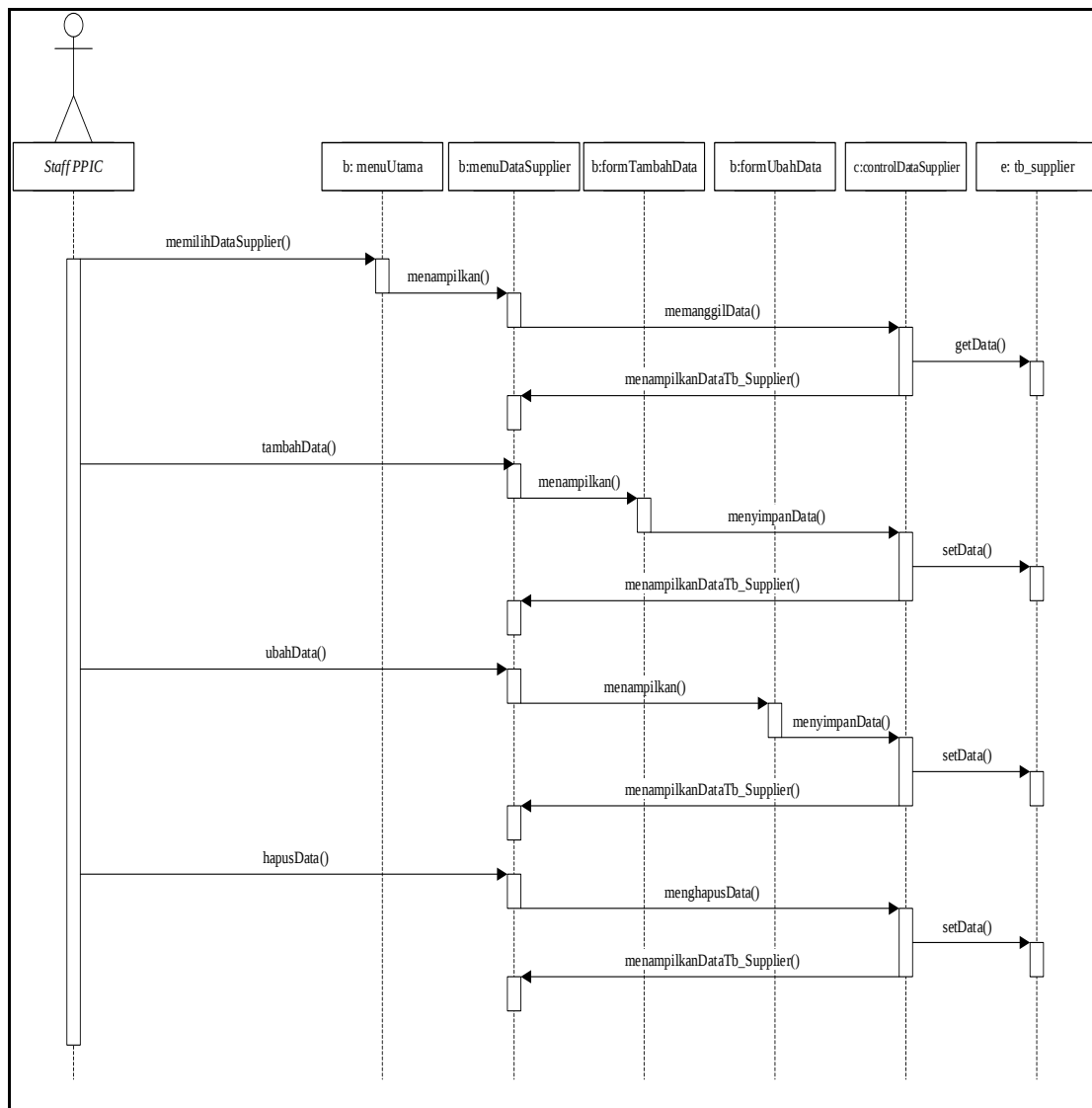
Sequence diagram mengelola *master data* Komponen Part menjelaskan sebuah *sequence diagram* dalam proses pengelolaan *master data* Komponen Part. Proses ini dilakukan oleh *user*, yaitu *Staff QA* agar dapat menyimpan, menambah, mengubah, maupun menghapus data. Berikut merupakan *sequence diagram* dari *use case* mengelola *master data* Komponen Part dapat dilihat pada Gambar V.16:



Gambar V.16 *Sequence Diagram* Mengelola *Master Data* Komponen Part
(Sumber: Hasil Analisis Data 2019)

4. Sequence Diagram Mengelola Master Data Supplier

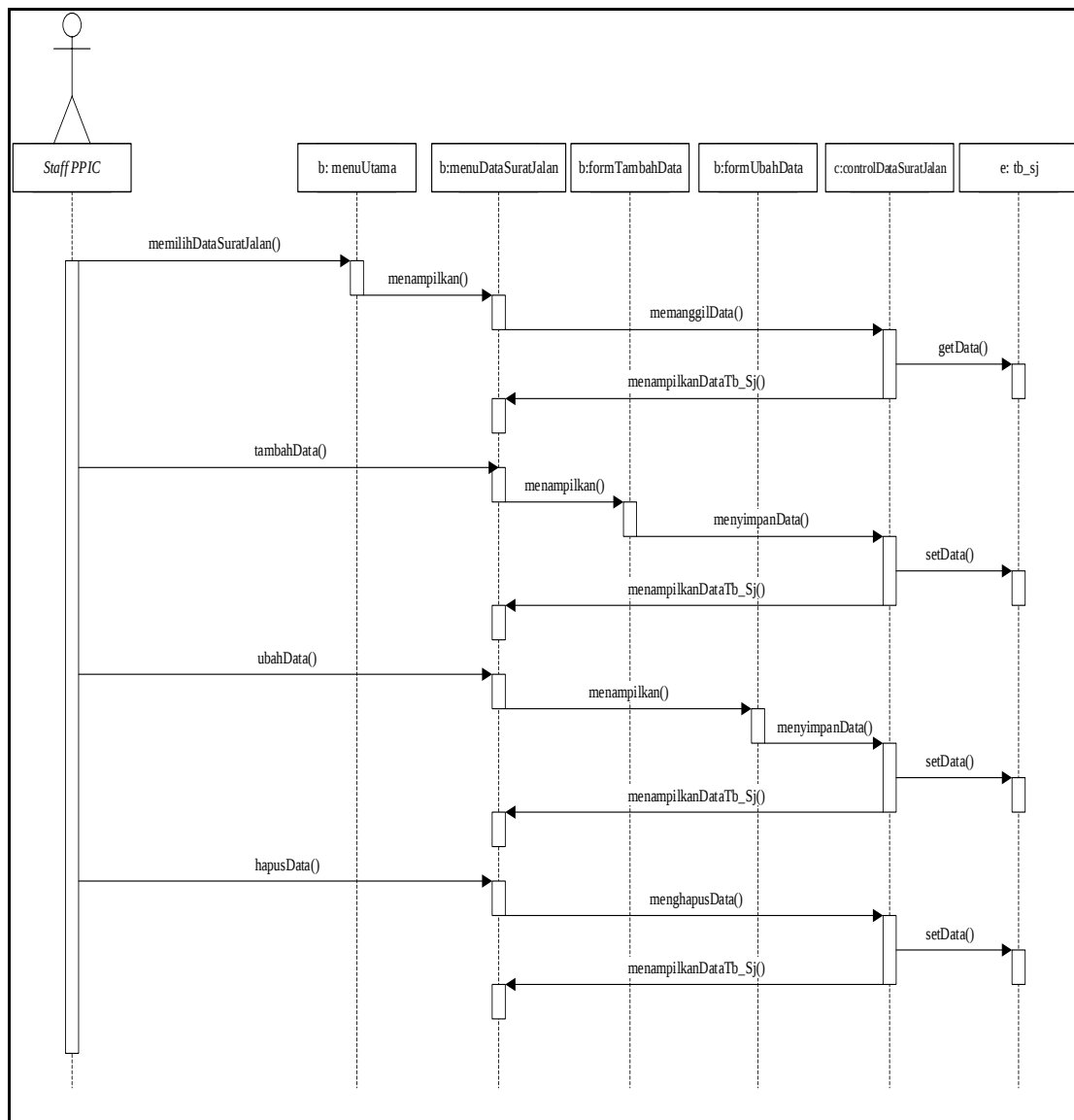
Sequence diagram mengelola master data supplier menjelaskan sebuah sequence diagram dalam proses pengelolaan master data supplier. Proses ini dilakukan oleh user, yaitu *Staff PPIC* agar dapat menyimpan, menambah, mengubah, maupun menghapus data. Berikut merupakan sequence diagram dari use case mengelola master data supplier dapat dilihat pada Gambar V.17:



Gambar V.17 Sequence Diagram Mengelola Master Data Supplier
(Sumber: Hasil Analisis Data 2019)

5. Sequence Diagram Mengelola Data Surat Jalan

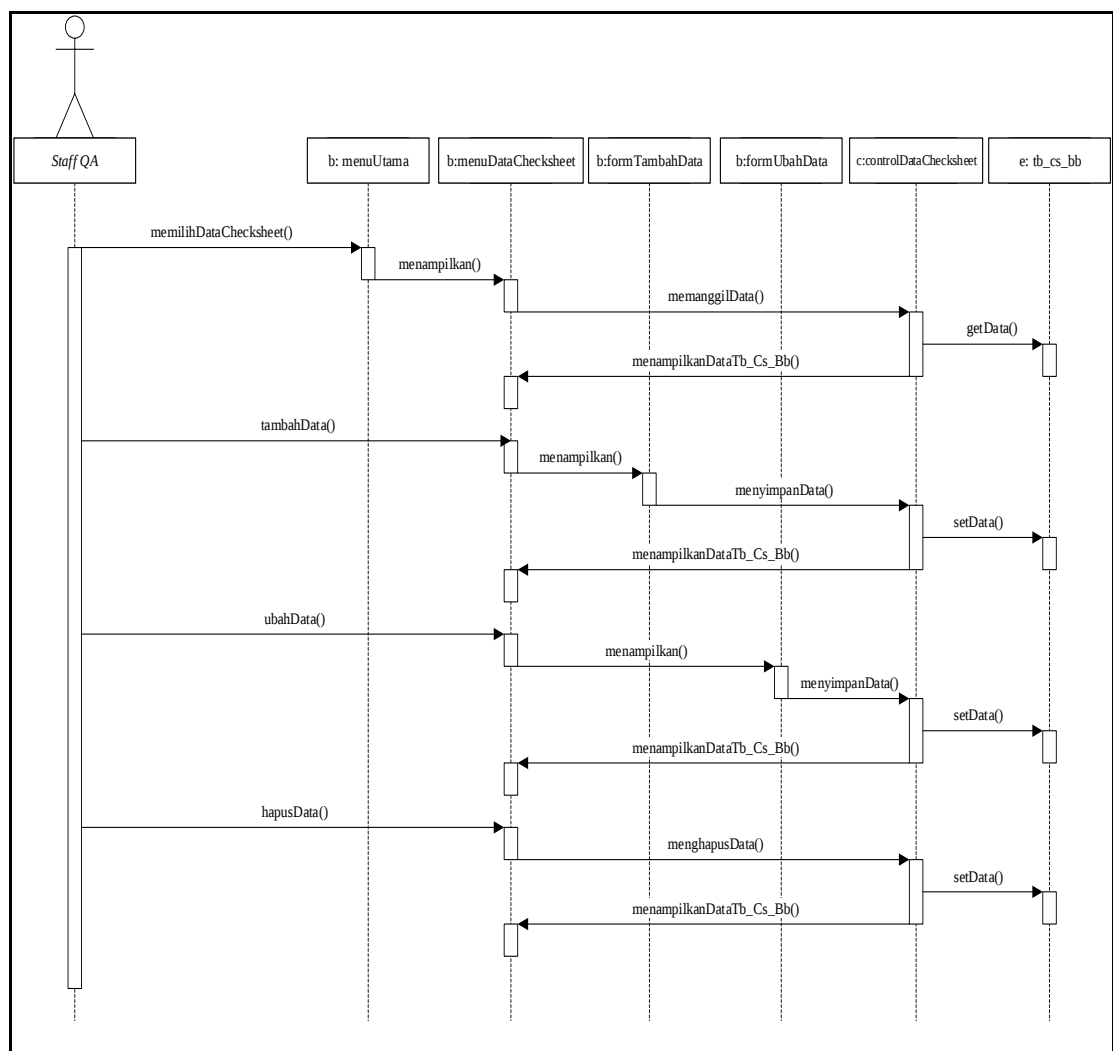
Sequence diagram mengelola data surat jalan menjelaskan sebuah *sequence diagram* dalam proses pengelolaan data surat jalan. Proses ini dilakukan oleh *user*, yaitu *Staff PPIC* agar dapat menyimpan, menambah, mengubah, maupun menghapus data. Berikut merupakan *sequence diagram* dari *use case* mengelola data surat jalan dapat dilihat pada Gambar V.18:



Gambar V.18 *Sequence Diagram* Mengelola Data Surat Jalan
(Sumber: Hasil Analisis Data 2019)

6. Sequence Diagram Mengelola Data Checksheet

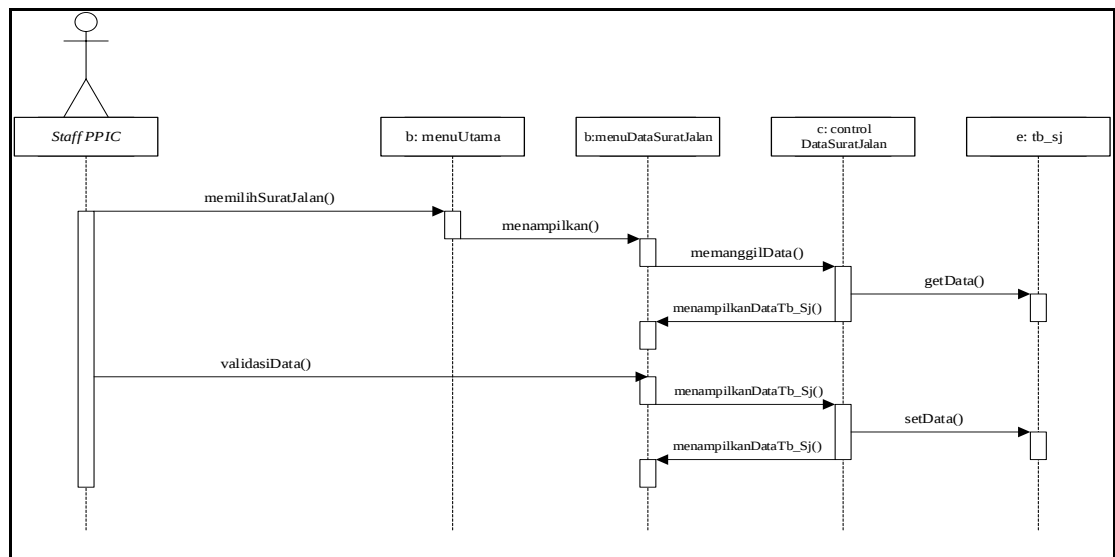
Sequence diagram mengelola data *checksheet* menjelaskan sebuah *sequence diagram* dalam proses pengelolaan data *checksheet*. Proses ini dilakukan oleh user, yaitu *Staff QA* agar dapat menyimpan, menambah, mengubah, maupun menghapus data. Berikut merupakan *sequence diagram* dari *use case* mengelola data *checksheet* dapat dilihat pada Gambar V.19:



Gambar V.19 Sequence Diagram Mengelola Data Checksheet
(Sumber: Hasil Analisis Data 2019)

7. *Sequence Diagram Melakukan Validasi Data Surat Jalan*

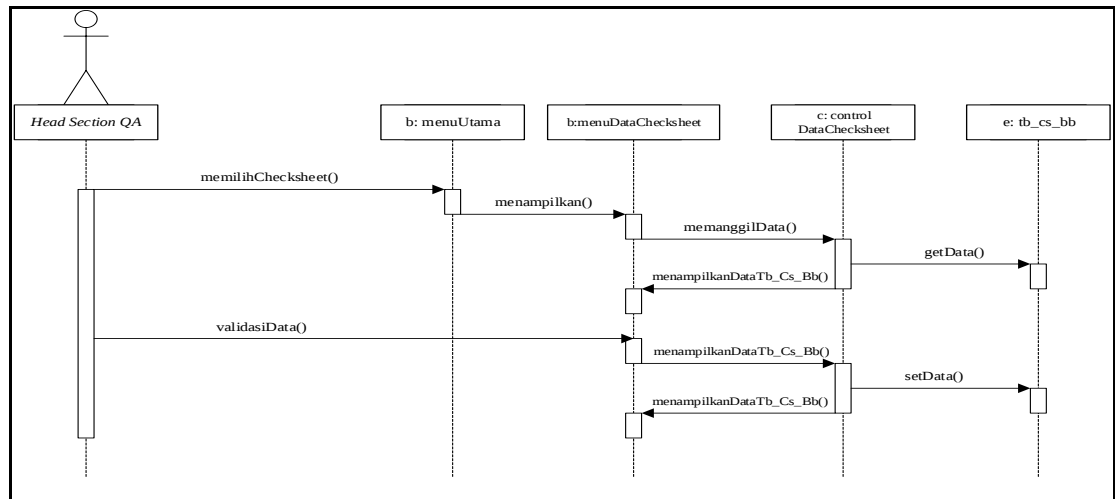
Sequence diagram melakukan data surat jalan menjelaskan sebuah *sequence diagram* dalam proses validasi data surat jalan. Proses ini dilakukan oleh *user*, yaitu *Staff PPIC* agar dapat memvalidasi data. Berikut merupakan *sequence diagram* dari *use case* melakukan validasi data surat jalan dapat dilihat pada Gambar V.20:



Gambar V.20 *Sequence Diagram Melakukan Validasi Data Surat Jalan*
(Sumber: Hasil Analisis Data 2019)

8. *Sequence Diagram Melakukan Validasi Data Checksheet*

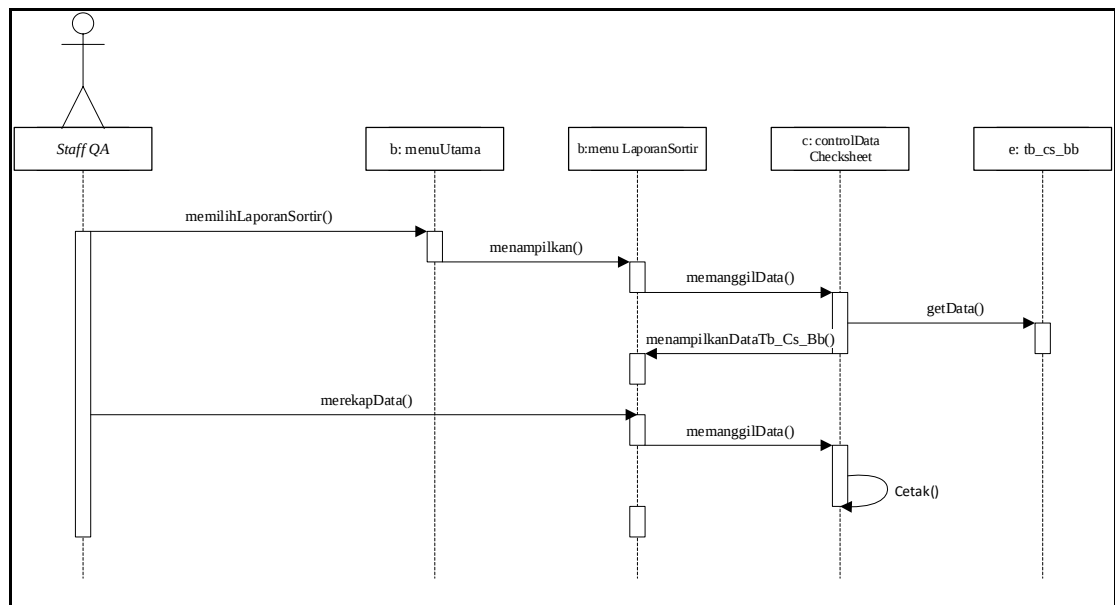
Sequence diagram melakukan validasi data surat jalan menjelaskan sebuah *sequence diagram* dalam proses validasi data surat jalan. Proses ini dilakukan oleh *user*, yaitu *Head Section QA* agar dapat memvalidasi data. Berikut merupakan *sequence diagram* dari *use case* melakukan validasi data surat jalan dapat dilihat pada Gambar V.21:



Gambar V.21 *Sequence Diagram* Melakukan Validasi Data Checksheet
(Sumber: Hasil Analisis Data 2019)

9. *Sequence Diagram* Merekap Laporan Sortir

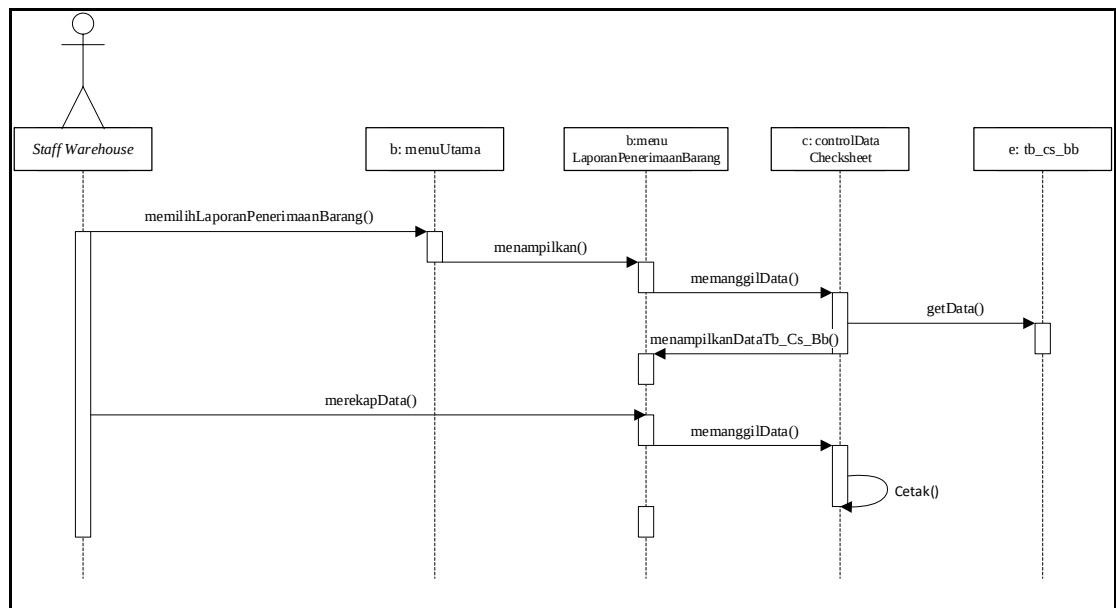
Sequence diagram merekap laporan sortir menjelaskan sebuah *sequence diagram* dalam proses pembuatan laporan sortir. Proses ini dilakukan oleh user, yaitu Staff QA agar dapat merekap laporan sortir. Berikut merupakan *sequence diagram* dari use case merekap laporan sortir dapat dilihat pada Gambar V.22:



Gambar V.22 *Sequence Diagram* Merekap Laporan Sortir
(Sumber: Hasil Analisis Data 2019)

10. *Sequence Diagram* Merekap Laporan Penerimaan Barang

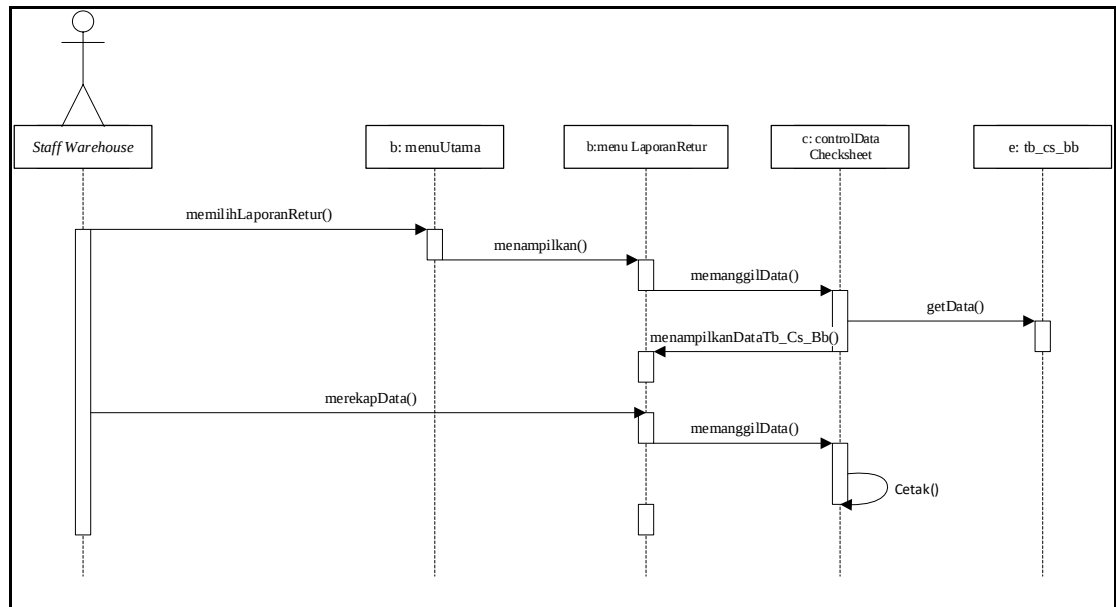
Sequence diagram merekap laporan penerimaan barang menjelaskan sebuah *sequence diagram* dalam proses pembuatan laporan penerimaan barang. Proses ini dilakukan oleh user, yaitu *Staff Warehouse* agar dapat merekap laporan penerimaan barang. Berikut merupakan *sequence diagram* dari *use case* merekap laporan penerimaan barang dapat dilihat pada Gambar V.23:



Gambar V.23 *Sequence Diagram* Merekap Laporan Penerimaan Barang
(Sumber: Hasil Analisis Data 2019)

11. *Sequence Diagram* Merekap Laporan Retur

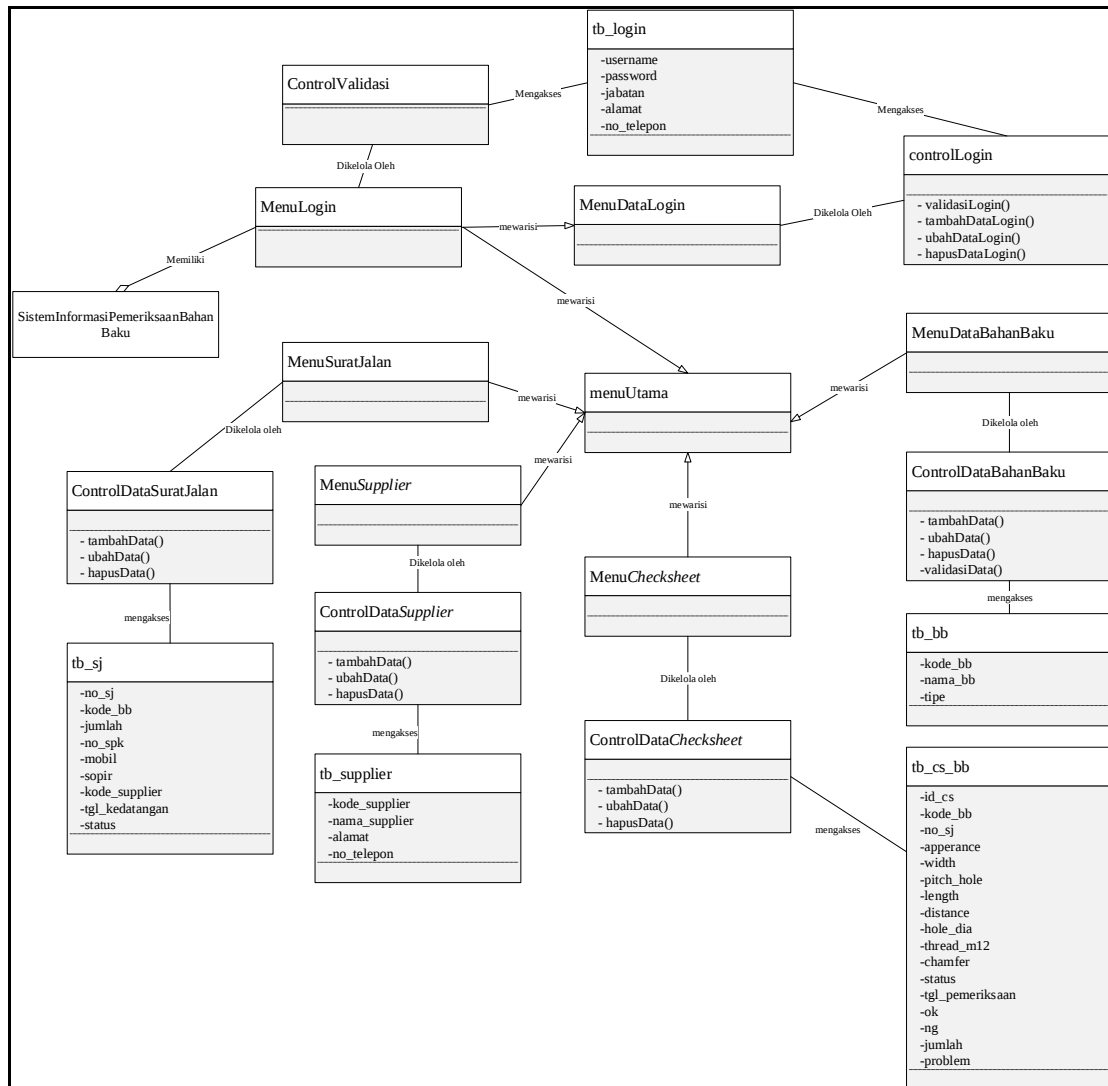
Sequence diagram merekap laporan retur menjelaskan sebuah *sequence diagram* dalam proses pembuatan laporan retur. Proses ini dilakukan oleh user, yaitu *Staff Warehouse* agar dapat merekap laporan retur. Berikut merupakan *sequence diagram* dari *use case* merekap laporan retur dapat dilihat pada Gambar V.24:



Gambar V.24 Sequence Diagram Merekap Laporan Retur
(Sumber: Hasil Analisis Data 2019)

5.7 Class Diagram

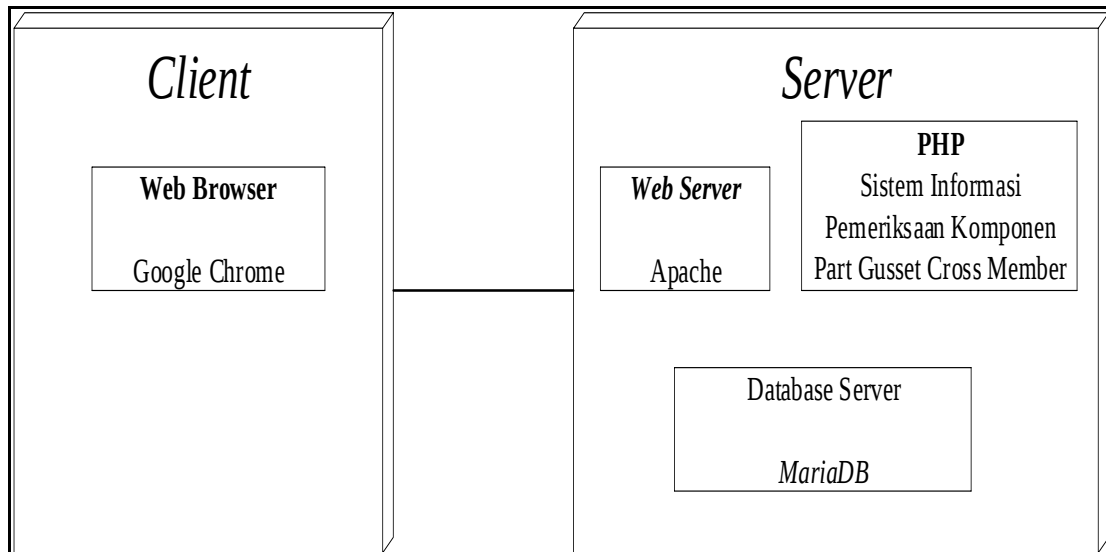
Class Diagram pada usulan sistem informasi pemeriksaan Komponen Part ini digunakan untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat. Untuk membangun sistem, *class diagram* dapat memperlihatkan hubungan antar kelas dan penjelasan detail pada setiap kelas di dalam model desain dari suatu sistem. Berikut merupakan *class diagram* usulan sistem informasi pemeriksaan Komponen Part :



Gambar V.25 *Class Diagram* Sistem Informasi Pemeriksaan Komponen Part
(Sumber: Hasil Analisis 2019)

5.8 Deployment Diagram

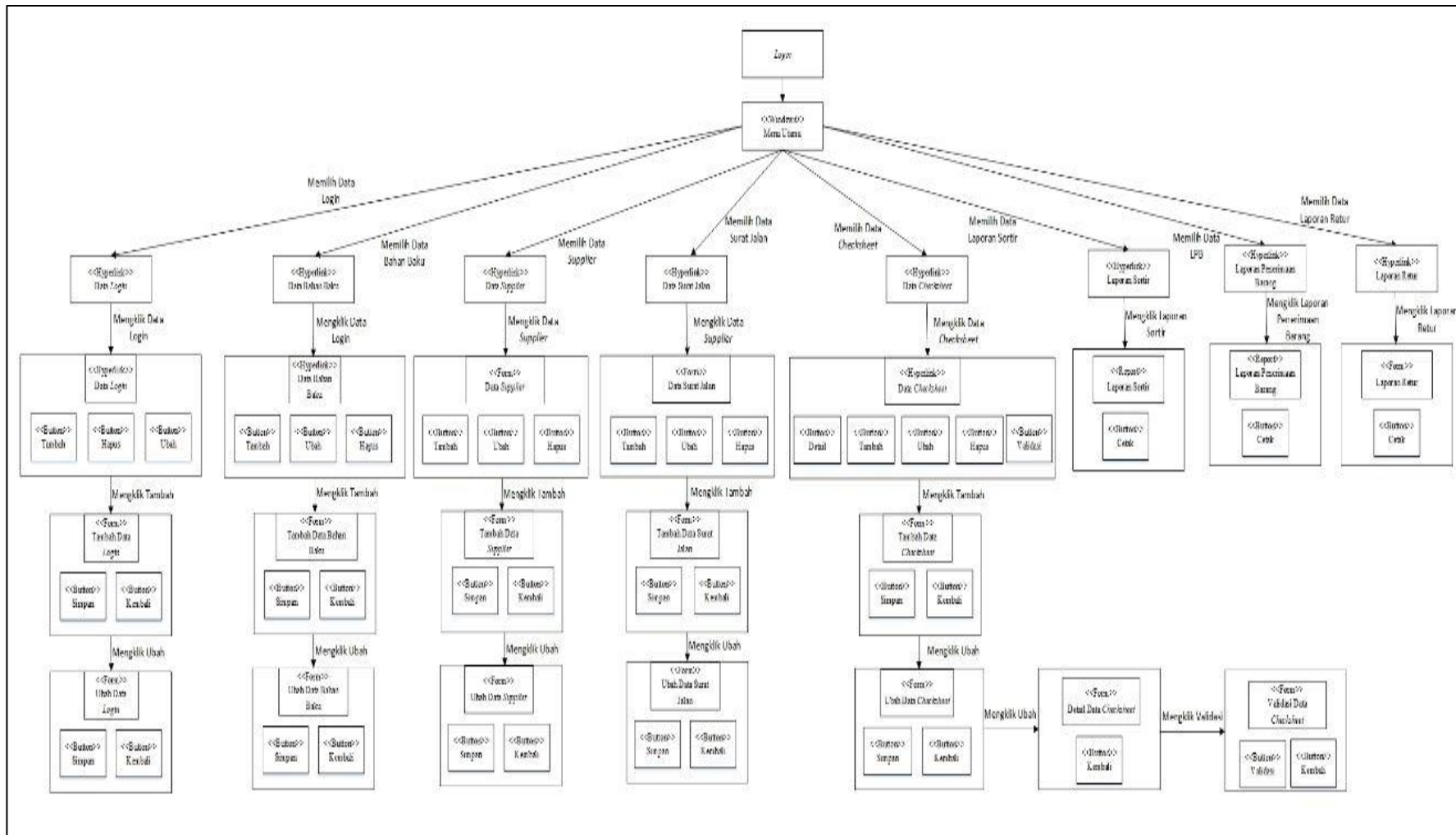
Deployment diagram pada usulan sistem informasi pemeriksaan Komponen Part ini digunakan untuk mewakili komponen-komponen perangkat lunak dan cara perangkat lunak tersebut ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. Berikut ini merupakan *deployment diagram* sistem informasi pemeriksaan Komponen Part yang dapat dilihat pada gambar V.26:



Gambar V.26 *Deployment Diagram* Sistem Informasi Pemeriksaan Komponen Part
(Sumber: Hasil Analisis 2019)

5.9 *Windows Navigation Diagram (WND)*

Windows Navigation Diagram pada sistem usulan digunakan untuk menunjukan bagaimana navigasi dari halaman-halaman yang terdapat di dalam aplikasi, berikut merupakan *Windows Navigation Diagram* sistem informasi pemeriksaan Komponen Part:



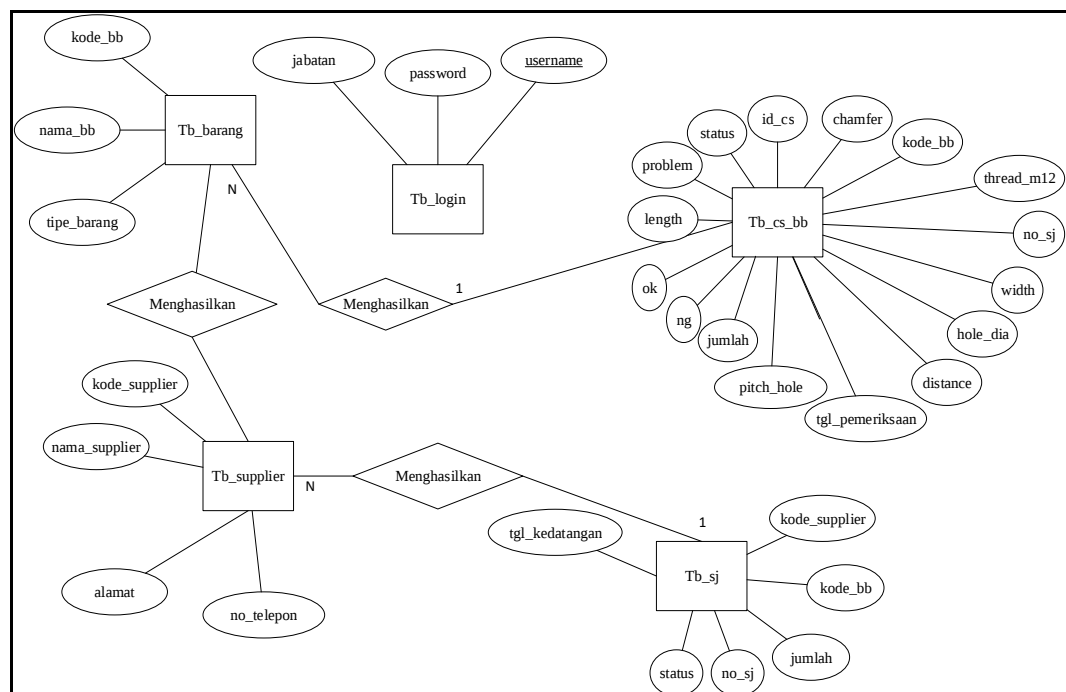
Gambar V.27 Windows Navigation Diagram Sistem Informasi Pemeriksaan Komponen Part
(Sumber: Hasil Analisis 2019)

5.10 Pemodelan Data Sistem Usulan

Pemodelan data pada sistem informasi pemeriksaan Komponen Part di PT Gemala Kempa Daya menggunakan dua cara, yakni *Entity Relationship Diagram* (ERD) yang digunakan untuk menjelaskan hubungan antar data dalam basis data dan kamus data yang digunakan untuk menjelaskan isi dari *database* yang digunakan dalam sistem usulan yang akan dibuat.

5.10.1 Entity Relationship Diagram(ERD)

Entity relationship diagram dapat membantu dalam merancang hubungan antara entitas dari sebuah *database*. Diagram ini berisi data-data yang berada di dalam *database* yang dapat membantu dalam mengetahui relasi keterhubungan antar tabel. *Entity relationship diagram* sistem informasi penjadwalan produksi dapat dilihat sebagai berikut:



Gambar V.27 Relationship Entity Diagram (ERD) Sistem Informasi Pemeriksaan Komponen Part

(Sumber: Hasil Analisis 2019)

5.10.2 Kamus Data

Kamus data membantu dalam pendefinisian data agar pendefinisian data tersebut dapat dilakukan dengan lengkap dan terstruktur. Pembentukan kamus data dilaksanakan dalam tahap analisis dan perancangan suatu sistem. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar penulisan). Berikut ini adalah kamus data yang terdapat dalam aplikasi sistem informasi pemeriksaan Komponen Part:

1. Spesifikasi Tabel Data *Login*

Nama Tabel : tb_login

Fungsi : Untuk menyimpan data *login*

Tipe : *File data master*

Tabel V.15 Tabel Data *Login*

No.	Nama Elemen	Akronim	Tipe	Width	Keterangan
1	<i>Username Login</i>	username	<i>varchar</i>	10	<i>Primary key</i>
2	<i>Password Login</i>	password	<i>Varchar</i>	20	
3	Jabatan	jabatan	<i>Varchar</i>	20	
4	Alamat Karyawan	Alamat	<i>Text</i>		
5	Nomor telepon	No_telepon	<i>Integer</i>	10	

(Sumber: Hasil Analisis Data, 2019)

2. Spesifikasi Tabel Data Komponen Part

Nama Tabel : tb_bb

Fungsi : Untuk menyimpan data Komponen Part

Tipe : *File data master*

Tabel V.16 Tabel Data Komponen Part

No.	Nama Elemen	Akronim	Tipe	Width	Keterangan
1	Kode Komponen Part	kode_bb	<i>Integer</i>	10	<i>Primary key</i>
2	Nama Komponen Part	nama_bb	<i>Varchar</i>	20	
3	Tipe Komponen Part	tipe	<i>Varchar</i>	10	

(Sumber: Hasil Analisis Data, 2019)

3. Spesifikasi Tabel Data *Supplier*

Nama Tabel : *tb_supllier*

Fungsi : Untuk menyimpan data *supplier*

Tipe : *File data master*

Tabel V.17 Tabel Data *Supplier*

No.	Nama Elemen	Akronim	Tipe	Width	Keterangan
1	Kode <i>Supplier</i>	kode_supplier	<i>Varchar</i>	10	<i>Primary key</i>
2	Nama <i>Supplier</i>	nama_supplier	<i>Varchar</i>	20	
3	Alamat <i>Supplier</i>	Alamat	<i>Text</i>		
4	Nomor Telepon	No_telepon	<i>Integer</i>	10	

(Sumber: Hasil Analisis Data, 2019)

4. Spesifikasi Tabel Data Surat Jalan

Nama Tabel : *tb_sj*

Fungsi : Untuk menyimpan data Surat Jalan

Tipe : *File data transaksi*

Tabel V.18 Tabel Data Surat Jalan

No.	Nama Elemen	Akronim	Tipe	Width	Keterangan
1	Nomor surat jalan	no_sj	<i>Varchar</i>	5	<i>Primary key</i>
2	Kode Komponen Part	kode_bb	<i>Varchar</i>	10	<i>Foreign key</i>
3	Kode <i>supplier</i>	kode_supplier	<i>Varchar</i>	10	<i>Foreign key</i>
4	Jumlah Komponen Part	jumlah	<i>Integer</i>	5	
5	Tanggal Kedatangan	tgl_kedatangan	<i>Date</i>		
6	Status Validasi	status	<i>Varchar</i>	5	

(Sumber: Hasil Analisis Data, 2019)

5. Spesifikasi Tabel Data *Checksheets* Komponen Part

Nama Tabel : tb_cs_bb

Fungsi : Untuk menyimpan data *checksheets* Komponen Part

Tipe : *File data transaksi*

Tabel V.19 Tabel Data *Checksheet* Komponen Part

No.	Nama Elemen	Akronim	Tipe	Width	Keterangan
1	Nomor surat jalan	no_sj	<i>Varchar</i>	5	<i>Foreign key</i>
2	Kode Komponen Part	kode_bb	<i>Varchar</i>	10	<i>Foreign key</i>
3	Kode <i>supplier</i>	kode_supplier	<i>Varchar</i>	10	<i>Foreign key</i>
4	Total Komponen Part	jumlah	<i>Integer</i>	5	
5	Jumlah OK	ok	<i>Varchar</i>	10	
6	Jumlah NG	ng	<i>Varchar</i>	10	
7	Tanggal Pemeriksaan	tgl_pemeriksaan	<i>Date</i>		
8	Status Validasi	status	<i>Varchar</i>	5	
9	Karakteristik	appearance	<i>Varchar</i>	5	
10	Karakteristik	width	<i>Varchar</i>	5	
11	Karakteristik	pitch_hole	<i>Varchar</i>	5	
12	Karakteristik	length	<i>Varchar</i>	5	
13	Karakteristik	distance	<i>Varchar</i>	5	
14	Karakteristik	hole_dia	<i>Varchar</i>	5	
15	Karakteristik	thread_m12	<i>Varchar</i>	5	
16	Karakteristik	chamfer	<i>Varchar</i>	5	
17	Kecacatan	problem	<i>Varchar</i>	5	

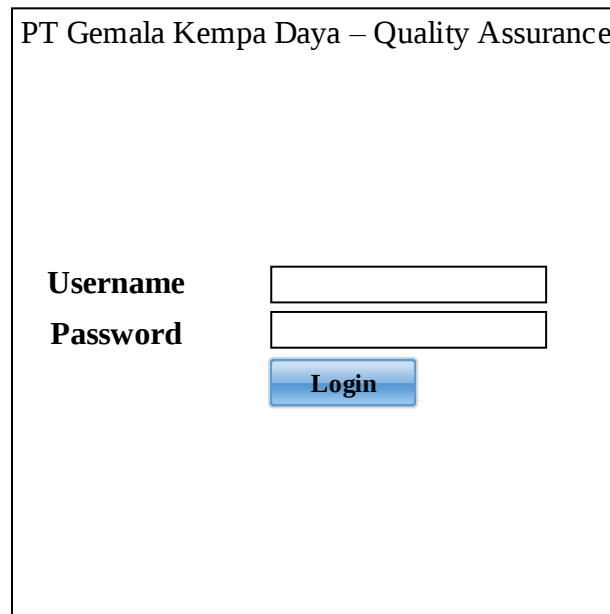
(Sumber: Hasil Analisis Data, 2019)

5.10.3 Perancangan *Interface* Sistem Informasi Pemeriksaan Komponen Part

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

1. *Form Login*

Form Login digunakan untuk membedakan hak akses *user*. Melalui *form* ini, pengguna yang dapat masuk kedalam sistem adalah pengguna yang mengetahui kode *login* dan *password* pengguna yang memiliki wewenang untuk menggunakan sistem informasi pemeriksaan Komponen Part.



PT Gemala Kempa Daya – Quality Assurance

Username

Password

Login

Gambar V.29 Rancangan *Form Login*
(Sumber: Hasil Analisis, 2019)

Keterangan:

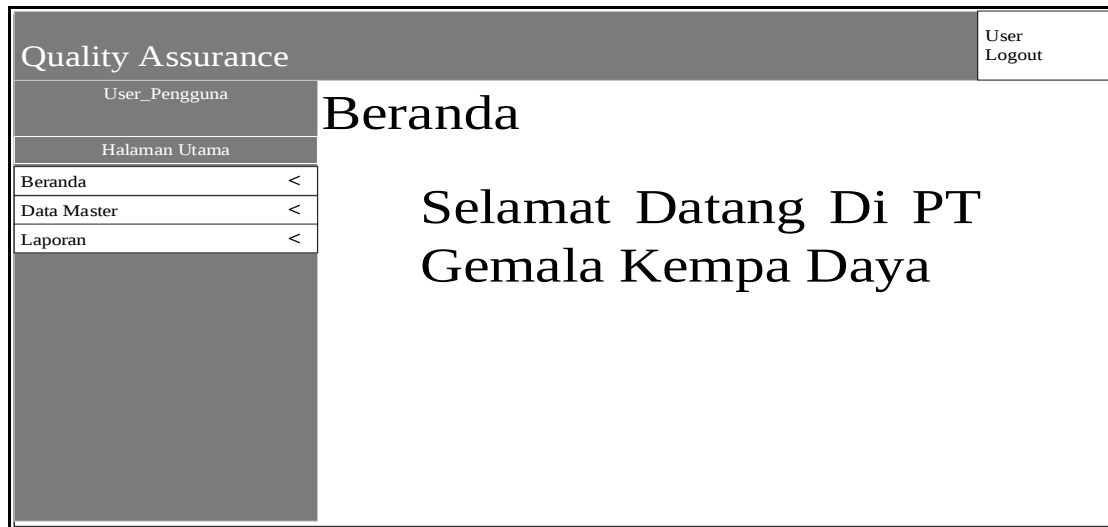
Username : untuk memasukkan *username*

Password : untuk memasukkan *password*

Button Login : Digunakan untuk proses masuk kedalam sistem.

2. Menu Utama

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

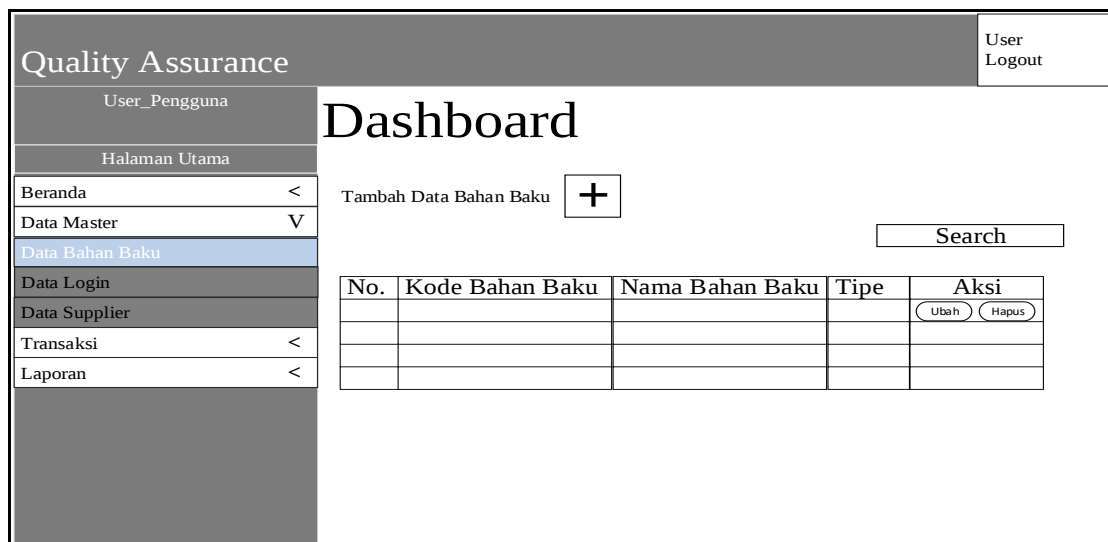


Gambar V.30 Rancangan Menu Utama
(Sumber: Hasil Analisis, 2019)

3. Data Master

Data *master* ini terdiri dari *Master* Komponen Part, data *login*, dan data *supplier*. Berikut penjelasan dari setiap data tersebut:

- 1) *Master* data Komponen Part merupakan *form master* yang digunakan untuk melihat dan memasukkan data *master* Komponen Part.



Gambar V.31 Rancangan *Form Master* Data Komponen Part
(Sumber: Hasil Analisis, 2019)

4. *Master data login* merupakan *form master* yang digunakan untuk melihat dan memasukkan data *login* yang berfungsi untuk hak akses.

Quality Assurance

User Pengguna

Halaman Utama

Beranda <

Data Master V

Data Bahan Baku

Data Login

Data Supplier

Transaksi <

Laporan <

Dashboard

Tambah Data Login +

Search

No.	Username	Password	Alamat	Jabatan	Aksi
					Ubah Hapus

User Logout

Gambar V.32 Rancangan *Form Master Data Login*
(Sumber: Hasil Analisis, 2019)

5. *Master data supplier* merupakan *form master* yang digunakan untuk melihat dan memasukkan data *supplier*.

Quality Assurance

User Pengguna

Halaman Utama

Beranda <

Data Master V

Data Bahan Baku

Data Login

Data Supplier

Transaksi <

Laporan <

Dashboard

Tambah Data Supplier +

Search

No.	Kode Supplier	Nama Supplier	Alamat	No Telepon	Aksi
					Ubah Hapus

User Logout

Gambar V.33 Rancangan *Form Master Data Supplier*
(Sumber: Hasil Analisis, 2019)

6. Data Surat Jalan

Data Surat Jalan adalah *form* data untuk menunjukkan surat jalan yang ada dan memasukkan data tersebut.

Quality Assurance

User Pengguna

Halaman Utama

Beranda <

Data Master <

Transaksi v

Data Surat Jalan

Data Checksheet

Laporan <

Dashboard

Tambah Data Surat Jalan +

Search

No.	Nomor Surat Jalan	Kode Bahan Baku	Kode Supplier	Jumlah	Mobil	Sopir	Tanggal Kedatangan	Sopir	Aksi
									Ubah Hapus

Gambar V.34 Rancangan *Form* Data Surat Jalan
(Sumber: Hasil Analisis, 2019)

7. Data Checksheet

Data *Checksheet* adalah *form* data untuk menunjukkan *checksheet* yang ada dan memasukkan data tersebut.

The screenshot shows a web application interface for Quality Assurance. The top header is 'Quality Assurance' with a 'User Logout' link. A sidebar on the left contains a menu with items: 'User_Pengguna', 'Halaman Utama', 'Beranda', 'Data Master', 'Transaksi', 'Data Surat Jalan', 'Data Checksheet' (highlighted), and 'Laporan'. The main content area is titled 'Dashboard' and features a 'Tambah Data Checksheet' button with a plus icon. Below this is a table with columns: 'No.', 'ID Checksheet', 'Tanggal Pemeriksaan', 'Kode Bahan Baku', 'Nomor Surat', 'Mobil', 'Status Validasi', and 'Aksi'. The 'Aksi' column contains buttons for 'Ubah', 'Detail', 'Validasi', and 'Hapus'. A search bar is located in the top right of the main area.

Gambar V.35 Rancangan *Form Data Checksheet*

(Sumber: Hasil Analisis, 2019)

8. *Form Data Laporan Sortir*

Form data laporan sortir adalah *form* yang digunakan untuk merekap data hasil dari proses pengendalian Komponen Part *Gusset Cross Member*. Rancang *form* laporan sortir dapat dilihat pada Gambar V.36:

The screenshot shows the same web application interface as Gambar V.35, but with the 'Laporan Sortir' menu item highlighted in the sidebar. The main content area is titled 'Dashboard' and features a 'Pilih Bulan' dropdown menu with a 'V' icon. Below this is a table with columns: 'No.', 'ID Checksheet', 'Nomor Surat Jalan', 'Kode Bahan Baku', 'OK', 'NG', and 'Jumlah'. A search bar is located in the top right of the main area.

Gambar V.36 Rancangan *Form Laporan Sortir*

(Sumber: Hasil Analisis, 2019)

9. Form Data Laporan Penerimaan Barang

Form data laporan penerimaan barang adalah *form* yang digunakan untuk merekap data hasil dari proses pengendalian Komponen Part Gusset Cross Member. Rancang *form* laporan penerimaan barang dapat dilihat pada Gambar V.37:

The screenshot shows a web application interface for Quality Assurance. On the left is a sidebar menu with the following items: Beranda, Data Master, Transaksi, Laporan, Laporan Sortir, Laporan Penerimaan Barang (highlighted), and Laporan Retur. The main content area is titled 'Dashboard' and contains a 'Pilih Bulan' dropdown menu set to 'V'. Below this is a table with the following columns: No., ID Checksheet, Nomor Surat Jalan, Kode Bahan Baku, Jumlah, and Keterangan. The table has four empty rows for data entry. A 'Search' button is located to the right of the table. In the top right corner, there are links for 'User' and 'Logout'.

Gambar V.37 Rancangan *Form* Laporan Penerimaan Barang
(Sumber: Hasil Analisis, 2019)

10. Form Data Laporan Retur

Form data laporan retur adalah *form* yang digunakan untuk merekap data hasil dari proses pengendalian Komponen Part Gusset Cross Member. Rancang *form* laporan retur dapat dilihat pada Gambar V.38:

This screenshot shows the same Quality Assurance Dashboard as Gambar V.37, but with the 'Laporan Retur' option highlighted in the sidebar menu. The main content area is identical, featuring the 'Pilih Bulan' dropdown, the data entry table with columns No., Nomor Surat Jalan, Kode Bahan Baku, Jumlah, and Keterangan, and the 'Search' button. The 'User' and 'Logout' links remain in the top right corner.

Gambar V.38 Rancangan *Form* Laporan Retur
(Sumber: Hasil Analisis, 2019)

5.10.4 Implementasi Sistem *Software* dan *Hardware*

Tahap ini adalah tahap pengkodean program menggunakan bahasa pemrograman *PHP* menggunakan *framework CodeIgniter* 3.1.8 dengan *Sublime Text* 3 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* 10
 - b. *Database Server* : *MariaDB* versi 10.1.36
 - c. Bahasa Pemrograman : *CodeIgniter* 3.1.8
2. Analisis Kebutuhan *Hardware*
 - a. *Processor* : Minimal *Processor Intel Core* i3 550
 - b. RAM : Minimal RAM 2 GB
 - c. *Harddisk* : Minimal *Harddisk* 64 GB
 - d. *Mouse, Keyboard, Monitor* sebagai peralatan muka.
3. Analisis Kebutuhan *Network*
 - a. Domain : Untuk alamat *website*
 - b. Hosting : Minimal 25Mb

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil sistem informasi yang telah dirancang mengenai sistem informasi pemeriksaan bahan baku pada PT Gemala Kempa Daya, dapat disimpulkan beberapa hal, yaitu:

1. Dengan adanya sistem informasi pemeriksaan bahan baku dapat membantu dalam proses penyimpanan data *checksheet* dan surat jalan, sehingga penyimpanan data dalam satu lokasi dapat mempermudah pengguna dalam proses pencarian dan penyimpanan data.
2. Dengan adanya *database* diharapkan dapat mengurangi terjadinya kehilangan data dan mempercepat proses pembuatan laporan pemeriksaan bahan baku.

6.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem informasi penjadwalan produksi ini selanjutnya yaitu:

1. Mengimplementasikan hasil dari penelitian berupa sistem informasi untuk proses pemeriksaan bahan baku *Gusset Cross Member* di divisi *Quality Assurance* terutama dilini *QA Receiving Inspection* pada PT Gemala Kempa Daya.
2. Untuk pengembangan sistem informasi, untuk penelitian selanjutnya diharapkan dapat mencakup lebih dari satu bahan baku.

DAFTAR PUSTAKA

AN-NISBAH ж 132, Vol. 01, No. 02, April 2015

Definisi dan Jenis Bahan Baku. <https://erwinnote.wordpress.com/2011/09/21/definisi-dan-jenis-bahan-baku/>. (Tanggal Akses: 15 Januari 2017).

Febriani. 2004. *Flowchart*. febriani.staff.gunadarma.ac.id/Downloads/files/56/Flowchart.pdf. (Tanggal Akses; 15 Februari 2017)

Goal, L, Jimmy. 2008. *Sistem Informasi Manajemen Pemahaman dan Aplikasi*. Jakarta : Penerbit PT Grasindo.

Jogiyanto. 2005. *Analisis & Desain Sistem Informasi*. Andi Yogyakarta.

Jogiyanto. HM. 1999. *Analisis & Desain Sistem Informasi, Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*. Yogyakarta: Andi.

Kamus Besar Bahasa Indonesia Online. Kbbi.web.id. (Tanggal Akses; 18 Januari 2018)

Menurut (Masiyal Kholmi 2003:172)

O'Brien, James. 2006. *Pengantar Sistem Informasi Prespektif Bisnis dan Managerial*. Jakarta: Salemba Empat.

Simatupang. 1994. *Dasar-Dasar Sistem*, Jakarta: Gramedia Pustaka Utama.

Sinulingga, Sukaria. *Perencanaan dan Pengendalian Produksi*. Cetakan Kedua. Yogyakarta: Candi Gerbang Permai.

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

Tjiptono, F. Fandi, dan Diana, A. *Total Quality Management*. Yogyakarta: Penerbit Andi, 2003.

Ukas, Maman. 2004. *Manajemen Konsep, Prinsip dan Aplikasi*. Bandung: Penerbit Agnini

DHJ PT. DUTA HITA JAYA
STEEL STRUCTURES & POLES FABRICATOR

SURAT JALAN

Kepada Yth.
PT. GEMALA KEMPA DAYA

No. SJ : 05081
Mobil :
Sopir : 33278
SPK No. :
PO. No. : J4987/PO-BKD/0318

Kami kirimkan barang-barang sebagai berikut :

No.	Nama Barang	Jumlah
	JASA JASA PRESS BUSHING BRKT SPRING FRONT AH 31 LPM	240 PCS

Barang-barang tersebut telah diterima dalam keadaan baik dan lengkap sesuai dengan jumlahnya.

Penyerima, Pengirim, Quality Control, Bag. Gudang, Keamanan, Sopir,

Nama Tgl. Nama Tgl. Nama Tgl. Nama Tgl. Nama Tgl. Nama Tgl.

22/10/18 04/10/18 04/10/18 04/10/18 04/10/18 22/10/18

PJ. Marketing, Merch. Marketing, Bju. Adm. Keuangan, Hjuk. Pusat, Kuning. Customer

TAMBUK

Lampiran 1 Formulir Surat Jalan
(Sumber : PT Gemala Kempa Daya, 2018)

SURAT JALAN

TGL S.J : 10/10/2018
 NO. S.J : TJ18B10P0108
 NO. PO :
 NO. KEND : B9557 LCC

KEPADA YTH
 PT. GEMALA KEMPA DAYA
 DI PULOGADUNG

NO	NAMA BARANG	PACKING	JUML. BARANG	KETERANGAN
1	NUT WELDING M10 H8 (9004A-17208)	6 DUS	10.000 PCS	6 DUS x 1500 Pcs + 2 KTG x 500
2	RIVET VOLVO D17 X 26.5 MM	50 BAG	20.000 PCS	50 BAG x 400 Pcs
3				
4				
5				
6				
7				
8				
9				
10				
11				

NB : UNTUK TUKAR GULING
 SURAT JALAN TIDAK DITAGIHKAN

6 DUS + 2 KTG + 50 BAG

RECEIVED
 10 OCT 2018
 WWS - GND

DIPERIKSA
 GDG SUP SEC

HORMAT KAMI
 BAG ORDER

Lampiran II Formulir Retur
 (Sumber : PT Gemala Kempa Daya, 2018)

PT. Garuda Kempa Daya Quality Assurance Department				CHECK SHEET				QCO / PRO Lampiran yang ditandatangani		Revisi 01		LH Date 1-May-08	
Part Name Gusset Cross Member no. 1, LH RH				Date May 15, 2008				Production Qty		Sample		Page (cont. 2)	
Brand / Type Mitsubishi / H. TD				Lot / Process MIS55277778				Methods		Inspection		Remarks	
Supplier P.T. TEM				Part No.				Product		Inspection		Remarks	
				No.		Product		Inspection		Remarks			
				1		Appearance		Visual		✓			
				2		Width		V-Caliper		✓			
				3		Pitch Hole		Height Gauge		✓			
				4		Length		V-Caliper		✓			
				5		Pitch Hole		Height Gauge		✓			
				6		Pitch Hole		Height Gauge		✓			
				7		Distance		Height Gauge		✓			
				8		Distance		Height Gauge		✓			
				9		Hole dia (2 pcs)		V-Caliper		✓			
				10		Distance		Height Gauge		✓			
				11		Thread M12 x 1.25 (2 pcs)		Thread Plug		✓			
				12		Thread M12 x 1.25 (2 pcs)		Thread Plug		✓			
				13		Hole dia		V-Caliper		✓			
				14		Chamber		Height Gauge		✓			
15		Hole dia (1 Pcs RH Only)		V-Caliper		✓							
Judgement (✓ = OK, X = NG)													
REVISI				EFFECTIVE DATE				DESCRIPTION					
1				09-July-2009				Base on Capillary Process Reinforcement CPM no.1					
Inspector				Checked				Remarks					

Lampiran III Formulir Checksheet
Komponen Part Gusset Cross Member
(Sumber : PT Gemala Kempa Daya, 2018)

 GEMALA KEMPA DAYA	LAPORAN SORTIR NO :/QA/GKD/...../.....		Doc. no	: FM/QAS/01-05
			Revision	: 02
			Eff. date	: 06-10-2017

NAMA SUPPLIER	:	
PART NAME	:	
MODEL	:	
PROBLEM	:	
TANGGAL SORTIR	:	
WAKTU SORTIR	:	
JUMLAH SORTIR	:	

Area Sortir	OK	NG	Total
Finish Good	pcs	pcs	pcs
In Proses/ Line	pcs	pcs	pcs
Ware House	pcs	pcs	pcs
Sub Total	pcs	pcs	pcs

Consumeable	:	1	Note:
	:	2	
	:	3	
	:	4	

Mengetahui		Supplier
PT. GKD		

Lampiran IV Laporan Sortir
(Sumber : PT Gemala Kempa Daya, 2018)

 PT. GEMALA KEMPA DAYA		LAPORAN PENERIMAAN BARANG / JASA		No. : 33778/000581/BA/10/11 Tgl. : 04-10-2018
DITERIMA DARI / SUPPLIER : DUTA HITA JAYA PT. No. SJ : 05081			PEMAKAI : PPIC GKD KPP/PP No. : JE507GPPC918 PO No. : J4997/PO-GKD/0918 LOKASI : OUT SOURCING	
NO.	BARANG / JASA	KODE BARANG	JUMLAH SATUAN	KETERANGAN PEMERIKSAAN
1	JASA PRESS BUSHING BRKT SPRING FRONT 31 U P/N: ISF- P652192090	806390010	13 240 PCS	240 / 2000
PENERIMA/PEMERIKSA : PENERIMA PEMERIKSA		MENGETAHUI :		CATATAN : Gudang : Plant gkd jakarta Tgl Delivery : 04-10-2018 Jam Cetak : 14:40:06 FM/PPC/020 Revisi : 01
LEMBAR 1 ACCOUNTING		LEMBAR 2 PPC		LEMBAR 3 GUDANG

Lampiran V Formulir Laporan
Penerimaan Barang
(Sumber : PT Gemala Kempa Daya, 2018)

TRANSKRIP WAWANCARA

Nama Narasumber : Bapak Yandi Fitra (Kepala Departemen IT)

Tanggal : 12 Juli 2018

Jam : 13.00 – 14.00 WIB

Tempat Wawancara : Ruangan *Office*

Topik Wawancara : Proses pemeriksaan bahan baku

Pewawancara : Proses pemeriksaan bahan baku yang datang dari *Supplier* pada PT Gemala Kempa Daya ini menggunakan sistem yang seperti apa pak ?

Narasumber : Proses pemeriksaannya di PT Gemala Kempa Daya masih belum dilakukan secara baik dan optimal.

Pewawancara : Untuk seluruh pemeriksaannya, Pak?

Narasumber : Iya benar, seluruh pemeriksaannya.

Pewawancara : Kenapa begitu ya, Pak?

Narasumber : Karena kami disini masih belum memiliki sistem yang terintegrasi untuk melakukan pemeriksaan secara menyeluruh, karena setiap divisi memiliki aspek pemeriksaan yang berbeda-beda.

Pewawancara : Bagaimana untuk pemeriksaan nya ketika barangnya sudah sampai Pak?

Narasumber : Sistem pemeriksaan yang ada masih belum jelas untuk pemeriksaan jumlahnya maupun kualitasnya. Kami sedang mengusahakannya.

Pewawancara : Oh seperti itu ya pak. Memang bagaimana proses pemeriksaan barangnya, Pak?

Narasumber : proses pemeriksaanya diawali dengan pemeriksaan jumlah, nama, dan surat jalannya, setelah sudah sesuai maka dilanjutkan dengan

pemeriksaan kualitasnya berdasarkan *Checksheets* dari *Supplier* kemudian setelah melalui pemeriksaan kualitasnya dipisahkan antara barang *OK* dan *NG*.

Pewawancara : Setelah barang dinyatakan kualitasnya sudah *OK* kemudian langsung disimpan digudang atau langsung diolah , Pak?

Narasumber : Setelah barang sudah dinyatakan kualitas nya *OK* tidak langsung diolah melainkan disimpan terlebih dahulu digudang dan dibuatkan laporan penerimaan barang.

Pewawancara : Kemudian bagaimana dengan barang yang dinyatakan *NG* Pak?

Narasumber : Untuk barang yang dinyatakan *NG* tetap sama akan disimpan digudang terlebih dahulu untuk dikembalikan kepada *Supplier*.

Pewawancara : Kira – kira apakah ada permasalahan yang ditimbulkan dari pemeriksaan barang ini, Pak?

Narasumber : Ada, biasanya ada dokumen atau data dari pemeriksaan itu rusak bahkan hilang.

Pewawancara : Kenapa bisa seperti itu, Pak?

Narasumber : Ya Karena yang pertama belum ada nya sistem pemeriksaan yang terintegrasi satu sama lain, yang kedua belum ada nya sistem keamanan untuk mengolah data nya di *Ms.Excel*, yang ketiga masih menggunakan media kertas untuk pengarsipannya kemudian pelaporannya harus mencari-cari dokumen dari pemeriksaannya. apalagi untuk pelaporan returnnya,disini untuk melakukan retur harus mencari dokumen surat jalannya.

Pewawancara : Baik pak kalau begitu, terima kasih atas informasi dan waktu luangnya pak.

Narasumber : Iya, sama – sama.

BLACK-BOX TESTING

1. *Form Login*

Deskripsi : Menguji fungsi login pada *Form Login*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol <i>Login</i> tanpa memasukkan <i>username</i> dan <i>password</i>	Sistem akan menampilkan pesan kolom <i>username</i> dan <i>password</i> harus diisi	Sistem menampilkan pesan dimana <i>user</i> harus mengisi kolom <i>username</i> dan <i>password</i>	Valid
2	Memasukkan <i>username</i> dengan benar sedangkan <i>password</i> salah atau sebaliknya, lalu mengklik tombol <i>Login</i>	Sistem akan menampilkan pesan kombinasi <i>username</i> dan <i>password</i> salah	Sistem menampilkan pesan kombinasi <i>username</i> dan <i>password</i> salah	Valid
3	Memasukkan <i>username</i> dan <i>password</i> dengan benar lalu mengklik tombol <i>Login</i>	Sistem akan menerima akses <i>login</i> dan menampilkan halaman utama	Sistem menerima akses <i>login</i> dan menampilkan halaman utama	Valid

2. Menu Data *Master Karyawan*

Deskripsi : Menguji fungsi tambah data pada Menu Data *Master Karyawan*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Tambah Data Karyawan	Sistem akan menampilkan <i>form</i> Tambah Data Karyawan	Sistem menampilkan <i>form</i> Tambah Data Karyawan	Valid

Deskripsi : Menguji fungsi ubah pada Menu Data *Master Karyawan*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Ubah	Sistem akan menampilkan <i>form</i> Ubah Data Karyawan	Sistem menampilkan <i>form</i> Ubah Data Karyawan	Valid

Deskripsi : Menguji fungsi detail pada Menu Data *Master Karyawan*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Detail	Sistem akan menampilkan <i>form</i> Detail Data Karyawan	Sistem menampilkan <i>form</i> Detail Data Karyawan	Valid

Deskripsi : Menguji fungsi hapus pada Menu Data *Master* Karyawan

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Hapus pada data yang dipilih	Sistem akan menghapus data yang dipilih	Sistem menghapus data yang dipilih	Valid

3. *Form* Tambah Data *Master* Karyawan

Deskripsi : Menguji fungsi simpan pada *Form* Tambah Data Karyawan

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Simpan dengan mengosongkan semua kolom	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
2	Mengklik tombol Simpan dengan kondisi salah satu kolom tidak diisi	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
3	Mengisi seluruh kolom isian lalu mengklik tombol Simpan	Data dapat disimpan	Data dapat disimpan	Valid

Deskripsi : Menguji fungsi kembali pada *Form* Tambah Data Karyawan

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Kembali	Sistem akan kembali ke <i>list</i> data karyawan	Sistem akan kembali ke <i>list</i> data karyawan	Valid

4. *Form* Ubah Data *Master* Karyawan

Deskripsi : Menguji fungsi ubah pada *Form* Ubah Data *Master* Karyawan

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
-----	-----------	-----------------	---------------	--------

1	Mengklik tombol Ubah dengan kondisi semua tidak ada yg diubah	Data tidak ada yang berubah	Data tidak ada yang berubah	Valid
2	Mengklik tombol Ubah dengan kondisi salah satu kolom diganti	Data dapat diubah	Data dapat diubah	Valid
3	Mengganti seluruh kolom isian lalu mengklik tombol Ubah	Data dapat diubah	Data dapat diubah	Valid

Deskripsi : Menguji fungsi kembali pada *Form Ubah Data Karyawan*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Kembali	Sistem akan kembali ke <i>list</i> data karyawan	Sistem akan kembali ke <i>list</i> data karyawan	Valid

5. Menu Data *Master Supplier*

Deskripsi : Menguji fungsi tambah data pada Menu Data *Master Supplier*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Tambah Data Kriteria	Sistem akan menampilkan <i>form</i> Tambah Data Kriteria	Sistem menampilkan <i>form</i> Tambah Data Kriteria	Valid

Deskripsi : Menguji fungsi ubah pada Menu Data *Master Supplier*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Ubah	Sistem akan menampilkan <i>form</i> Ubah Data Kriteria	Sistem menampilkan <i>form</i> Ubah Data Kriteria	Valid

Deskripsi : Menguji fungsi detail pada Menu Data *Master Supplier*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Detail	Sistem akan menampilkan <i>form</i> menu list data subkriteria	Sistem akan menampilkan <i>form</i> menu list data subkriteria	Valid

Deskripsi : Menguji fungsi hapus pada Menu Data *Master Supplier*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Hapus pada data yang dipilih	Sistem akan menghapus data yang dipilih	Sistem menghapus data yang dipilih	Valid

6. *Form Tambah Data Master Supplier*

Deskripsi : Menguji fungsi simpan pada *Form Tambah Data Master Supplier*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Simpan dengan mengosongkan semua kolom	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
2	Mengklik tombol Simpan dengan kondisi salah satu kolom tidak diisi	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
3	Mengisi seluruh kolom isian lalu mengklik tombol Simpan	Data dapat disimpan	Data dapat disimpan	Valid

Deskripsi : Menguji fungsi Kembali pada *Form Tambah Data Master Supplier*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Kembali	Sistem akan kembali ke <i>list</i> data kriteria	Sistem akan kembali ke <i>list</i> data kriteria	Valid

7. *Form Ubah Data Master Barang*

Deskripsi : Menguji fungsi ubah pada *Form Ubah Data Barang*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Ubah dengan kondisi semua kolom tidak diisi	Data tidak ada yang berubah	Data tidak ada yang berubah	Valid
2	Mengklik tombol Ubah dengan kondisi salah satu kolom diganti	Data dapat diubah	Data dapat diubah	Valid
3	Mengganti seluruh kolom isian lalu mengklik tombol Ubah	Data dapat diubah	Data dapat diubah	Valid

Deskripsi : Menguji fungsi Kembali pada *Form Ubah Data Master Supplier*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Kembali	Sistem akan kembali ke <i>list data Supplier</i>	Sistem akan kembali ke <i>list data Supplier</i>	Valid

8. *Menu Data Surat Jalan*

Deskripsi : Menguji fungsi tambah data pada *Menu Data Surat Jalan*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Tambah Data Surat Jalan	Sistem akan menampilkan <i>form</i> Tambah Data Surat Jalan	Sistem menampilkan <i>form</i> Tambah Data Surat Jalan	Valid

9. *Form Tambah Data Surat Jalan*

Deskripsi : Menguji fungsi simpan pada *Form Tambah Data Surat Jalan*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Simpan dengan mengosongkan semua kolom	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid

2	Mengklik tombol Simpan dengan kondisi salah satu kolom tidak diisi	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
3	Mengisi seluruh kolom isian lalu mengklik tombol Simpan	Data dapat disimpan	Data dapat disimpan	Valid

Deskripsi : Menguji fungsi kembali pada *Form* Tambah Data Surat Jalan

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Kembali	Sistem akan kembali ke <i>list</i> data Surat Jalan	Sistem akan kembali ke <i>list</i> data Surat Jalan	Valid

10. Menu Data Hasil Surat Jalan

Deskripsi : Menguji fungsi hapus pada Menu Data Hasil Surat Jalan

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Hapus di atas tabel	Sistem akan menghapus data Surat Jalan	Sistem menghapus data Surat Jalan	Valid

Deskripsi : Menguji fungsi normalisasi pada Menu Data Surat Jalan

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol Normalisasi pada data yang dipilih	Sistem akan menghitung normalisasi dari data Surat Jalan dan kemudian menampilkannya	Sistem menghitung normalisasi dan menampilkannya	Valid

11. Menu Data *Checksheets*

Deskripsi : Menguji fungsi *Checksheets* pada Menu Data *Checksheets*

Penguji : Bagus Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol <i>Checksheets</i> di atas tabel	Sistem akan me <i>Checksheets</i> hasil normalisasi	Sistem me <i>Checksheets</i> hasil normalisasi	Valid

		dan menampilkannya pada tabel	dan menampilkannya pada tabel	
--	--	-------------------------------------	-------------------------------------	--

Deskripsi : Menguji fungsi hapus pada Menu Data *Checksheet*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol hapus	Sistem akan menghapus data <i>Checksheet</i> dari tabel	Sistem menghapus data <i>Checksheet</i> dari tabel	Valid

Deskripsi : Menguji fungsi cetak pada Menu Data *Checksheet*

Penguji : Bagas Eko Saputro (1315051)

No.	Test Case	Expected Result	Actual Record	Result
1	Mengklik tombol cetak	Sistem akan menampilkan format laporan <i>Checksheet</i> Surat Jalan kinerja karyawan dan kemudian mencetaknya	Sistem menampilkan format laporan <i>Checksheet</i> Surat Jalan kinerja karyawan dan kemudian mencetaknya	Valid

KODE PROGRAM

1.Login

a. Auth.php (Control)

<?php

```
class Auth extends CI_Controller
```

```
{
```

```
    public function __construct()
```

```
    {
```

```
        parent::__construct();
```

```
        $this->load->library('form_validation');
```

```
        $this->load->model('M_login');
```

```
    }
```

```
    public function index()
```

```
    {
```

```
        // jika button di klik (yg ada di view)
```

```
        if(isset($_POST['submit'])){
```

```
            //proses login disini
```

```
            $username = $this->input->post('username');
```

```
            $password = $this->input->post('password');
```

```
            $hasil = $this->M_login->login($username,$password);
```

```
            $jabatan=$this->db->
```

```
            >get_where('tb_login',array('username'=>$username))->row_array();
```

```
            if($hasil==1)
```

```

        {
            $this->session->set_userdata($jabatan);

            // $this->session-
            >set_userdata(array('status_login'=>'oke','username'=>$username,'jabatan'=>$jabatan));

            redirect ('Dashboard');
        }
        else{
            $this->session->set_flashdata('message', '<div class="alert alert-
danger" role="alert">

            Username atau password yang ada masukan salah <button
type="button" class="close" data-dismiss="alert" aria-label="Close">

            <span aria-hidden="true">&times;</span>

            </button>

            </div>');

            // die;

            redirect('Auth');
        }
    }else{
        $this->load->view('form_login');

    }
}

public function logout()
{

    redirect('Auth');

}
}

```

b. M_login (Model)

```
<?php
```

```
class M_login extends CI_model {
```

```
    public function tampilkan_data()
```

```
    {
```

```
        return $this->db->get('tb_login');
```

```
    }
```

```
    public function simpan($data)
```

```
    {
```

```
        $this->db->insert('tb_login', $data);
```

```
    }
```

```
    public function hapus($id)
```

```
    {
```

```
        $this->db->delete('tb_login', ['username' => $id]);
```

```
    }
```

```
    public function get_data($id)
```

```
    {
```

```
        $vans = array('username' => $id);
```

```
        return $this->db->get_where('tb_login', $vans);
```

```
    }
```

```
    public function detail($id)
```

```
    {
```

```

        return $this->db->get_where('tb_login', ['username' => $id])->row_array();
    }

    function update($data, $id)
    {
        $this->db->where('username', $id);
        $this->db->update('tb_login', $data);
    }

    function login($username,$password)
    {
        $chek=$this->db-
>get_where('tb_login',array('username'=>$username,'password'=>$password));

        // untuk check data username dan password ada atau tidak
        if ($chek->num_rows()>0)
        {
            return 1;
        }
        else {
            return 0;
        }
    }
}

```

c. formlogin (View)

```

<html>
<head>
    <title>Login</title>
</head>

```

```

<body>
<h1>Silahkan Login</h1>
<form action="<?php echo base_url('login/cek_login'); ?>" method="post">
  <table>
    <tr>
      <td>Username</td>
      <td><input type="text" name="username"></td>
    </tr>
    <tr>
      <td>Password</td>
      <td><input type="password" name="password"></td>
    </tr>
    <tr>
      <td></td>
      <td><input type="submit" value="Login"></td>
    </tr>
  </table>
</form>
</body>
</html>

```

2.Checksheet

a. Checksheet.php (Controller)

```
<?php
```

```

class Checksheet extends CI_Controller
{
    public function __construct()

```

```

{
    parent::__construct();
    $this->load->model('M_cs');
    $this->load->library('form_validation');
}

public function index()
{
    $data['record'] = $this->M_cs->tampilkan_data();
    $data['kode_bb'] = $this->M_cs->get();
    $this->template->load('template', 'checksheet/lihat_data' , $data);
}

public function tambah()
{
    $data['record'] = $this->M_cs->tampilkan_data();
    $data['kode_bb'] = $this->M_cs->get();
    $data['no_sj'] = $this->M_cs->got();
    $this->template->load('template', 'checksheet/form_input', $data);
}

public function detail($id)
{
    $data['checksheet'] = $this->M_cs->tampilkan_data();
    $data['judul'] = 'Detail Data Checksheet';
    $data['checksheet'] = $this->M_cs->detail($id);
}

```

```

        $this->template->load('template', 'checksheet/detail_data', $data);
    }

```

```

public function simpan()

```

```

{
    $id                = $this->input->post('id_cs');
    $kode_bb           = $this->input->post('kode_bb');
    $no_sj             = $this->input->post('no_sj');
    $status            = $this->input->post('status');
    $problem           = $this->input->post('problem');
    $apperance         = $this->input->post('apperance');
    $width             = $this->input->post('width');
    $pitch_hole        = $this->input->post('pitch_hole');
    $length            = $this->input->post('length');
    $distance          = $this->input->post('distance');
    $hole_dia          = $this->input->post('hole_dia');
    $thread_m12        = $this->input->post('thread_m12');
    $chamfer           = $this->input->post('chamfer');
    $tgl_pemeriksaan   = $this->input->post('tgl_pemeriksaan');
    $ok                = $this->input->post('ok');
    $ng                = $this->input->post('ng');
    $jumlah            = $this->input->post('jumlah');

```

```

        $data = array('id_cs' => $id, 'kode_bb' => $kode_bb, 'no_sj' => $no_sj, 'status' =>
$status, 'problem' => $problem, 'apperance' => $apperance, 'width' => $width, 'pitch_hole' =>
$pitch_hole, 'length' => $length, 'distance' => $distance, 'hole_dia' => $hole_dia, 'thread_m12'
=> $thread_m12, 'chamfer' => $chamfer, 'tgl_pemeriksaan' => $tgl_pemeriksaan, 'ok' => $ok,
'ng' => $ng, 'jumlah' => $jumlah);

```

```

    $simpan = $this->M_cs->simpan($data);

```

```

        $this->session->set_flashdata('message', '<div class="alert alert-success"
role="alert">

                Data Berhasil Di tambahkan <button type="button" class="close"
data-dismiss="alert" aria-label="Close">

                        <span aria-hidden="true">&times;</span>

                </button>

                </div>');

        redirect('checksheet');

    }

    public function hapus($id)
    {
        $id = $this->uri->segment(3);
        $this->M_cs->hapus($id);
        $this->session->set_flashdata('message', '<div class="alert alert-danger"
role="alert">

                Data Berhasil Di Hapus <button type="button" class="close" data-
dismiss="alert" aria-label="Close">

                        <span aria-hidden="true">&times;</span>

                </button>

                </div>');

        redirect('checksheet');

    }

    public function ubah()
    {
        $id = $this->uri->segment(3);
        $data['record'] = $this->M_cs->get_data($id)->row_array();
        $this->template->load('template', 'checksheet/form_edit', $data);

    }

```

```

public function validasi()
{
    $id = $this->uri->segment(3);
    $data['record'] = $this->M_cs->get_data($id->row_array();
    $this->template->load('template', 'checksheet/form_validasi', $data);
}

```

```

public function edit()

```

```

{
    $id = $this->input->post('id_cs');
    $kode_bb = $this->input->post('kode_bb');
    $no_sj = $this->input->post('no_sj');
    $status = $this->input->post('status');
    $problem = $this->input->post('problem');
    $apperance = $this->input->post('apperance');
    $width = $this->input->post('width');
    $pitch_hole = $this->input->post('pitch_hole');
    $length = $this->input->post('length');
    $distance = $this->input->post('distance');
    $hole_dia = $this->input->post('hole_dia');
    $thread_m12 = $this->input->post('thread_m12');
    $chamfer = $this->input->post('chamfer');
    $tgl_pemeriksaan = $this->input->post('tgl_pemeriksaan');
    $ok = $this->input->post('ok');
    $ng = $this->input->post('ng');
    $jumlah = $this->input->post('jumlah');
}

```

```

        $data = array('id_cs' => $id, 'kode_bb' => $kode_bb, 'no_sj' => $no_sj, 'status' =>
$status, 'problem' => $problem, 'apperance' => $apperance, 'width' => $width, 'pitch_hole' =>
$pitch_hole, 'length' => $length, 'distance' => $distance, 'hole_dia' => $hole_dia, 'thread_m12'
=> $thread_m12, 'chamfer' => $chamfer, 'tgl_pemeriksaan' => $tgl_pemeriksaan, 'ok' => $ok,
'ng' => $ng, 'jumlah' => $jumlah);

```

```

        $simpan = $this->M_cs->update($data, $id);

```

```

        $this->session->set_flashdata('message', '<div class="alert alert-success"
role="alert">

```

```

        Data Berhasil Di Ubah <button type="button" class="close" data-
dismiss="alert" aria-label="Close">

```

```

        <span aria-hidden="true">&times;</span>

```

```

        </button>

```

```

        </div>');

```

```

        redirect('checksheet');

```

```

    }

```

```

}

```

b. M_cs.php (Model)

```

<?php

```

```

class M_cs extends CI_model {

```

```

    public function tampilkan_data()

```

```

    {

```

```

        $this->db->select('*');

```

```

        $this->db->from('tb_cs_bb');

```

```

        $this->db->join('tb_bb', 'tb_bb.kode_bb = tb_cs_bb.kode_bb');

```

```

        return $this->db->get();

```

```

    }

```

```

public function get()
{
    $query = $this->db->query('SELECT * FROM tb_bb');
    return $query->result();
}

public function got()
{
    $query = $this->db->query('SELECT * FROM tb_sj');
    return $query->result();
}

public function simpan($data)
{
    $this->db->insert('tb_cs_bb', $data);
}

public function hapus($id)
{
    $this->db->delete('tb_cs_bb', ['id_cs' => $id]);
}

public function get_data($id)
{
    $vans = array('id_cs' => $id);
    return $this->db->get_where('tb_cs_bb', $vans);
    $query = $this->db->query('SELECT * FROM tb_bb');
return $query->result();
}

public function detail($id)

```

```

    {
        return $this->db->get_where('tb_cs_bb', ['id_cs' => $id])->row_array();
    }
    function update($data, $id)
    {
        $this->db->where('id_cs', $id);
        $this->db->update('tb_cs_bb', $data);
    }
}

```

c. Lihat_data (view)

```

<!-- DataTales Example -->
<h1>Data Checksheet</h1>
<div class="card shadow mb-4">
<div class="card-header py-3">
    <a href="<?php echo base_url(); ?>Checksheet/tambah" class="btn btn-primary">Tambah Data
    Checksheet</a>
</div>
<div class="card-body">
    <div class="table-responsive">
        <?php echo $this->session->flashdata('message');?>
        <table class="table table-bordered" id="dataTable" width="100%" cellspacing="0">
        <thead>
            <tr>
                <th>ID Checksheet</th>
                <th>Tanggal Pemeriksaan</th>
                <th>Kode Bahan Baku</th>
                <th>No Surat Jalan</th>
                <th>Status Validasi</th>
                <th>Action</th>

```

```

        </tr>
    </thead>
    <tbody>
        <?php
            $no = 1;
            foreach ($record->result() as $cs) {
                echo "<tr>
                    <td>$cs->id_cs</td>
                    <td>$cs->tgl_pemeriksaan</td>
                    <td>$cs->kode_bb</td>
                    <td>$cs->no_sj</td>
                    <td>$cs->status</td>
                    <td>
                        ".anchor('Checksheetsheet/ubah/'. $cs->id_cs,'Edit',array('class' => 'fas fa-user-edit text-info'))."
                        ".anchor('Checksheetsheet/validasi/'. $cs->id_cs,'Validasi',array('class' => 'fas fa-check'))."
                        ".anchor('Checksheetsheet/detail/'. $cs->id_cs,'Detail',array('class' => 'btn btn fas fa-search'))."
                        ".anchor('Checksheetsheet/hapus/'. $cs->id_cs,'Hapus',array('class' => 'text-danger fas fa-trash'))."
                    </td>
                </tr>";
                $no++;
            }
        ?>
    </tbody>
</table>
</div>
<!-- Modal Input -->
</div>
</div>

```

```
</div>
```

```
</div>
```

d. form_input.php (view)

```
<!-- DataTales Example -->
```

```
<h1>Tambah Data Checksheet</h1>
```

```
<div class="row">
```

```
<div class="col-lg-6">
```

```
<div class="card shadow mb-4">
```

```
<div class="card-body">
```

```
<?=
```

```
form_open('checksheet/simpan');
```

```
?>
```

```
<div class="box-body">
```

```
<div class="form-group">
```

```
<div class="form-group">
```

```
<label>ID Checksheet</label>
```

```
<input type="text" name="id_cs" class="form-control" id="" placeholder="ID  
Checksheet" required="true">
```

```
</div>
```

```
<label>Tanggal Pemeriksaan</label>
```

```
<input type="date" name="tgl_pemeriksaan" class="form-control" id=""  
placeholder="Tanggal Pemeriksaan" required="true">
```

```
</div>
```

```
<div class="form-group">
```

```
<label>Kode Bahan Baku</label>
```

```
<div>
```

```
<select class="form-control" required name="kode_bb">
```

```
<option value="" disabled="nama" selected>Nama Bahan Baku</option>
```

```
<?php
```

```

        foreach ($kode_bb as $key) {
            echo "<option value='$key->kode_bb'$key->nama_bb</option>";
        }
    ?>
</select>
</div>

<div class="form-group">
    <label>No Surat Jalan</label>
    <div>
        <select class="form-control" required name="no_sj">
            <option value="" disabled="nama" selected>Kode Supplier</option>
            <?php
                foreach ($no_sj as $key) {
                    echo "<option value='$key->no_sj'$key->kode_supplier</option>";
                }
            ?>
        </select>
    </div>

    <div class="form-group">
        <input type="hidden" name="status" class="form-control" id="" placeholder="Status"
value="Belum Divalidasi">
    </div>

    <div class="form-group">
        <label>Apperance</label>
        <input type="text" name="apperance" class="form-control" id=""
placeholder="Apperance" required="true"></div>

    <div class="form-group">
        <label>Width</label>

```

```

        <input type="text" name="width" class="form-control" id="" placeholder="Width"
required="true"></div>

        <div class="form-group">

            <label>Pitch Hole</label>

            <input type="text" name="pitch_hole" class="form-control" id="" placeholder="Pitch
Hole" required="true"></div>

            <div class="form-group">

                <label>Length</label>

                <input type="text" name="length" class="form-control" id="" placeholder="Length"
required="true"></div>

                <div class="form-group">

                    <label>Distance</label>

                    <input type="text" name="distance" class="form-control" id="" placeholder="Distance"
required="true"></div>

                    <div class="form-group">

                        <label>Hole Dia</label>

                        <input type="text" name="hole_dia" class="form-control" id="" placeholder="Hole
Dia" required="true"></div>

                        <div class="form-group">

                            <label>Thread M12 x 1.25</label>

                            <input type="text" name="thread_m12" class="form-control" id=""
placeholder="Thread M12 x 1.25" required="true"></div>

                            <div class="form-group">

                                <label>Chamfer</label>

                                <input type="text" name="chamfer" class="form-control" id="" placeholder="Chamfer"
required="true">

                                </div>

                            <div class="form-group">

                                <label>OK</label>

                                <input type="text" name="ok" class="form-control" id="" placeholder="OK"
required="true"></div>

```

```

        <div class="form-group">
            <label>NG</label>
            <input type="text" name="ng" class="form-control" id="" placeholder="NG"
required="true"></div>
        <div class="form-group">
            <label>Jumlah</label>
            <input type="text" name="jumlah" class="form-control" id="" placeholder="Jumlah"
required="true"></div>
    </div>
    <div class="form-group">
        <label>Problem</label>
        <input type="text" name="problem" class="form-control" id="" placeholder="Problem"
required="true">
    </div>
</div><!-- /.box-body -->
<div class="box-footer">
    <button type="submit" class="btn btn-primary">Submit</button>
    <a href="<?php echo base_url(); ?>checksheet" class="btn btn-danger">Kembali</a>
</div>
</form>
</div>
</div>
</div>
</div>

```

e. form_edit.php (view)

```

<!-- DataTales Example -->
<h1>Ubah Data Checksheet</h1>
<div class="row">

```

```

<div class="col-lg-6">

<div class="card shadow mb-4">

<div class="card-body">

    <?=
        form_open('checksheet/edit');
    ?>

    <div class="box-body">

        <table class="table table-bordered" id="dataTable" width="100%" cellspacing="0">

            <div class="form-group">

                <label>Tanggal Pemeriksaan</label>

                <input type="text" name="tgl_pemeriksaan" class="form-control" id=""
placeholder="Tanggal Pemeriksaan" value="<?= $record['tgl_pemeriksaan']; ?>"readonly>

                <div class="form-group">

                    <label>ID Checksheet</label>

                    <input type="text" name="id_cs" class="form-control" id="" placeholder="ID
Checksheet" value="<?= $record['id_cs']; ?>"readonly>

                    <div class="form-group">

                        <label>Kode Bahan Baku</label>

                        <input type="text" name="kode_bb" class="form-control" id="" placeholder="Kode
Bahan Baku" value="<?= $record['kode_bb']; ?>"readonly >

                    </div>

                    <div class="form-group">

                        <label>No Surat Jalan</label>

                        <input type="text" name="no_sj" class="form-control" id="" placeholder="No Surat
Jalan" value="<?= $record['no_sj']; ?>">

                    </div>

                    <div class="form-group">

                        <label></label>

                        <input type="text" hidden="" name="status" class="form-control" id=""
placeholder="Status" value="<?= $record['status']; ?>">

```

```

</div>

<div class="form-group">

  <label>Apperance</label>

  <input type="text" name="apperance" class="form-control" id=""
placeholder="Apperance" value="<?=$record['apperance']; ?>">

  <div class="form-group">

    <label>Width</label>

    <input type="text" name="width" class="form-control" id="" placeholder="Width"
value="<?=$record['width']; ?>">

    <div class="form-group">

      <label>Pitch Hole</label>

      <input type="text" name="pitch_hole" class="form-control" id="" placeholder="Pitch
Hole" value="<?=$record['pitch_hole']; ?>">

      <div class="form-group">

        <label>Length</label>

        <input type="text" name="length" class="form-control" id="" placeholder="Length"
value="<?=$record['length']; ?>">

        <div class="form-group">

          <label>Distance</label>

          <input type="text" name="distance" class="form-control" id="" placeholder="Distance"
value="<?=$record['distance']; ?>">

          <div class="form-group">

            <label>Hole Dia</label>

            <input type="text" name="hole_dia" class="form-control" id="" placeholder="Hole
Dia" value="<?=$record['hole_dia']; ?>">

            <div class="form-group">

              <label>Thread M12 x 1.25</label>

              <input type="text" name="thread_m12" class="form-control" id=""
placeholder="Thread M12 x 1.25" value="<?=$record['thread_m12']; ?>">

              <div class="form-group">

                <label>Chamfer</label>

```

```

        <input type="text" name="chamfer" class="form-control" id="" placeholder="Chamfer"
value="<?= $record['chamfer']; ?>">
    </div>

    <div class="form-group">

        <label>OK</label>

        <input type="text" name="ok" class="form-control" id="" placeholder="OK"
value="<?= $record['ok']; ?>">
    </div>

    <div class="form-group">

        <label>NG</label>

        <input type="text" name="ng" class="form-control" id="" placeholder="NG"
value="<?= $record['ng']; ?>">
    </div>

    <div class="form-group">

        <label>Jumlah</label>

        <input type="text" name="jumlah" class="form-control" id="" placeholder="Jumlah"
value="<?= $record['jumlah']; ?>">

        <label>Problem</label>

        <input type="text" name="problem" class="form-control" id="" placeholder="Problem"
value="<?= $record['problem']; ?>">
    </div>
</div><!-- /.box-body -->

<div class="box-footer">

    <button type="submit" class="btn btn-primary">Submit</button>

    <a href="<?php echo base_url(); ?>checksheet" class="btn btn-danger">Kembali</a>
</div>
</form>
</div>
</div>

```

</div>

</div>

f. detail_data.php (view)

<!-- DataTales Example -->

<h1>Ubah Data Checksheet</h1>

<div class="row">

<div class="col-lg-6">

<div class="card shadow mb-4">

<div class="card-body">

<?=

form_open('checksheet/edit');

?>

<div class="box-body">

<table class="table table-bordered" id="dataTable" width="100%" cellpadding="0">

<div class="form-group">

<label>Tanggal Pemeriksaan</label>

<input type="text" name="tgl_pemeriksaan" class="form-control" id=""
placeholder="Tanggal Pemeriksaan" value="<?= \$record['tgl_pemeriksaan']; ?>"readonly>

<div class="form-group">

<label>ID Checksheet</label>

<input type="text" name="id_cs" class="form-control" id="" placeholder="ID
Checksheet" value="<?= \$record['id_cs']; ?>"readonly>

<div class="form-group">

<label>Kode Bahan Baku</label>

<input type="text" name="kode_bb" class="form-control" id="" placeholder="Kode
Bahan Baku" value="<?= \$record['kode_bb']; ?>"readonly >

</div>

<div class="form-group">

```

    <label>No Surat Jalan</label>

    <input type="text" name="no_sj" class="form-control" id="" placeholder="No Surat
Jalan" value="<?=$record['no_sj']; ?>">
</div>

<div class="form-group">

    <label></label>

    <input type="text" hidden="" name="status" class="form-control" id=""
placeholder="Status" value="<?=$record['status']; ?>">
</div>

<div class="form-group">

    <label>Apperance</label>

    <input type="text" name="apperance" class="form-control" id=""
placeholder="Apperance" value="<?=$record['apperance']; ?>">

    <div class="form-group">

        <label>Width</label>

        <input type="text" name="width" class="form-control" id="" placeholder="Width"
value="<?=$record['width']; ?>">

        <div class="form-group">

            <label>Pitch Hole</label>

            <input type="text" name="pitch_hole" class="form-control" id="" placeholder="Pitch
Hole" value="<?=$record['pitch_hole']; ?>">

            <div class="form-group">

                <label>Length</label>

                <input type="text" name="length" class="form-control" id="" placeholder="Length"
value="<?=$record['length']; ?>">

                <div class="form-group">

                    <label>Distance</label>

                    <input type="text" name="distance" class="form-control" id="" placeholder="Distance"
value="<?=$record['distance']; ?>">

                    <div class="form-group">

                        <label>Hole Dia</label>

```

```

        <input type="text" name="hole_dia" class="form-control" id="" placeholder="Hole
Dia" value="<?=$record['hole_dia']; ?>">

        <div class="form-group">

            <label>Thread M12 x 1.25</label>

            <input type="text" name="thread_m12" class="form-control" id=""
placeholder="Thread M12 x 1.25" value="<?=$record['thread_m12']; ?>">

            <div class="form-group">

                <label>Chamfer</label>

                <input type="text" name="chamfer" class="form-control" id="" placeholder="Chamfer"
value="<?=$record['chamfer']; ?>">

            </div>

            <div class="form-group">

                <label>OK</label>

                <input type="text" name="ok" class="form-control" id="" placeholder="OK"
value="<?=$record['ok']; ?>">

            </div>

            <div class="form-group">

                <label>NG</label>

                <input type="text" name="ng" class="form-control" id="" placeholder="NG"
value="<?=$record['ng']; ?>">

            </div>

            <div class="form-group">

                <label>Jumlah</label>

                <input type="text" name="jumlah" class="form-control" id="" placeholder="Jumlah"
value="<?=$record['jumlah']; ?>">

                <label>Problem</label>

                <input type="text" name="problem" class="form-control" id="" placeholder="Problem"
value="<?=$record['problem']; ?>">

            </div>

        </div><!-- /.box-body -->

```

```

<div class="box-footer">
    <button type="submit" class="btn btn-primary">Submit</button>
    <a href="<?php echo base_url(); ?>checksheet" class='btn btn-danger'>Kembali</a>
</div>
</form>
</div>
</div>
</div>
</div>

```

3.Surat Jalan (Controller)

a. <?php

```

class Suratjalan extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load->model('M_suratjalan');
        $this->load->library('form_validation');
    }

    public function index()
    {
        $data['record'] = $this->M_suratjalan->tampilkan_data();
        $this->template->load('template', 'suratjalan/lihat_data' , $data);
    }
}

```

```

public function tambah()
{
    $data['record'] = $this->M_suratjalan->tampilkan_data();
    $data['kode_supplier'] = $this->M_suratjalan->get();
    $data['kode_bb'] = $this->M_suratjalan->got();
    $this->template->load('template', 'suratjalan/form_input', $data);
}

public function simpan()
{
    $no_sj                = $this->input->post('no_sj');
    $mobil                = $this->input->post('mobil');
    $sopir                = $this->input->post('sopir');
    $kode_bb              = $this->input->post('kode_bb');
    $jumlah               = $this->input->post('jumlah');
    $kode_supplier        = $this->input->post('kode_supplier');
    $status               = $this->input->post('status');
    $tgl_kedatangan       = $this->input->post('tgl_kedatangan');

    $data = array('no_sj' => $no_sj, 'mobil' => $mobil, 'sopir' => $sopir, 'kode_bb' =>
$kode_bb, 'jumlah' => $jumlah, 'kode_supplier' => $kode_supplier, 'status' => $status,
'tgl_kedatangan' => $tgl_kedatangan);

    $simpan = $this->M_suratjalan->simpan($data);

    $this->session->set_flashdata('message', '<div class="alert alert-success"
role="alert">

        Data Berhasil Di tambahkan <button type="button" class="close"
data-dismiss="alert" aria-label="Close">

            <span aria-hidden="true">&times;</span>

```

```

        </button>

        </div>');

    redirect('suratjalan');
}

public function hapus($id)
{
    $id = $this->uri->segment(3);
    $this->M_suratjalan->hapus($id);
    $this->session->set_flashdata('message', '<div class="alert alert-danger"
role="alert">
        Data Berhasil Di Hapus <button type="button" class="close" data-
dismiss="alert" aria-label="Close">
            <span aria-hidden="true">&times;</span>
        </button>
    </div>');
    redirect('suratjalan');
}

public function ubah()
{
    $id = $this->uri->segment(3);
    $data['record'] = $this->M_suratjalan->get_data($id)->row_array();
    $this->template->load('template', 'suratjalan/form_edit', $data);
}

public function validasi()
{
    $id = $this->uri->segment(3);
    $data['record'] = $this->M_suratjalan->get_data($id)->row_array();

```

```

        $this->template->load('template', 'suratjalan/form_validasi', $data);
    }

    public function edit()
    {
        $no_sj                = $this->input->post('no_sj');
        $mobil                = $this->input->post('mobil');
        $sopir                = $this->input->post('sopir');
        $kode_bb              = $this->input->post('kode_bb');
        $jumlah                = $this->input->post('jumlah');
        $kode_supplier         = $this->input->post('kode_supplier');
        $status                = $this->input->post('status');
        $tgl_kedatangan        = $this->input->post('tgl_kedatangan');

        $data = array('no_sj' => $no_sj, 'mobil' => $mobil, 'sopir' => $sopir, 'kode_bb' =>
        $kode_bb, 'jumlah' => $jumlah, 'kode_supplier' => $kode_supplier, 'status' => $status,
        'tgl_kedatangan' => $tgl_kedatangan);

        $simpan = $this->M_suratjalan->update($data, $no_sj);

        $this->session->set_flashdata('message', '<div class="alert alert-success"
        role="alert">

                Data Berhasil Di Ubah <button type="button" class="close" data-
        dismiss="alert" aria-label="Close">

                <span aria-hidden="true">&times;</span>

                </button>

                </div>');

        redirect('suratjalan');
    }
}

b.      M_suratjalan.php (model)
<?php

```

```

class M_suratjalan extends CI_model {

    public function tampilkan_data()
    {
        $this->db->select('*');
        $this->db->from('tb_sj');
        $this->db->join('tb_supplier', 'tb_supplier.kode_supplier = tb_sj.kode_supplier');
        return $this->db->get();
    }

    public function get()
    {
        $query = $this->db->query('SELECT * FROM tb_supplier');
        return $query->result();
    }

    public function got()
    {
        $query = $this->db->query('SELECT * FROM tb_bb');
        return $query->result();
    }

    public function simpan($data)
    {
        $this->db->insert('tb_sj', $data);
    }

    public function hapus($id)
    {

```

```

        $this->db->delete('tb_sj', ['no_sj' => $id]);
    }

    public function get_data($id)
    {
        $vans = array('no_sj' => $id);
        return $this->db->get_where('tb_sj', $vans);
    }

    function update($data, $id)
    {
        $this->db->where('no_sj', $id);
        $this->db->update('tb_sj', $data);
    }
}

c.      lihat_data (view)
<?php

class M_suratjalan extends CI_model {

    public function tampilkan_data()
    {
        $this->db->select('*');
        $this->db->from('tb_sj');
        $this->db->join('tb_supplier', 'tb_supplier.kode_supplier = tb_sj.kode_supplier');
        return $this->db->get();
    }
}

```

```

public function get()
{
    $query = $this->db->query('SELECT * FROM tb_supplier');
    return $query->result();
}

public function got()
{
    $query = $this->db->query('SELECT * FROM tb_bb');
    return $query->result();
}

public function simpan($data)
{
    $this->db->insert('tb_sj', $data);
}

public function hapus($id)
{
    $this->db->delete('tb_sj', ['no_sj' => $id]);
}

public function get_data($id)
{
    $vans = array('no_sj' => $id);
    return $this->db->get_where('tb_sj', $vans);
}

function update($data, $id)
{

```

```

        $this->db->where('no_sj', $id);
        $this->db->update('tb_sj', $data);
    }
}

d.    form_input.php (view)
<!-- DataTales Example -->
<h1>Tambah Data Surat Jalan</h1>
<div class="row">
    <div class="col-lg-6">
        <div class="card shadow mb-4">
            <div class="card-body">
                <?=
                    form_open('suratjalan/simpan');
                ?>
                <div class="box-body">
                    <div class="form-group">
                        <label>Nomor Surat Jalan</label>
                        <input type="text" name="no_sj" class="form-control" id="" placeholder="Nomor
Surat Jalan" required="true">
                    </div>
                    <div class="box-body">
                        <div class="form-group">
                            <label>Kode Supplier</label>
                            <div>
                                <select class="form-control" required name="kode_supplier">
                                    <option value="" disabled="nama" selected>Nama Supplier</option>
                                <?php
                                    foreach ($kode_supplier as $key) {
                                        echo "<option value='$key->kode_supplier'$key->nama_supplier</option>";

```

```

    }
    ?>
</select>
</div>
<div class="form-group">
    <label>Kode Bahan Baku</label>
    <div>
        <select class="form-control" required name="kode_bb">
            <option value="" disabled="nama" selected>Nama Bahan Baku</option>
            <?php
                foreach ($kode_bb as $key) {
                    echo "<option value='$key->kode_bb'$key->nama_bb</option>";
                }
            ?>
        </select>
    </div>
    <div class="form-group">
        <label>Tanggal Kedatangan</label>
        <input type="date" name="tgl_kedatangan" class="form-control" id=""
placeholder="Tanggal Kedatangan" required="true">
        <div class="form-group">
            <label>Jumlah</label>
            <input type="text" name="jumlah" class="form-control" id="" placeholder="Jumlah"
required="true">
            <div class="box-body">
                <div class="form-group">
                    <label>Mobil</label>
                    <textarea type="text" name="mobil" class="form-control" id="" placeholder="Mobil"
required="true"></textarea>

```

```

</div>

<div class="form-group">

    <label>Supir</label>

    <input type="text" name="sopir" class="form-control" id="" placeholder="Supir"
required="true">

</div>

<div class="form-group">

    <input type="hidden" name="status" class="form-control" id="" placeholder="Status"
value="Belum Divalidasi">

</div>

</div><!-- /.box-body -->

<div class="box-footer">

    <button type="submit" class="btn btn-primary">Submit</button>

    <a href="<?php echo base_url(); ?>suratjalan" class="btn btn-danger">Kembali</a>

</div>

</form>

</div>

</div>

</div>

</div>

```

e. form_edit.php (view)

```

<!-- DataTales Example -->

<h1>Ubah Data Surat Jalan</h1>

<div class="row">

    <div class="col-lg-6">

        <div class="card shadow mb-4">

            <div class="card-body">

                <?=

```

```

form_open('suratjalan/edit');
?>
<div class="box-body">
  <div class="form-group">
    <label>Nomor Surat Jalan</label>

    <input type="text" name="no_sj" class="form-control" id="" placeholder="Nomor
Surat jalan" value="<?=$record['no_sj']; ?>" readonly>

  </div>

  <div class="form-group">
    <label>Kode Supplier</label>

    <input type="text" name="kode_supplier" class="form-control" id=""
placeholder="Kode Supplier" value="<?=$record['kode_supplier']; ?>">

  </div>

  <div class="form-group">
    <label>Kode Bahan Baku</label>

    <input type="text" name="kode_bb" class="form-control" id="" placeholder="Kode
Bahan Baku" value="<?=$record['kode_bb']; ?>">

  </div>

  <div class="form-group">
    <label>Tanggal Kedatangan</label>

    <input type="date" name="tgl_kedatangan" class="form-control" id=""
placeholder="Tanggal Kedatangan" value="<?=$record['tgl_kedatangan']; ?>">

  </div>

  <div class="form-group">
    <label>Jumlah</label>

    <input type="text" name="jumlah" class="form-control" id="" placeholder="Jumlah"
value="<?=$record['jumlah']; ?>">

  </div>

  <div class="form-group">
    <label>Mobil</label>

```

```

        <input type="text" name="mobil" class="form-control" id="" placeholder="Mobil"
value="<?= $record['mobil']; ?>">
    </div>

    <div class="form-group">

        <label>Supir</label>

        <input type="text" name="sopir" class="form-control" id="" placeholder="Supir"
value="<?= $record['sopir']; ?>">
    </div>

    <div class="form-group">

        <label></label>

        <input type="text" hidden="" name="status" class="form-control" id=""
placeholder="Status" value="<?= $record['status']; ?>">
    </div>

</div><!-- /.box-body -->

<div class="box-footer">

    <button type="submit" class="btn btn-primary">Submit</button>

    <a href="<?php echo base_url(); ?>suratjalan" class="btn btn-danger">Kembali</a>

</div>

</form>

</div>

</div>

</div>
</div>

```

f. form_validasi (view)

```

<!-- DataTales Example -->
<h1>Data Surat Jalan</h1>
<div class="row">
    <div class="col-lg-6">

```

```

<div class="card shadow mb-4">
  <div class="card-body">
    <?=
      form_open('suratjalan/edit');
    ?>
    <div class="box-body">
      <table class="table table-bordered" id="dataTable" width="100%" cellspacing="0">
        <div class="form-group">
          <label>Tanggal Kedatangan</label>
          <input type="text" name="tgl_kedatangan" class="form-control" id=""
placeholder="Tanggal Kedatangan" value="<?= $record['tgl_kedatangan']; ?>"readonly>
          <div class="form-group">
            <label>No Surat Jalan</label>
            <input type="text" name="no_sj" class="form-control" id="" placeholder="Nomor
Surat Jalan" value="<?= $record['no_sj']; ?>"readonly>
            <div class="form-group">
              <div class="form-group">
                <label>Kode Supplier</label>
                <input type="text" name="kode_supplier" class="form-control" id=""
placeholder="Kode Supplier" value="<?= $record['kode_supplier']; ?>"readonly >
              </div>
              <div class="form-group">
                <label>Kode Bahan Baku</label>
                <input type="text" name="kode_bb" class="form-control" id="" placeholder="Kode
Bahan Baku" value="<?= $record['kode_bb']; ?>"readonly >
              </div>
            <div class="form-group">
              <label>Tanggal Kedatangan</label>
              <input type="date" name="tgl_kedatangan" class="form-control" id=""
placeholder="Tanggal Kedatangan" value="<?= $record['tgl_kedatangan']; ?>"readonly >
            </div>
          </div>
        </div>
      </table>
    </div>
  </div>

```

```

</div>

</div>

<div class="form-group">

    <label>Jumlah</label>

    <input type="text" name="jumlah" class="form-control" id="" placeholder="Jumlah"
value="<?= $record['jumlah']; ?>"readonly >

</div>

<div class="form-group">

    <label>Mobil</label>

    <input type="text" name="mobil" class="form-control" id="" placeholder="Mobil"
value="<?= $record['mobil']; ?>"readonly >

    <div class="form-group">

        <label>Sopir</label>

        <input type="text" name="sopir" class="form-control" id="" placeholder="Sopir"
value="<?= $record['sopir']; ?>"readonly >

    </div>

    <div class="form-group">

        <input type="hidden" name="status" class="form-control" id="" placeholder="Status"
value="Sudah Divalidasi">

    </div><!-- /.box-body -->

    <div class="box-footer">

        <button type="submit" class="btn btn-primary">Validasi</button>

        <a href="<?php echo base_url(); ?>suratjalan" class='btn btn-danger'>Kembali</a>

    </div>

</form>

</div>

</div>

</div>

```