

No. Doc: 6271

Copy: 1

D 698.560 285  
AUD  
S

**SISTEM INFORMASI MONITORING HASIL PRODUKSI  
CLUTCH HOUSING PADA PT MITSUBISHI KRAMA  
YUDHA MOTORS AND MANUFACTURING**

**TUGAS AKHIR**

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

**OLEH  
SIFA AUDITA  
1316022**

<b>DATA BUKU PERPUSTAKAAN</b>	
<b>Tgl Terima</b>	13 15 / 2020
<b>No Induk Buku</b>	263/5110/STMI/JAKA



**POLITEKNIK STMI JAKARTA  
KEMENTERIAN PERINDUSTRIAN RI  
JAKARTA  
2020**

**MILIK PERPUSTAKAAN STMI**  
Membaca : Ibadah, Mengambil : Dosa

**POLITEKNIK STMI JAKARTA**  
**KEMENTERIAN PERINDUSTRIAN RI**

**LEMBAR PENGESAHAN**

**JUDUL TUGAS AKHIR:**

**SISTEM INFORMASI *MONITORING* HASIL PRODUKSI *CLUTCH*  
*HOUSING* PADA PT MITSUBISHI KRAMA YUDHA MOTORS AND  
MANUFACTURING**

**Disusun Oleh:**

Nama : Sifa Audita

NIM : 1316022

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 R.I. pada hari Kamis, 10 September 2020

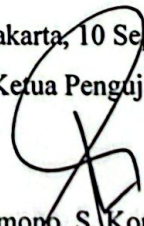
Jakarta, 10 September 2020

Dosen Pembimbing



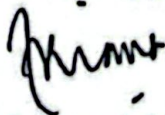
Ahmad Juniar, S. Kom., MT  
NIP.197906052006041002

Ketua Penguji



Ahlan Ismono, S. Kom., MMSI  
NIP.197901072000641002

Dosen Penguji



Triana Fatmawati, S.T., MT  
NIP.198005142005022001

Dosen Penguji



Lucky Herivanto, S.T., MTI  
NIP.197908202009011009

**POLITEKNIK STMI JAKARTA**  
**KEMENTERIAN PERINDUSTRIAN RI**

**TANDA PERSETUJUAN DOSEN PEMBIMBING**

JUDUL TUGAS AKHIR:

**SISTEM INFORMASI *MONITORING* HASIL PRODUKSI *CLUTCH*  
*HOUSING* PADA PT MITSUBISHI KRAMA YUDHA MOTORS AND  
MANUFACTURING**

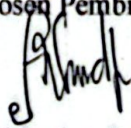
**Disusun Oleh:**

Nama : Sifa Audita  
Nim : 1316022  
Program Studi : Sistem Informasi Industri Otomotif  
Tanggal Seminar : 24 Agustus 2020  
Tanggal Sidang : 10 September 2020  
Tanggal Lulus : 10 September 2020

Telah diperiksa dan disetujui untuk diajukan dan dipertahankan dalam  
Ujian Tugas Akhir Politeknik STMI Jakarta

Jakarta, 10 September 2020












Dosen Pembimbing,



Ahmad Juniar, S. Kom., MT.  
NIP.197906052006041002

## LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR


Nama : Sifa Audita  
 NIM : 1316022  
 Judul TA : Sistem Informasi *Monitoring* Hasil Produksi *Clutch*  
*Housing* Pada PT Mitsubishi Krama Yudha Motors  
*and Manufacturing.*  
 Pembimbing : Ahmad Juniar, S.Kom., MT.

Tanggal	Bab	Keterangan	Paraf
28 Feb 2020	I	Pengajuan Cover Judul dan Bab I	
20 maret 2019	I & II	Revisi Bab I, Pengajuan Ganti Judul, Bab II	
13 April 2020	II & III	Revisi Bab II dan Pengajuan Bab III	
30 April 2020	III & IV	Revisi Bab III, penambahan teori, Pengajuan Bab IV.	
07 Mei 2020	IV	Revisi Bab IV	
21 Mei 2020	V	Pengajuan Bab V (Use Case dan Activity Diagram)	
08 Juni 2020	V	Pengajuan Bab V (Class, Deployment, dan WND Diagram)	
11 Juni 2020	V	Pengajuan Bab V (ERD dan Kamus Data)	
15 Juni 2020	V	Pengajuan Bab V (Interface dan Black Box Testing)	
22 Juni 2020	VI	Pengajuan Bab 6 dan lampiran	
03 Juli 2020		Bimbingan Program	

Mengetahui,  
 Ketua Program Studi Sistem Informasi  
 Industri Otomotif

Dosen Pembimbing

Lucky Heriyanto, S.T., MTI  
 NIP. 197811212009012003

  
Ahmad Juniar, S.Kom.MT.  
 NIP. 197906052006041002

## LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Sifa Audita

NIM : 1316022

Berstatus mahasiswa Program Studi Sistem Informasi Industri Otomotif di Politeknik STMI Jakarta. Kementrian Perindustrian Republik Indonesia. Dengan ini menyatakan bahwa hasil karya Tugas Akhir yang saya buat dengan judul:

**“Sistem Informasi *Monitoring* Hasil Produksi *Clutch Housing* Pada PT Mitsubishi Krama Yudha Motors and Manufacturing”**

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

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

Jakarta, 10 September 2020

Sifa Audita

## ABSTRAK

PT Mitsubishi Krama Yudha Motors and Manufacturing merupakan perusahaan manufaktur yang bergerak di bidang otomotif. Perusahaan ini memproduksi kompone untuk kendaraan khususnya Colt L300 dan Mitsubishi Fuso. Pada saat memproduksi Mitsubishi Fuso terdapat komponen poros gigi yang bernama *Clutch Housing (CH)*. CH diproduksi melalui enam mesin dengan dua mesin *backup* yang pada setiap mesinnya memiliki standar pengecekan kualitas (*checksheet*). Lembar *checksheet* hanya untuk *material* yang kondisinya OK, sedangkan *material* yang kondisinya *Not Good (NG)* diisi di lembar kertas *list part reject* atau *list part pending*. Permasalahan yang terjadi pada PT Mitsubishi Krama Yudha Motors and Manufacturing yaitu pengisian *checksheet*, *list part pending*, dan *list part reject* masih menggunakan lembar *form* kertas dan diarsipkan ke dalam ordner. Untuk mengatasi masalah tersebut diperlukan suatu proses *monitoring* hasil produksi yang mana pengisian dan pengarsipannya sudah terotomatisasi. Metodologi pengembangan sistem yang digunakan pada penelitian adalah *waterfall*. Perancangan menggunakan pemodelan sistem berupa *use case diagram*, *activity diagram*, *class diagram*, *deployment diagram*, dan *windows navigation diagram* serta pemodelan data berupa *Entity Relationship Diagram (ERD)* dan kamus data. Sistem informasi ini dibangun menggunakan PHP dengan *Bootstrap* sebagai *framework css* dan *Mariadb* sebagai *database*.

**Kata Kunci:** *Monitoring Hasil Produksi, Bootstrap, Mariadb*

## KATA PENGANTAR

Puji Syukur dipanjatkan kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga dapat diselesaikan Tugas Akhir dengan judul “**Sistem Informasi *Monitoring* Hasil Produksi Clutch Housing Pada PT Mitsubishi Krama Yudha Motors and Manufacturing**”.

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

Telah disadari bahwa Tugas Akhir ini tidak akan terwujud tanpa adanya bantuan, bimbingan serta dukungan dari berbagai pihak. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Allah SWT atas rahmat serta kemudahan yang diberikan.
2. Kedua orang tua dan adik saya yang selalu memberikan dukungan, sehingga atas ridhonya saya dapat menjalankan pendidikan di Politeknik STMI Jakarta.
3. Bapak Dr. Mustofa, ST., MT, selaku Direktur Politeknik STMI Jakarta.
4. Bapak Lucky Heriyanto, ST., MTI, selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
5. Bapak Ahmad Juniar, S.Kom., MT, selaku Dosen Pembimbing yang telah membantu, meluangkan waktu untuk mengorganisir jalannya penulisan dan penyusunan tugas akhir.
6. Seluruh dosen yang ada di Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
7. Bapak Agus selaku HRD dan bapak Faikar selaku *Trans Assy and Engine Test Production Department* di PT Mitsubishi Krama Yudha Motors and Manufacturing.

8. Bapak Sarwono, Bapak Ahmad, dan *Operator Line* selaku pembimbing yang telah membantu mengarahkan dan membimbing selama Kerja Lapangan di PT Mitsubishi Krama Yudha Motors and Manufacturing.
9. Seluruh karyawan di PT Mitsubishi Krama Yudha Motors and Manufacturing yang telah membantu memberikan arahan serta informasi dalam pengumpulan data.
10. Dr. Agung Priatmaja, Sp.Kj., M.Kes yang telah memberikan dukungan dalam kesehatan mental saya.
11. Bapak Hario Broto dan Hartanti Kusuma Dewi yang telah membantu saya memberikan berbagai saran dalam pembuatan program.
12. Alfath Febriadi, Atikah Restiwahyuni, Deni Arbianto, Mohammad Fadhel Hasbullah, Zeus Christian Fernando, dan Dwiky Radhifan selaku sahabat saya yang selalu memberikan saya dukungan serta saran-saran dalam pembuatan tugas akhir.
13. Seluruh teman-teman mahasiswa/i Politeknik STMI Jakarta Program Studi Sistem Informasi Industri.

Semoga Allah SWT membalas segala kebaikan dan ketulusan serta memberikan kemudahan dalam segala urusan kepada pihak-pihak yang telah membantu. Telah disadari bahwa penulisan Laporan Tugas Akhir ini masih jauh dari sempurna, untuk itu dimohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan di masa yang akan datang.

Jakarta, 10 September 2020

Penulis

## DAFTAR ISI

	Halaman
<b>LEMBAR PENGESAHAN .....</b>	<b>i</b>
<b>TANDA PERSETUJUAN DOSEN PEMBIMBING.....</b>	<b>ii</b>
<b>LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR .....</b>	<b>iii</b>
<b>LEMBAR PERNYATAAN KEASLIAN .....</b>	<b>iv</b>
<b>ABSTRAK .....</b>	<b>v</b>
<b>KATA PENGANTAR.....</b>	<b>vi</b>
<b>DAFTAR ISI.....</b>	<b>viii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiii</b>
<b>DAFTAR TABEL.....</b>	<b>xvii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Pokok Permasalahan .....	2
1.3 Tujuan Penelitian .....	2
1.4 Batasan Masalah.....	2
1.5 Manfaat Tugas Akhir .....	3
1.6 Sistematika Penulisan.....	3
<b>BAB II LANDASAN TEORI .....</b>	<b>5</b>
2.1 Konsep Dasar Sistem .....	5
2.1.1 Definisi Sistem .....	6

2.2 Konsep Dasar Informasi.....	6
2.2.1 Definisi Informasi .....	7
2.3 Definisi Sistem Informasi.....	7
2.3.1 Tipe-Tipe Sistem Informasi .....	7
2.3.2 Daur Hidup Sistem Informasi .....	9
2.3.3 Pengelolaan Sistem Informasi .....	11
2.4 Definisi Produksi.....	12
2.4.1 Proses Produksi .....	12
2.5 Definisi <i>Monitoring</i> .....	12
2.5.1 Sistem <i>Monitoring</i> .....	12
2.6 <i>System Development Life Cycle (SDLC)</i> .....	13
2.6.1 Metode Pengembangan Sistem <i>Waterfall</i> .....	17
2.7 Analisis dan Desain Berorientasi Objek.....	19
2.8 <i>Flowchart</i> .....	20
2.9 <i>Flowmap</i> .....	24
2.10 <i>Unified Modeling Language (UML)</i> .....	26
2.10.1 Diagram UML .....	26
2.10.2 <i>Use Case Diagram</i> .....	28
2.10.3 <i>Sequence Diagram</i> .....	29
2.10.4 <i>Activity Diagram</i> .....	32
2.10.4 <i>Class Diagram</i> .....	34
2.10.5 <i>Deployment Diagram</i> .....	36
2.10.6 <i>Windows Navigation Diagram (WND)</i> .....	37

2.11 <i>Entity Relationship Diagram (ERD)</i> .....	37
2.11.1 Kardinalitas .....	38
2.12 Kamus Data .....	39
2.13 <i>Hypertext Preprocessor (PHP)</i> .....	40
2.14 XAMPP .....	40
2.15 <i>Bootstrap</i> .....	41
2.16 <i>Framework</i> .....	41
2.17 MariaDB.....	42
2.18 <i>Black Box Testing</i> .....	42
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>43</b>
3.1 Metodologi Penelitian .....	43
3.2 Jenis dan Sumber Data .....	43
3.3 Metode Pengumpulan Data .....	44
3.4 Metodologi Pengembangan Sistem.....	45
3.5 Kerangka Penelitian .....	46
<b>BAB IV PENGUMPULAN DAN PENGOLAHAN DATA .....</b>	<b>50</b>
4.1 Sejarah Perusahaan.....	50
4.2 Profil Perusahaan.....	52
4.3 Visi dan Misi Perusahaan.....	54
4.4 Struktur Organisasi .....	54
4.5 Struktur Organisasi Departemen Produksi .....	55
4.6 Kegiatan Usaha Perusahaan .....	56
4.7 Pemasaran dan Sistem Distribusi .....	57

4.8 Produk Yang Dihasilkan .....	57
4.9 Dokumen Terkait Pada Prosedur Pengendalian Kualitas.....	59
4.11 Pengarsipan <i>Checksheet</i> .....	66
4.12 Prosedur Proses Bisnis <i>Monitoring</i> Hasil Produksi yang Sedang Berjalan .....	68
<b>BAB 5 HASIL ANALISIS DAN PEMBAHASAN .....</b>	<b>70</b>
5.1 Analisis Kebutuhan Sistem Prosedur .....	70
5.2 Proses Bisnis <i>Monitoring</i> Hasil Produksi Usulan .....	72
5.3 Analisis Sistem Usulan.....	74
5.3.1 <i>Use Case Diagram</i> Usulan.....	74
5.3.2 <i>Activity Diagram</i> Usulan.....	89
5.3.3 <i>Sequence Diagram</i> .....	105
5.3.4 <i>Class Diagram</i> Usulan .....	119
5.3.5 <i>Deployment Diagram</i> Usulan.....	120
5.3.6 <i>Windows Navigation Diagram (WND)</i> .....	121
5.4 Pemodelan Data Sistem Usulan .....	122
5.4.1 <i>Entity Relationship Diagram (ERD)</i> .....	122
5.4.2 Kamus Data .....	123
5.5 Perancangan Sistem Antarmuka.....	129
5.6 Pengujian <i>Black Box Testing</i> .....	163
<b>BAB VI KESIMPULAN DAN SARAN.....</b>	<b>181</b>
6.1 Kesimpulan.....	181
6.2 Saran.....	181

<b>DAFTAR PUSTAKA.....</b>	<b>182</b>
<b>LAMPIRAN A TAMPILAN PROGRAM.....</b>	<b>L-1</b>
<b>LAMPIRAN B KODE PROGRAM.....</b>	<b>L-20</b>

## DAFTAR GAMBAR

	Halaman
Gambar II.1 Gambar Daur Hidup Sistem Informasi .....	11
Gambar II.2 Fase Pengembangan Sistem Informasi .....	13
Gambar II.3 Fase Pengembangan Sistem <i>Waterfall</i> .....	19
Gambar II.4 Struktur <i>UML Diagram</i> .....	27
Gambar II.5 Contoh Windows Navigation Diagram .....	37
Gambar III.1 Kerangka Penelitian .....	49
Gambar IV.1 Logo PT MKM.....	50
Gambar IV.2 <i>Shareholder</i> PT MKM .....	51
Gambar IV.3 Luas PT MKM .....	53
Gambar IV.4 Struktur Organisasi PT MKM.....	55
Gambar IV.5 Struktur Organisasi Departemen Produksi.....	55
Gambar IV.6 Contoh <i>Checksheets Clutch Housing</i> .....	60
Gambar IV.7 Contoh <i>List Reject Process Clutch Housing</i> .....	61
Gambar IV.8 Contoh <i>List Reject Material Clutch Housing</i> .....	62
Gambar IV. 9 Contoh <i>List Part Pending M025</i> .....	64
Gambar IV.10 <i>List Part Pending M035</i> .....	65
Gambar IV.7 Tempat Pengarsipan Alat Sortir.....	67
Gambar IV.8 Tempat Pengarsipan <i>Ordner</i> .....	67
Gambar IV.9 <i>Flowmap Monitoring</i> Hasil Produksi.....	69
Gambar V.1 <i>Flowmap Proses Monitoring</i> Hasil Produksi Usulan .....	73
Gambar V.2 Sistem <i>Monitoring</i> Hasil Produksi Usulan.....	74
Gambar V.3 <i>Activity Diagram Login</i> .....	89
Gambar V.4 <i>Activity Diagram Mengelola Data User</i> .....	91
Gambar V.5 <i>Activity Diagram Mengelola Data Role</i> .....	92

Gambar V.6 <i>Activity Diagram</i> Mengelola Data <i>Model</i> .....	93
Gambar V.7 <i>Activity Diagram</i> Mengelola Data <i>Operator</i> .....	94
Gambar V.8 <i>Activity Diagram</i> Mengelola Alat Ukur .....	95
Gambar V.9 <i>Activity Diagram</i> Mengelola Item Check .....	96
Gambar V.10 <i>Activity Diagram</i> Input Data <i>Checksheet</i> .....	97
Gambar V.11 <i>Activity Diagram</i> Melihat Data <i>Checksheet Operator</i> .....	98
Gambar V.12 <i>Activity Diagram</i> Melihat Data <i>Checksheet As. Foreman dan Foreman</i> .....	99
Gambar V.13 <i>Activity Diagram</i> Mengaktifkan Data <i>Checksheet</i> .....	100
Gambar V.14 <i>Activity Diagram</i> Validasi Data <i>Checksheet</i> .....	101
Gambar V.15 <i>Activity Diagram</i> Input List <i>Reject or Pending</i> .....	102
Gambar V.16 <i>Activity Diagram</i> Validasi List <i>Reject dan Pending</i> .....	103
Gambar V.17 <i>Activity Diagram</i> Rekap LPP dan LPR.....	104
Gambar V.18 <i>Sequence Diagram</i> Login.....	105
Gambar V.19 <i>Sequence Diagram</i> Mengelola Data <i>User</i> .....	106
Gambar V.20 <i>Sequence Diagram</i> Mengelola Data <i>Role</i> .....	107
Gambar V.21 <i>Sequence Diagram</i> Mengelola Data <i>Model</i> .....	108
Gambar V.22 <i>Sequence Diagram</i> Mengelola Data <i>Operator</i> .....	109
Gambar V.23 <i>Sequence Diagram</i> Mengelola Alat Ukur .....	110
Gambar V.24 <i>Sequence Diagram</i> Mengelola Item Check.....	111
Gambar V.25 <i>Sequence Diagram</i> Input Data <i>Checksheet</i> .....	112
Gambar V.26 <i>Sequence Diagram</i> Melihat Data <i>Checksheet Operator</i> .....	113
Gambar V.27 <i>Sequence Diagram</i> Melihat Data <i>Checksheet As. Foreman dan</i> <i>Foreman</i> .....	114
Gambar V.28 <i>Sequence Diagram</i> Mengaktifkan Data <i>Checksheet</i> .....	115
Gambar V.29 <i>Sequence Diagram</i> Validasi Data <i>Checksheet</i> .....	116
Gambar V.30 <i>Sequence Diagram</i> Input List <i>Reject or Pending</i> .....	117
Gambar V.31 <i>Sequence Diagram</i> Validasi Data List <i>Reject or Pending</i> .....	118
Gambar V.32 <i>Sequence Diagram</i> Rekap LPP dan LPR .....	119

Gambar V.33 <i>Class Diagram</i> Sistem Informasi <i>Monitoring</i> Hasil Produksi <i>Clutch Housing</i> .....	120
Gambar V.34 <i>Deployment Diagram</i> Sistem Informasi <i>Monitoring</i> Hasil Produksi <i>Clutch Housing</i> .....	121
Gambar V.35 <i>Windows Navigation Diagram</i> Sistem Informasi <i>Monitoring</i> Hasil Produksi <i>Clutch Housing</i> .....	122
Gambar V.36 <i>Entity Relationship Diagram</i> Usulan Sistem Informasi <i>Monitoring</i> Hasil Produksi <i>Clutch Housing</i> .....	123
Gambar V.22 Tampilan Halaman <i>Login</i> .....	130
Gambar V.23 Tampilan Halaman Utama Admin .....	131
Gambar V.24 Tampilan Halaman Utama <i>Operator</i> .....	132
Gambar V.25 Tampilan Halaman Utama <i>As. Foreman</i> .....	133
Gambar V.26 Tampilan Halaman Utama <i>Foreman</i> .....	134
Gambar V.27 Tampilan Halaman <i>User</i> .....	135
Gambar V.28 Tampilan Halaman <i>Add User</i> .....	136
Gambar V.29 Tampilan Halaman <i>Edit User</i> .....	137
Gambar V.30 Tampilan Halaman <i>Role</i> .....	138
Gambar V.32 Tampilan Halaman <i>Add Role</i> .....	139
Gambar V.33 Tampilan Halaman <i>Edit Role</i> .....	140
Gambar V.34 Tampilan Halaman <i>Model</i> .....	141
Gambar V.35 Tampilan Halaman <i>Add Model</i> .....	142
Gambar V.36 Tampilan Halaman <i>Edit Model</i> .....	143
Gambar V.37 Tampilan Halaman <i>Operator</i> .....	144
Gambar V.38 Tampilan Halaman <i>Add Operator</i> .....	145
Gambar V.39 Tampilan Halaman <i>Edit Operator</i> .....	146
Gambar V.40 Tampilan Halaman <i>Alat Ukur</i> .....	147
Gambar V.41 Tampilan Halaman <i>Add Alat Ukur</i> .....	148
Gambar V.42 Tampilan Halaman <i>Edit Alat Ukur</i> .....	149
Gambar V.43 Tampilan Halaman <i>Item Check</i> .....	150

Gambar V.44 Tampilan Halaman <i>Item Check</i> .....	151
Gambar V.45 Tampilan Halaman <i>Edit Item Check</i> .....	152
Gambar V.46 Tampilan Halaman <i>Checksheet</i> .....	153
Gambar V.47 Tampilan Halaman <i>Edit Checksheet</i> .....	154
Gambar V.48 Tampilan Halaman Laporan <i>Checksheet</i> .....	155
Gambar V.49 Tampilan Halaman <i>Checksheet Shift</i> Belum Disetujui .....	156
Gambar V.50 Tampilan Halaman <i>Checksheet</i> Aktif.....	157
Gambar V.51 Tampilan Halaman <i>Checksheet Shift</i> Sudah Disetujui .....	158
Gambar V.52 Tampilan Halaman <i>Data List Reject dan Pending</i> M025 dan M035 .	159
Gambar V.53 Tampilan Halaman <i>Add Reject</i> M025 dan M035 .....	160
Gambar V.54 Tampilan Halaman <i>Edit Reject dan Pending</i> M025 dan M035.....	161
Gambar V.46 Tampilan Halaman <i>List Reject dan Pending</i> Aktif.....	162
Gambar V.55 Tampilan Halaman Rekapian <i>List Reject dan Pending</i> .....	163

## DAFTAR TABEL

	Halaman
Tabel II.1 Simbol <i>Flowchart Flow Direction</i> .....	21
Tabel II.2 Simbol <i>Flowchart Processing</i> .....	22
Tabel II.3 Simbol <i>Flowchart Input</i> atau <i>Output</i> .....	23
Tabel II.3 Simbol <i>Flowchart Input</i> atau <i>Output</i> (lanjutan).....	24
Tabel II.4 Simbol-simbol <i>Flowmap</i> .....	25
Tabel II.5 Simbol-Simbol <i>Use Case Diagram</i> .....	28
Tabel II.6 Simbol-Simbol <i>Sequence Diagram</i> .....	30
Tabel II.7 Simbol Macam-macam <i>Class</i> pada <i>Sequence Diagram</i> .....	32
Tabel II.8 Simbol-Simbol <i>Activity Diagram</i> .....	33
Tabel II.9 Simbol-Simbol <i>Class Diagram</i> .....	35
Tabel II.10 Simbol-Simbol <i>Deployment Diagram</i> .....	36
Tabel II.11 Simbol-Simbol <i>Entity Relationship Diagram</i> .....	38
Tabel II.12 Contoh Kamus Data .....	40
Tabel IV.1 Daftar <i>Customer</i> Dalam Negeri dan Luar Negeri.....	57
Tabel III.2 Produk Yang Dihasilkan PT MKM .....	58
Tabel III.2 Produk Yang Dihasilkan PT MKM (lanjutan).....	59
Tabel V.1 Analisis Kebutuhan Sistem Usulan.....	70
Tabel V.1 Analisis Kebutuhan Sistem Usulan (lanjutan).....	71
Tabel V.2 Kebutuhan Sistem ( <i>Non Functional Requirement</i> ).....	71
Tabel V.3 Definisi Aktor <i>Use Case Diagram</i> Usulan.....	75
Tabel V.4 Definisi <i>Use Case Diagram</i> Usulan.....	76
Tabel V.5 Skenario <i>Use Case Login</i> .....	77
Tabel V.6 Skenario <i>Use Case Mengelola Data User</i> .....	78
Tabel V.7 Skenario <i>Use Case Mengelola Data Role</i> .....	79

Tabel V.8 Skenario <i>Use Case</i> Mengelola Data <i>Model</i> .....	80
Tabel V.9 Skenario <i>Use Case</i> Mengelola Data <i>Operator</i> .....	81
Tabel V.10 Skenario <i>Use Case</i> Mengelola Alat Ukur .....	82
Tabel V.11 Skenario <i>Use Case</i> Mengelola <i>Item Check</i> .....	83
Tabel V.12 Skenario <i>Use Case</i> <i>Input Data Checksheet</i> .....	84
Tabel V.13 Skenario <i>Use Case</i> Melihat Data <i>Checksheet</i> .....	85
Tabel V.14 Skenario <i>Use Case</i> Mengaktifkan Data <i>Checksheet</i> .....	85
Tabel V.14 Skenario <i>Use Case</i> Mengaktifkan Data <i>Checksheet</i> (lanjutan).....	86
Tabel V.15 Skenario <i>Use Case</i> Validasi Data <i>Checksheet</i> .....	86
Tabel V.16 Skenario <i>Use Case</i> <i>Input List Reject or Pending</i> .....	87
Tabel V.17 Skenario <i>Use Case</i> Validasi Data <i>List Reject or Pending</i> .....	87
Tabel V.18 Skenario <i>Use Case</i> Rekap LPP dan LPR .....	88
Tabel V.16 Tabel Kamus Data <i>User</i> .....	124
Tabel V.17 Tabel Kamus Data <i>Role</i> .....	124
Tabel V.18 Tabel Kamus Data <i>Check</i> .....	125
Tabel V.19 Tabel Kamus Data <i>Operator</i> .....	125
Tabel V.20 Tabel Kamus Data <i>Model</i> .....	125
Tabel V.21 Tabel Kamus Data <i>Alat</i> .....	126
Tabel V.22 Tabel Kamus Data <i>Detail Monitor</i> .....	126
Tabel V.23 Tabel Kamus Data <i>Shop</i> .....	127
Tabel V.24 Tabel Kamus Data <i>Monitor</i> .....	127
Tabel V.25 Tabel Kamus Data <i>Detail Reject</i> .....	128
Tabel V.26 Kamus Data <i>Reject</i> .....	128
Tabel V.27 Kamu Data <i>List</i> .....	129
Tabel V.28 <i>Test Case</i> <i>Login</i> .....	164
Tabel V.29 <i>Test Case</i> Mengelola Data <i>User</i> .....	165
Tabel V.30 <i>Test Case</i> Mengelola Data <i>Role</i> .....	167
Tabel V.31 <i>Test Case</i> Mengelola Data <i>Model</i> .....	168
Tabel V.32 <i>Test Case</i> Mengelola Data <i>Operator</i> .....	170

Tabel V.33 <i>Test Case</i> Mengelola Alat Ukur.....	171
Tabel V.34 <i>Test Case</i> Mengelola <i>Item Check</i> .....	173
Tabel V.35 <i>Test Case</i> <i>Input Data Checksheet</i> .....	174
Tabel V.36 <i>Test Case</i> Melihat <i>Data Checksheet</i> .....	175
Tabel V.37 <i>Test Case</i> Mengaktifkan <i>Data Checksheet</i> .....	175
Tabel V.38 <i>Test Case</i> Validasi <i>Data Checksheet</i> .....	177
Tabel V.39 <i>Test Case</i> <i>Input List Reject or Pending</i> .....	178
Tabel V.40 <i>Test Case</i> Validasi <i>List Reject or Pending</i> .....	179
Tabel V.41 <i>Test Case</i> Rekap LPP dan LPR.....	180

## BAB I PENDAHULUAN

### 1.1 Latar Belakang

PT Mitsubishi Krama Yudha Motors and Manufacturing (PT MKM) merupakan perusahaan manufaktur yang bergerak di bidang otomotif yang berdiri sejak tahun 1973. Perusahaan ini memproduksi komponen-komponen untuk kendaraan, baik itu badan kendaraan maupun komponen untuk mesin kendaraan khususnya Colt L300 dan Mitsubishi Fuso. Perusahaan tersebut terbagi menjadi dua *factory* yang diantara MKM 1 (*Stamping Factory*) yang bergerak dalam memproduksi komponen badan kendaraan, sedangkan MKM 2 (*Engine Factory*) memproduksi komponen mesin kendaraan.

Departemen produksi pada *engine factory* perusahaan ini mengolah *blank material* menjadi barang jadi. Departemen ini dengan *line Transmission TD Machining* khususnya produk *clutch housing* sendiri merupakan poros gigi Mitsubishi Fuso yang pada saat proses produksi harus melalui enam mesin dengan dua mesin *back up* yang pada setiap mesinnya memiliki standar pengecekan kualitas atau yang biasa disebut dengan *checksheet*. *Checksheet* sendiri hanya untuk produk yang kualitasnya OK, sedangkan untuk produk yang kualitasnya *Not Good (NG)* akan diidentifikasi untuk mengetahui *material* tersebut merupakan *part reject* atau *part pending*. Departemen produksi ini pun harus menginput banyaknya jumlah produksi pada setiap harinya melalui sistem informasi yang diterapkan. Sementara yang lainnya seperti *checksheet*, *list part reject*, dan *list part pending* masih mengandalkan form kertas. *Operator machining* sendiri telah menerapkan pengisian form dan pengarsipan dokumen tersebut ke dalam ordner, akan tetapi dokumen tersebut rentan hilang dan dapat memakan waktu ketika mencarinya karena semua dokumen yang telah diisi akan diarsipkan ke dalam ordner.

Berdasarkan informasi diatas, untuk mengatasi masalah tersebut, diperlukan aplikasi yang terintegrasi untuk memaksimalkan penerapan sistem *monitoring* hasil produksi pada mesin produksi *Clutch Housing* pada PT MKM. Adapun judul tugas akhir adalah “**Sistem Informasi *Monitoring* Hasil Produksi *Clutch Housing* Pada PT Mitsubishi Krama Yudha Motors and Manufacturing**”

### 1.2 Pokok Permasalahan

Permasalahan yang terjadi pada Bagian Produksi dalam proses produksi *Clutch Housing* pada PT Mitsubishi Krama Yudha Motors and Manufacturing adalah sebagai berikut:

1. Pengarsipan *checksheet*, *list part pending*, *list part reject* yang dilakukan oleh *machining operator* masih menggunakan selembar form kertas sehingga rentan kehilangan dokumen.
2. Terdapat banyaknya waktu yang terbuang dikarenakan mencari data *checksheet*, *list part pending*, dan *list part reject* pada tumpukan kertas di ordner jika adanya blank material dengan kondisi *Not Good (NG)*

### 1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah merancang dan membangun sistem informasi *monitoring* hasil produksi *clutch housing* yang mampu:

1. Memudahkan pengarsipan pada data *checksheet*, *list part pending*, dan *list part reject*.
2. Membuat waktu menjadi lebih efisien pada saat pencarian dokumen *checksheet*, *list part pending*, dan *list part reject*.

### 1.4 Batasan Masalah

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

1. Departemen atau bagian yang diteliti adalah Departemen Produksi pada seksi *Machining* khususnya *Transmission TD Machining* dengan hasil produksi *clutch housing*.
2. Selama proses penelitian dilakukan pengamatan pada pengisian dan pengarsipan *checksheet*, *list part pending*, dan *list part reject*.
3. Sistem ini hanya di rancang bangun dan belum diimplementasikan di dalam perusahaan.

### 1.5 Manfaat Tugas Akhir

Adapun manfaat bagi perusahaan yang dapat diambil dari penelitian Tugas Akhir ini adalah perusahaan memiliki aplikasi *monitoring* hasil produksi yang sudah terotomatisasi dengan media penyimpanan berupa database dan pengelolaan data dengan hak akses. Hal ini dapat membantu kinerja perusahaan dalam melakukan proses bisnis penyimpanan data guna mengatasi rentannya kehilangan data.

### 1.6 Sistematika Penulisan

Laporan ini disusun berdasarkan hal-hal yang berhubungan erat dengan hasil penelitian sehingga dapat memperoleh gambaran yang lebih jelas mengenai isi laporan dengan praktik kerja lapangan yang dilaksanakan. Adapun tahapan-tahapan dalam laporan ini adalah sebagai berikut:

#### **BAB I        PENDAHULUAN**

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

#### **BAB II        LANDASAN TEORI**

Bab ini menjelaskan tentang berbagai teori yang dirangkum dari jurnal, buku-buku atau berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori kajian penelitian yang dipaparkan pada laporan ini adalah seputar metode *Statistical*, *monitoring hasil produksi*,

artikel jurnal yang sesuai dengan topik yang diambil, metodologi pengembangan sistem, *flowchart*, *Unified Modeling Language (UML)*, *Entity Relationship Diagram (ERD)*, *Hypertext Preprocessor (PHP)*, dan MariaDB.

### **BAB III      METODOLOGI PENELITIAN**

Pada bab ini membahas tentang metode pengumpulan data, serta langkah-langkah yang akan dilakukan dalam perumusan dan pemecahan masalah termasuk metodologi pengembangan sistem yang digunakan.

### **BAB IV      PENGUMPULAN DAN PENGOLAHAN DATA**

Pada bab ini akan membahas mengenai data yang telah diperoleh berdasarkan hasil pengamatan di PT Mitsubishi Krama Yudha Motors and Manufacturing.

### **BAB V        ANALISIS DAN PEMBAHASAN**

Pada bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis sistem yang meliputi diagram alir sistem yang berjalan, perancangan basis data, perancangan tampilan layar, perancangan UML, perancangan hierarki menu, dan pembuatan spesifikasi proses melalui metode yang diterapkan.

### **BAB VI      PENUTUP**

Dalam bab penutup ini dikemukakan kesimpulan-kesimpulan dari hasil penelitian dan saran-saran untuk pihak perusahaan dalam berbagai hal yang berhubungan dengan proses *monitoring* hasil produksi.

## BAB II

### LANDASAN TEORI

#### 2.1 Konsep Dasar Sistem

Terdapat dua kelompok pendekatan di dalam pendefinisian sistem, yaitu kelompok yang menekankan pada prosedur dan kelompok yang menekankan pada elemen atau komponennya. Pendekatan yang menekankan pada prosedur mendefinisikan sistem sebagai suatu jaringan kerja prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponen mendefinisikan sistem sebagai kumpulan elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

Teori sistem melahirkan konsep-konsep *futuristic*, antara lain yang terkenal adalah konsep sibernetika (*cybernetics*). Konsep atau bidang kajian ilmiah ini terutama berkaitan dengan upaya-upaya untuk menerapkan berbagai disiplin ilmu, yaitu ilmu perilaku, fisika, biologi, dan teknik. Oleh karena itu sibernetika biasanya berkaitan dengan usaha-usaha otomasi tugas-tugas yang dilakukan oleh manusia, sehingga melahirkan studi-studi tentang robotika, kecerdasan buatan (*artificial intelligence*) dan lain sebagainya. Unsur-unsur yang mewakili suatu sistem secara umum adalah masukan (*input*), pengolahan (*processing*), dan keluaran (*output*).

Konsep lain yang terkandung di dalam definisi sistem adalah konsep sinergi. Konsep ini mengandaikan bahwa di dalam suatu sistem, *output* dari suatu organisasi diharapkan lebih besar daripada *output* individual atau *output* masing-masing bagian. (Sutabri, 2012)

### 2.1.1 Definisi Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukansasaran yang tertentu (Hutahaean, 2015). Sistem adalah kumpulan orang yang saling bekerja sama dengan ketentuan-ketentuan aturan yang sistematis dan terstruktur untuk membentuk satu kesatuan yang melaksanakan suatu fungsi untuk mencapai tujuan. (Anggraeni, 2017)

### 2.2 Konsep Dasar Informasi

Informasi merupakan proses lebih lanjut dari data yang sudah memiliki nilai tambah. Informasi dapat dikelompokkan menjadi tiga bagian, yaitu:

1. **Informasi Strategis.** Informasi ini digunakan untuk mengambil keputusan jangka panjang yang mencakup informasi eksternal, rencana perluasan perusahaan, dan sebagainya.
2. **Informasi Taktis.** Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti informasi tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.
3. **Informasi Teknis.** Informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stok, retur penjualan, dan laporan kas harian.

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan.

Teori informasi lebih tepat disebut sebagai teori matematika komunikasi yang memberikan pandangan yang berguna bagi sistem informasi, di mana konsep usia

informasi menunjukkan hubungan interval informasi, jenis data, dan penundaan pengolahan dalam menentukan usia informasi. Sumber dari informasi adalah data. (Sutabri, 2012)

### **2.2.1 Definisi Informasi**

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Data kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian (*event*) adalah kejadian yang terjadi pada saat tertentu. (Hutahaean, 2015)

Informasi merupakan data yang sudah diolah yang ditujukan untuk seseorang, organisasi ataupun siapa saja yang membutuhkan. Informasi akan menjadi berguna apabila objek menerima informasi membutuhkan informasi tersebut. (Sutabri, 2012)

### **2.3 Definisi Sistem Informasi**

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan (Anggraeni, 2017).

Menurut Sutabri (2012), sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

#### **2.3.1 Tipe-Tipe Sistem Informasi**

Sistem informasi dibagi menjadi beberapa tipe, khususnya pada *Computer Based Information System* (CBIS) atau dalam bahasa Indonesia disebut Sistem

Informasi Berbasis Komputer menurut Fatta (2007) dibedakan menjadi beberapa tipe aplikasi, yaitu:

1. *Transaction Processing System (TPS)*

*Transaction Processing System (TPS)* atau Sistem Pemrosesan Transaksi adalah sistem informasi terkomputerisasi yang dikembangkan untuk memproses sejumlah besar data untuk transaksi bisnis rutin. Adapun hal-hal yang bisa dilakukan dalam sistem ini meliputi:

- a. Mengotomasi penanganan data-data aktivitas bisnis dan transaksi, yang bisa dianggap sebagai kejadian diskrit dalam kehidupan organisasi.
  - b. Menangkap data dari setiap transaksi.
  - c. Memverifikasi transaksi untuk diterima atau ditolak.
  - d. Menyimpan transaksi yang telah divalidasi untuk pengumpulan data berikutnya.
  - e. Menghasilkan laporan untuk menyediakan rangkuman dari setiap transaksi.
- Memungkinkan memindah transaksi dari satu proses ke proses yang lainnya untuk menangani seluruh aspek bisnis.

2. *Management Information System (MIS)*

Sistem Informasi Manajemen (SIM) adalah sebuah sistem informasi pada level manajemen yang berfungsi untuk membantu perencanaan, pengendalian, dan pengambilan keputusan dan menyediakan *resume* rutin dan laporan-laporan tertentu. SIM mengambil data mentah dari TPS dan mengubahnya menjadi kumpulan data yang lebih berarti yang dibutuhkan manajer untuk menjalankan tanggung jawabnya. Untuk mengembangkan suatu SIM, diperlukan pemahaman yang baik tentang informasi apa saja yang dibutuhkan manajer dan bagaimana mereka menggunakan informasi tersebut.

3. *Decision Support System (DSS)*

*Decision Support System* merupakan sistem informasi pada level manajemen dari suatu organisasi yang mengombinasikan data dan model analisis canggih atau peralatan data analisis untuk mendukung pengambilan yang semi terstruktur

dan tidak terstruktur. DSS dirancang untuk membantu pengambilan keputusan organisasional. DSS biasanya tersusun dari:

- a. *Database* (bisa diekstraksi dari TPS atau MIS).
- b. Model grafis atau matematis, yang digunakan untuk proses bisnis.
- c. Antarmuka pengguna, yang digunakan oleh pengguna untuk berkomunikasi dengan DSS.

#### 4. *Expert System and Artificial Intelligence (ES & AI)*

*Expert System (ES)* merupakan representasi pengetahuan yang menggambarkan cara seorang ahli dalam mendekati suatu masalah. ES lebih berpusat pada bagaimana mengodekan dan memanipulasi pengetahuan dari informasi (misalnya aturan *if...then*). Adapun cara kerja ES sebagai berikut:

- a. Pengguna berkomunikasi dengan sistem menggunakan dialog interaktif.
- b. ES menanyakan pertanyaan (yang akan ditanyakan seorang pakar) dan pengguna memberikan jawaban.
- c. Jawaban digunakan untuk menentukan aturan mana yang dipakai dan ES sistem menyediakan rekomendasi berdasarkan aturan yang telah disimpan.
- d. Seorang *knowledge engineer* bertanggung jawab pada bagaimana melakukan akuisisi pengetahuan, sama seperti seorang analis tetapi dilatih untuk menggunakan teknik yang berbeda.

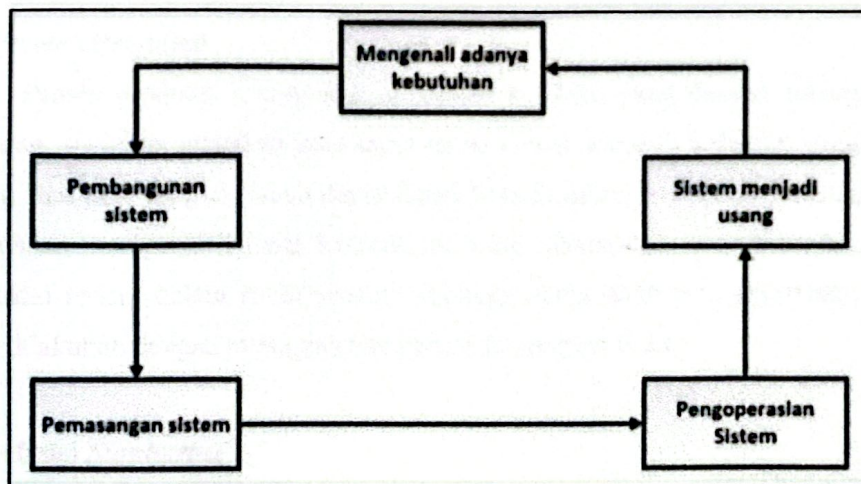
### 2.3.2 Daur Hidup Sistem Informasi

Siklus hidup sistem (*system life cycle*) adalah proses evolusioner yang diikuti dalam penerapan sistem atau subsistem informasi berbasis computer. Siklus hidup sistem sering disebut sebagai pendekatan air terjun (*waterfall approach*) bagi pembangunan dan pengembangan sistem. (Sutabri, 2012)

Pembangunan sistem hanyalah salah satu dari rangkaian daur hidup suatu sistem. Meskipun demikian proses ini merupakan aspek yang sangat penting. Berikut beberapa fase/ tahapan daur hidup suatu sistem, antara lain:

1. **Mengenali adanya kebutuhan**  
Sebelum segala sesuatu terjadi, pastilah terlebih dahulu timbul suatu kebutuhan atau problema yang harus dapat dikenali sebagaimana adanya. Kebutuhan dapat terjadi sebagai hasil perkembangan organisasi. Volume kebutuhan itu meningkat melebihi kapasitas dari sistem yang ada. Semua kebutuhan ini harus dapat didefinisikan dengan jelas. Tanpa adanya kejelasan mengenai kebutuhan yang ada, pembangunan sistem akan kehilangan arah dan efektivitasnya.
2. **Pembangunan sistem**  
Suatu proses atau seperangkat prosedur yang harus diikuti guna menganalisis kebutuhan yang timbul dan membangun suatu sistem untuk memenuhi kebutuhan tersebut.
3. **Pemasangan sistem**  
Setelah tahap pembangunan sistem selesai, kemudian sistem akan dioperasikan. Pemasangan sistem merupakan tahap yang penting dalam daur hidup sistem, di mana yang merupakan langkah akhir dari suatu pembangunan sistem.
4. **Pengoperasian sistem**  
Program-program computer dan prosedur-prosedur pengoperasian yang membentuk suatu sistem informasi semuanya bersifat statis, sedangkan organisasi yang ditunjang oleh sistem informasi selalu mengalami perubahan karena pertumbuhan kegiatan, perubahan peraturan dan kebijaksanaan, ataupun teknologi. Untuk mengatasi perubahan-perubahan tersebut, sistem harus diperbaiki atau diperbaharui.
5. **Sistem menjadi *using***  
Kadang-kadang perubahan yang terjadi begitu drastis sehingga tidak dapat diatasi hanya dengan melakukan perbaikan pada sistem yang sedang berjalan. Tiba saat di mana secara ekonomis dan teknis, sistem yang ada sudah tidak layak lagi untuk dioperasikan dan sistem yang baru perlu dibangun untuk menggantikannya.

Sistem informasi kemudian akan melanjutkan daur hidupnya. Sistem dibangun untuk memenuhi kebutuhan. Sistem beradaptasi terhadap aneka perubahan lingkungannya yang dinamis hingga kemudian sampai pada kondisi di mana sistem tidak dapat lagi beradaptasi. Sistem baru kemudian dibangun untuk menggantikannya. Daur hidup sistem dapat dilihat pada Gambar II.3



Gambar II.1 Gambar Daur Hidup Sistem Informasi  
Sumber: Sutabri (2012)

### 2.3.3 Pengelolaan Sistem Informasi

Pengelolaan sistem informasi merupakan bagian yang tak terpisahkan dari *studi* manajemen, sebagaimana halnya pengelolaan ketenagaan, keuangan, organisasi, tata laksana, dan lain sebagainya. Barang kali dapat diasumsikan bahwa pengelolaan sistem informasi merupakan faktor kunci bagi keberhasilan dan terlaksananya manajemen. Hal ini dapat dimengerti mengingat semua sub sistem manajemen bertopang pada unsur manusia, baik sebagai manajer maupun sebagai bawahan yang ditentukan dengan cara bertingkah laku atau melakukan perbuatan tertentu yang terarah untuk mencapai tujuan manajemen.

## 2.4 Definisi Produksi

Menurut Assauri (2008), produksi dikerjakan untuk menambah nilai guna suatu benda atau menciptakan benda baru sehingga lebih bermanfaat dalam memenuhi kebutuhan. Kegiatan menambah daya guna suatu benda tanpa mengubah bentuknya dinamakan produksi jasa, sedangkan kegiatan menambah daya guna suatu benda dengan mengubah sifat dan bentuknya dinamakan produksi barang.

### 2.4.1 Proses Produksi

Proses produksi merupakan rangkaian kegiatan yang dengan menggunakan peralatan, sehingga masukan atau input dapat diolah menjadi keluaran yang berupa barang atau jasa yang akhirnya dapat dijual kepada pelanggan untuk memungkinkan perusahaan memperoleh hasil keuntungan yang diharapkan. Proses produksi yang dilakukan terkait dalam suatu sistem, sehingga pengolahan atau pentransformasian dapat dilakukan dengan menggunakan peralatan yang dimiliki.

## 2.5 Definisi *Monitoring*

Menurut Moerdiyanto (2009), *monitoring* merupakan aktivitas yang dilakukan pimpinan untuk melihat, memonitor jalannya organisasi selama kegiatan berlangsung, dan menilai ketercapaian tujuan, melihat faktor pendukung dan penghambat pelaksanaan program. Dalam *monitoring* dikumpulkan data dan dianalisis, hasil analisis diinterpretasikan dan dimaknakan sebagai masukan bagi pimpinan untuk mengadakan perbaikan.

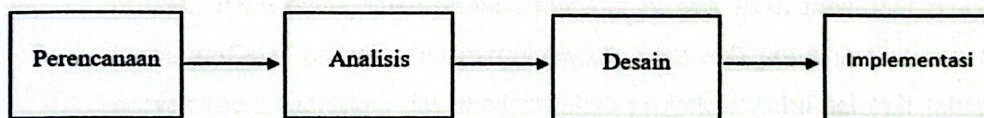
### 2.5.1 Sistem *Monitoring*

Menurut Wrihatnolo (2008), sistem *monitoring* dapat memberikan informasi keberlangsungan proses untuk menetapkan langkah menuju ke arah perbaikan yang berkesinambungan. Pada pelaksanaannya, *monitoring* dilakukan ketika suatu proses sedang berlangsung.

## 2.6 System Development Life Cycle (SDLC)

Menurut Dennis dkk. (2015) dalam membangun sistem informasi mirip dengan membangun rumah. Pertama, menjelaskan visi rumah kepada pengembang. Kedua, ide ini ditransformasikan menjadi sketsa dan gambar yang diperlihatkan kepada pemilik sampai pemilik setuju bahwa gambar-gambar yang ada menggambarkan apa yang dia inginkan. Ketiga, serangkaian cetak biru yang terperinci dikembangkan hingga menyajikan lebih banyak informasi yang lebih spesifik mengenai rumah yang dibuat. Terakhir, rumah dibangun mengikuti cetak biru dan sering kali dengan adanya beberapa perubahan dan keputusan yang dibuat oleh pemilik ketika rumah itu didirikan.

Membangun sistem informasi SDLC mengikuti serangkaian empat fase dasar yang serupa, yaitu perencanaan, analisis, desain, dan implementasi. Fase pengembangan sistem informasi dapat dilihat pada Gambar II.3.



Gambar II.2 Fase Pengembangan Sistem Informasi  
Sumber: Dennis dkk (2015)

### 1. Perencanaan

Tahap perencanaan adalah proses dasar untuk memahami mengapa sistem informasi harus dibangun dan menentukan bagaimana tim proyek akan membangun sebuah sistem informasi. Terdapat dua tahapan dalam perencanaan pengembangan sistem informasi, yaitu:

- a. Dalam masa inisiasi atau permulaan awal sebuah proyek, nilai bisnis terhadap organisasi harus diidentifikasi terlebih dahulu. Contohnya: Bagaimana hal itu (sistem informasi) dapat menurunkan dan meningkatkan pendapatan ?. Sebagian besar ide untuk sistem baru datang dari luar sistem informasi itu sendiri ( dari departemen pemasaran, departemen akuntansi,

dll) dalam bentuk sebuah permintaan sistem. Sebuah permintaan sistem menyajikan ringkasan singkat dari kebutuhan bisnis, dan menjelaskan bagaimana sebuah sistem yang mendukung kebutuhan dapat menciptakan sebuah nilai bisnis. Departemen sistem informasi bekerja sama dengan seseorang atau departemen yang menghasilkan permintaan (disebut sponsor proyek) untuk melakukan analisis kelayakan. Analisis kelayakan menguji aspek-aspek kunci dari proyek yang diusulkan, yaitu:

- 1) Kelayakan Teknis (Bisakah kita membangunnya ?).
- 2) Kelayakan Ekonomi (Apakah bisa memberikan nilai bisnis ?).
- 3) Kelayakan Organisasi (Apakah akan digunakan ?).

Permintaan sistem dan analisis kelayakan diberikan kepada *approval committee* yang memutuskan apakah proyek akan dilaksanakan atau tidak.

- b. Setelah proyek disetujui, proyek tersebut akan masuk ke tahap manajemen proyek. Pada manajemen proyek, manajer proyek akan membuat rencana kerja, staf-staf proyek, dan menempatkan para ahli untuk membantu tim proyek mengendalikan dan mengarahkan proyek melalui seluruh tahapan SDLC. Hasil dari manajemen proyek adalah rencana proyek yang mendeskripsikan bagaimana tim proyek akan mengembangkan sebuah sistem.

## 2. Analisis

Pada tahap analisis adalah tahap menjawab pertanyaan mengenai siapa yang akan menggunakan sistem, apa yang akan sistem lakukan, dan dimana serta kapan sistem itu akan digunakan. Selama dalam tahap ini, tim proyek akan menyelidiki setiap sistem, mengidentifikasi peluang peningkatan, dan pengembangan konsep untuk sistem baru. Pada tahap analisis terdapat tiga langkah, yaitu:

- a. Strategi Analisis dikembangkan untuk memandu atau mengarahkan tim proyek. Strategi seperti ini biasanya mencakup studi mengenai sistem yang ada (sistem apa adanya) serta masalah-masalah yang ada pada sistem

tersebut dan membayangkan cara untuk merancang sistem baru (sistem yang akan datang).

- b. Pengumpulan persyaratan, misalnya dengan cara melalui wawancara, lokakarya (*workshop*) kelompok, atau kuesioner. Informasi ini dianalisis bersamaan dengan masukan dari sponsor proyek dan banyak orang lainnya untuk mengarahkan dalam pengembangan konsep sistem baru. Konsep sistem ini kemudian digunakan sebagai dasar untuk mengembangkan serangkaian model analisa bisnis yang menggambarkan bagaimana bisnis akan beroperasi jika sistem baru telah dikembangkan.
- c. Analisis, konsep sistem, dan model-model digabungkan menjadi sebuah dokumen yang disebut proposal sistem, yang disajikan kepada sponsor proyek dan kepada pembuatan keputusan. Tujuannya untuk mendapatkan persetujuan dari sponsor proyek dan *approval committee* apakah proyek akan terus berlanjut.

Proposal sistem adalah hasil awal yang menggambarkan persyaratan bisnis apa yang harus dipenuhi oleh sistem yang baru.

### 3. Desain

Pada tahap ini diputuskan bagaimana suatu sistem akan beroperasi dalam hal perangkat keras, perangkat lunak, dan infrastruktur jaringan yang akan dibuat, seperti antarmuka pengguna, formulir, laporan yang akan digunakan, program khusus, *database*, dan *file* yang akan dibutuhkan. Meskipun sebagian besar keputusan strategis mengenai sistem yang dibuat dalam pengembangan konsep sistem pada tahap analisis, tetapi pada tahap desain ditentukan dengan tepat bagaimana sistem akan beroperasi. Terdapat empat fase dalam tahap desain, yaitu:

- a. Strategi desain harus ditentukan terlebih dahulu, hal ini bertujuan untuk mengklarifikasi apakah sistem akan dikembangkan oleh *programmer* perusahaan sendiri, apakah pengembangannya akan diserahkan kepada

perusahaan lain (biasanya perusahaan konsultan), atau apakah perusahaan akan membeli *software package* yang sudah ada.

- b. Mengarah pada pengembangan desain arsitektur dasar sistem yang menggambarkan *hardware*, *software*, dan *network infrastructure* yang akan digunakan. Dalam banyak kasus dimana sistem akan menambahkan atau mengubah infrastruktur yang sudah ada di dalam organisasi. Desain antarmuka menentukan bagaimana pengguna akan bergerak melalui sistem (misalnya dengan metode navigasi seperti menu dan tombol di layar) dan formulir serta laporan yang akan digunakan oleh sistem.
- c. Pengembangan *Database* dan spesifikasi file. Hal ini bertujuan untuk mendefinisikan dengan tepat data apa yang akan disimpan dan dimana data akan disimpan.
- d. Tim analis mengembangkan desain program, dimana tim analis mendefinisikan program apa yang perlu ditulis dan apa yang sebenarnya setiap program akan lakukan.

Kumpulan hasil kerja ini (desain arsitektur, antarmuka pengguna, *database* dan spesifikasi *file*, dan desain program) adalah spesifikasi sistem yang diserahkan kepada tim *programming* untuk implementasi ke depannya. Pada akhir tahap desain, analisis kelayakan dan rencana proyek diperiksa ulang dan diperbaiki. Keputusan lainnya yang dibuat oleh sponsor proyek dan *approval committee* tentang apakah akan menghentikan proyek atau melanjutkannya.

#### 4. Implementasi

Implementasi adalah tahap terakhir dalam SLDC, dimana sistem sebenarnya dibangun (atau dibeli, dalam kasus *packaged software design*). Tahap ini biasanya merupakan tahap yang mendapat perhatian paling besar, karena bagi kebanyakan sistem implementasi merupakan bagian yang paling lama dan paling mahal dari proses pengembangan. Tahap implementasi memiliki tiga fase, yaitu:

- a. Membangun sistem, sistem ini dibangun dan diuji untuk memastikan kinerjanya sesuai dengan desain yang dibuat. Karena biaya *bug* bisa sangat

besar, dan pengujian adalah salah satu langkah paling penting dalam implementasi. Sebagian besar organisasi memberikan perhatian dan waktu lebih untuk pengujian daripada menulis program dari awal.

- b. Instalasi sistem, instalasi adalah proses dimana sistem lama dimatikan dan sistem baru dihidupkan. Salah satu aspek konversi yang paling penting adalah pengembangan rencana pelatihan untuk mengajarkan pengguna bagaimana menggunakan sistem baru dan membantu mengelola perubahan yang disebabkan oleh sistem baru.
- c. Tim analis membentuk rencana dukungan untuk sistem. Perencanaan ini biasanya mencakup tinjauan formal atau informal pasca-implementasi serta sebagai cara sistematis untuk mengidentifikasi perubahan besar dan kecil yang diperlukan untuk sistem.

### 2.6.1 Metode Pengembangan Sistem *Waterfall*

Terdapat 5 langkah dalam metodologi pengembangan sistem waterfall yang diantaranya adalah:

#### 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 yang cukup banyak. Kesalahan pada tahap ini akan mengakibatkan keuntungan yang diperoleh tidak maksimal, bahkan bisa rugi. Pada tahapan ini peran manajemen sistem informasi berpengalaman sangat dibutuhkan.

#### 2. Analisis (*Analysis*)

Tahap kedua, adalah tahap analisis, yaitu 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*)

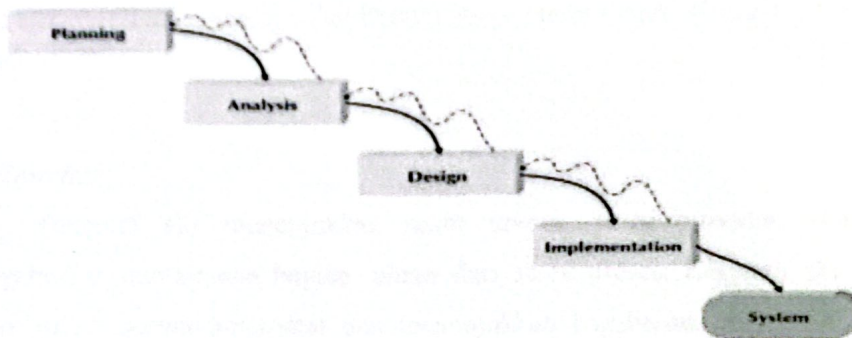
Tahap Perancangan (*design*), mencoba mencari solusi permasalahan yang didapat dari tahap analisis. Tahapan mengubah kebutuhan yang masih berupa konsep menjadi spesifikasi sistem yang nyata untuk diimplementasikan. Jika pada tahapan analisis (*form requirement to specification*), maka tahapan desain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

4. Implementasi (*Implementation*)

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

5. Sistem (*System*)

Pada tahapan sistem ini dilakukan pengujian (*testing*) dan pemeliharaan, yang dapat digunakan untuk menentukan apakah sistem/perangkat lunak yang dibuat sudah sesuai dengan kebutuhan pengguna atau belum. Jika belum, proses selanjutnya adalah bersifat *iterative*, yaitu kembali ke tahap sebelumnya. Tahap pemeliharaan dan perawatan dimana mulai melakukan pengoperasian sistem dan jika diperlukan melakukan perbaikan-perbaikan kecil. Kemudian jika waktu penggunaan sistem habis, maka akan masuk lagi pada tahap perencanaan (*design*).



Gambar II.3 Fase Pengembangan Sistem *Waterfall*  
 Sumber: Dennis dkk. (2015)

## 2.7 Analisis dan Desain Berorientasi Objek

Analisis Berorientasi Objek atau *Object Oriented Analysis (OOA)* adalah tahapan untuk menganalisis spesifikasi atau kebutuhan akan sistem yang akan dibangun dengan konsep berorientasi objek. OOA biasanya untuk membangun kelas-kelas yang akan digunakan atau menggunakan *Unified Modeling Language (UML)* pada bagian diagram *use case*, diagram kelas, dan diagram objek. (Rosa & Shalahuddin, 2018)

Desain Berorientasi Objek atau *Object Oriented Design (OOD)* adalah tahapan perantara untuk memetakan spesifikasi atau kebutuhan sistem yang akan dibangun dengan konsep berorientasi objek ke desain pemodelan agar lebih mudah diimplementasikan dengan pemrograman berorientasi objek. Pemodelan berorientasi objek biasanya dituangkan dalam dokumentasi perangkat lunak dengan menggunakan perangkat pemodelan berorientasi objek, diantaranya adalah UML. Kendala dan permasalahan pembangunan sistem berorientasi objek biasanya dapat dikenali dalam tahap ini.

OOA dan OOD dalam proses yang berulang-ulang seringkali memiliki batasan yang samar, sehingga kedua tahapan ini sering juga disebut *Object Oriented and*

*Design (OOAD) atau Analisis dan Desain Berorientasi Objek. (Rosa & Shalahuddin, 2018)*

## **2.8 Flowchart**

Diagram alir menunjukkan suatu urutan, suatu prosedur, atau seperti menyebutkan nama-nama bagian, aliran dari suatu proses. Diagram alir biasanya digambarkan secara horizontal dan menunjukkan bagaimana perbedaan aktivitas-aktivitas, unsur-unsur, atau prosedur-prosedur untuk menggabungkan sesuatu menjadi suatu keseluruhan. Diagram Alir dibagi menjadi 2 model, yaitu:

### **1. System Flowchart**

*System Flowchart* menggambarkan suatu sistem peralatan komputer yang digunakan dalam proses pengolahan data serta hubungan antar-peralatan tersebut dan tidak digunakan untuk menggambarkan urutan langkah untuk memecahkan masalah serta hanya untuk menggambarkan prosedur dalam sistem yang dibentuk.

### **2. Program Flowchart**

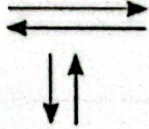

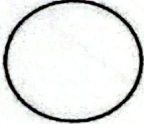
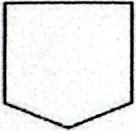
*Program Flowchart* menggambarkan urutan logika dari suatu prosedur pemecahan masalah. Ada dua jenis program chart, yaitu *conseptual flowchat*, menggambarkan alur pemecahan masalah secara global, dan *detail flowchart*, menggambarkan alur pemecahan masalah secara terperinci.

Menurut Jalinus & Ambiyar (2016), simbol-simbol *flowchart* dibagi menjadi tiga, yaitu:

#### **1. Flow Direction Symbols**

*Flow direction symbol* digunakan untuk menghubungkan simbol satu dengan yang lain dan disebut juga *connecting line*. Simbol diagram alir *flow direction* dapat dilihat pada Tabel II.1.

Tabel II.1 Simbol *Flowchart Flow Direction*

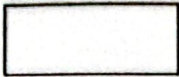

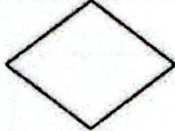
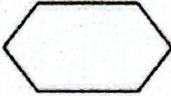
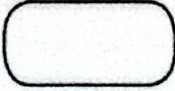
No.	Simbol	Makna
1		Simbol arus atau <i>flow</i> Menyatakan jalannya arus suatu proses.
2		Simbol <i>communication link</i> Menyatakan transmisi data dari satu lokasi ke lokasi lain.
3		Simbol <i>connector</i> Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
4		Simbol <i>offline connector</i> Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.

Sumber: Jalinus & Ambiyar (2016)

## 2. *Processing Symbols*

*Processing symbol* menunjukkan jenis operasi pengolahan dalam suatu proses atau prosedur. Simbol diagram alir *processing* dapat dilihat pada Tabel II.2.

Tabel II.2 Simbol *Flowchart Processing*

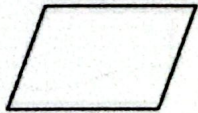
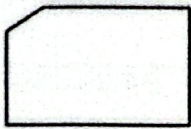
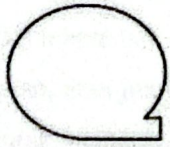
No.	Simbol	Makna
1		Simbol Proses Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
2		Simbol <i>Manual</i> Menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.
3		Simbol Decision Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban ya atau tidak.
4		Simbol <i>Predefined Process</i> Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5		Simbol <i>Terminal</i> Menyatakan permulaan atau akhir suatu program.

Sumber: Jalinus &amp; Ambiyar (2016)

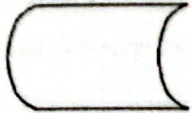
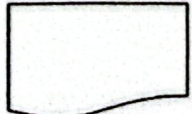
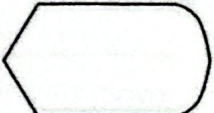
### 3. *Input atau Output Symbols*

*Input atau output symbol* menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*. Simbol diagram alir *input* atau *output* dapat dilihat pada Tabel II.3.

Tabel II.3 Simbol *Flowchart Input atau Output*

No.	Simbol	Makna
1		Simbol <i>Input</i> atau <i>Output</i> Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya.
2		Simbol <i>Punched Card</i> Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu.
3		Simbol <i>Magnetic Tape</i> Menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis.

Tabel II.3 Simbol *Flowchart Input* atau *Output* (lanjutan)

No.	Simbol	Makna
4		Simbol <i>Disk Storage</i> Menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
5		Simbol <i>Document</i> Mencetak keluaran dalam bentuk dokumen.
6		Simbol <i>Display</i> Mencetak keluaran dalam layar monitor.

Sumber: Jalinus & Ambiyar (2016)

## 2.9 Flowmap

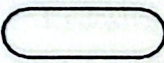

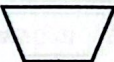

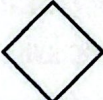
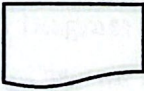
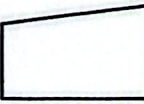
*Flowmap* adalah campuran peta dan *flowchart* yang menunjukkan pergerakan benda dari suatu lokasi ke lokasi lain, seperti jumlah orang dalam migrasi, jumlah barang yang diperdagangkan, atau jumlah paket dalam jaringan. *Flowmap* menolong analis dan *programmer* untuk memecahkan masalah kedalam segmen yang lebih kecil dan menolong dalam menganalisis alternatif lain dalam pengoperasian. Adapun pedoman-pedoman dalam pembuatan *flowmap* adalah sebagai berikut:

1. *Flowmap* sebaiknya digambarkan dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam *flowmap* harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhir.
4. Masing-masing kegiatan didalam *flowmap* sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.

5. Masing-masing kegiatan didalam *flowmap* harus didalam urutan yang semestinya.
  6. Kegiatan yang terpotong dan akan disambung ditempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
  7. Gunakan simbol simbol *flowmap* yang standar.
- (Jogiyanto, 2010).

Adapun simbol-simbol yang sering digunakan dalam *flowmap* dapat dilihat pada tabel II.4 berikut ini:

Tabel II.4 Simbol-simbol *Flowmap*

Gambar	Nama	Keterangan
	Mulai/Akhir	Menunjukkan dimulainya/akhir dari sebuah proses.
	Proses	Menunjukkan kegiatan proses dari operasi komputer.
	Kegiatan Manual	Menunjukkan kegiatan manual.
	Garis Alur	Menunjukkan arus dari proses.
	Keputusan	Keputusan dalam suatu program.
	Dokumen	Menunjukkan dokumen <i>input/ouput</i> baik untuk proses manual, mekanik, atau komputer.
	Keyboard	Menunjukkan <i>input</i> yang menggunakan <i>on-line keyboard</i> .

Sumber: (Jogiyanto, 2010)

## **2.10 Unified Modeling Language (UML)**

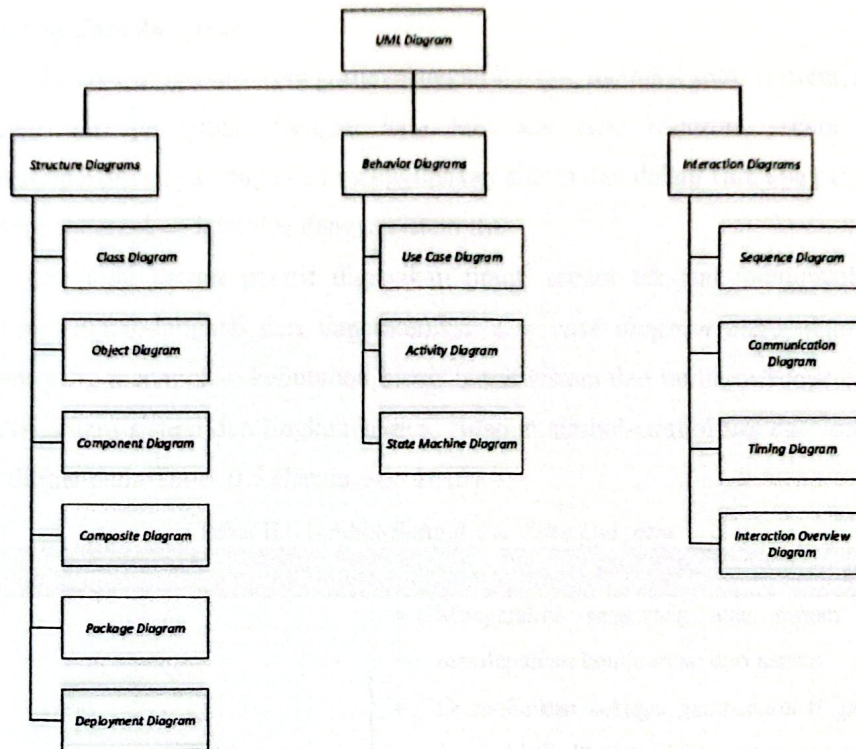
UML (*Unified Modeling Language*) merupakan kosakata umum berbasis objek dan diagram teknik yang cukup efektif untuk memodelkan setiap proyek pengembangan sistem mulai tahap analisis sampai tahap desain dan implementasi. UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk (Dennis dkk. 2015):

1. Merancang perangkat lunak.
2. Sarana komunikasi
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

Blok pembangunan utama UML adalah diagram. Beberapa diagram ada yang rinci dan lainnya ada yang bersifat umum (misalnya diagram kelas). Para pengembang sistem berorientasi objek menggunakan bahasa model untuk menggambarkan, membangun dan mendokumentasikan sistem yang mereka rancang. UML memungkinkan para anggota tim untuk bekerja sama dengan bahasa model yang sama dalam mengaplikasikan beragam sistem. Intinya, UML merupakan alat komunikasi yang konsisten dalam mensupport para pengembang sistem saat ini (Dennis dkk.2015).

### **2.10.1 Diagram UML**

Diagram UML terdiri dari tiga belas macam yang dikelompokkan dalam tiga kategori (dapat dilihat pada Gambar II.6)



Gambar II.4 Struktur *UML Diagram*  
 Sumber: Rosa & Shalahuddin (2018)

Berikut ini penjelasan singkat dari pembagian kategori tersebut:

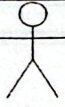
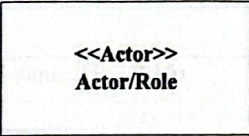
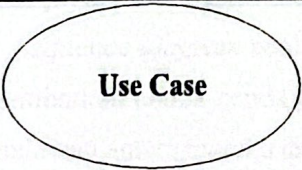
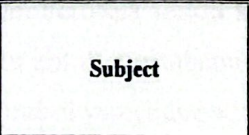
1. *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction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

### 2.10.2 Use Case Diagram


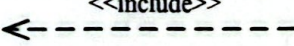
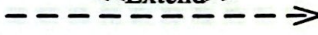

*Use case diagram* secara grafis menggambarkan, interaksi secara sistem, sistem eksternal dan pengguna. Dengan kata lain *use case diagram* secara grafis mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna (*user*) mengharapkan interaksi dengan sistem itu.

*Use case* secara naratif digunakan untuk secara tekstual menggambarkan sekuensi langkah-langkah dari tiap interaksi. *Use case diagram* merupakan suatu diagram yang menangkap kebutuhan bisnis untuk sistem dan untuk menggambarkan interaksi antara sistem dan lingkungannya. Adapun simbol-simbol *use case diagram* dapat dilihat pada Tabel II.5 (Denis dkk. 2015)

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

Simbol	Deskripsi
 <p>Actor/Role</p> 	<ul style="list-style-type: none"> <li>• Menyatakan seseorang atau sistem yang mendapatkan keuntungan dari sistem.</li> <li>• Digambarkan sebagai gambar <i>stick</i>/ gambar orang (<i>default</i>) atau jika bukan seorang aktor manusia, digambarkan dengan suatu kotak dengan tanda <code>&lt;&lt;actor&gt;&gt;</code> di dalamnya (alternatif).</li> </ul>
 <p>Use Case</p>	<ul style="list-style-type: none"> <li>• Merepresentasikan bagian utama dari fungsionalitas suatu sistem.</li> <li>• Dapat berupa perluasan <i>use case</i> lain.</li> <li>• Dapat termasuk di dalam <i>use case</i> lain.</li> <li>• Diletakkan di dalam batas sistem.</li> <li>• Dinamakan dengan frasa kata kerja.</li> </ul>
 <p>Subject</p>	<ul style="list-style-type: none"> <li>• Menyertakan nama subjek di dalam maupun di atas.</li> <li>• Merepresentasikan ruang lingkup dari subjek, sistem atau proses bisnis.</li> </ul>

Tabel II.5 Simbol-Simbol *Use Case Diagram* (lanjutan)

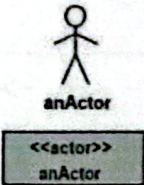
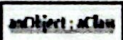


Simbol	Deskripsi
<p><i>Association</i></p> 	<ul style="list-style-type: none"> <li>Menghubungkan suatu aktor dengan <i>use case</i> dengan interaksi antara keduanya.</li> </ul>
<p><i>&lt;&lt;include&gt;&gt;</i></p> 	<ul style="list-style-type: none"> <li>Merepresentasikan fungsionalitas suatu <i>use case</i> dengan <i>use case</i> lainnya.</li> <li>Disimbolkan dengan anak panah dari sebuah <i>use case</i> dasar ke <i>use case</i> yang digunakan.</li> </ul>
<p><i>&lt;&lt;Extend&gt;&gt;</i></p> 	<ul style="list-style-type: none"> <li>Merepresentasikan perluasan (<i>extend</i>) dari <i>use case</i> lain untuk menyertakan perilaku opsional/<i>optional</i> (tidak wajib).</li> </ul>
 <p>Generalisasi</p>	<ul style="list-style-type: none"> <li>Merepresentasikan <i>use case</i> khusus ke satu <i>use case</i> yang lebih umum.</li> <li>Disimbolkan dengan anak panah yang digambarkan dari <i>use case</i> khusus ke <i>use case</i> umum.</li> </ul>

Sumber: Dennis dkk. (2015)

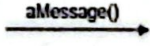
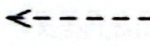
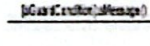
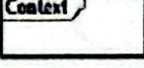

### 2.10.3 *Sequence Diagram*

*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. Berikut simbol yang ada pada *Sequence Diagram* dapat dilihat pada Tabel II.5 (Dennis dkk. 2015):

Tabel II.6 Simbol-Simbol *Sequence Diagram*

Simbol	Nama	Deskripsi
	Aktor	<ul style="list-style-type: none"> <li>• Seseorang atau sistem yang mendapat manfaat dan bersifat eksternal dari sistem.</li> <li>• Berpartisipasi dalam urutan dengan mengirim dan / atau menerima pesan.</li> <li>• Digambarkan baik sebagai stik (<i>default</i>) atau, jika aktor bukan manusia terlibat, sebagai persegi panjang dengan &lt;&lt;actor&gt;&gt; di dalamnya (alternatif).</li> </ul>
	Objek	<ul style="list-style-type: none"> <li>• Berpartisipasi secara berurutan oleh mengirim dan/atau menerima pesan.</li> <li>• Ditempatkan dibagian atas diagram.</li> </ul>
	<i>Lifeline</i>	<ul style="list-style-type: none"> <li>• Menunjukkan kehidupan sebuah objek selama <i>Sequence</i>.</li> <li>• Berisi sebuah X pada titik dimana kelas tidak lagi berinteraksi</li> </ul>
	<i>Execution occurrence</i>	<ul style="list-style-type: none"> <li>• Persegi panjang sempit yang ditempatkan di atas garis hidup.</li> <li>• Menunjukkan ketika suatu objek mengirim atau menerima pesan.</li> </ul>




Tabel II.6 Simbol-Simbol *Sequence Diagram* (lanjutan)

Simbol	Nama	Deskripsi
	<i>Message</i>	<ul style="list-style-type: none"> <li>Menyampaikan informasi dari satu objek ke objek lainnya.</li> </ul>
	<i>Message (return)</i>	<ul style="list-style-type: none"> <li>Pesan yang dikirim untuk diri sendiri</li> </ul>
	<i>Guard condition</i>	<ul style="list-style-type: none"> <li>Merupakan tes yang harus dipenuhi untuk pesan yang akan dikirim.</li> </ul>
	<i>Frame</i>	<ul style="list-style-type: none"> <li>Menunjukkan konteks dalam <i>sequence diagram</i></li> </ul>
	<i>Object destruction</i>	<ul style="list-style-type: none"> <li>Merupakan sebuah X ditempatkan pada akhir suatu garis hidup untuk menunjukkan bahwa itu akan keluar dari eksistensi.</li> </ul>

Sumber: Dennis dkk. (2015)

Selain simbol-simbol diatas *sequence diagram* memiliki *class* dengan fungsi yang masing-masing berbeda, berikut merupakan *class* yang terdapat dalam *sequence diagram* (Richardson & Thies, 2013):

Tabel II.7 Simbol Macam-macam *Class* pada *Sequence Diagram*

Simbol	Deskripsi
<p><i>Boundary Class</i></p> 	<i>Boundary Class</i> bertanggung jawab terhadap penanganan interaksi antara aktor dengan sistem.
<p><i>Entity Class</i></p> 	<i>Entity Class</i> merupakan simbol penyimpanan, objek yang dihasilkan sebagian besar berupa data dalam sistem.
<p><i>Control Class</i></p> 	<i>Control Class</i> merupakan koordinator dari sistem, setidaknya harus terdapat satu <i>control class</i> dalam setiap <i>use case</i> .


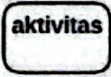



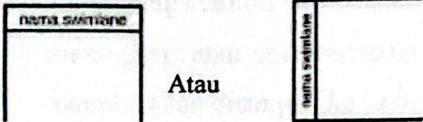
Sumber: Richardson & Thies (2013)

#### 2.10.4 *Activity Diagram*

Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. 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/ *userinterface* 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 ditampilkan pada perangkat lunak.

Tabel II.8 Simbol-Simbol *Activity Diagram*

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

Sumber: Rosa &amp; Shalahuddin (2018)

#### 2.10.4 Class Diagram

Diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

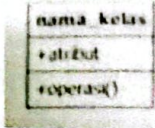






1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Susunan struktur kelas yang baik pada diagram kelas memiliki jenis-jenis kelas berikut:

1. Kelas Main, kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan
2. Kelas yang menangani tampilan sistem (*view*), kelas yang mendefinisikan dan mengatur tampilan ke pemakai
3. Kelas yang diambil dari pendefinisian *use case* (*controller*), kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.
4. Kelas yang diambil dari pendefinisian data (*model*), kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

(Rosa & Shalahuddin, 2018)

Tabel II.9 Simbol-Simbol *Class Diagram*

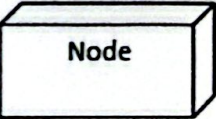

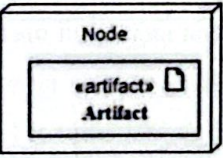
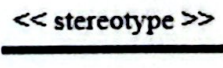
Simbol	Nama	Deskripsi
	Kelas	Kelas pada struktur sistem
	<i>Interface/</i> antarmuka	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
	Asosiasi	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
	Asosiasi berarah	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
	Generalisasi	Relasi antarkelas dengan makna generalisasi spesialisasi (umum khusus).
	Kebergantungan/ <i>dependency</i>	Relasi antarkelas dengan makna kebergantungan antarkelas.
	Agregasi/ <i>aggregation</i>	Relasi antarkelas dengan makna semua bagian.

Sumber: Rosa &amp; Shalahuddin (2018)

### 2.10.5 Deployment Diagram

*Deployment diagram* digunakan untuk mewakili hubungan antara komponen-komponen *hardware* yang digunakan dalam infrastruktur fisik sistem informasi. Misalnya, ketika membuat suatu sistem informasi terdistribusi yang akan menggunakan jaringan luas, *deployment diagram* dapat digunakan untuk menunjukkan hubungan komunikasi antara node yang berbeda dalam jaringan. *Deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. Dalam hal ini, *deployment diagram* mewakili lingkungan pembuatan *software*. Berikut simbol-simbol yang digunakan pada *deployment diagram*. (Dennis dkk. 2015):

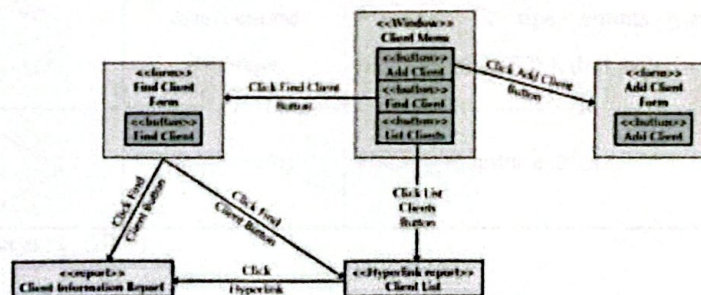
Tabel II.10 Simbol-Simbol Deployment Diagram

Simbol	Nama	Deskripsi
	<i>Node</i>	<ul style="list-style-type: none"> <li>Merupakan sumber daya komputasi, contoh: komputer klien, <i>server</i>, jaringan terpisah, atau perangkat jaringan individu.</li> </ul>
	<i>Artifact</i>	<ul style="list-style-type: none"> <li>Untuk menggambarkan spesifikasi dari <i>software</i> atau <i>database</i>, misalnya <i>file</i> sumber, tabel <i>database</i>, <i>executable file</i>.</li> </ul>
	<i>Node with a Deployed Artifact</i>	<ul style="list-style-type: none"> <li>Untuk menggambarkan <i>artifact</i> yang ditempatkan pada node fisik. Mendukung pemodelan distribusi perangkat lunak melalui jaringan.</li> </ul>
	<i>Communication Path</i>	<ul style="list-style-type: none"> <li>Untuk menggambarkan hubungan antara dua node untuk bertukar pesan.</li> </ul>

Sumber: Dennis dkk. (2015)

### 2.10.6 Windows Navigation Diagram (WND)

Menurut Dennis dkk.(2015) *Windows Navigation Diagram* merupakan struktur navigasi yang menentukan cara kerja masing-masing *interface* untuk menyediakan fungsionalitas pengguna. *Windows Navigation Diagram* digunakan untuk menunjukkan bagaimana semua *interface*, *form*, dan *report* yang digunakan oleh sistem terkait dan bagaimana pengguna berpindah dari satu *interface* ke *interface* yang lain. Contoh gambar *windows navigation diagram* dapat dilihat pada Gambar II.5.

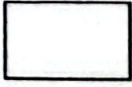


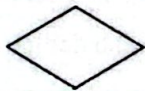


Gambar II.5 Contoh Windows Navigation Diagram  
Sumber: (Dennis dkk. (2015))

### 2.11 Entity Relationship Diagram (ERD)

Terdapat sebuah model perancangan hubungan antar entitas (tabel) dari sebuah basis data (*database*). Istilah untuk frase ini biasa dikenal dengan nama *Entity Relationship Model*. Model hubungan ini seterusnya akan berlanjut menjadi sebuah diagram hubungan antar entitas yang biasa dikenal dengan nama *Entity Relationship Diagram* (Hoffer et al., 2007). Simbol-simbol ERD dapat dilihat secara rinci pada Tabel II.11.

Tabel II.11 Simbol-Simbol *Entity Relationship Diagram*

Simbol	Nama	Deskripsi
	<i>Strong Entity</i>	Setiap hal dunia nyata (orang, tempat, objek, konsep, aktivitas) tentang suatu perusahaan mencatat data
	<i>Attribute</i>	Properti atau karakteristik tipe entitas.
	<i>Multivalued Attribute</i>	Karakteristik tipe entitas yang dapat memiliki nilai lebih dari satu.
	<i>Relationship</i>	Hubungan antar entitas.

Sumber: Hoffer et al., (2007)

### 2.11.1 Kardinalitas

*Relationship* mempunyai tiga tipe kardinalitas yang mana tiap tipe menunjukkan jumlah *record* dari setiap tabel yang direlasikan ke *record* pada tabel lain. Ketiga tipe tersebut adalah sebagai berikut (Nugroho, 2017):

1. Hubungan satu ke satu (*One to one relationship*)

Hubungan antara file pertama dan file kedua berbanding satu. Dalam hubungan ini, tiap *record* dalam tabel A hanya memiliki satu *record* yang cocok dalam tabel B dan tiap *record* dalam tabel B hanya memiliki satu *record* dalam tabel A. Logika penalaran matematika dari *one to one relationship* adalah pemetaan dengan “perkawan satu-satu” atau sering disebut dengan korespondensi satu-satu. Contoh *one to one relationship* adalah satu pasien mempunyai satu tempat tidur.

2. Hubungan satu ke banyak (*One to many relationship*)

Hubungan antara file pertama dengan file kedua adalah satu berbanding banyak. Dalam hubungan ini tiap *record* dalam tabel A memiliki beberapa *record* yang

cocok dalam tabel B. Logika penalaran matematika dari *one to many relationship* adalah “Perkawanan satu ke banyak”. Contoh *one to many relationship* adalah satu dosen mengajar banyak mata kuliah.

3. Hubungan banyak ke banyak (*Many to many relationship*)

Hubungan antara file pertama dengan file kedua adalah banyak berbanding banyak. Dalam hubungan ini tiap *record* dalam tabel A memiliki beberapa *record* yang cocok dalam tabel B dan tiap *record* dalam tabel B hanya memiliki satu *record* yang cocok dalam tabel A. Logika penalaran matematika dari *many to many relationship* adalah pemetaan “Perkawanan banyak ke banyak”. Contoh *many to many relationship* adalah banyak mahasiswa memiliki banyak mata kuliah dan banyak mata kuliah memiliki banyak mahasiswa. Hubungan *many to many* tidak dapat diimplementasikan ke dalam *database relationship* sehingga hubungan ini harus dipecah menjadi hubungan *one to many*.

## 2.12 Kamus Data

Kamus Data merupakan katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi (Hartono, 2010). Berdasarkan definisi tersebut, maka dapat disimpulkan bahwa kamus data merupakan suatu bantuan yang berguna untuk kebutuhan informasi dari suatu sistem informasi.

Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi:

1. Nama : nama dari data.
2. Digunakan pada : merupakan proses-proses yang terkait data.
3. Deskripsi : merupakan deskripsi data.
4. Informasi tambahan : seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

Tabel II.12 Contoh Kamus Data

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

Sumber: (Hartono, 2010)

### 2.13 Hypertext Preprocessor (PHP)

PHP yaitu Bahasa pemrograman *web-server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan *Hypertext Markup Language (HTML)* dan berada pada *server (serverside HTML embedded scripting)*. PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru. Semua *script* PHP dieksekusi pada *server* di mana *script* tersebut dijalankan. (Anhar, 2010)

### 2.14 XAMPP

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

Fungsi lainnya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X, Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU

*General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. (Kadir, 2014)

### 2.15 *Bootstrap*

*Bootstrap* adalah paket aplikasi siap pakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, *bootstrap* adalah template desain web dengan fitur *plus*. *Bootstrap* diciptakan untuk mempermudah proses desain web bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman.

Paket *bootstrap* berisi sekumpulan *file Cascading Style Sheets (CSS)*, *font*, dan *JavaScript* yang siap diintegrasikan ke sebuah dokumen *Hypertext Markup Language (HTML)* menggunakan kaidah-kaidah tertentu. Dokumen HTML yang dihasilkan pun secara dinamis akan tampil dalam layout yang disesuaikan dengan layar piranti pengunjung.

*Bootstrap* merupakan proyek *open source* dan tentunya pemakaiannya bersifat gratis. Saat ini, *bootstrap* menjadi aplikasi paling populer di situs layanan *hosting software open source* GitHub.com.

### 2.16 *Framework*

*Framework* adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class* dan *function*) yang dapat membantu *developer* dalam menangani berbagai masalah-masalah dalam pemrograman, seperti koneksi ke *database*, pemanggilan *variable*, *file* sehingga pekerjaan *developer* lebih fokus dan lebih cepat dalam membangun aplikasi. *Framework* adalah komponen pemrograman yang siap digunakan ulang kapan saja sehingga programmer tidak harus membuat *script* yang sama untuk tugas yang sama. (Yudhanto dan Prasetyo, 2018)

### 2.17 MariaDB

MariaDB adalah *Database Management System* (DBMS) hasil *forking* dari DMBSMySQL. Jadi *syntaxquery* yang digunakan hampir sama. MariaDB menerapkan lisensi bebas GNU/GPL. Nama MariaDB diturunkan dari nama putri ketiga Michael “Monty” Widenius, pendiri Monty Program dan MySQL. MariaDB memiliki banyak fitur yang sejak lama telah dipelihara komunitas sehingga menurut Monty seharusnya sangat stabil.

MariaDB telah mengganti InnoDB dengan XtraDB yang juga telah mendapat perbaikan dari Google dan Percona. ExtraDB diklaim secara signifikan bisa jalan lebih cepat dibandingkan InnoDB, disamping ia juga bisa digunakan tanpa perlu memasang Plugin seperti halnya pada MySQL. (Putra, 2018)

### 2.18 Black Box Testing

Menurut Sukanto & Shalahuddin (2011) *Black Box Testing*, yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian ini dimaksud untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian *Black Box* dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Pengujian *Black Box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses *database* eksternal
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

## BAB III

# METODOLOGI PENELITIAN

### 3.1 Metodologi Penelitian

Menurut istilah “Metodologi” berasal dari Bahasa Yunani yakni *metodhos* dan *logos*, *metodhos* berarti cara, kiat, dan seluk beluk yang berkaitan dengan upaya menyelesaikan sesuatu, sementara *logos* berarti ilmu pengetahuan, cakrawala, dan wawasan. Dengan sedemikian metodologi adalah metode atau cara-cara yang berlaku dalam kajian atau penelitian. Metodologi penelitian adalah suatu cabang ilmu pengetahuan yang membicarakan mengenai cara-cara melaksanakan penelitian (meliputi kegiatan-kegiatan mencari, mencatat, merumuskan, menganalisis, sampai menyusun laporannya) berdasarkan fakta-fakta secara ilmiah. (Fitrah dan Luthfiyah, 2017)

### 3.2 Jenis dan Sumber Data

Data yang diperlukan dalam penelitian ini ada dua jenis, yaitu data primer dan data sekunder. Sumber dari data-data ini berasal dari tempat yang diamati pada praktek kerja lapangan di PT Mitsubishi Krama Yudha Motors and Manufacturing:

#### 1. Data Primer

Data Primer adalah data yang diperoleh secara langsung dari PT Mitsubishi Krama Yudha Motors and Manufacturing, dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data-data tersebut adalah data yang digunakan dalam proses *monitoring* hasil produksi diantaranya, analisis dokumen yang 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

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 pada Bagian Produksi di PT Mitsubishi Krama Yudha Motors and Manufacturing dengan mengamati proses pengisian *checksheet*.
- b. Wawancara, yaitu mencari data yang dibutuhkan secara langsung dengan memberikan pertanyaan yang ingin ditanyakan terhadap segala hal yang diperlukan pada penyusunan Tugas Akhir ini. Wawancara ini dilakukan kepada Bagian Produksi khususnya *machining operator* di PT Mitsubishi Krama Yudha Motors and Manufacturing.
- c. Analisis Dokumen  
Menganalisis dokumen-dokumen yang berkaitan dengan proses pengisian dan penyimpanan *checksheet*, *list part reject*, dan *list part pending* pada PT Mitsubishi Krama Yudha Motors and Manufacturing.

#### 2. Studi Kepustakaan

Studi kepustakaan dilakukan dengan membaca buku dan literatur dalam lingkup perkuliahan maupun di luar lingkup perkuliahan yang berhubungan dengan judul dan permasalahan, sehingga dapat menunjang dalam penulisan tugas akhir ini.

Studi kepustakaan yang dilakukan adalah dengan menggunakan buku yang dimiliki, buku yang dipinjam dari perpustakaan dan mencari data yang diperlukan melalui internet.

### 3.4 Metodologi Pengembangan Sistem

Metode pengembangan sistem informasi *motoring* hasil produksi ini menggunakan metode pengembangan sistem *waterfall* dikarenakan persyaratan sudah dipahami dengan baik pada awal pengembangan sehingga tidak ada perubahan saat proses pengembangan. Metode *waterfall* ini terdiri dari tahap *planning*, *analysis*, *design*, *implementation*, dan *system*. Berikut ini akan dijelaskan secara singkat mengenai tahapan-tahapan yang dilakukan dalam pengembangan sistem dengan metode *waterfall*.

#### 1. Tahap Perencanaan (*Planning*)

Tahap perencanaan adalah proses dasar untuk memahami mengapa sistem informasi harus dibangun dan menentukan bagaimana akan membangunnya.

#### 2. Tahap Analisis (*Analysis*)

Tahap analisis adalah untuk memahami dan mendokumentasikan kebutuhan bisnis dan persyaratan dari sistem baru. Menganalisa kebutuhan sebagai bahan dalam membuat spesifikasi di tahapan selanjutnya.

#### 3. Tahap Desain (*Design*)

Membuat pemodelan sistem dengan *Unified Modeling Language* (UML) yaitu menggunakan *usecase diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *windows navigation diagram*. Data-data yang didapat dari tahap analisis akan diterapkan dalam tahap desain.

#### 4. Tahap Implementasi (*Implementation*)

Mengimplementasikan perancangan sistem dengan pemilihan perangkat keras dan penyusunan perangkat lunak aplikasi (pengkodean). Bahasa pemrograman yang digunakan adalah PHP dan *database* MariaDB.

#### 5. Sistem (*System*)

Di tahap sistem ini dilakukan pengujian yang dilakukan untuk mengetahui apakah *software* yang dibuat telah sesuai dengan desainnya.

### 3.5 Kerangka Penelitian

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

#### 1. Studi Pendahuluan

Studi Pendahuluan merupakan langkah awal yang dilakukan untuk memulai penelitian. Studi pendahuluan dilakukan dengan turun langsung ke lokasi untuk mengetahui gambaran yang jelas mengenai sistem yang sedang berjalan pada Bagian Produksi. Studi dapat dilakukan dengan melakukan observasi, wawancara, dan studi pustaka.

#### 2. Identifikasi Masalah

Pada Tahap ini, peneliti menganalisis proses proses pengisian *checksheet*, *list part pending*, dan *list part reject* yang sedang berjalan guna *memonitoring* hasil produksi, mengidentifikasi permasalahan yang ada pada proses bisnis yang sedang berjalan dengan melakukan wawancara dan observasi.

#### 3. Identifikasi Solusi

Pada tahap ini dilakukan penetapan identifikasi tujuan penelitian dan batasan masalah penelitian.

#### 4. Batasan Masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya. Berikut batasan dalam penulisan tugas akhir ini:

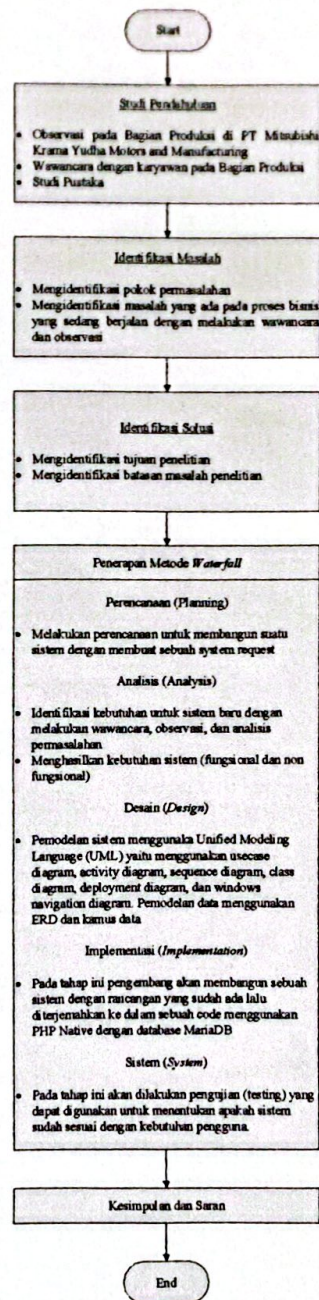
- a. Penelitian ini dilakukan selama dua bulan yang bertempat di PT Mitsubishi Krama Yudha Motors and Manufacturing tepatnya di Jalan Raya Bekasi KM 21, Kel. Rawa Terate, Kec. Cakung, Jakarta Timur, 13920.
  - b. Departemen atau bagian yang diteliti adalah Departemen Produksi pada seksi *Machining* khususnya *Transmission TD Machining*.
  - c. Selama proses penelitian dilakukan pengamatan pada pengisian dan pengarsipan checksheet, list part pending, dan list part reject.
  - d. Sistem informasi monitoring hasil produksi ini dapat membantu untuk mengurangi terbuangnya waktu sehingga *machining operator* bisa tetap menjalankan mesin produksi.
5. Penerapan Metode *Waterfall*
- a. Perencanaan (*Planning*)  
Tahap perencanaan adalah proses dasar untuk memahami mengapa sistem informasi harus dibangun dan menentukan bagaimana akan membangunnya.
  - b. Melakukan Analisis (*Analysis*)  
Tahap analisis adalah untuk memahami dan mendokumentasikan kebutuhan bisnis dan persyaratan dari sistem baru. Menganalisa kebutuhan sebagai bahan dalam membuat spesifikasi di tahapan selanjutnya.
  - c. Membuat Desain (*Design*)  
Membuat pemodelan sistem dengan *Unified Modeling Language* (UML) yaitu menggunakan *usecase diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *windows navigation diagram*. Membuat pemodelan data dengan *Entity Relationship Diagram* (ERD) dan kamus data serta data-data yang didapat dari tahap analisis akan diterapkan dalam tahap desain.
  - d. Implementasi (*Implementation*)  
Mengimplementasikan perancangan sistem dengan pemilihan perangkat keras dan penyusunan perangkat lunak aplikasi (pengkodean). Bahasa pemrograman yang digunakan adalah PHP dan *database* MariaDB.

e. **Sistem (*System*)**

Pada tahapan sistem ini dilakukan pengujian (*testing*) untuk menentukan apakah sistem atau perangkat lunak yang dibuat sudah sesuai dengan kebutuhan pengguna atau belum.

6. **Kesimpulan dan Saran**

Kesimpulan digunakan untuk membandingkan hasil penelitian atau pengembangan sistem dengan sistem sebelumnya. Sedangkan saran berisi masukan untuk pengembangan sistem selanjutnya. Berikut merupakan diagram alir kerangka penelitian yang digunakan dalam laporan tugas akhir ini dapat dilihat pada Gambar III.1.



Gambar III.1 Kerangka Penelitian  
Sumber: Pengolahan Data (2020)

## BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

### 4.1 Sejarah Perusahaan

PT Mitsubishi Krama Yudha Motors and Manufacturing merupakan perusahaan manufaktur yang bergerak di bidang otomotif. Perusahaan tersebut memproduksi komponen mesin kendaraan *Colt Diesel* dan *Fuso Truck*.



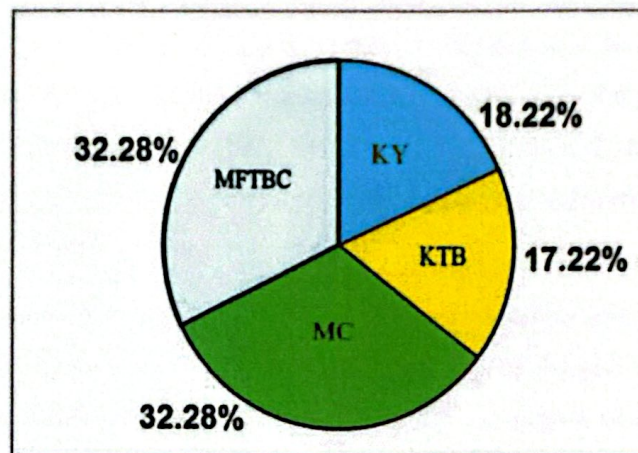
**PT MITSUBISHI KRAMA YUDHA MOTORS AND MANUFACTURING**

Gambar IV.1 Logo PT MKM  
Sumber: PT MKM (2019)

Pada tanggal 18 Januari 1973 terjadi persetujuan usaha patungan (*Joint Venture*) antara PT Krama Yudha, Mitsubishi Corporation, dan Mitsubishi Motor Corporation yang kemudian pada tanggal 19 Mei 1973 didirikan PT Mitsubishi Krama Yudha Motors and Manufacturing (MKM) yang mempunyai modal dasar \$ 42.866.250 dan modal di setor sebesar \$ 42.866.250. Pada tanggal 3 Agustus 1973 bertempat di Jakarta berdasarkan Akte Notaris Eliza Pondang No 17, PT MKM secara resmi berdiri.

Akte pendirian beserta perubahannya telah disetujui oleh Menteri Kehakiman dengan surat No. Y.A.5/362/19 tanggal 11 Juni 1981 dan dimuat dalam tambahan berita Negara No. 13 tanggal 6 Desember 1981, tambahan No. 1029/1981. PT MKM ini merupakan perusahaan dengan penanaman modal asing (PMA) yang bekerja sama dengan Jepang dengan pemegang sahamnya terdiri dari:

- |  |          |
|--|----------|
| 1. PT Krama Yudha, Indonesia                         | : 18,22% |
| 2. PT Krama Yudha Tiga Berlian Motors, Indonesia     | : 17,22% |
| 3. Mitsubishi Corporation, Jepang                    | : 32,28% |
| 4. Mitsubishi Fuso Truck and Bus Corporation, Jepang | : 32,28% |



Gambar IV.2 *Shareholder* PT MKM  
Sumber: PT MKM (2019)

Tahap-tahap pengembangan diawali dengan pembangunan mesin pabrik dimulai tanggal 14 Januari dan selesai dalam waktu empat bulan yang dilanjutkan dengan pemasukan mesin-mesin dan peralatan pada bulan Mei. Produksi percobaan dimulai pada pertengahan bulan Oktober 1974 yang berlangsung untuk beberapa bulan. Sedangkan, produksi komersial dimulai pada tanggal 6 Januari 1975.

Pada bulan Desember 1982 didirikan PT *Colt Engine and Manufacturing* yang merupakan usaha patungan antara Indonesia dan Jepang yaitu PT Mitsubishi Corporation dan Mitsubishi Motors Corporation. PT *Colt Engine and Manufacturing* mulai berproduksi secara komersial pada tahun 1985.

Pada tanggal 1 Januari 1988 PT MKM melakukan *merger* dengan PT *Colt Engine and Manufacturing* yang kemudian disebut PT MITSUBISHI KRAMA

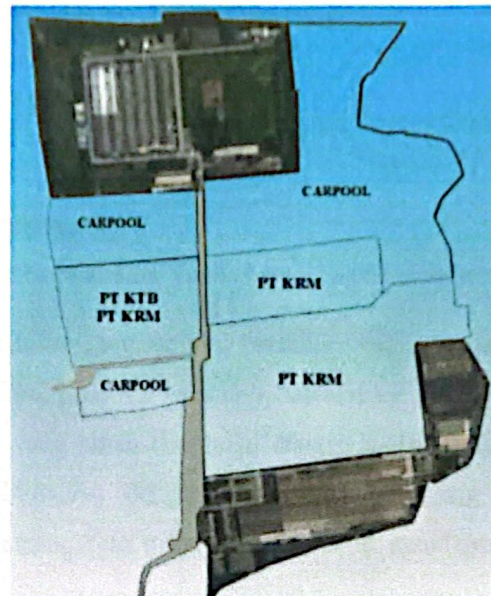
YUDHA MOTORS AND MANUFACTURING. Alasan dilakukan *merger* antara lain untuk efisiensi. Pada saat *merger* perbandingan saham antara Indonesia dan Jepang sebesar 34,5% dan 64,6%. MKM 1 (*Stamping Factory*) berdiri di atas tanah seluas 45.970 m<sup>2</sup> dengan luas bangunan 19.703m<sup>2</sup>.

MKM 1 (*Stamping Factory*) bergerak dalam memproduksi komponen badan kendaraan, sedangkan MKM 2 (*Engine Factory*) memproduksi komponen mesin kendaraan. Setelah dapat membuat komponen mesin seperti *Crankshaft*, *Connecting Rod*, *Cylinder Head*, dan *Camshaft*. Pada tahun 1997 MKM 2 mulai membuat komponen yang lain, yaitu: *Transmission Case* dan *Extension Housing* untuk jenis kendaraan *Colt L 300* dan Kuda.

Pada tahun 1988, dimulai ekspor komponen mesin antara lain *Cylinder Head*, *Crankshaft*, dan *Connecting Rod* ke Jepang (MMC Kytoo). Kemudian, pada tahun 1999 mengexport komponen *body* ke Philipina (MMPC). Dan pada tahun 2000 export dilakukan untuk transmisi ke Philipina (ATC).

#### 4.2 Profil Perusahaan

PT MKM berlokasi di Jalan Raya Bekasi KM 21, Kel. Rawa Terate, Kec. Cakung, Jakarta Timur, 13920. Perusahaan tersebut memiliki luas tanah 86.465m<sup>2</sup> dengan luas bangunan 19.548m<sup>2</sup> untuk *Factory Visit (Engine Factory)* dan luas tanah 45.970m<sup>2</sup> dengan luas bangunan 19.703m<sup>2</sup> untuk *Factory Visit (Stamping Factory)* (dapat dilihat pada Gambar IV.3).



Gambar IV.3 Luas PT MKM  
Sumber: PT MKM (2019)

Adapun profil perusahaan lebih rinci yaitu sebagai berikut:

<b>Nama Perusahaan</b>	: PT Mitsubishi Krama Yudha Motors And Manufacturing
<b>Alamat</b>	: Jalan Raya Bekasi KM 21, Kel. Rawa Terate, Kec. Cakung, Jakarta Timur, 13920.
<b>Telepon</b>	: (021) 4602908
<b>Fax</b>	: (021) 4602915
<b>Jenis Usaha</b>	: Produsen Komponen Otomotif
<b>Produk</b>	: Komponen Kendaraan
<b>Status</b>	: Perseroan Terbatas
<b>Pimpinan Perusahaan</b>	: T. Hirotso ( <i>President Director</i> ) K. Akimoto ( <i>Vice President Director</i> ) Patoni J. ( <i>Director GA</i> ) R. Takoda ( <i>Director Controlling/ IT CFO</i> )

Tony S (*Director PPC*)

S. Yamada (*Director QE*)

R. Nagoshi (*P/ E Engine, T/ M Maintenance*)

#### 4.3 Visi dan Misi Perusahaan

Visi dari PT Mitsubishi Krama Yudha Motors and Manufacturing, yaitu:

1. Menjadikan perusahaan yang mampu bersaing dengan perusahaan sejenis di Indonesia dengan hasil produksi yang baik dan berkualitas tinggi.
2. Mengelola pabrik yang aman dan maju dengan melaksanakan *control QCD (Quality, Cost, Delivery)* dengan mempunyai tanggung jawab terhadap lingkungan dan menempatkan prioritas utama untuk mendapatkan kepercayaan konsumen.
3. Meningkatkan kepuasan kepada pemilik saham, pemerintah, direktur, dan seluruh karyawan.

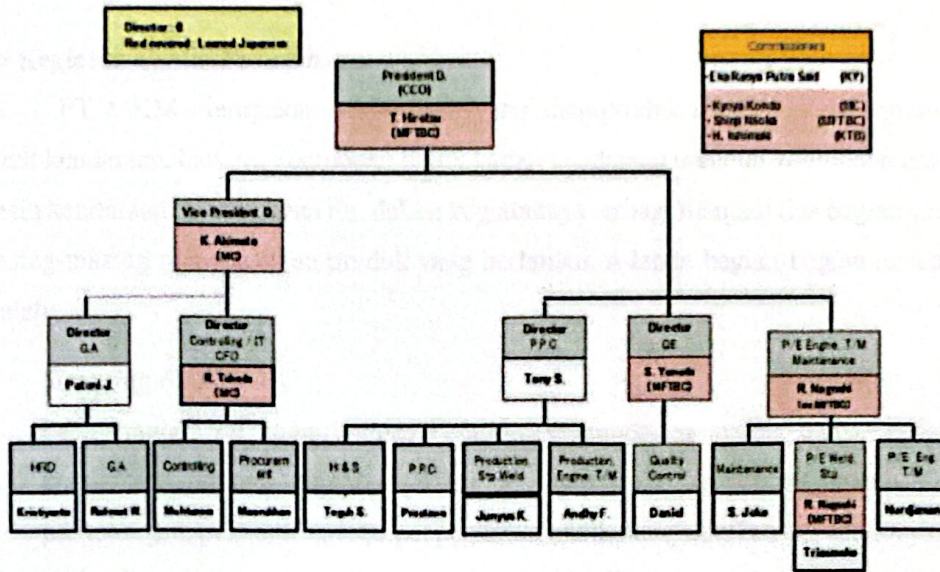
Misi PT Mitsubishi Krama Yudha Motors and Manufacturing, yaitu:

1. Penurunan biaya.
2. Peningkatan kualitas.
3. Pengawasan terhadap jadwal pengiriman.
4. Mengadakan persiapan yang lancar dan baik untuk produk model baru.
5. Peningkatan dalam bidang manajemen, keselamatan, dan lingkungan.

#### 4.4 Struktur Organisasi

Struktur organisasi merupakan kerjasama untuk mencapai tujuan organisasi yang bersangkutan. Suatu bagan organisasi dapat menggambarkan pembagian tugas dan fungsi yang ada dalam perusahaan PT MKM. Pimpinan tertinggi pada PT MKM adalah seorang Presiden Direktur, di dalam menjalankan tugasnya dibantu oleh 2 orang Wakil Presiden Direktur.

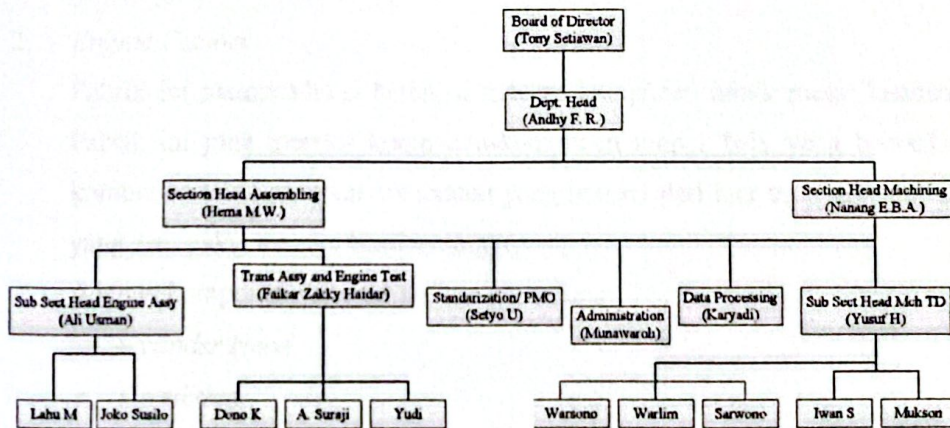
Berikut struktur organisasi PT MKM (dapat dilihat pada Gambar IV.4)



Gambar IV.4 Struktur Organisasi PT MKM  
Sumber: PT MKM (2019)

#### 4.5 Struktur Organisasi Departemen Produksi

Berikut struktur organisasi departemen produksi (dapat dilihat pada Gambar IV.5)



Gambar IV.5 Struktur Organisasi Departemen Produksi  
Sumber: PT MKM (2019)

#### 4.6 Kegiatan Usaha Perusahaan

PT MKM merupakan perusahaan yang memproduksi komponen-komponen untuk kendaraan, baik itu komponen untuk badan kendaraan maupun komponen untuk mesin kendaraan. Oleh karena itu, dalam kegiatannya terbagi menjadi dua bagian yang masing-masing menghasilkan produk yang berlainan. Adapun bagian-bagian tersebut adalah:

##### 1. *Stamping Factory*

Pada bagian ini memproduksi komponen-komponen untuk badan rangka kendaraan, dimana kegiatannya adalah mengelola lempengan logam dari pemotongan, pengepresan sampai perakitan untuk menghasilkan kendaraan yang berkualitas tinggi. Adapun komponen yang dihasilkan antara lain:

- a. *Body Kendaraan*
- b. *Chasing*
- c. *Steering*
- d. *Fuel tank*
- e. *Exhaust Pipe*
- f. *Muffler*

##### 2. *Engine Factory*

Pabrik ini memproduksi berbagai macam komponen untuk mesin kendaraan. Pabrik ini juga merakit komponen-komponen mesin, baik yang berasal dari komponen lokal maupun komponen yang berasal dari luar menjadi satu mesin yang siap pakai dengan kualitas tinggi.

Adapun komponen yang dihasilkan antara lain:

- a. *Cylinder Head*
- b. *Crankshaft*
- c. *Cylinder Block*
- d. *Connection Rod*

#### 4.8 Produk Yang Dihasilkan

Berikut produk yang dihasilkan oleh PT MKM dapat dilihat pada Tabel III.

No	Customer Dalam Negeri	Customer Luar Negeri
1	PT Krama Yudha Tiga Berlian	Philipina
2	PT Krama Yudha Ratu Motor	Jepang

Sumber: PT MKM (2019)

Tabel IV.1 Daftar Customer Dalam Negeri dan Luar Negeri

MKM dapat dilihat pada Tabel III.1.







PT MKM mendistribusikan produknya untuk *customer* dalam negeri dan untuk *customer* luar negeri. Untuk *customer* dalam negeri, dan luar negeri yang dimiliki PT

#### 4.7 Pemasaran dan Sistem Distribusi



Dalam menjalankan kegiatan produksinya, PT MKM memproduksi komponen-komponen tersebut berdasarkan jumlah pesanan yang diterima dari distributor tunggal kendaraan Mitsubishi di Indonesia, yang dipegang oleh PT Krama Yudha Tiga Berlian (KTB) yang didirikan pada tanggal 27 April 1973. Produk yang dihasilkan oleh PT MKM didistribusikan ke PT Krama Yudha Ratu Motor (KRM) untuk jenis T120SS, L30 dan Truk. PT KRM sendiri didirikan pada tanggal 2 Juni 1973. Ketiga perusahaan tersebut (PT KTB, PT MKM, PT KRM) tergabung ke dalam Krama Yudha Group, yang dalam menjalankan usahanya saling berkaitan dan saling membantu antara perusahaan yang satu dengan perusahaan yang lainnya.

- e. *Camshaft*
- f. *Transmission Case*
- g. *Extension Housing*

Tabel III.2 Produk Yang Dihasilkan PT MKM

No	Nama Produk	Gambar Produk
1	<i>Cylinder Head</i>	
2	<i>Crankshaft</i>	
3	<i>Cylinder Block</i>	
4	<i>Connection Rod</i>	
5	<i>Camshaft</i>	
6	<i>Transmission Case</i>	

Tabel III.2 Produk Yang Dihasilkan PT MKM (lanjutan)

No	Nama Produk	Gambar Produk
7	<i>Transmission TD</i>	
8	<i>Clutch Housing</i>	

Sumber: PT MKM (2019)

#### 4.9 Dokumen Terkait Pada Prosedur Pengendalian Kualitas

Terdapat dokumen yang terkait dalam prosedur pengendalian kualitas di *line* produksi *Clutch Housing* yang diantaranya:

##### 1. *Checksheet*

Terdapat dokumen yang terkait dalam proses pengendalian kualitas di *line* produksi *Clutch Housing* yang diantaranya adalah *Checksheet*. *Checksheet* merupakan lembar data yang digunakan sebagai *quality control part* dari produksi untuk menghindari adanya part rusak sampai terkirim ke *assy line*. Pengisian *checksheet* dilakukan setiap hari dan sesuai dengan ketentuan *interval* pada setiap mesin. *Checksheet* berisi spesifikasi *part* yang harus diukur sesuai dengan spesifikasi yang telah ditentukan pada lembar pengendalian kualitas, beserta alat ukurnya (dapat dilihat pada Gambar IV.6).







- a. *Shop* : Berisi nama *line shop*.
- b. *Model* : Berisi *model shopnya*.
- c. *Bulan* : Berisi bulan terbuatnya *list part reject*.
- d. *Tanggal* : Berisi tanggal dibuat *list part reject*.
- e. *OP.NO* : Berisi nomor mesin produksi mana yang terdapat *part reject*.
- f. *Tgl/Shift* : Berisi tanggal dan keterangan waktu kerja *operator*.
- g. *Qty Reject (man, PLN Off, M/C, Tool)* : Berisi keterangan *alasan reject proses*.
- h. *No. REM/ NCR (reject pending)* : Berisi No. NCR yang terdapat pada label *part reject*.
- i. *Paraf Person yang List Up* : Berisi paraf Assistant Foreman..
- j. *Problem* : Berisi deskripsi tentang *part reject*.
- k. *Penyebab* : Berisi penyebab terjadinya *part reject*.
- l. *Action* : Berisi tentang tindakan apa yang akan dilakukan.

### 3. *List Part Pending*

*List Reject* merupakan lembar data yang digunakan sebagai daftar produk yang mengalami *reject*. *List reject* dilakukan setiap hari sekalipun pada hari itu tidak ada *part reject*. *List* tersebut juga terbagi menjadi dua, yaitu:

- a. *List Part Pending M025*(dapat dilihat pada Gambar IV.9)
- b. *List Part Pending M035* (dapat dilihat pada Gambar IV.10)

### LIST - PART PENDING

GROUP	C/INTK PENDING	Date	Signature											
MODEL	M025			Su A. L. M. P. O.										
SIARAN	PERAWATAN	Target	01.03.2019											
PRODUCTION DEPT ENGINE														
DRAWING														
No	QTY	MATERIAL	QTY PENDING				NO. SKR	PROBLEM	PROGRESS	ACTION	JUDGMENT			
			REKAM	SIARAN	TRUK	DI BANGUN					QTY	PARAF	REVISI	INDO
							DESCRIBE							
TOTAL							NONE							
REVISI														
REVISI														
REVISI														

CS Scanned with CamScanner

Gambar IV. 9 Contoh List Part Pending M025  
 Sumber: PT MKM (2019)



- a. *Shop* : Berisi nama *line shop*.
- b. *Model* : Berisi *model shopnya*.
- c. *Bulan* : Berisi bulan terbuatnya *list part reject*.
- d. *Tanggal* : Berisi tanggal dibuat *list part reject*.
- e. *OP.NO* : Berisi nomor mesin produksi mana yang terdapat *part reject*.
- f. *Tgl/Shift* : Berisi tanggal dan keterangan waktu kerja *operator*.
- g. *Qty Reject (man, PLN Off, M/C, Tool)* : Berisi keterangan *alasan reject proses*.
- h. *No. REM/ NCR (reject pending)* : Berisi No. NCR yang terdapat pada label *part reject*.
- i. *Paraf Person yang List Up* : Berisi paraf Assistant Foreman..
- j. *Problem* : Berisi deskripsi tentang *part reject*.
- k. *Penyebab* : Berisi penyebab terjadinya *part reject*.
- l. *Action* : Berisi tentang tindakan apa yang akan dilakukan.

#### 4.11 Pengarsipan *Checksheet*

Pada *Line Transmission TD Machining tipe Clutch Housing*, dalam melaksanakan pengendalian kualitas produk masih dilakukan dengan bantuan *Checksheet*, sehingga untuk melakukan *update* tiap produk membutuhkan waktu lebih lama. Penyebabnya dikarenakan pengecekan produk harus sesuai dengan alat ukur yang digunakan, ukuran yang telah ditetapkan, dan pengisian *checksheets* yang masih manual.

Setelah *check sheet* diisi, laporan atas kualitas produk tersebut dilaporkan kepada *Foreman* dan diarsipkan ke dalam alat sortir sesuai dengan *checksheets* masing-

masing mesin (dapat dilihat pada Gambar IV.7). Setelah *checksheet* terkumpul, data tersebut dimasukkan ke dalam *ordner* (dapat dilihat pada Gambar IV.8)



Gambar IV.7 Tempat Pengarsipan Alat Sortir  
Sumber: PT MKM (2019)



Gambar IV.8 Tempat Pengarsipan *Ordner*  
Sumber: PT MKM (2019)

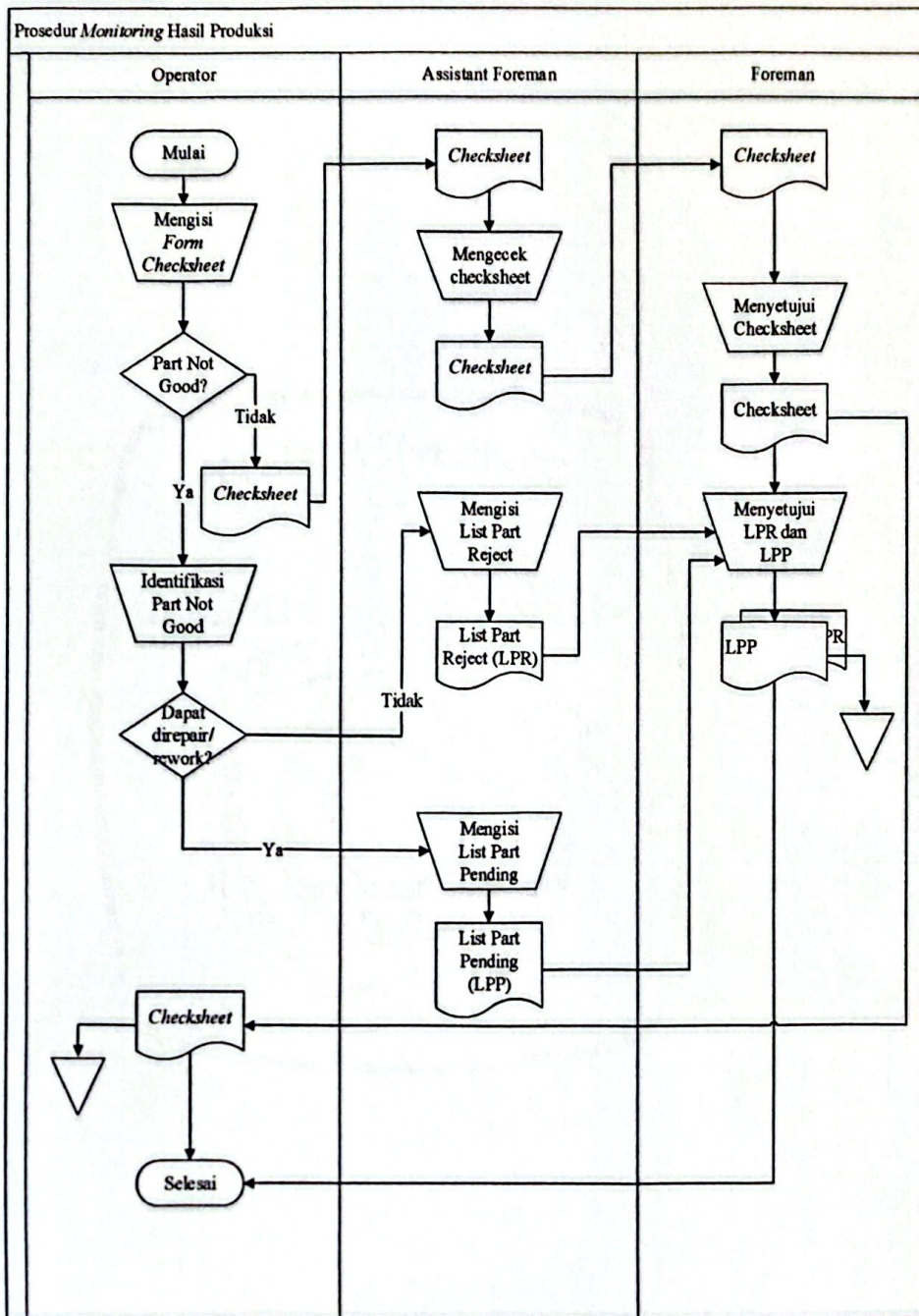
#### 4.12 Prosedur Proses Bisnis *Monitoring* Hasil Produksi yang Sedang Berjalan

Proses bisnis pada sistem pengendalian kualitas mesin *Clutch Housing* di PT MKM dijelaskan melalui *flowmap* (dapat dilihat pada Gambar IV.9) dengan beberapa departemen yang terlibat, diantaranya:

1. *Operator*
2. *Assistant Foreman*
3. *Foreman*

Berikut penjelasan singkat terkait Gambar IV.12 yaitu:

1. *Operator* mengisi *form checksheet (CS)*, lalu mengidentifikasi apakah ada *part Not Good (NG)*. Jika tidak, *checksheet* diberikan kepada *Ass.Foreman* untuk mengecek *checksheet* tersebut.
2. Setelah ditemukan adanya *part NG*, operator akan menganalisis untuk mengetahui *part* tersebut dapat *direpair/rework* atau tidak. Jika tidak bisa, *Ass.Foreman* akan mengisi *list part reject* dan diajukan kepada *Foreman* untuk disetujui.
3. Setelah dianalisis hasilnya dapat di *rework/repair*, *Ass.Foreman* akan mengisi *list part pending* dan diajukan kepada *Foreman* untuk disetujui.
4. Dokumen *list part reject* dan *list part pending* diarsipkan ke dalam ordner oleh *Foreman*.
5. Setelah *Ass. Foreman* mengecek *checksheet* tersebut, *checksheet* diberikan kepada *Foreman* untuk disetujui.
6. Setelah disetujui oleh *Foreman*, *checksheet* tersebut diberikan kembali kepada *Operator* untuk diarsipkan ke dalam ordner.



Gambar IV.9 *Flowmap* Monitoring Hasil Produksi  
 Sumber: Hasil Analisis (2020)

## BAB 5

### HASIL ANALISIS DAN PEMBAHASAN

#### 5.1 Analisis Kebutuhan Sistem Prosedur

Analisis kebutuhan sistem untuk proses bisnis *monitoring* hasil produksi dapat dilihat secara rinci pada Tabel V.1 dan Tabel V.2 sebagai berikut:

Tabel V.1 Analisis Kebutuhan Sistem Usulan

No	Permasalahan	Kebutuhan User	Solusi	Kebutuhan Sistem (Functional Requirement)
1	Pengisian dokumen <i>checksheet</i> masih dilakukan dengan mengisi di selembar kertas	Sistem yang dapat melakukan pengisian dokumen secara terkomputerisasi	Merancang dan membangun sistem pengisian dokumen yang sudah terkomputerisasi	Sistem memiliki fitur yang dapat <i>menginput</i> dokumen yang dibuat serta mengintegrasikannya
2	Penyerahan dokumen <i>checksheet</i> ke <i>Assistant Foreman</i> masih dilakukan dengan memberikannya secara langsung	Sistem otomasi mengenai penyerahan dokumen <i>checksheet</i> ke <i>Assistant Foreman</i>	Merancang dan membangun sistem otomasi penyerahan dokumen <i>checksheet</i>	Sistem memiliki fitur yang dapat menyampaikan atau mengirimkan informasi mengenai <i>checksheet</i> ke <i>Assistant Foreman</i>
3	Penyerahan dokumen <i>checksheet</i> ke <i>Foreman</i> masih dilakukan dengan memberikannya secara langsung	Sistem otomasi mengenai penyerahan dokumen <i>checksheet</i> ke <i>Foreman</i>	Merancang dan membangun sistem otomasi penyerahan dokumen <i>checksheet</i>	Sistem memiliki fitur yang dapat menyampaikan atau mengirimkan informasi mengenai <i>checksheet</i> ke <i>Foreman</i>

Tabel V.1 Analisis Kebutuhan Sistem Usulan (lanjutan)

No	Permasalahan	Kebutuhan User	Solusi	Kebutuhan Sistem (Functional Requirement)
4.	Pengarsipan dokumen <i>checksheet</i> masih dilakukan dengan menyimpan di ordner	Sistem yang dapat melakukan penyimpanan dokumen secara terkomputerisasi	Merancang dan membangun sistem pengisian dokumen yang sudah terkomputerisasi	Sistem memiliki fitur yang dapat menyimpan informasi yang dibuat serta mengintegrasikannya
5.	Pengisian dokumen <i>list part pending</i> dan <i>list part reject</i> dilakukan di selembar kertas.	Sistem yang dapat melakukan pengisian dokumen secara terkomputerisasi	Merancang dan membangun sistem pengisian dokumen yang sudah terkomputerisasi	Sistem memiliki fitur yang dapat meng-input dokumen yang dibuat serta mengintegrasikannya
6.	Pengarsipan dokumen <i>list part pending</i> dan <i>list part reject</i> masih dilakukan dengan menyimpan di ordner	Sistem yang dapat melakukan penyimpanan dokumen secara terkomputerisasi	Merancang dan membangun sistem pengisian dokumen yang sudah terkomputerisasi	Sistem memiliki fitur yang dapat menyimpan dokumen yang dibuat serta mengintegrasikannya

Sumber: Hasil Analisis (2020)

Tabel V.2 Kebutuhan Sistem (Non Functional Requirement)

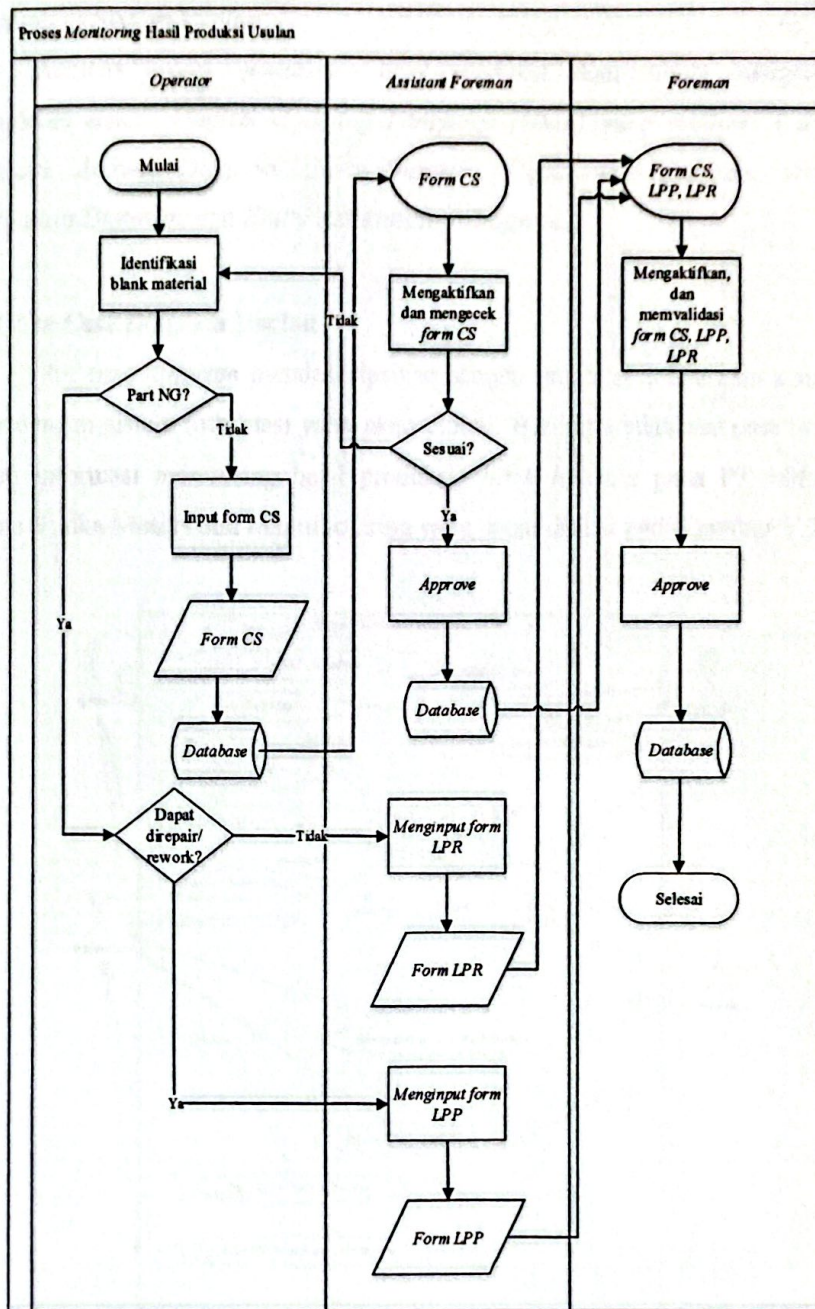
No	Kebutuhan Sistem (Non Functional Requirement)
1	Aplikasi dapat beroperasi dalam lintas <i>platform</i> , seperti Linux.
2	Sistem dapat dijalankan oleh beberapa <i>software web browser</i> , seperti <i>Internet Explorer, Google Chrome, Mozilla Friefox</i> .
3	Pengguna sistem harus mengidentifikasi diri dengan sebuah <i>username</i> dan <i>password</i> . Hanya pengguna yang memiliki wewenang untuk dapat mengakses data dalam sistem.

Sumber: Hasil Analisis (2020)

## 5.2 Proses Bisnis *Monitoring* Hasil Produksi Usulan

Proses bisnis *monitoring* hasil produksi yang diusulkan melalui beberapa tahapan sebagai berikut yang dapat dilihat pada Gambar V.1:

1. *Operator* mengidentifikasi blank material untuk mengetahui adanya part reject/pending atau tidak. Jika tidak, *operator* akan mengisi *form checksheet*. Jika ya, *operator* akan mengidentifikasi kembali apakah material tersebut dapat *direpair/rework*.
2. Jika material tersebut dapat *direpair/rework*, *As. Foreman* akan mengisi *form LPP*. Jika tidak, *As. Foreman* akan mengisi *form LPR*.
3. *Checksheet* sudah diinput akan terkirim secara otomatis ke *As. Foreman* untuk dicek. Apabila sesuai, *checksheet* tersebut akan dikirimkan kepada *Foreman* untuk di validasi. Begitupun untuk *form LPP dan LPR* yang sudah diisi oleh *As. Foreman* akan dikirimkan secara otomatis ke *Foreman* dan divalidasi.



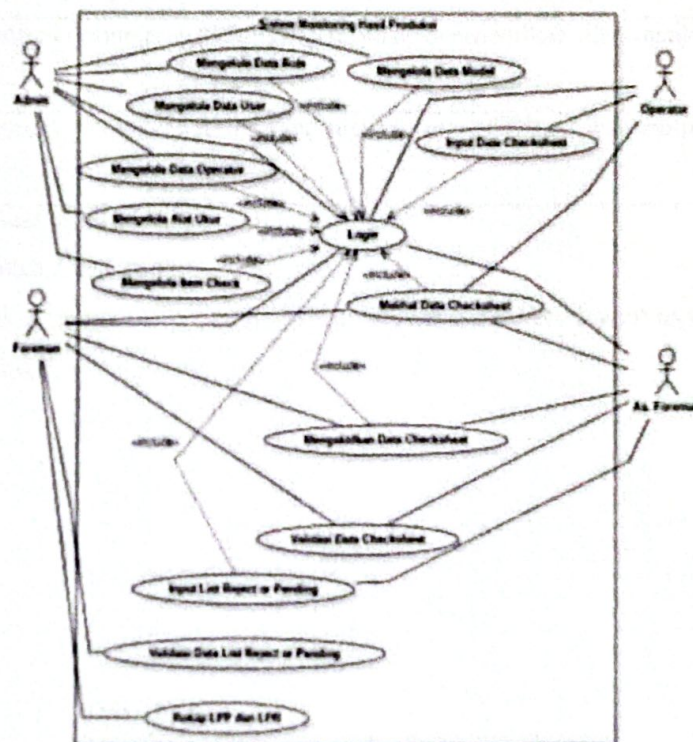
**Gambar V.1 Flowmap Proses Monitoring Hasil Produksi Usulan**  
 Sumber: Hasil Analisis (2020)

### 5.3 Analisis Sistem Usulan

Analisis proses *monitoring* hasil produksi akan dibuat menggunakan pemodelan sistem *Unified Modeling Language (UML)* yang meliputi *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, *Deployment Diagram*, *Windows Navigation Diagram*, dan *Entity Relationship Diagram*.

#### 5.3.1 Use Case Diagram Usulan

*Use case diagram* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Berikut adalah *use case diagram* sistem informasi *monitoring* hasil produksi *clutch housing* pada PT Mitsubishi Krama Yudha Motors and Manufacturing yang dapat dilihat pada Gambar V.2.



Gambar V.2 Sistem *Monitoring* Hasil Produksi Usulan  
Sumber: Hasil Analisis (2020)

Berikut penjelasan secara rinci mengenai *use case diagram* sistem *monitoring* hasil produksi, yaitu:

### 1. Definisi Aktor

Dalam sistem *monitoring* hasil produksi terdapat empat aktor. Adapun detail analisis *use case diagram* sebagai berikut:

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

No	Aktor	Definisi
1	Admin	Aktor admin tidak terdapat dalam proses bisnis usulan namun admin bertugas mengelola (membuat, menghapus, dan mengubah) data <i>user</i> yang terdapat pada proses bisnis usulan.
2	<i>Operator</i>	Aktor yang dapat menginput data <i>checksheet</i> dan melihat data <i>checksheet</i> .
3	<i>Assistant Foreman</i>	Aktor yang melihat, mengaktifkan, dan memvalidasi data <i>checksheet</i> .
4	<i>Foreman</i>	Aktor yang melihat, mengaktifkan, dan memvalidasi data <i>checksheet</i> .

Sumber: Hasil Analisis (2020)

### 2. Definisi *Use Case*

Berikut ini pendefinisian rancangan *use case diagram* sistem *monitoring* hasil produksi.

Tabel V.4 Definisi *Use Case Diagram* Usulan

No	Use Case	Definisi
1	<i>Login</i>	Proses verifikasi pengguna pada sistem.
2	Mengelola Data <i>User</i>	Proses pengelolaan ( <i>create, update, delete</i> ) data pengguna sistem sesuai dengan fungsinya.
3	Mengelola Data <i>Role</i>	Proses pengelolaan ( <i>create, update, delete</i> ) data pemberian hak akses.
4	Mengelola Data Model	Proses pengelolaan ( <i>create, update, delete</i> ) data model produk.
5	Mengelola Data <i>Operator</i>	Proses pengelolaan ( <i>create, update, delete</i> ) data no <i>operator</i> mesin.
6	Mengelola Alat Ukur	Proses pengelolaan ( <i>create, update, delete</i> ) data alat ukur.
7	Mengelola <i>Item Check</i>	Proses pengelolaan ( <i>create, update, delete</i> ) data <i>item check</i> .
8	<i>Input Data Checksheet</i>	Proses penginputan data <i>checksheet</i> yang dilakukan oleh <i>Operator</i> .
9	Melihat Data <i>Checksheet</i>	Data <i>checksheet</i> yang telah diinput dapat dilihat oleh <i>Operator, Assistant Foreman, dan Foreman</i> .
10	Mengaktifkan Data <i>Checksheet</i>	Proses pengaktifan data <i>checksheet</i> oleh <i>As. Foreman dan Foreman</i> agar dapat melihat data <i>checksheet</i> .
11	Validasi Data <i>Checksheet</i>	Proses validasi data <i>checksheet</i> yang telah diinput oleh <i>As. Foreman dan Foreman</i> .
12	<i>Input List Reject or Pending</i>	Proses penginputan data <i>list reject</i> atau <i>pending</i> yang dilakukan oleh <i>As. Foreman</i> .
13	Validasi Data <i>List Reject or Pending</i>	Proses validasi data <i>list reject</i> atau <i>pending</i> yang telah diinput oleh <i>As. Foreman</i>

Sumber: Hasil Analisis (2020)

### 3. Skenario *Use Case Diagram*

Berikut adalah skenario masing-masing *use case* pada sistem *monitoring* hasil produksi usulan, diantaranya:

a. *Use Case Login*

Berikut adalah skenario *use case login* yang dapat dilihat pada Tabel V.5

Tabel V.5 Skenario *Use Case Login*

<i>Nama Use Case</i>	<i>Login</i>
<i>Primary Actor</i>	Admin, Operator, As. Foreman, dan Foreman.
<i>Use Case Description</i>	<i>Use case login</i> merupakan proses otentikasi data <i>user</i> berupa <i>username</i> dan <i>password</i> .
<i>Relationship</i>	<i>Include: Mengelola Data User, Input Data Checksheet, Melihat Data Checksheet, Mengaktifkan Data Checksheet.</i> <i>Assosiation: Admin, Operator, As. Foreman, Foreman.</i>
<i>Normal Flow of Event</i>	1. <i>User</i> membuka aplikasi. 2. Sistem menampilkan halaman <i>login</i> . 3. <i>User</i> menginput <i>username</i> dan <i>password</i> pada <i>form login</i> . 4. Sistem akan mengecek otentikasi data.

Sumber: Hasil Analisis (2020)

b. *Use Case Mengelola Data User*

Berikut adalah *use case* mengelola data *user* dapat dilihat pada Tabel V.6

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

<i>Nama Use Case</i>	Mengelola Data <i>User</i>
<i>Primary Actor</i>	Admin
<i>Use Case Description</i>	<i>Use case</i> mengelola data <i>user</i> merupakan proses mengelola data <i>user</i> seperti menambah, mengubah, dan menghapus data <i>user</i> .
<i>Relationship</i>	<i>Include</i> : Login <i>Assosiation</i> : Admin
<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User</i> melakukan login.</li> <li>2. <i>User</i> memilih menu <i>user</i>.</li> <li>3. Jika <i>user</i> ingin membuat <i>user</i> baru maka <i>user</i> klik <i>add user</i> dan sistem akan menampilkan <i>form</i> tambah <i>user</i>.</li> <li>4. <i>User</i> menginput data <i>user</i> sesuai dengan kolom yang ada dan klik simpan.</li> <li>5. Sistem akan memvalidasi data <i>user</i> yang di <i>input</i> lalu menyimpannya.</li> <li>6. Jika <i>user</i> ingin mengubah data <i>user</i> maka <i>user</i> klik <i>edit</i> dan memilih data mana yang akan diubah lalu klik <i>update</i>.</li> <li>7. Jika <i>user</i> menghapus data <i>user</i> maka <i>user</i> klik <i>delete</i> pada data yang ingin dihapus.</li> <li>8. Sistem akan menghapus data yang <i>user</i> yang telah dipilih.</li> </ol>

Sumber: Hasil Analisis (2020)

c. *Use Case Mengelola Data Role*

Berikut adalah *use case* mengelola data *role* yang dapat dilihat pada Tabel V.7

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

Nama <i>Use Case</i>	Mengelola Data <i>Role</i>
Primary Actor	Admin
<i>Use Case Description</i>	<i>Use case</i> mengelola data <i>role</i>
Relationship	Include: Login Assosiation: admin
Normal Flow of Event	<p>User melakukan login.</p> <p>2. User memilih menu <i>role</i>.</p> <p>3. Jika <i>user</i> ingin membuat <i>role</i>/hak akses baru maka <i>user</i> klik <i>add role</i> dan sistem akan menampilkan <i>form</i> tambah <i>role</i>.</p> <p>4. User menginput data <i>role</i> sesuai dengan kolom yang ada dan klik simpan.</p> <p>5. Jika <i>user</i> ingin mengubah data <i>role</i> maka <i>user</i> klik <i>edit</i> dan mengganti data <i>role</i> yang akan diubah lalu klik <i>update</i>.</p> <p>6. Jika <i>user</i> menghapus data maka <i>user</i> klik <i>delete</i> pada data yang ingin dihapus.</p> <p>7. Sistem akan menghapus data yang <i>user</i> yang telah dipilih.</p>

Sumber: Hasil Analisis (2020)

d. *Use Case Mengelola Data Model*

Berikut adalah *use case* mengelola data *model* yang dapat dilihat pada Tabel V.8

Tabel V.8 Skenario *Use Case* Mengelola Data *Model*

Nama <i>Use Case</i>	Mengelola Data <i>Model</i>
Primary Actor	<i>Admin</i>
<i>Use Case Description</i>	<i>Use case</i> mengelola data <i>model</i>
Relationship	Include: <i>Login</i> Assosiation: <i>admin</i>
Normal Flow of Event	<ol style="list-style-type: none"> <li>1. <i>User</i> melakukan <i>login</i>.</li> <li>2. <i>User</i> memilih menu <i>model</i>.</li> <li>3. Jika <i>user</i> ingin membuat <i>model</i> baru maka <i>user</i> klik <i>add model</i> dan sistem akan menampilkan <i>form</i> tambah <i>model</i>.</li> <li>4. <i>User</i> menginput data <i>model</i> sesuai dengan kolom yang ada dan klik <i>simpan</i>.</li> <li>5. Sistem akan memvalidasi data <i>model</i> yang di <i>input</i> lalu menyimpannya.</li> <li>6. Jika <i>user</i> ingin mengubah data <i>model</i> maka <i>user</i> klik <i>edit</i> dan memilih data mana yang akan diubah lalu klik <i>update</i>.</li> <li>7. Jika <i>user</i> menghapus data <i>model</i> maka <i>user</i> klik <i>delete</i> pada data yang ingin dihapus.</li> <li>8. Sistem akan menghapus data yang <i>user</i> yang telah dipilih.</li> </ol>

Sumber: Hasil Analisis (2020)

e. *Use Case* Mengelola Data *Operator*

Berikut adalah *use case* mengelola data *operator* yang dapat dilihat pada Tabel V.9

Tabel V.9 Skenario *Use Case* Mengelola Data Operator

<i>Nama Use Case</i>	Mengelola Data Operator
<i>Primary Actor</i>	Admin
<i>Use Case Description</i>	Use case mengelola data operator
<i>Relationship</i>	Include: Login Assosiation: admin
<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. User melakukan login.</li> <li>2. User memilih menu operator.</li> <li>3. Jika user ingin membuat operator baru maka user klik <i>add operator</i> dan sistem akan menampilkan <i>form</i> tambah operator.</li> <li>4. User menginput data operator sesuai dengan kolom yang ada dan klik simpan.</li> <li>5. Sistem akan memvalidasi data operator yang di input lalu menyimpannya.</li> <li>6. Jika user ingin mengubah data operator maka user klik <i>edit</i> dan memilih data mana yang akan diubah lalu klik <i>update</i>.</li> <li>7. Jika user menghapus data operator maka user klik <i>delete</i> pada data yang ingin dihapus.</li> <li>8. Sistem akan menghapus data yang user yang telah dipilih.</li> </ol>

Sumber: Hasil Analisis (2020)

f. *Use Case* Mengelola Alat Ukur

Berikut adalah *use case* mengelola alat ukur yang dapat dilihat pada Tabel V.10

Tabel V.10 Skenario *Use Case* Mengelola Alat Ukur

Nama <i>Use Case</i>	Mengelola Alat Ukur
Primary Actor	Admin
Use Case Description	Use case mengelola alat ukur
Relationship	Include: Login Assosiation: admin
Normal Flow of Event	<ol style="list-style-type: none"> <li>1. User melakukan login.</li> <li>2. User memilih menu alat ukur.</li> <li>3. Jika user ingin membuat alat ukur baru maka user klik <i>add</i> alat ukur dan sistem akan menampilkan <i>form</i> tambah alat ukur.</li> <li>4. User menginput data alat ukur sesuai dengan kolom yang ada dan klik simpan.</li> <li>5. Sistem akan memvalidasi data alat ukur yang di <i>input</i> lalu menyimpannya.</li> <li>6. Jika user ingin mengubah data alat ukur maka user klik <i>edit</i> dan memilih data mana yang akan diubah lalu klik <i>update</i>.</li> <li>7. Jika user menghapus data alat ukur maka user klik <i>delete</i> pada data yang ingin dihapus.</li> <li>8. Sistem akan menghapus data yang user yang telah dipilih.</li> </ol>

Sumber: Hasil Analisis (2020)

g. *Use Case* Mengelola *Item Check*

Berikut adalah *use case* mengelola *Item Check* yang dapat dilihat pada Gambar V.11

Tabel V.11 Skenario *Use Case* Mengelola *Item Check*

<i>Nama Use Case</i>	Mengelola <i>Item Check</i>
<i>Primary Actor</i>	<i>Admin</i>
<i>Use Case Description</i>	<i>Use case</i> mengelola <i>item check</i>
<i>Relationship</i>	<i>Include: Login</i> <i>Assosiation: admin</i>
<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User</i> melakukan <i>login</i>.</li> <li>2. <i>User</i> memilih menu <i>item check</i>.</li> <li>3. Jika <i>user</i> ingin membuat <i>item check</i> baru maka <i>user</i> klik <i>add item check</i> dan sistem akan menampilkan <i>form</i> tambah <i>model</i>.</li> <li>4. <i>User</i> menginput <i>item check</i> sesuai dengan kolom yang ada dan klik <i>simpan</i>.</li> <li>5. Sistem akan memvalidasi <i>item check</i> yang di <i>input</i> lalu menyimpannya.</li> <li>6. Jika <i>user</i> ingin mengubah <i>item check</i> maka <i>user</i> klik <i>edit</i> dan memilih data mana yang akan diubah lalu klik <i>update</i>.</li> <li>7. Jika <i>user</i> menghapus <i>item check</i> maka <i>user</i> klik <i>delete</i> pada data yang ingin dihapus.</li> <li>8. Sistem akan menghapus data yang <i>user</i> yang telah dipilih.</li> </ol>

Sumber: Hasil Analisis (2020)

#### h. *Use Case Input Data Checksheet*

Berikut adalah *use case input data checksheet* yang dapat dilihat pada Tabel V.12

Tabel V.12 Skenario *Use Case Input Data Checksheet*

<i>Nama Use Case</i>	<i>Input Data Checksheet</i>
<i>Primary Actor</i>	<i>Operator</i>
<i>Use Case Description</i>	<i>Use case menginput data checksheet.</i>
<i>Relationship</i>	<i>Include: Login</i> <i>Assosiation: Operator</i>
<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User melakukan login.</i></li> <li>2. <i>User memilih menu checksheet.</i></li> <li>3. <i>User mengklik Day/ Night sebagai shift.</i></li> <li>4. <i>User memilih Model Checksheet dan memilih OP. No checksheet tersebut.</i></li> <li>4. <i>User menginput data sesuai dengan dengan item check yang telah tersedia pada form.</i></li> <li>5. <i>Jika data tidak berupa angka, user dapat mengklik complete yang nantinya terbaca sebagai "OK" oleh sistem.</i></li> </ol>

Sumber: Hasil Analisis (2020)

i. *Use Case Melihat Data Checksheet*

Berikut adalah *use case* melihat data *checksheet* yang dapat dilihat pada Tabel V.13

Tabel V.13 Skenario *Use Case* Melihat Data *Checksheet*

<i>Nama Use Case</i>	Melihat Data <i>Checksheet</i>
<i>Primary Actor</i>	<i>Operator, As. Foreman, Foreman.</i>
<i>Use Case Description</i>	<i>Use case</i> melihat data <i>checksheet</i> digunakan untuk melihat data <i>checksheet</i> yang telah diinput oleh <i>Operator</i> .
<i>Relationship</i>	<i>Include: Login, Mengaktifkan Data Checksheet.</i> <i>Assosiation: Operator, As. Foreman, Foreman.</i>
<i>Normal Flow of Event</i>	1. <i>User</i> melakukan <i>login</i> . 2. <i>User</i> memilih menu laporan <i>checksheet</i> . 3. <i>User</i> memilih Model <i>Checksheet</i> dan memilih OP. No <i>checksheet</i> tersebut. 4. <i>Checksheet</i> yang telah diinput akan terlihat.

Sumber: Hasil Analisis (2020)

j. *Use Case* Mengaktifkan Data *Checksheet*

Berikut adalah *use case* mengaktifkan data *checksheet* yang dapat dilihat pada Tabel V.14

Tabel V.14 Skenario *Use Case* Mengaktifkan Data *Checksheet*

<i>Nama Use Case</i>	Mengaktifkan Data <i>Checksheet</i>
<i>Primary Actor</i>	<i>As. Foreman, Foreman.</i>
<i>Use Case Description</i>	<i>Use case</i> mengaktifkan data <i>checksheet</i> digunakan untuk mengaktifkan data tersebut agar <i>As. Foreman</i> dan <i>Foreman</i> dapat melihat data <i>checksheet</i> .
<i>Relationship</i>	<i>Include: Login.</i> <i>Assosiation: As. Foreman, Foreman.</i>

Tabel V.14 Skenario *Use Case* Mengaktifkan Data *Checksheet* (lanjutan)

<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User</i> melakukan <i>login</i>.</li> <li>2. <i>User</i> memilih menu <i>Checksheet Shift</i> yang belum disetujui.</li> <li>3. <i>User</i> memilih <i>shift Day/ Night</i>.</li> <li>4. <i>User</i> memilih <i>detail</i> untuk mengaktifkan <i>checksheet</i> tersebut.</li> </ol>
-----------------------------	---

Sumber: Hasil Analisis (2020)

k. *Use Case* Validasi Data *Checksheet*

Berikut adalah *use case* validasi data *checksheet* yang dapat dilihat pada Tabel V.15

Tabel V.15 Skenario *Use Case* Validasi Data *Checksheet*

Nama <i>Use Case</i>	Validasi Data <i>Checksheet</i>
<i>Primary Actor</i>	<i>As. Foreman, Foreman</i> .
<i>Use Case Description</i>	<i>Use case</i> validasi data <i>checksheet</i> digunakan untuk mengapproved data <i>checksheet</i> yang telah diinput oleh <i>Operator</i> .
<i>Relationship</i>	<i>Include: Mengaktifkan Data Checksheet.</i> <i>Assosiation: As. Foreman, Foreman.</i>
<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User</i> melakukan <i>login</i>.</li> <li>2. <i>User</i> memilih menu <i>Checksheet Shift</i> yang belum disetujui.</li> <li>3. <i>User</i> memilih <i>shift Day/ Night</i>.</li> <li>4. <i>User</i> memilih <i>Approved</i> untuk validasi <i>checksheet</i> tersebut.</li> </ol>

Sumber: Hasil Analisis (2020)

l. *Use Case Input List Reject or Pending*

Berikut adalah *use case input list reject or pending* yang dapat dilihat pada Tabel V.16.

Tabel V.16 Skenario *Use Case Input List Reject or Pending*

<i>Nama Use Case</i>	<i>Input List Reject or Pending</i>
<i>Primary Actor</i>	<i>As. Foreman</i>
<i>Use Case Description</i>	<i>Use case menginput list reject or pending.</i>
<i>Relationship</i>	<i>Include: Login</i> <i>Assosiation: As. Foreman</i>
<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User melakukan login.</i></li> <li>2. <i>User memilih menu list reject dan pending.</i></li> <li>3. <i>User mengklik model.</i></li> <li>4. <i>User mengklik add reject untuk melakukan penginputan.</i></li> <li>4. <i>User menginput data sesuai dengan dengan list reject or pending yang telah tersedia pada form.</i></li> </ol>

Sumber: Hasil Analisis (2020)

m. *Use Case Validasi Data List Reject or Pending*

Berikut adalah *use case validasi data list reject or pending* yang dapat dilihat pada Tabel V.17.

Tabel V.17 Skenario *Use Case Validasi Data List Reject or Pending*

<i>Nama Use Case</i>	<i>Validasi Data List Reject or Pending</i>
<i>Primary Actor</i>	<i>Foreman</i>
<i>Use Case Description</i>	<i>Use case validasi data list reject or pending.</i>
<i>Relationship</i>	<i>Include: Input List Reject or Pending</i> <i>Assosiation: Foreman</i>

Tabel V.17 Skenario *Use Case* Validasi Data *List Reject or Pending* (lanjutan)

<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User</i> melakukan <i>login</i>.</li> <li>2. <i>User</i> memilih menu <i>List reject dan pending</i> yang pilih data <i>reject</i></li> <li>3. <i>User</i> klik <i>active</i> untuk mengaktifkan.</li> <li>4. <i>User</i> memilih menu belum disetujui untuk melihat data <i>list reject dan pendingnya</i></li> <li>5. kembali memilih menu <i>list reject dan pending</i> yang pilih data <i>reject</i>.</li> <li>6. klik <i>approve</i>, dan data akan berpindah ke <i>list reject dan pending</i> yang sudah disetujui.</li> </ol>
-----------------------------	---

Sumber: Hasil Analisis (2020)

n. *Use Case* Rekap LPP dan LPR

Berikut adalah *use case* rekap LPP dan LPR yang dapat dilihat pada

Tabel V.18

Tabel V.18 Skenario *Use Case* Rekap LPP dan LPR

Nama <i>Use Case</i>	Rekap LPP dan LPR
<i>Primary Actor</i>	<i>Foreman</i>
<i>Use Case Description</i>	<i>Use case</i> menginput <i>list reject or pending</i> .
<i>Relationship</i>	<i>Include: Validasi Data List Reject or Pending</i> <i>Assosiation: Foreman</i>
<i>Normal Flow of Event</i>	<ol style="list-style-type: none"> <li>1. <i>User</i> melakukan <i>login</i>.</li> <li>2. <i>User</i> memilih menu laporan <i>checksheet</i>.</li> <li>3. <i>User</i> mengklik rekap.</li> <li>4. <i>User</i> mengklik cetak.</li> </ol>

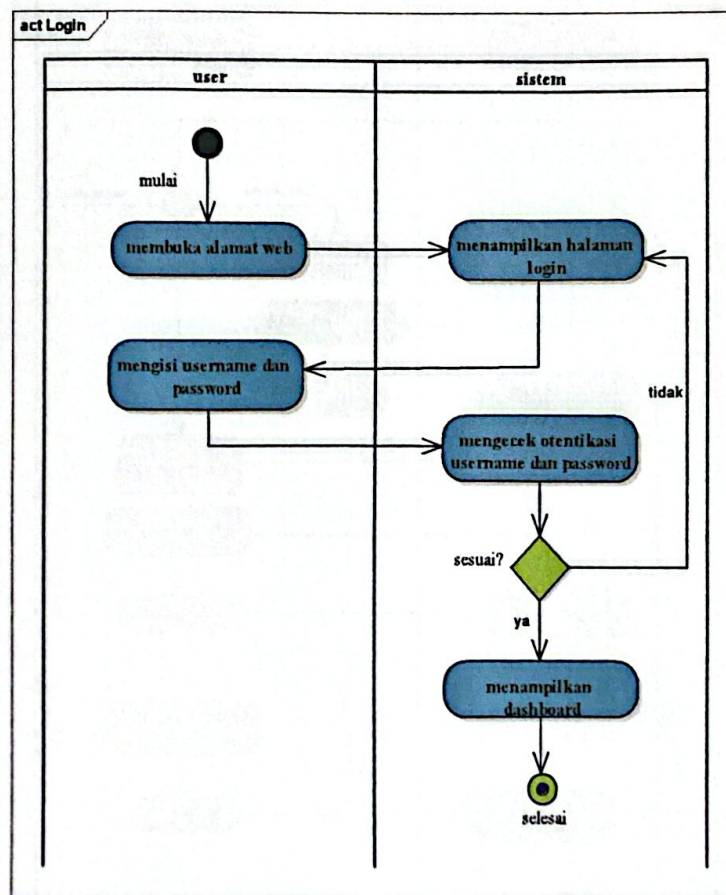
Sumber: Hasil Analisis (2020)

### 5.3.2 Activity Diagram Usulan

*Activity Diagram* merupakan sebuah diagram yang menggambarkan urutan aktivitas proses bisnis pada suatu sistem. Berikut *activity diagram* usulan yang terdapat pada sistem *monitoring* hasil produksi:

#### 1. *Activity Diagram Login*

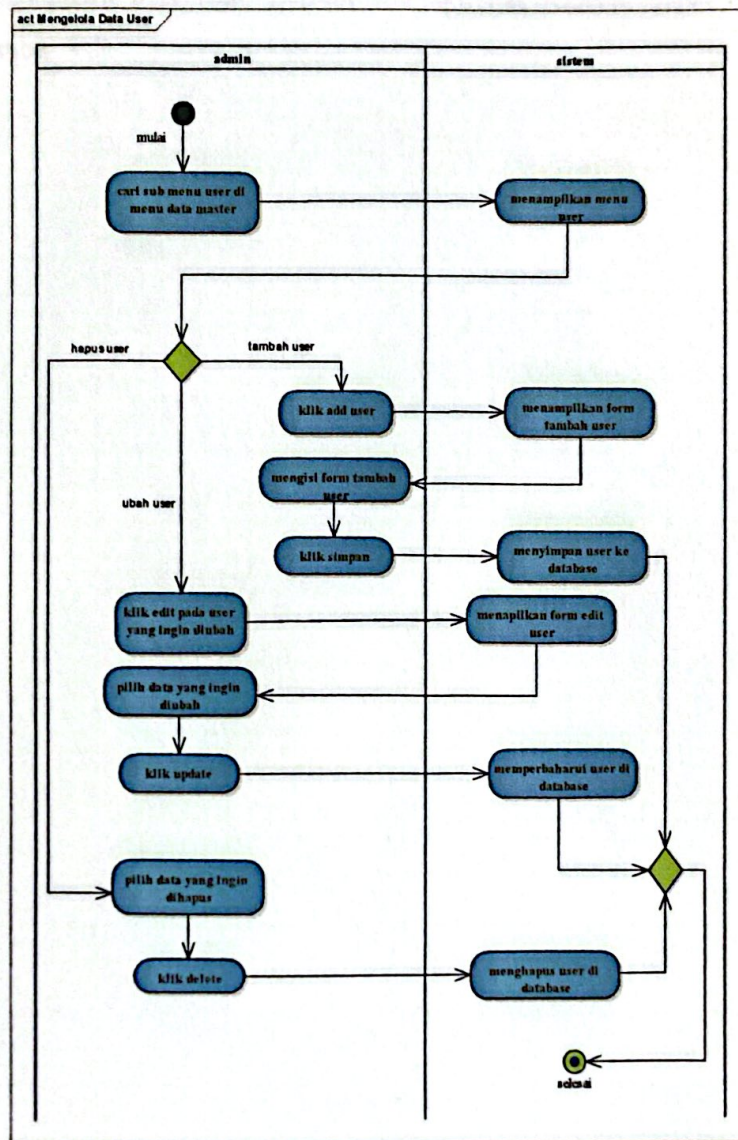
*Activity diagram login* menjelaskan alur aktivitas yang dilakukan *user* untuk mendapatkan hak akses ke dalam sistem. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.3.



Gambar V.3 *Activity Diagram Login*  
Sumber: Hasil Analisis (2020)

## 2. Activity Diagram Mengelola Data User

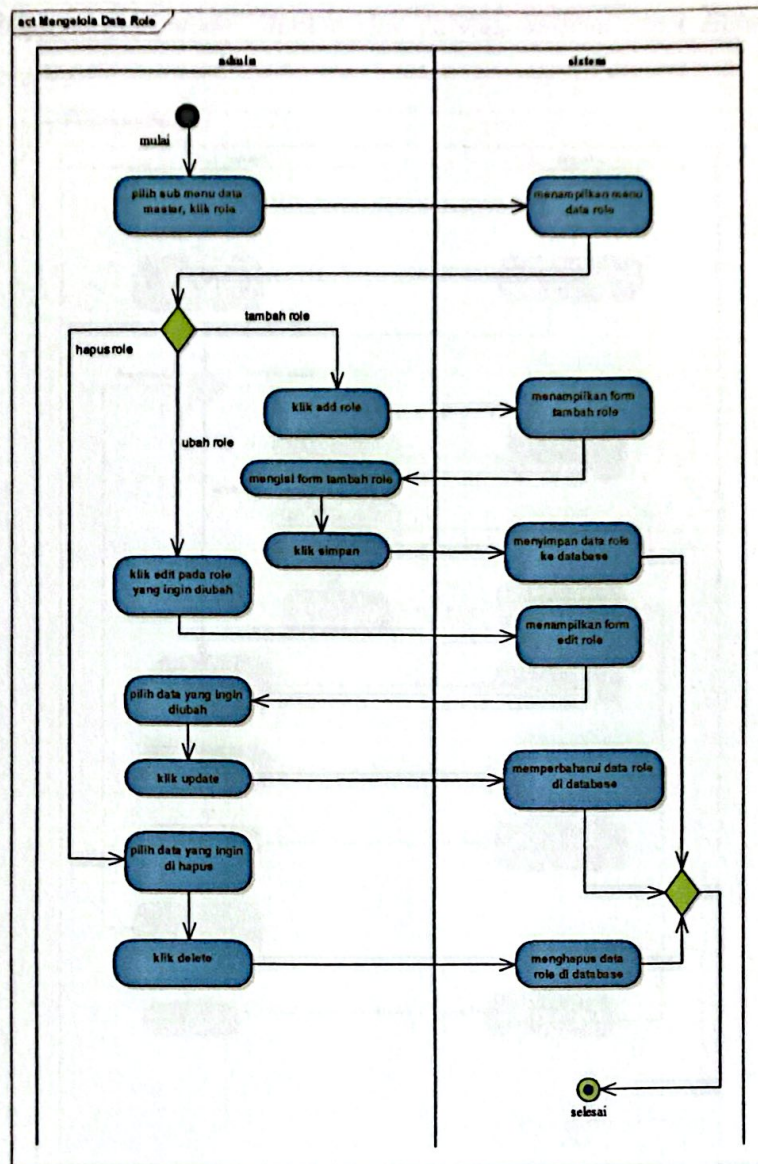
Activity diagram mengelola data user menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data user. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.4.



Gambar V.4 *Activity Diagram* Mengelola Data User  
Sumber: Hasil Analisis (2020)

3. *Activity Diagram* Mengelola Data Role

*Activity diagram* mengelola data *role* menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data *role*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.5.

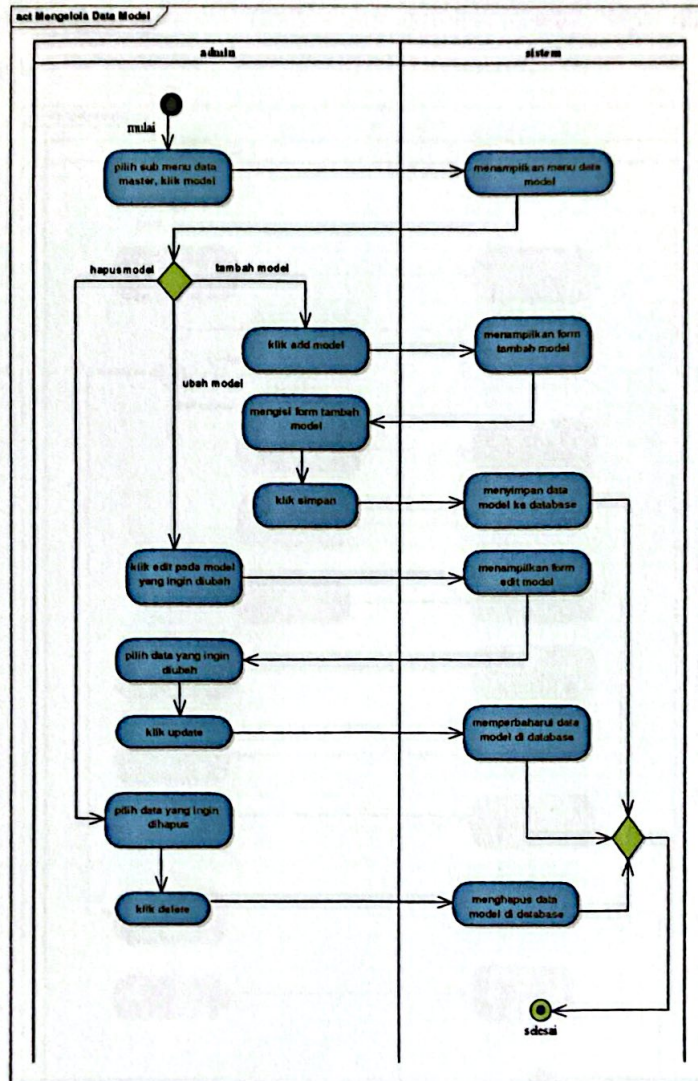


Gambar V.5 Activity Diagram Mengelola Data Role  
Sumber: Hasil Analisis (2020)

#### 4. Activity Diagram Mengelola Data Model

Activity diagram mengelola data model menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan

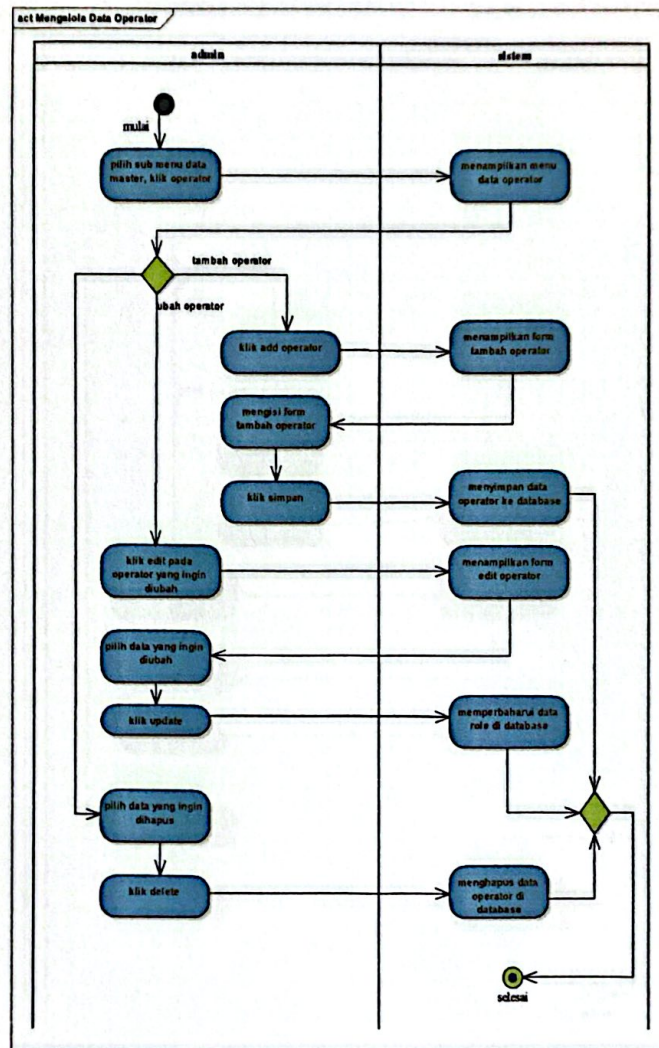
penghapusan data *model*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.6.



Gambar V.6 Activity Diagram Mengelola Data Model  
Sumber: Hasil Analisis (2020)

## 5. Activity Diagram Mengelola Data Operator

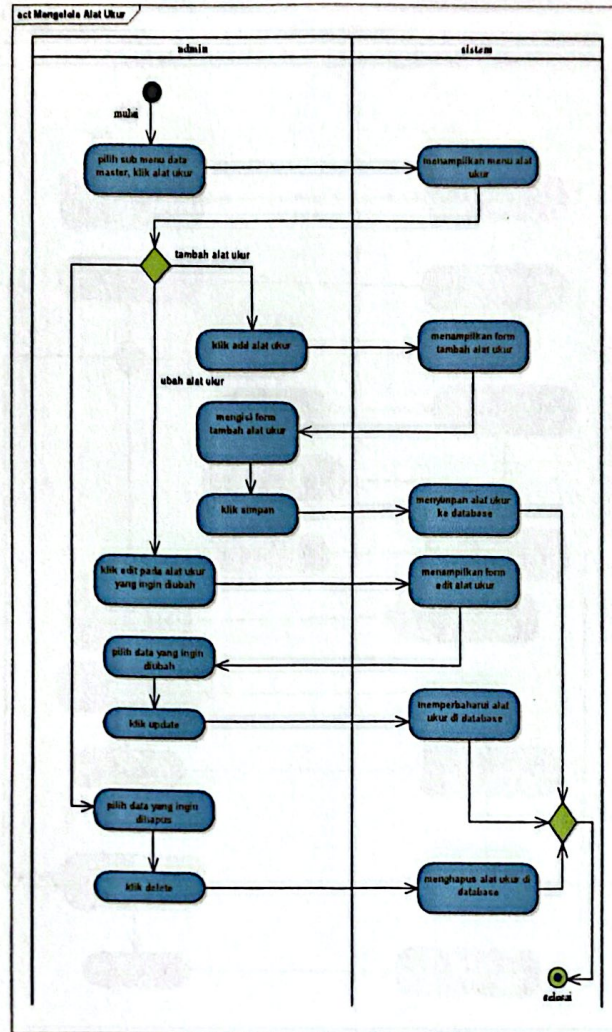
*Activity diagram* mengelola data *operator* menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data *operator*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.7.



Gambar V.7 *Activity Diagram* Mengelola Data Operator  
Sumber: Hasil Analisis (2020)

## 6. *Activity Diagram* Mengelola Alat Ukur

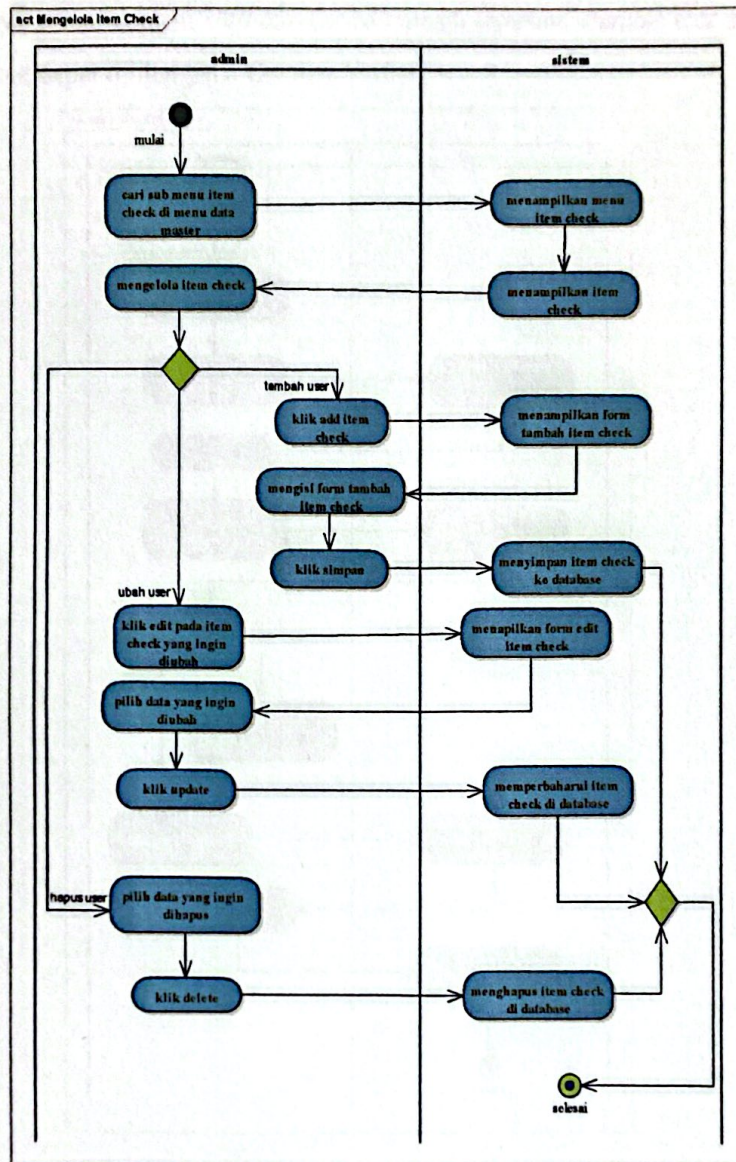
*Activity diagram* mengelola data alat ukur menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data alat ukur. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.8.



Gambar V.8 *Activity Diagram* Mengelola Alat Ukur  
Sumber: Hasil Analisis (2020)

## 7. *Activity Diagram* Mengelola Item Check

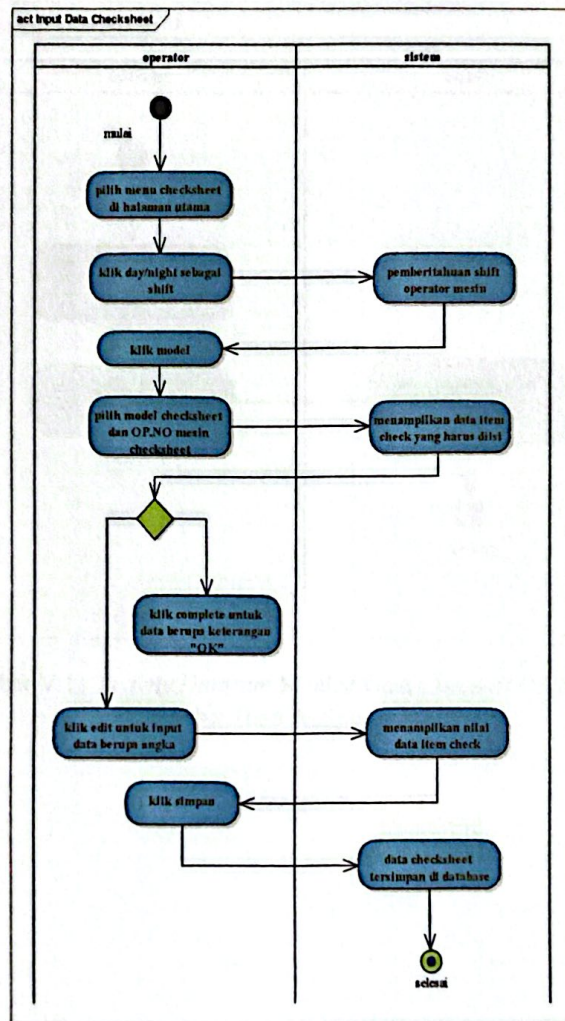
*Activity diagram* mengelola data *user* menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data *user*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.9.



Gambar V.9 Activity Diagram Mengelola Item Check  
Sumber: Hasil Analisis (2020)

### 8. Activity Diagram Input Data Checksheet

Activity diagram input data checksheet menjelaskan alur aktivitas yang dilakukan oleh Operator untuk melakukan penginputan data checksheet setiap harinya sesuai dengan shift dan OP.NO mesin tersebut. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.10.

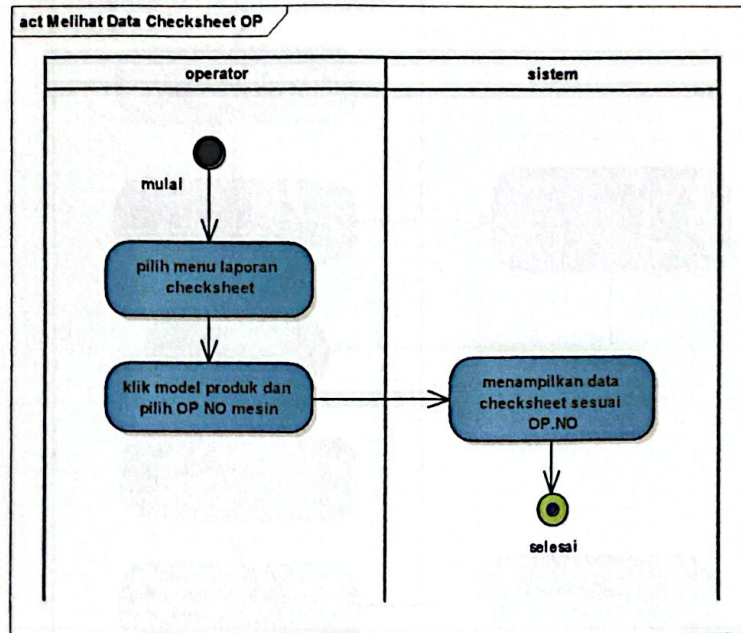


Gambar V.10 Activity Diagram Input Data Checksheet  
Sumber: Hasil Analisis (2020)

9. *Activity Diagram Melihat Data Checksheet*

*Activity diagram* melihat data *checksheet* menjelaskan alur aktivitas yang dilakukan *Operator, As. Foreman, dan Foreman* untuk melihat data *checksheet* yang telah diinput. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.11 dan Gambar V.12.

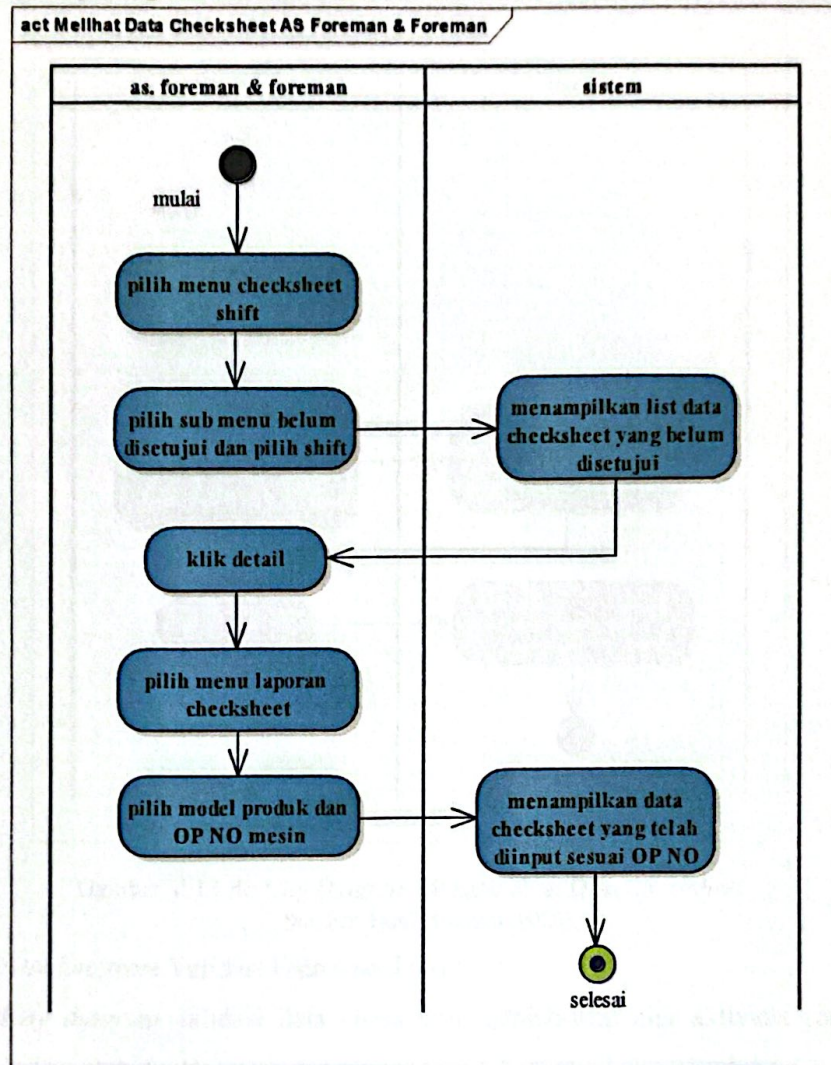
a. *Activity Diagram Melihat Data Checksheet Operator*



Gambar V.11 *Activity Diagram Melihat Data Checksheet Operator*

Sumber: Hasil Analisis (2020)

b. *Activity Diagram* Melihat Data *Checksheet* As. Foreman dan Foreman



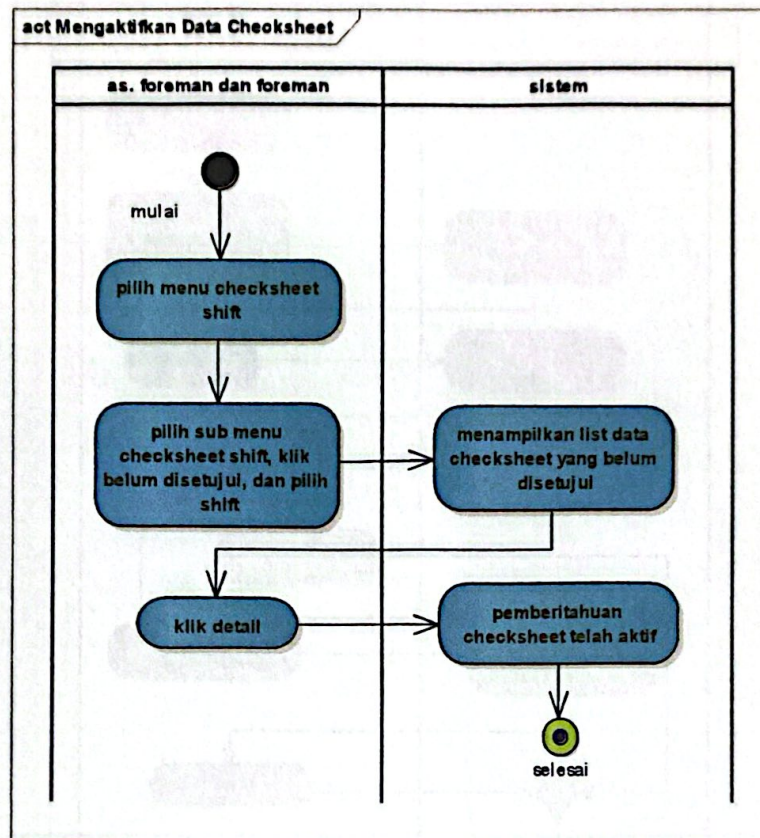
Gambar V.12 *Activity Diagram* Melihat Data *Checksheet* As. Foreman dan Foreman

Sumber: Hasil Analisis (2020)

10. *Activity Diagram* Mengaktifkan Data *Checksheet*

*Activity diagram* mengaktifkan data *checksheet* menjelaskan alur aktivitas yang dilakukan oleh As. Foreman dan Foreman untuk mengaktifkan guna

melihat *checksheet* yang telah diinput. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.13.

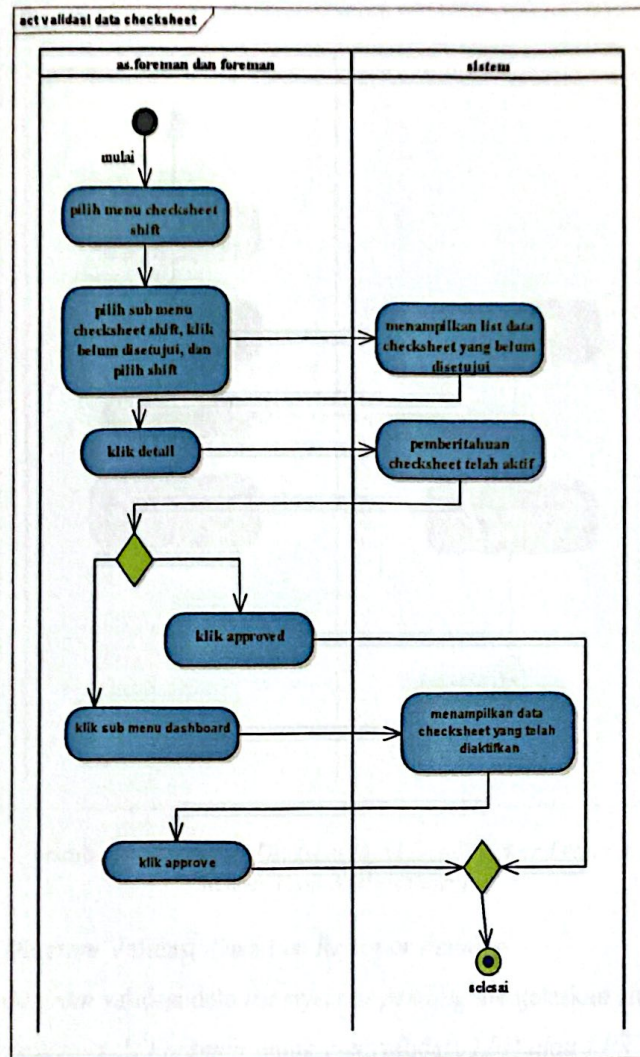


Gambar V.13 Activity Diagram Mengaktifkan Data Checksheet

Sumber: Hasil Analisis (2020)

#### 11. Activity Diagram Validasi Data Checksheet

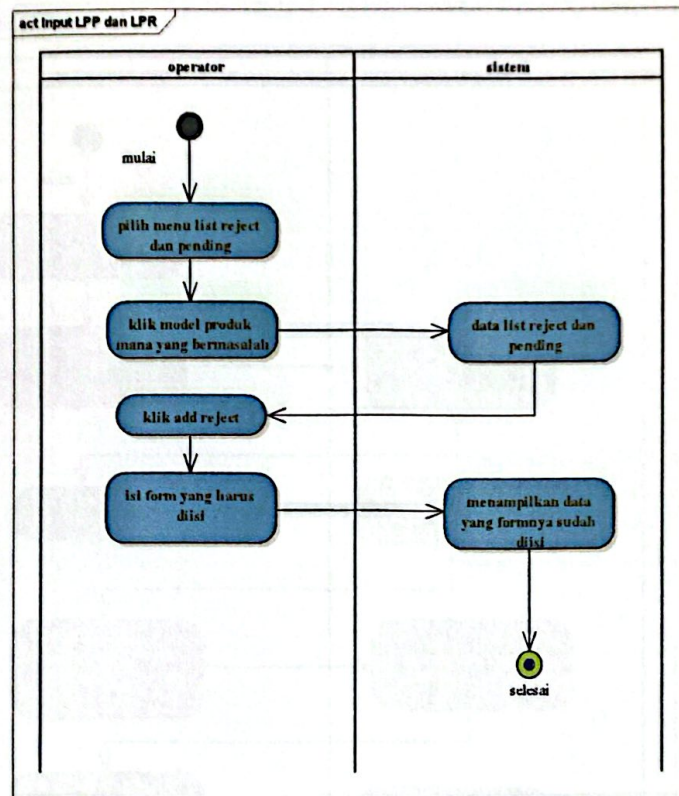
Activity diagram validasi data *checksheet* menjelaskan alur aktivitas yang dilakukan oleh As. Foreman dan Foreman untuk memvalidasi *checksheet* yang telah diinput oleh Operator. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.14.



Gambar V.14 Activity Diagram Validasi Data Checklist  
Sumber: Hasil Analisis (2020)

12. Activity Diagram Input List Reject or Pending

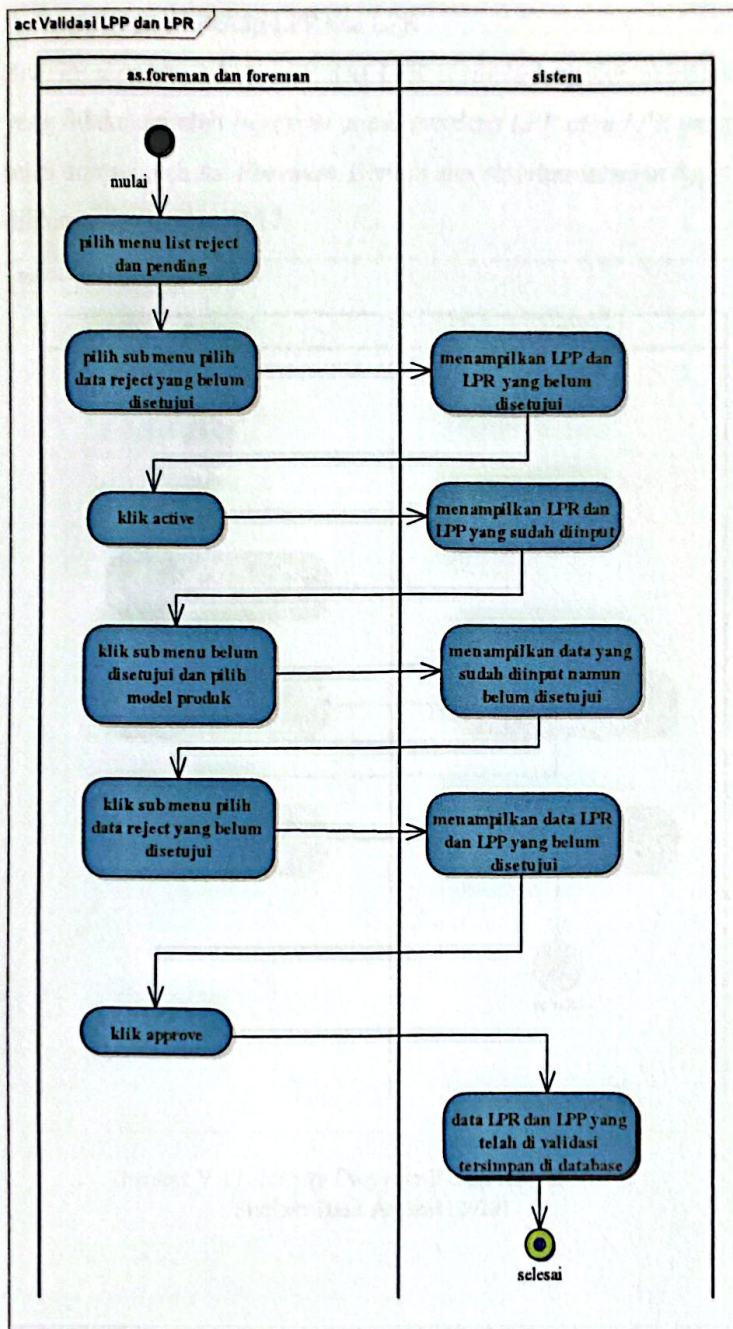
Activity diagram input list reject or pending menjelaskan alur aktivitas yang dilakukan oleh As. Foreman untuk melakukan penginputan data LPR dan LPP form tersebut. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.15.



Gambar V.15 Activity Diagram Input List Reject or Pending  
Sumber: Hasil Analisis (2020)

13. Activity Diagram Validasi Data List Reject or Pending

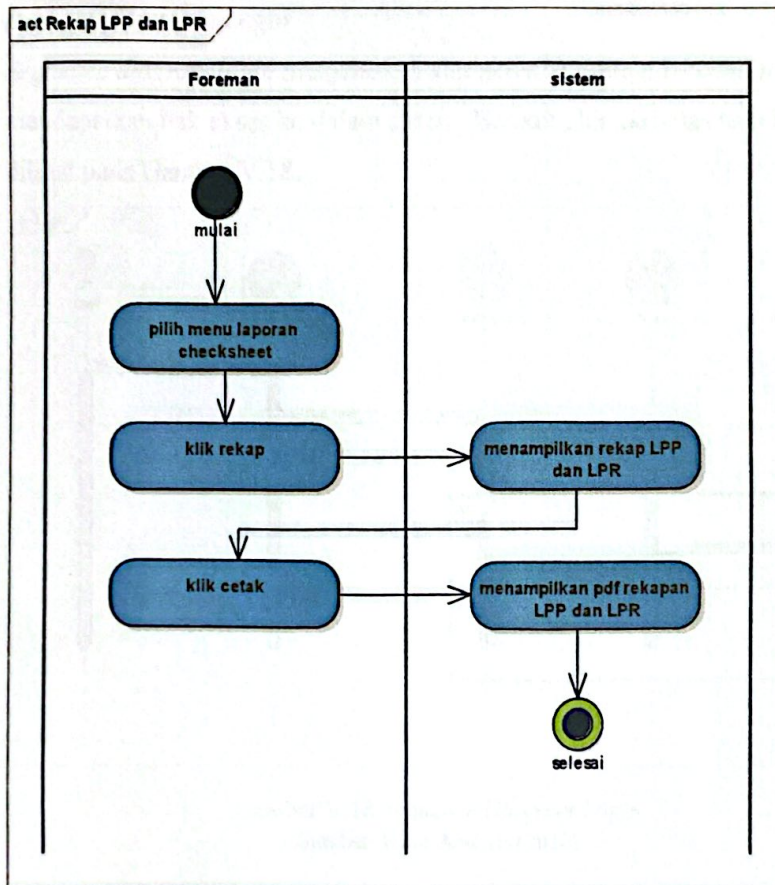
Activity diagram validasi data list reject or pending menjelaskan alur aktivitas yang dilakukan oleh Foreman untuk memvalidasi LPP atau LPR yang telah diinput oleh As. Foreman. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.16.



Gambar V.16 Activity Diagram Validasi List Reject dan Pending  
Sumber: Hasil Analisis (2020)

14. *Activity Diagram* Rekap LPP dan LPR

*Activity diagram* rekap LPP dan LPR menjelaskan alur aktivitas yang dilakukan oleh *Foreman* untuk merekap *LPP* atau *LPR* yang telah diinput oleh *As. Foreman*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.17.



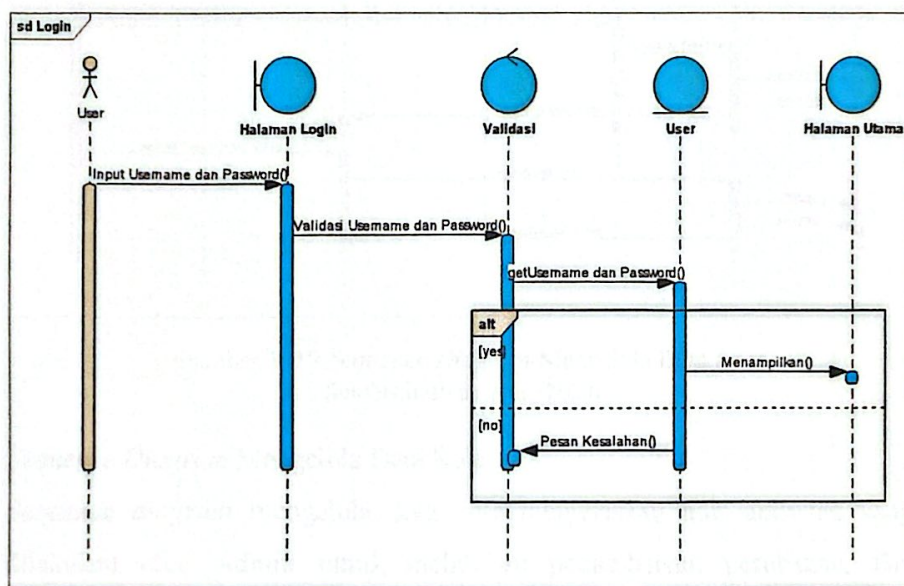
Gambar V.17 *Activity Diagram* Rekap LPP dan LPR  
Sumber: Hasil Analisis (2020)

### 5.3.3 Sequence Diagram

*Sequence Diagram* menggambarkan interaksi yang terjadi pada objek ketika melakukan proses tertentu, dimana urutan proses dapat diketahui dengan melihat gambaran pada *sequence diagram*. Berikut merupakan *sequence diagram* usulan dari sistem informasi *monitoring* hasil produksi:

#### 1. *Sequence Diagram Login*

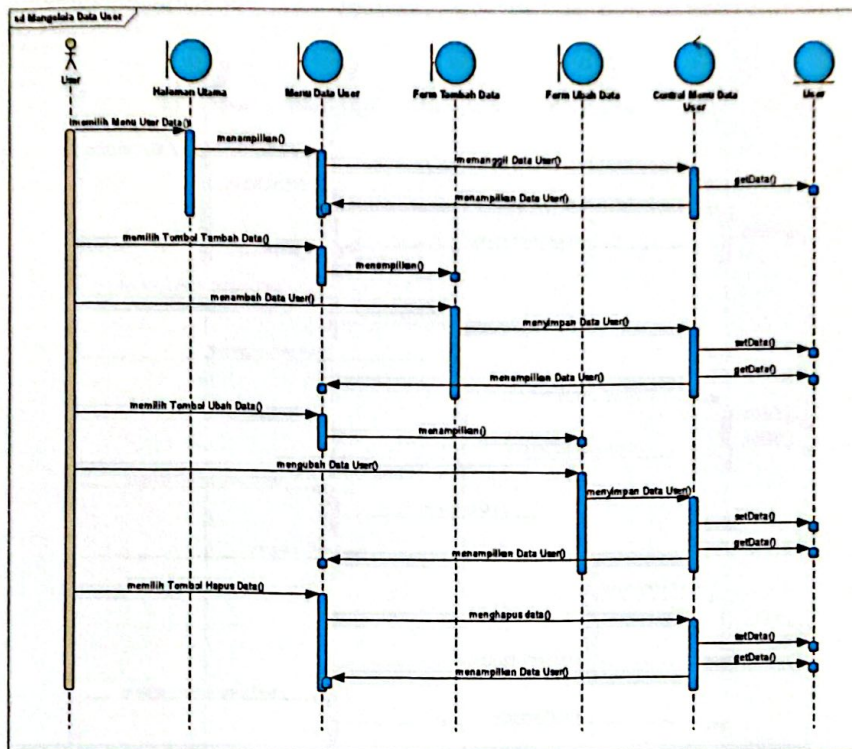
*Sequence diagram login* menjelaskan alur aktivitas yang dilakukan *user* untuk mendapatkan hak akses ke dalam sistem. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.18.



Gambar V.18 *Sequence Diagram Login*  
Sumber: Hasil Analisis (2020)

#### 2. *Sequence Diagram Mengelola Data User*

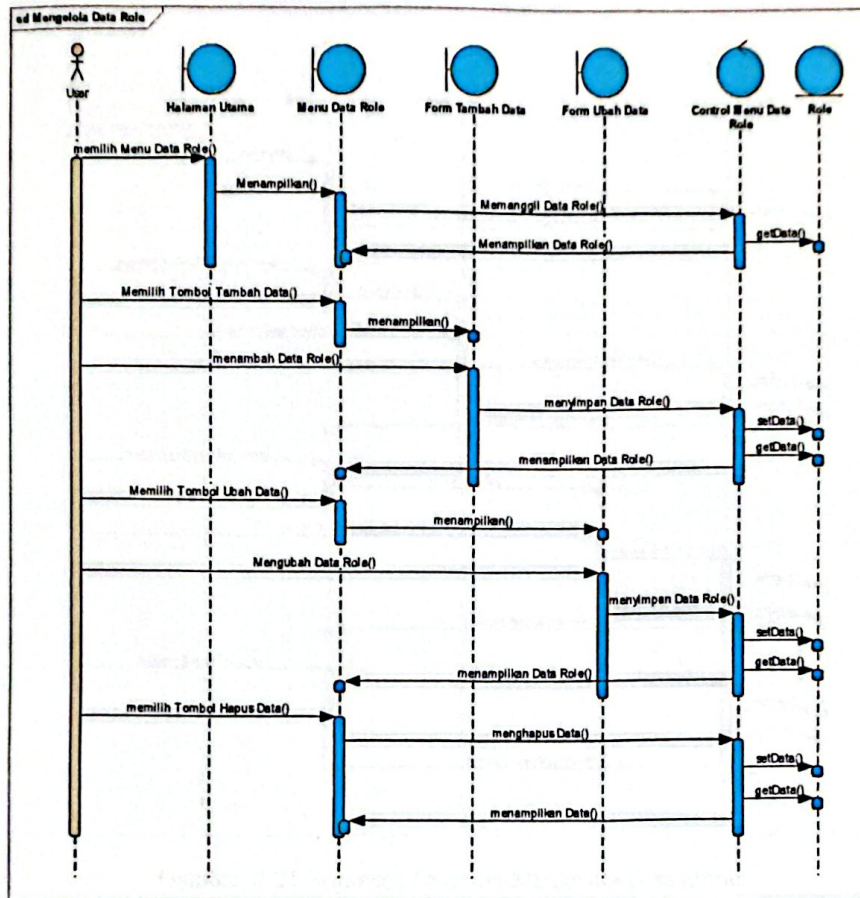
*Sequence diagram* mengelola data *user* menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data *user*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.19.



Gambar V.19 *Sequence Diagram* Mengelola Data User  
Sumber: hasil analisis (2020)

### 3. *Sequence Diagram* Mengelola Data Role

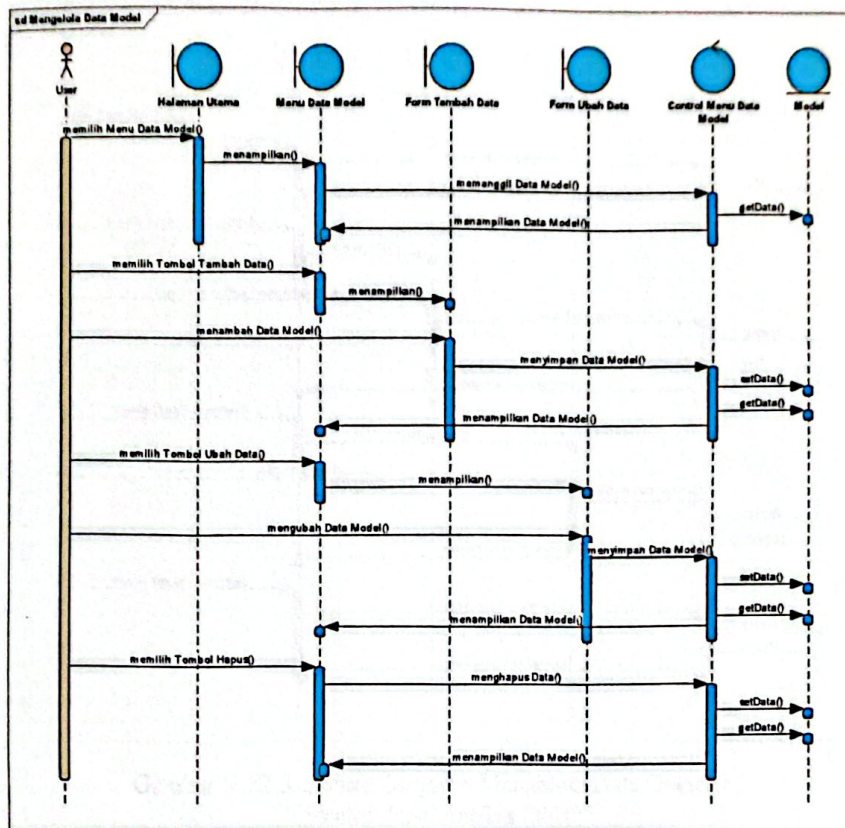
*Sequence diagram* mengelola data role menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data role. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.20.



Gambar V.20 *Sequence Diagram* Mengelola Data Role  
Sumber: Hasil Analisis (2020)

#### 4. *Sequence Diagram* Mengelola Data Model

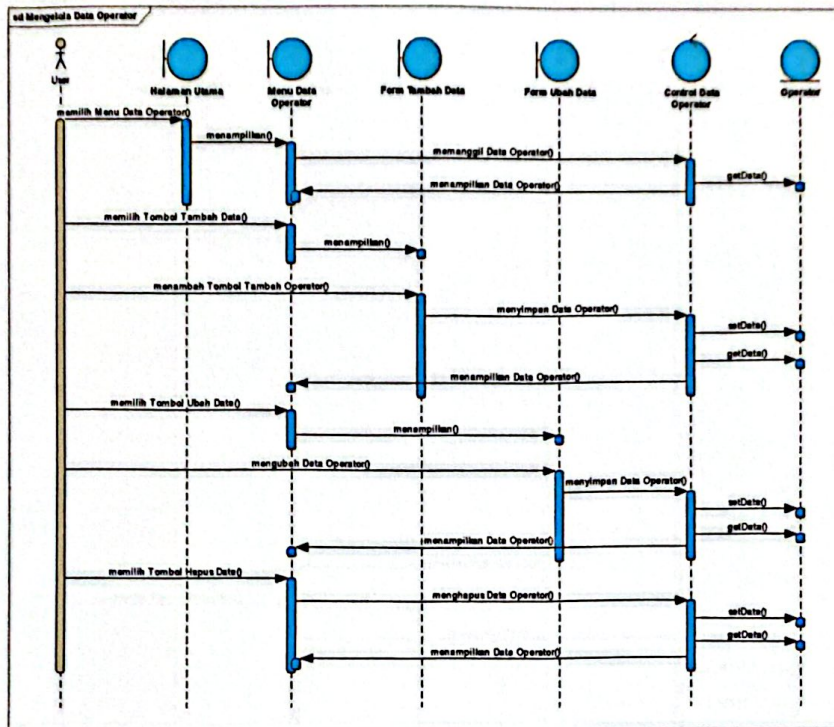
*Sequence diagram* mengelola data *model* menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data *model*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.21.



Gambar V.21 *Sequence Diagram* Mengelola Data Model  
 Sumber: Hasil Analisis (2020)

##### 5. *Sequence Diagram* Mengelola Data Operator

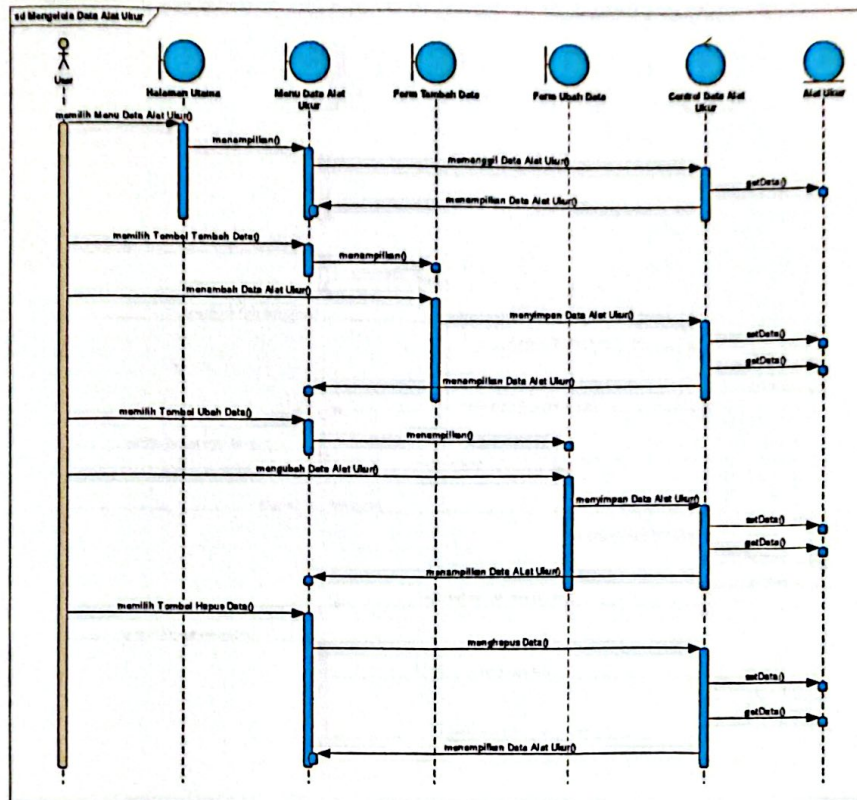
*Sequence diagram* mengelola data operator menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data operator. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.22.



Gambar V.22 Sequence Diagram Mengelola Data Operator  
 Sumber: Hasil Analisis (2020)

6. Sequence Diagram Mengelola Alat Ukur

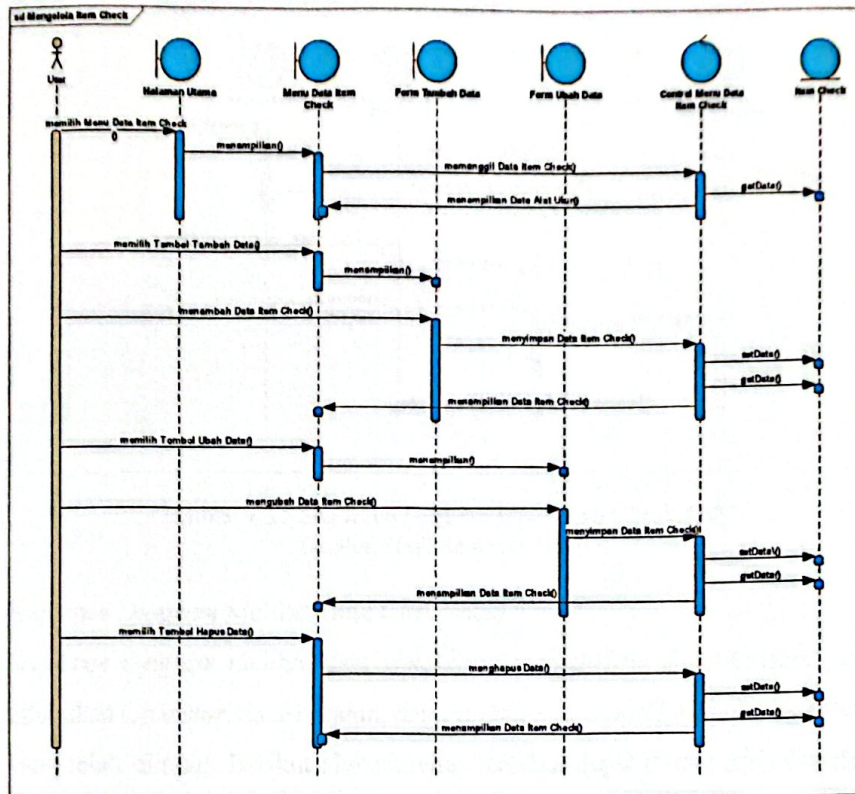
Sequence diagram mengelola data alat ukur menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data alat ukur. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.23.



Gambar V.23 Sequence Diagram Mengelola Alat Ukur  
Sumber: Hasil Analisis (2020)

#### 7. Sequence Diagram Mengelola Item Check

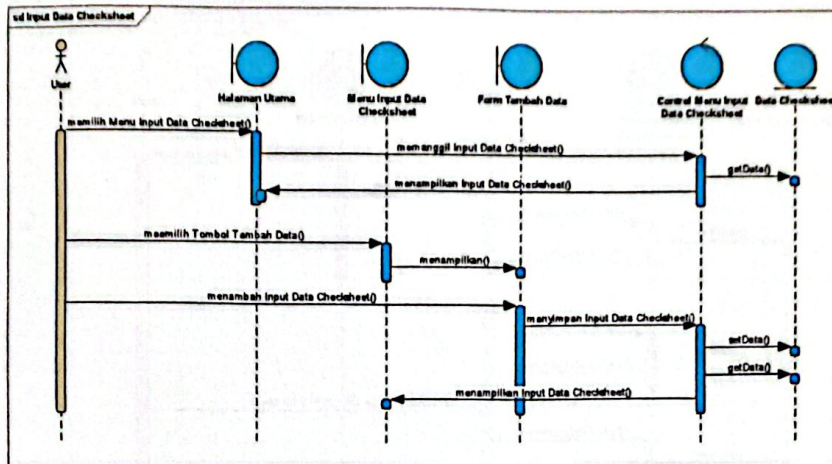
Sequence diagram mengelola data user menjelaskan alur aktivitas yang dilakukan oleh Admin untuk melakukan penambahan, perubahan, dan penghapusan data user. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.24.



Gambar V.24 Sequence Diagram Mengelola Item Check  
Sumber: Hasil Analisis (2020)

#### 8. Sequence Diagram Input Data Checksheet

Sequence diagram input data checksheet menjelaskan alur aktivitas yang dilakukan oleh Operator untuk melakukan penginputan data checksheet setiap harinya sesuai dengan shift dan OP.NO mesin tersebut. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.25.

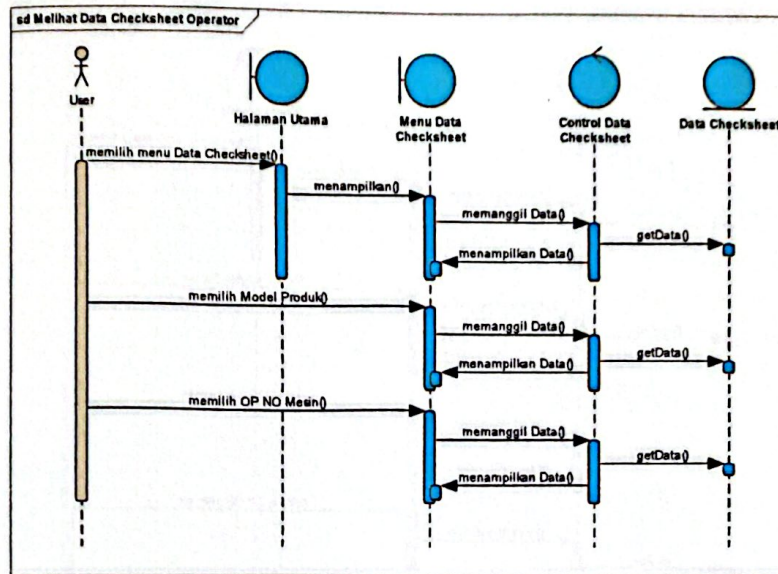


Gambar V.25 Sequence Diagram Input Data Checksheet  
Sumber: Hasil Analisis (2020)

#### 9. Sequence Diagram Melihat Data Checksheet

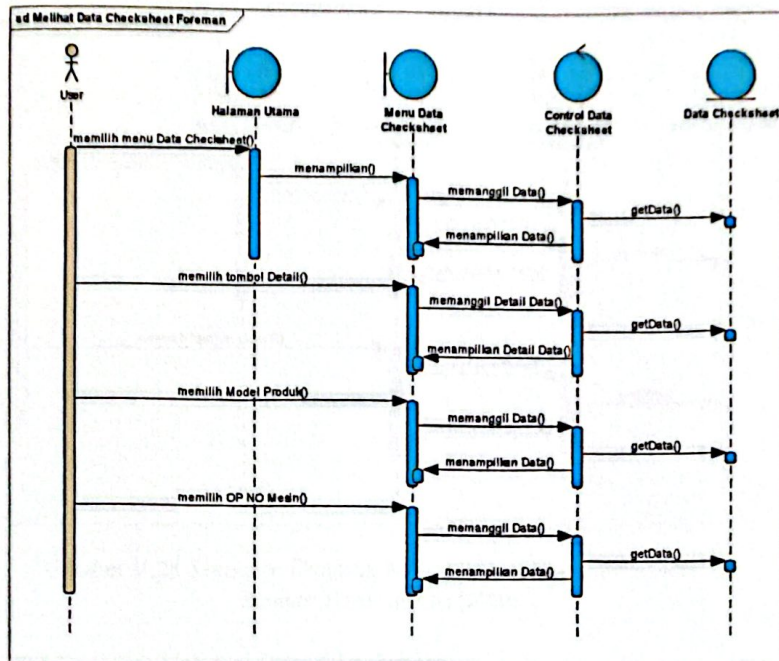
Sequence diagram melihat data *checksheet* menjelaskan alur aktivitas yang dilakukan Operator, As. Foreman, dan Foreman untuk melihat data *checksheet* yang telah diinput. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.26 dan Gambar V.27.

##### a. Melihat Data *Checksheet* Operator



Gambar V.26 Sequence Diagram Melihat Data Checksheet Operator  
Sumber: Hasil Analisis (2020)

b. Melihat Data *Checksheet* As. Foreman dan Foreman

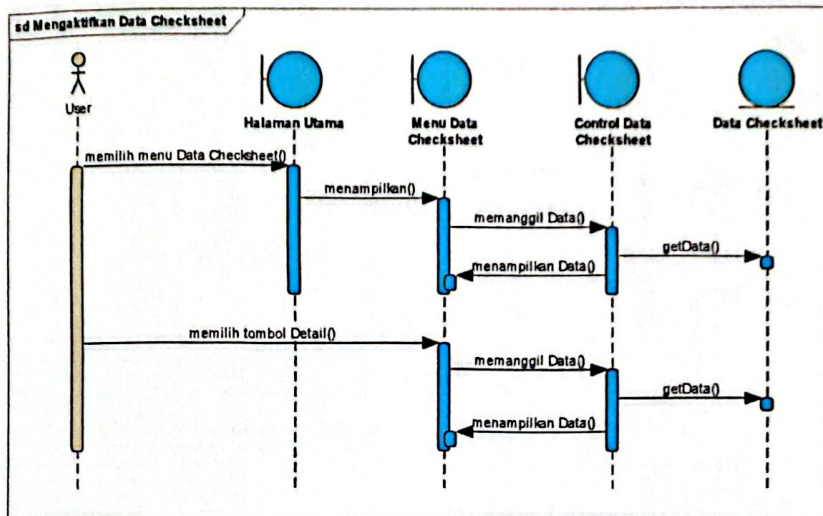


Gambar V.27 Sequence Diagram Melihat Data Checksheet As. Foreman dan Foreman

Sumber: Hasil Analisis (2020)

#### 10. Sequence Diagram Mengaktifkan Data Checksheet

Sequence diagram mengaktifkan data *checksheet* menjelaskan alur aktivitas yang dilakukan oleh As. Foreman dan Foreman untuk mengaktifkan guna melihat *checksheet* yang telah diinput. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.28.

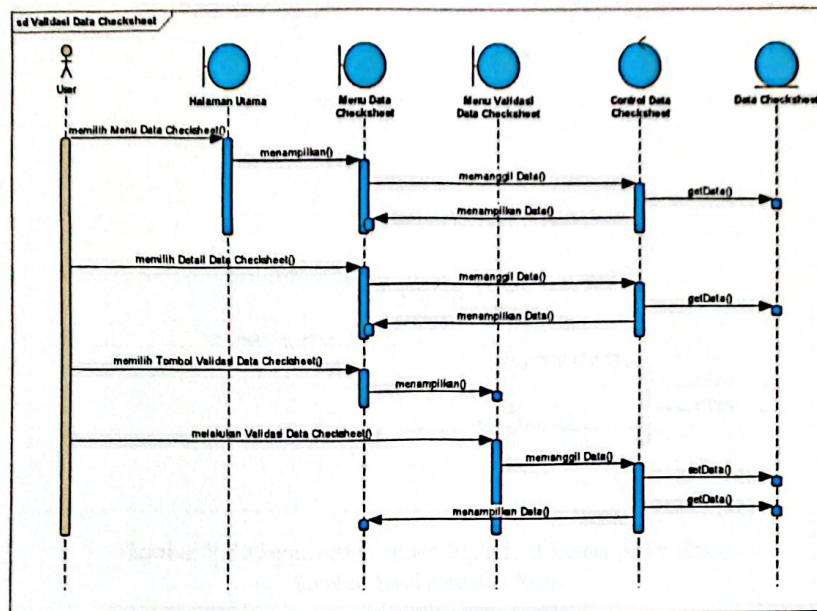


Gambar V.28 *Sequence Diagram* Mengaktifkan Data Checksheet

Sumber: Hasil Analisis (2020)

#### 11. *Sequence Diagram* Validasi Data Checksheet

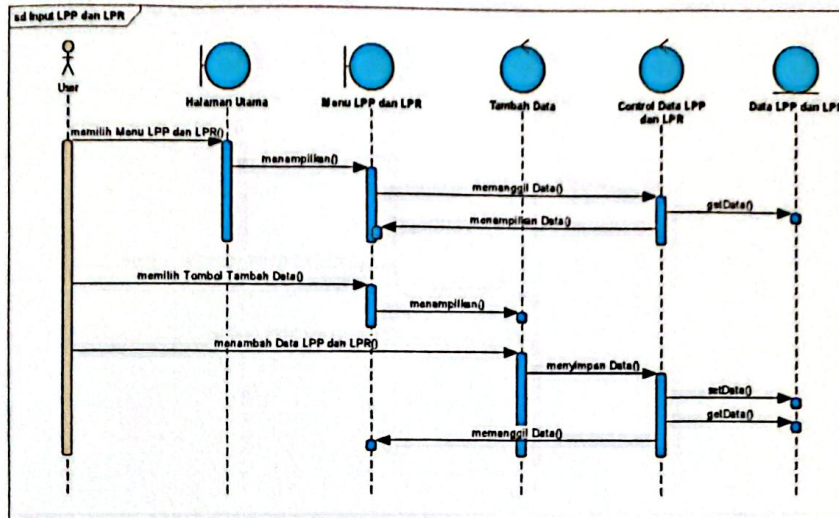
*Sequence diagram* validasi data checksheet menjelaskan alur aktivitas yang dilakukan oleh *As. Foreman* dan *Foreman* untuk memvalidasi checksheet yang telah diinput oleh *Operator*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.29.



Gambar V.29 Sequence Diagram Validasi Data Checksheet  
Sumber: Hasil Analisis (2020)

12. *Sequence Diagram Input List Reject or Pending*

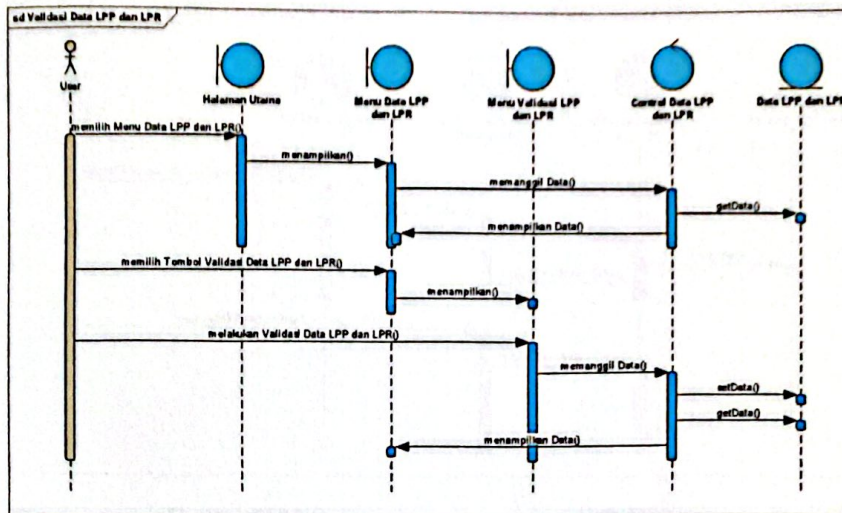
*Sequence diagram input list reject or pending* menjelaskan alur aktivitas yang dilakukan oleh *As. Foreman* untuk melakukan penginputan data *LPR dan LPP form* tersebut. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.30.



Gambar V.30 Sequence Diagram Input List Reject or Pending  
Sumber: Hasil Analisis (2020)

### 13. Sequence Diagram Validasi Data List Reject or Pending

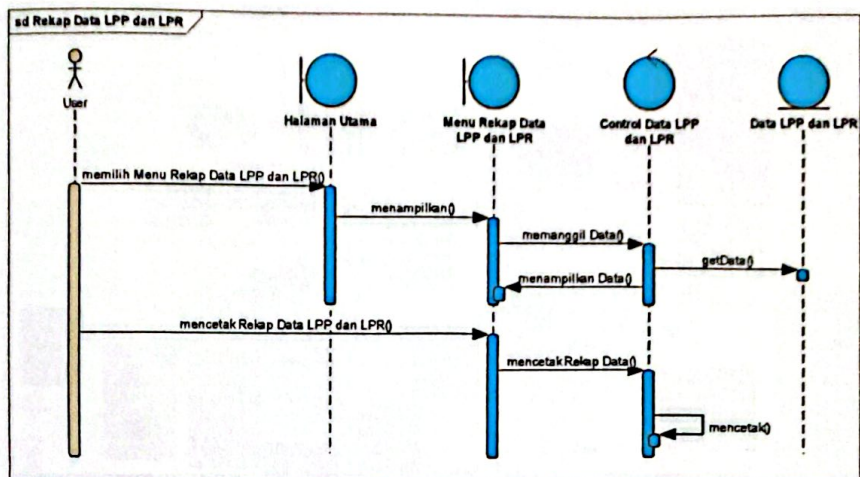
Sequence diagram validasi data list reject or pending menjelaskan alur aktivitas yang dilakukan oleh Foreman untuk memvalidasi LPP atau LPR yang telah diinput oleh As. Foreman. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.31.



Gambar V.31 Sequence Diagram Validasi Data List Reject or Pending  
Sumber: Hasil Analisis (2020)

#### 14. Sequence Diagram Rekap LPP dan LPR

Sequence diagram rekap LPP dan LPR menjelaskan alur aktivitas yang dilakukan oleh *Foreman* untuk merekap *LPP* atau *LPR* yang telah diinput oleh *As. Foreman*. Berikut alur aktivitas tersebut dapat dilihat pada Gambar V.32.

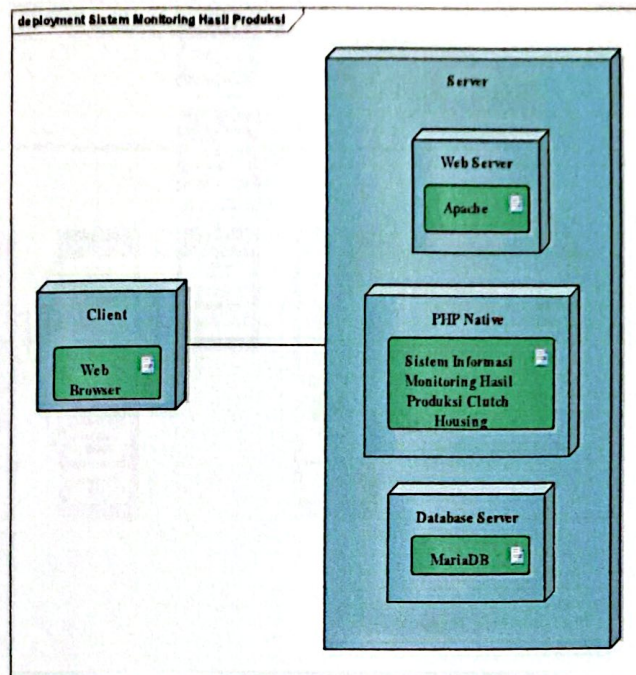


Gambar V.32 *Sequence Diagram* Rekap LPP dan LPR  
 Sumber: Hasil Analisis (2020)

#### 5.3.4 *Class Diagram* Usulan

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian setiap kelas-kelas. *Class diagram* juga memperlihatkan hubungan antar kelas dan penjelasan tiap-tiap kelas dari suatu sistem. Berikut ini merupakan *class diagram* sistem informasi *monitoring* hasil produksi *clutch housing* yang dapat dilihat pada Gambar V.33.

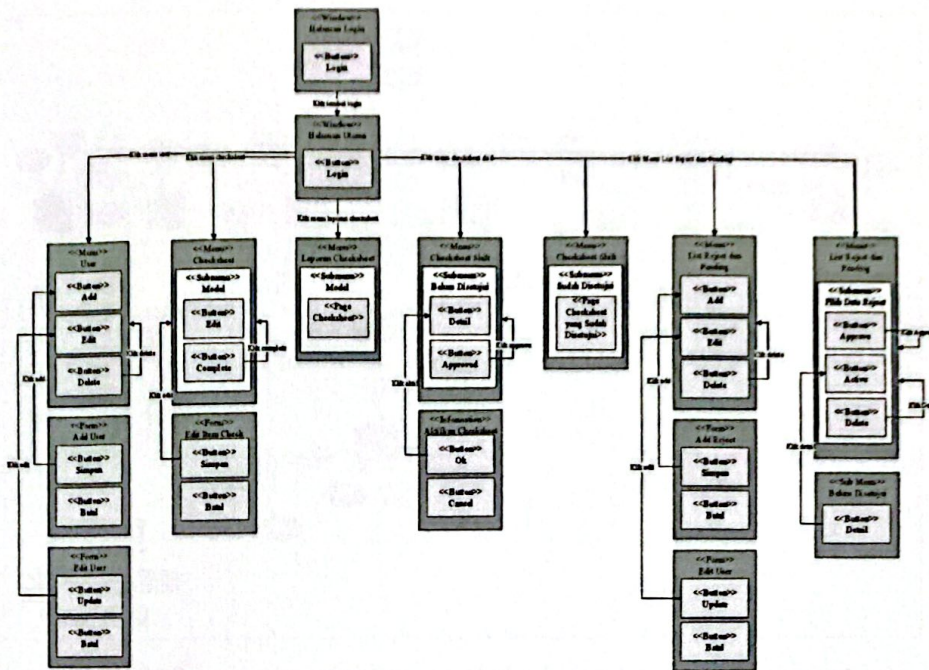




Gambar V.34 *Deployment Diagram* Sistem Informasi  
*Monitoring Hasil Produksi Clutch Housing*  
 Sumber: Hasil Analisis (2020)

### 5.3.6 *Windows Navigation Diagram (WND)*

*Windows Navigation Diagram* pada sistem usulan digunakan untuk menunjukkan bagaimana navigasi dari halaman-halaman yang terdapat di dalam aplikasi, berikut merupakan *Windows Navigation Diagram* sistem informasi *monitoring hasil produksi clutch housing* yang dapat dilihat pada Gambar V.35.



Gambar V.35 *Windows Navigation Diagram* Sistem Informasi *Monitoring Hasil Produksi Clutch Housing*  
 Sumber: Hasil Analisis (2020)

**5.4 Pemodelan Data Sistem Usulan**

Pemodelan data pada sistem informasi *monitoring* hasil produksi *clutch housing* pada PT Mitsubishi Krama Yudha Motors and Manufacturing dengan menggunakan dua cara yaitu *Entity Relationship Diagram (ERD)* dan Kamus Data. Berikut pemodelan data tersebut.

**5.4.1 Entity Relationship Diagram (ERD)**

*Entity Relationship Diagram* merupakan suatu model untuk menjelaskan hubungan antar data dalam database berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Berikut ERD yang diusulkan dapat dilihat pada Gambar V.36.



Tabel V.16 Tabel Kamus Data *User*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID User	id_user	Varchar	6	<i>Primary Key</i>
2	Username	username	Varchar	15	
3	Password	password	Varchar	50	
4	ID Hak Akses	id_role	Varchar	6	

Sumber: Hasil Analisis (2020)

2. Spesifikasi tabel *role*
  - a. Nama tabel: *tb\_role*
  - b. Fungsi: untuk memberikan hak akses
  - c. Tipe: file data master

Tabel V.17 Tabel Kamus Data *Role*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID Hak Akses	id_role	Varchar	6	<i>Primary Key</i>
2	Hak Akses	role	Varchar	15	

Sumber: Hasil Analisis (2020)

3. Spesifikasi tabel *check*
  - a. Nama tabel: *tb\_check*
  - b. Fungsi: untuk menyimpan data *item check*
  - c. Tipe: file data master

Tabel V.18 Tabel Kamus Data *Check*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID <i>Check</i>	id_check	Int	3	<i>Primary Key</i>
2	<i>Item Check</i>	item	Varchar	100	
3	Spesifikasi	spec	Varchar	25	
4	ID Nama Alat	Id_alat	Varchar	6	<i>Foreign Key</i>
5	<i>Interval</i>	inter	Varchar	10	
6	ID <i>Operator</i>	id_operator	Varchar	6	<i>Foreign Key</i>
7	ID Model Produk	Id_model	Varchar	6	<i>Foreign Key</i>

Sumber: Hasil Analisis (2020)

4. Spesifikasi tabel *operator*
  - a. Nama tabel: tb\_operator
  - b. Fungsi: untuk menyimpan data OP. NO mesin *operator*
  - c. Tipe: file data master

Tabel V.19 Tabel Kamus Data *Operator*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID <i>Operator</i>	id_operator	Varchar	6	<i>Primary Key</i>
2	OP.NO <i>Operator</i>	operator	Varchar	10	

Sumber: Hasil Analisis (2020)

5. Spesifikasi tabel model
  - a. Nama tabel: tb\_model
  - b. Fungsi: untuk menyimpan data model dari produk
  - c. Tipe: file data master

Tabel V.20 Tabel Kamus Data Model

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID Model	id_model	Varchar	6	<i>Primary Key</i>
2	Model Produk	model	Varchar	10	

Sumber: Hasil Analisis (2020)

6. Spesifikasi tabel alat
  - a. Nama tabel: *tb\_alat*
  - b. Fungsi: untuk menyimpan data alat ukur
  - c. Tipe: file data master

Tabel V.21 Tabel Kamus Data Alat

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID Nama Alat	id_alat	Varchar	6	<i>Primary Key</i>
2	Alat Ukur	alat	Varchar	20	

Sumber: Hasil Analisis (2020)

7. Spesifikasi tabel *detail monitor*
  - a. Nama tabel: *tb\_detail\_monitor*
  - b. Fungsi: untuk menyimpan data yang telah diinput
  - c. Tipe: file data transaksi

Tabel V.22 Tabel Kamus Data *Detail Monitor*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID <i>Detail Monitor</i>	id_detail_monitor	Int	10	<i>Primary Key</i>
2	ID <i>Monitor</i>	id_monitor	Varchar	6	<i>Foreign Key</i>
3	ID Check	id_check	Int	3	<i>Foreign Key</i>
4	Keterangan	keterangan	Varchar	5	

Sumber: Hasil Analisis (2020)

8. Spesifikasi tabel *shop*
  - a. Nama tabel: *tb\_shop*
  - b. Fungsi: untuk menyimpan data *shop*
  - c. Tipe: file data transaksi

Tabel V.23 Tabel Kamus Data *Shop*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID Shop	id_shop	Varchar	6	<i>Primary Key</i>
2	ID Operator	id_operator	Varchar	6	<i>Foreign Key</i>
3	ID Model	id_model	Varchar	6	<i>Foreign Key</i>

Sumber: Hasil Analisis (2020)

9. Spesifikasi tabel *monitor*

- a. Nama tabel: *tb\_monitor*
- b. Fungsi: untuk menyimpan validasi data
- c. Tipe: file data transaksi

Tabel V.24 Tabel Kamus Data *Monitor*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID <i>Monitor</i>	id_monitor	Varchar	6	<i>Primary Key</i>
2	ID <i>User</i>	id_user	Varchar	6	<i>Foreign Key</i>
3	Tanggal	tanggal	Date		
4	<i>Operator</i>	operator	Varchar	10	<i>Foreign Key</i>
5	Diperiksa	diperiksa	Varchar	6	
6	Disetujui	disetujui	Varchar	6	

Sumber: Hasil Analisis (2020)

10. Spesifika tabel *detail reject*

- a. Nama tabel: *tb\_detail\_reject*
- b. Fungsi: untuk menyimpan data *reject* dan *pending* yang diinput
- c. Tipe: file data transaksi

Tabel V.25 Tabel Kamus Data Detail *Reject*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID <i>Detail Reject</i>	id_detail_reject	Int	4	<i>Primary Key</i>
2	ID <i>Reject</i>	id_reject	Varchar	7	<i>Foreign Key</i>
3	ID List	id_list	Varchar	7	<i>Foreign Key</i>
4	Model produk	model	Varchar	7	
5	ID <i>Operator</i>	Id_operator	Varchar	7	<i>Foreign Key</i>
6	Tanggal	tgl	date		
7	Kode	kode	Varchar	15	
8	Jumlah	jumlah	Int	3	
9	Keterangan	keterangan	Varchar	100	
10	Penyebab	penyebab	Varchar	100	
11	Tindakan	tindakan	Varchar	100	

Sumber: Hasil Analisis (2020)

#### 11. Spesifikasi Tabel *Reject*

- a. Nama tabel: tb\_detail\_reject
- b. Fungsi: untuk menyimpan data *reject* dan *pending* yang diinput
- c. Tipe: file data transaksi

Tabel V.26 Kamus Data *Reject*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID <i>Reject</i>	id_reject	Varchar	7	<i>Primary Key</i>
2	ID <i>User</i>	id_user	Varchar	7	<i>Foreign Key</i>
3	Tanggal	tanggal	Date		
5	Diperiksa	diperiksa	Varchar	20	

Sumber: Hasil Analisis (2020)

## 12. Spesifikasi Tabel *List*

- a. Nama tabel: *tb\_list*
- b. Fungsi: tabel untuk menyimpan data *List*
- c. Tipe: file data transaksi

Tabel V.27 Kamu Data *List*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID <i>List</i>	id_list	Varchar	7	<i>Primary Key</i>
2	List	list	Varchar	30	<i>Foreign Key</i>

Sumber: Hasil Analisis (2020)

## 5.5 Perancangan Sistem Antarmuka

Rancangan *interface* (antarmuka) dari sistem informasi *monitoring* hasil produksi ini bertujuan untuk menggambarkan sistem yang telah dibuat. Rancangan antarmuka ini merupakan bagian yang berhubungan langsung antara pengguna dengan sistem, berikut rancangan antarmuka sistem informasi *monitoring* hasil produksi, yaitu:

### 1. Halaman *Login*

Halaman *login* merupakan halaman awal yang akan dilihat oleh user pada sistem *monitoring* ini. User yang dapat mengakses hanyalah user yang sudah terdaftar dan memiliki hak akses sesuai kebutuhan dengan menginput *username* dan *password*. Berikut rancangan antarmuka halaman login yang dapat dilihat pada Gambar V.22.



**PT MITSUBISHI KRAMA YUDHA MOTORS AND MANUFACTURING**

Login Monitoring Clutch Housing

Username

Password

Login

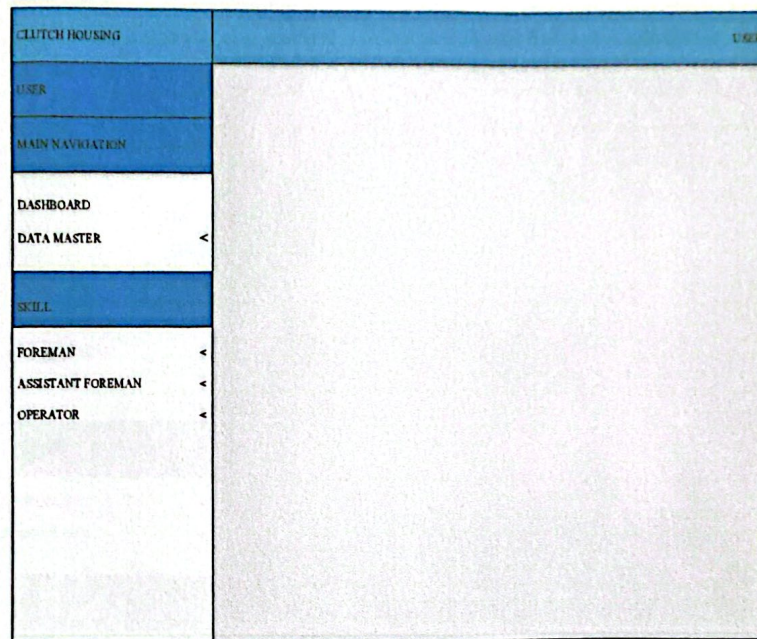
Gambar V.22 Tampilan Halaman *Login*  
Sumber: Hasil Analisis (2020)

## 2. Halaman Utama

Halaman utama adalah halaman tampilan utama ketika *user* masuk ke dalam sebuah sistem. Sistem *monitoring* hasil produksi memiliki halaman utama yang dibagi ke dalam beberapa tampilan sesuai dengan hak akses *user*, yaitu;

### a. Halaman Utama Admin

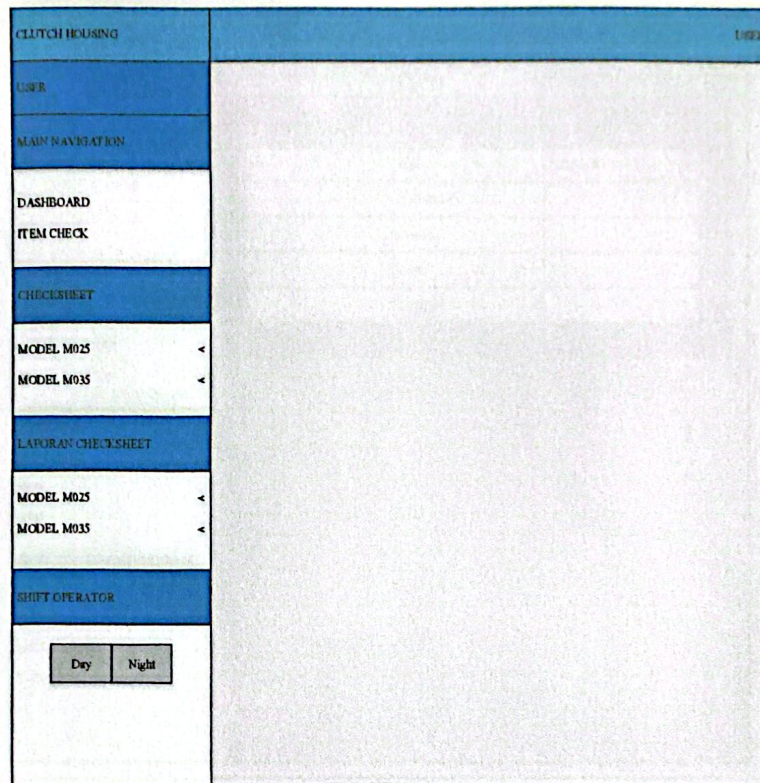
Halaman utama admin adalah halaman utama yang akan muncul ketika *user* dengan hak akses sebagai admin yang memiliki hak akses penuh terhadap semua menu yang ada. Berikut halaman utama admin yang dapat dilihat pada Gambar V.23.



Gambar V.23 Tampilan Halaman Utama Admin  
Sumber: Hasil Analisis (2020)

**b. Halaman Utama Operator**

Halaman utama *operator* adalah halaman utama yang akan muncul ketika user dengan hak akses sebagai *operator* yang memiliki hak akses terhadap menu untuk *input* dan melihat *checksheet*. Berikut halaman utama *operator* yang dapat dilihat pada Gambar V.24.



Gambar V.24 Tampilan Halaman Utama *Operator*  
 Sumber: Hasil Analisis (2020)

c. Halaman Utama *Assistant Foreman*

Halaman utama *As. Foreman* adalah halaman utama yang akan muncul ketika user dengan hak akses sebagai *as.foreman* memiliki hak akses terhadap menu untuk melihat, mengaktifkan, dan memvalidasi *checksheet*. Berikut halaman utama *as. foreman* yang dapat dilihat pada Gambar V.25.

CLUTCH HOUSING		USER
USER		
MAIN NAVIGATION		
DASHBOARD		
ITEM CHECK		
CHECKSHEET SHEET		
SUDAH DIBETUJUI <		
BELUM DIBETUJUI <		
LIST REJECT DAN PENDING		
M025 <		
M035 <		
LAPORAN CHECKSHEET		
MODEL M025 <		
MODEL M035 <		

Data Checklist		
No.	Bagian	Keterangan
1	No Monitor	
2	Operator	
3	Shift	
	Tanggal	

Gambar V.25 Tampilan Halaman Utama *As. Foreman*  
 Sumber: Hasil Analisis (2020)

d. Halaman Utama *Foreman*

Halaman utama *Foreman* adalah halaman utama yang akan muncul ketika user dengan hak akses sebagai *foreman* memiliki hak akses terhadap menu untuk melihat, mengaktifkan, dan memvalidasi *checksheet*. Berikut halaman utama *foreman* yang dapat dilihat pada Gambar V.26.

CLUTCH HOUSING		USER																			
USER		<table border="1"> <thead> <tr> <th colspan="3">Data Checklist</th> </tr> <tr> <th>No</th> <th>Bagian</th> <th>Keterangan</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>No Monitor</td> <td></td> </tr> <tr> <td>2</td> <td>Operator</td> <td></td> </tr> <tr> <td>3</td> <td>Shif</td> <td></td> </tr> <tr> <td></td> <td>Tanggal</td> <td></td> </tr> </tbody> </table>		Data Checklist			No	Bagian	Keterangan	1	No Monitor		2	Operator		3	Shif			Tanggal	
Data Checklist																					
No	Bagian			Keterangan																	
1	No Monitor																				
2	Operator																				
3	Shif																				
	Tanggal																				
MAIN NAVIGATION																					
DASHBOARD																					
ITEM CHECK																					
CHECKSHEET SHIFT																					
SUDAH DISETUJUI <																					
BELUM DISETUJUI <																					
LIST REJECT DAN PENDING																					
Pilih Data Reject <																					
Sudah Dietujui <																					
Belum Dietujui <																					
LAPORAN CHECKSHEET																					
Rakap																					
MODEL M025 <																					
MODEL M035 <																					

Gambar V.26 Tampilan Halaman Utama *Foreman*

Sumber: Hasil Analisis (2020)

### 3. Halaman *User*

Halaman user adalah halaman yang dapat diakses dengan hak akses sebagai admin, pada halaman user dapat melihat *list* data user yang ada. Berikut halaman *user* yang dapat dilihat pada Gambar V.27.

CLUTCH HOUSING	USER																																		
USER	DATA USER <span style="float: right;">Add User</span>																																		
MAIN NAVIGATION	<table border="1"> <thead> <tr> <th>No.</th> <th>Id User</th> <th>Username</th> <th>Role</th> <th>Foto</th> <th>Opsi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td> <input type="button" value="Edit"/> <input type="button" value="Delete"/> </td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>					No.	Id User	Username	Role	Foto	Opsi						<input type="button" value="Edit"/> <input type="button" value="Delete"/>																		
No.						Id User	Username	Role	Foto	Opsi																									
										<input type="button" value="Edit"/> <input type="button" value="Delete"/>																									
DASHBOARD																																			
DATA MASTER <																																			
SE.LL.																																			
FOREMAN <																																			
ASSISTANT FOREMAN <																																			
OPERATOR <																																			

Gambar V.27 Tampilan Halaman *User*  
 Sumber: Hasil Analisis (2020)

#### 4. Halaman *Add User*

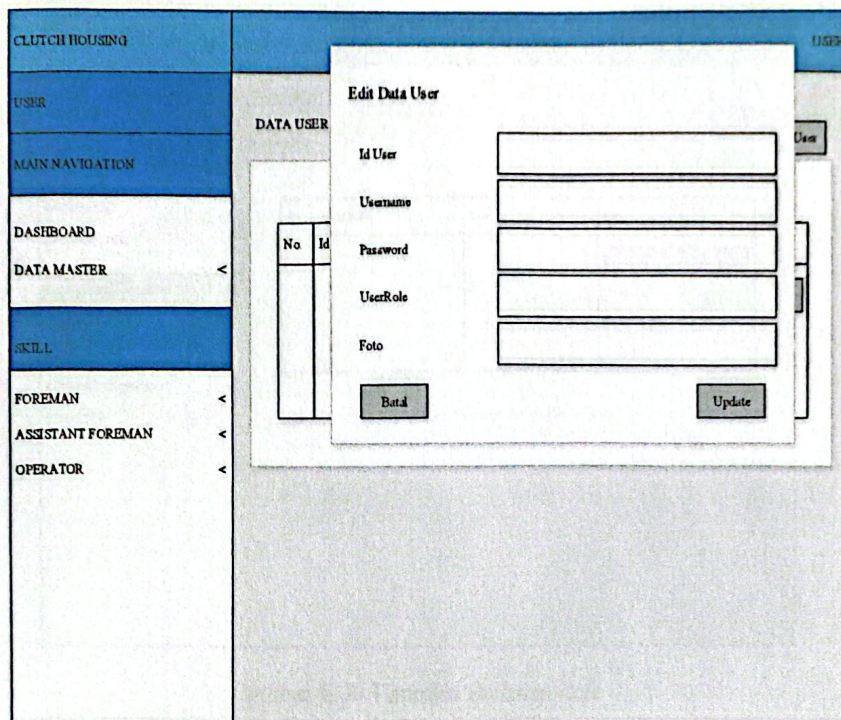
Halaman *add user* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *add user*, admin dapat membuat *user* baru dengan mengisi *form* yang ada. Berikut halaman *add user* yang dapat dilihat pada Gambar V.28.

The image shows a web application interface for user management. On the left is a vertical navigation menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area is titled 'DATA USER' and contains a table with columns 'No.' and 'Id'. A modal window titled 'Registrasi User Baru' is open over the table. The modal contains the following fields: 'Id User', 'Username', 'Password', 'UserRole', and 'Foto'. Below these fields are two buttons: 'Batal' (Cancel) and 'Simpan' (Save).

Gambar V.28 Tampilan Halaman *Add User*  
Sumber: Hasil Analisis (2020)

##### 5. Halaman *Edit User*

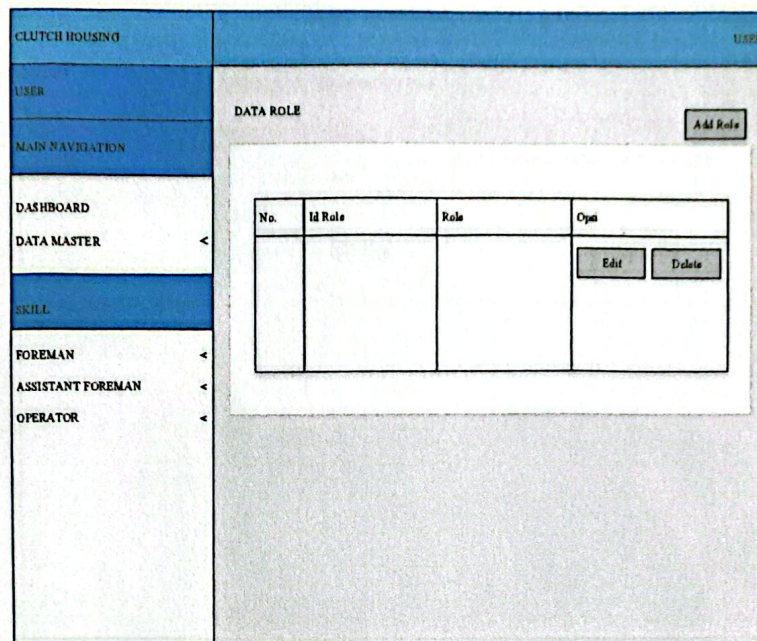
Halaman *edit user* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *edit user*, admin dapat mengubah data *user* dengan mengisi *form* yang ingin diubah. Berikut halaman *edit user* yang dapat dilihat pada Gambar V.29.



Gambar V.29 Tampilan Halaman *Edit User*  
 Sumber: Hasil Analisis (2020)

#### 6. Halaman *Role*

Halaman *role* adalah halaman yang dapat diakses dengan hak akses sebagai admin, pada halaman *role* dapat melihat *list* data *role* yang ada. Berikut halaman *role* yang dapat dilihat pada Gambar V.30.



Gambar V.30 Tampilan Halaman *Role*  
 Sumber: Hasil Analisis (2020)

#### 7. Halaman *Add Role*

Halaman *add role* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *add role*, admin dapat membuat *role* baru dengan mengisi *form* yang ada. Berikut halaman *add role* yang dapat dilihat pada Gambar V.31.

The image shows a web application interface. On the left is a sidebar menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATEIN, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area is titled 'DATA ROLE' and contains a table with columns 'No.', 'Id Role', and 'Role'. A modal window titled 'Registrasi Role Baru' is open over the table, containing two input fields for 'Id Role' and 'Role', and two buttons: 'Batal' and 'Simpan'.

Gambar V.32 Tampilan Halaman *Add Role*  
Sumber: Hasil Analisis (2020)

#### 8. Halaman *Edit Role*

Halaman *edit role* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *edit role*, admin dapat mengubah data *role* dengan mengisi *form* yang ingin diubah. Berikut halaman *edit role* yang dapat dilihat pada Gambar V.33.

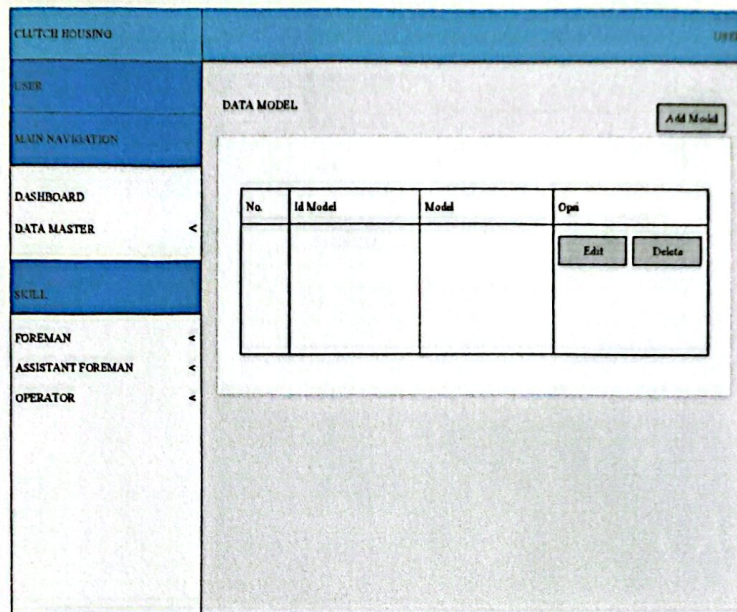
The image shows a web application interface for editing a data role. On the left is a sidebar menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area is titled 'Edit Data Role' and contains a form with the following elements:

- Form title: Edit Data Role
- Field: Id Role
- Field: Role
- Buttons: Batal, Update
- Table with columns: No., Role, Update

Gambar V.33 Tampilan Halaman *Edit Role*  
Sumber: Hasil Analisis (2020)

## 9. Halaman Model

Halaman *model* adalah halaman yang dapat diakses dengan hak akses sebagai admin, pada halaman *model* dapat melihat *list* data model yang ada. Berikut halaman model yang dapat dilihat pada Gambar V.34.



Gambar V.34 Tampilan Halaman Model  
Sumber: Hasil Analisis (2020)

#### 10. Halaman *Add Model*

Halaman *add model* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *add model*, admin dapat membuat model baru dengan mengisi *form* yang ada. Berikut halaman *add model* yang dapat dilihat pada Gambar V.35.

The screenshot shows a web application interface. On the left is a sidebar menu with the following items: CLUTCH HOUSING, USER, ALAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area is titled 'DATA MODEL' and contains a modal window titled 'Registarsi Model Baru'. This modal has three input fields: 'No.', 'Id Model', and 'Model'. Below these fields are two buttons: 'Batal' and 'Simpan'.

Gambar V.35 Tampilan Halaman *Add Model*  
 Sumber: Hasil Analisis (2020)

#### 11. Halaman *Edit Model*

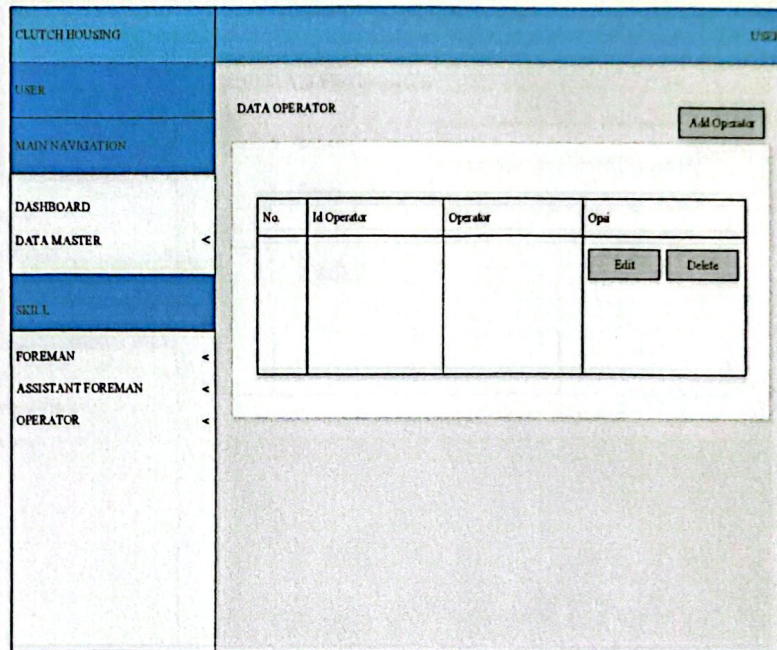
- Halaman *edit model* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *edit model*, admin dapat mengubah data model dengan mengisi *form* yang ingin diubah. Berikut halaman *edit model* yang dapat dilihat pada Gambar V.36.

The screenshot displays a web application interface. On the left, there is a vertical navigation menu with a blue header and white body. The menu items are: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area is a light gray background. A white modal window titled 'Edit Data Model' is centered on the screen. The modal contains a form with two input fields: 'Id Model' and 'Model'. Below the input fields are two buttons: 'Reset' and 'Update'. The modal is overlaid on a background that shows a table with columns and rows, partially obscured by the modal window.

Gambar V.36 Tampilan Halaman *Edit Model*  
Sumber: Hasil Analisis (2020)

## 12. Halaman *Operator*

Halaman *operator* adalah halaman yang dapat diakses dengan hak akses sebagai admin, pada halaman *operator* dapat melihat *list* data *operator* yang ada. Berikut halaman *operator* yang dapat dilihat pada Gambar V.37.



Gambar V.37 Tampilan Halaman *Operator*  
 Sumber: Hasil Analisis (2020)

### 13. Halaman *Add Operator*

Halaman *add operator* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *add operator*, admin dapat membuat *operator* baru dengan mengisi *form* yang ada. Berikut halaman *add operator* yang dapat dilihat pada Gambar V.38.

The image shows a web application interface with a sidebar menu on the left and a main content area on the right. The sidebar menu includes the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area displays a table with columns for 'No.', 'Data', and 'Simpan'. A modal form titled 'Registrasi Operator Baru' is overlaid on the table. The form contains the following fields and buttons: 'Id Operator' (text input), 'Operator' (text input), 'No.' (text input), 'Data' (button), and 'Simpan' (button). The form is positioned over the table, partially obscuring the 'No.' and 'Data' columns.

Gambar V.38 Tampilan Halaman *Add Operator*  
Sumber: Hasil Analisis (2020)

#### 14. Halaman *Edit Operator*

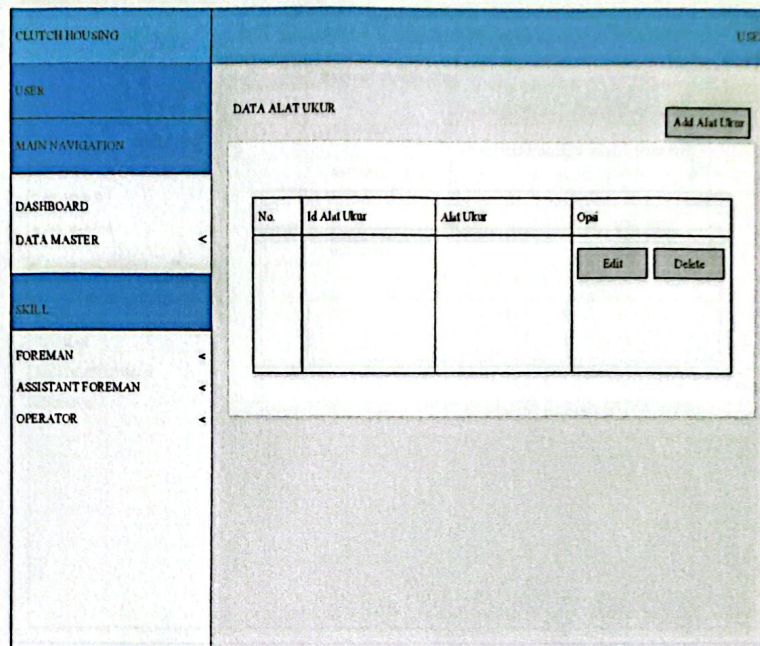
Halaman *edit operator* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *edit operator*, admin dapat mengubah data *operator* dengan mengisi *form* yang ingin diubah. Berikut halaman *edit operator* yang dapat dilihat pada Gambar V.39

The image shows a web application interface. On the left is a sidebar menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area displays a form titled 'Edit Data Operator'. The form contains two input fields: 'Id Operator' and 'Operator'. Below these fields are two buttons: 'Batal' and 'Update'. In the background, a table is partially visible with columns labeled 'No.', 'Id Operator', and 'Operator'. The table has several empty rows.

Gambar V.39 Tampilan Halaman *Edit Operator*  
Sumber: Hasil Analisis (2020)

#### 15. Halaman Alat Ukur

Halaman alat ukur adalah halaman yang dapat diakses dengan hak akses sebagai admin, pada halaman alat ukur dapat melihat *list* data alat ukur yang ada. Berikut halaman alat ukur yang dapat dilihat pada Gambar V.40.



Gambar V.40 Tampilan Halaman Alat Ukur  
Sumber: Hasil Analisis (2020)

#### 16. Halaman *Add* Alat Ukur

Halaman *add* alat ukur adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *add* alat ukur, admin dapat membuat alat ukur baru dengan mengisi *form* yang ada. Berikut halaman *add* alat ukur yang dapat dilihat pada Gambar V.41.

The image shows a web application interface. On the left is a vertical sidebar menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main area is titled 'DATA ALAT' and contains a table with columns 'No.', 'Id Alat Ukur', and 'Alat Ukur'. A modal window titled 'Registrasi Alat Ukur Baru' is open over the table. The modal contains two input fields: 'Id Alat Ukur' and 'Alat Ukur'. Below these fields are two buttons: 'Batal' (Cancel) and 'Simpan' (Save). The modal also has a 'Baru' button in the top right corner.

Gambar V.41 Tampilan Halaman *Add* Alat Ukur  
Sumber: Hasil Analisis (2020)

#### 17. Halaman *Edit Alat Ukur*

Halaman *edit* alat ukur adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *edit* alat ukur, admin dapat mengubah data alat ukur dengan mengisi *form* yang ingin diubah. Berikut halaman *edit* alat ukur yang dapat dilihat pada Gambar V.42.

The screenshot shows a web application interface. On the left is a sidebar menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area is titled 'Edit Data Alat Ukur'. It contains two input fields: 'Id Alat Ukur' and 'Alat Ukur'. Below these fields are two buttons: 'Batal' and 'Update'. The background of the main content area is a light gray color.

Gambar V.42 Tampilan Halaman *Edit Alat Ukur*  
 Sumber: Hasil Analisis (2020)

#### 18. Halaman *Item Check*

Halaman *item check* adalah halaman yang dapat diakses dengan hak akses sebagai admin, pada halaman alat ukur dapat melihat *list* data *item check* yang ada. Berikut halaman *item check* yang dapat dilihat pada Gambar V.43.

No.	Item check	Spec	Alat Ukur	Interval	Operator	Model	Opsi
							<input type="button" value="Edit"/> <input type="button" value="Delete"/>

Gambar V.43 Tampilan Halaman *Item Check*  
 Sumber: Hasil Analisis (2020)

#### 19. Halaman *Add Item Check*

Halaman *add item check* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *add item check*, admin dapat membuat *item check* baru dengan mengisi *form* yang ada. Berikut halaman *add item check* yang dapat dilihat pada Gambar V.44.

The image shows a web application interface. On the left is a sidebar menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area is titled 'DATA ITEM' and contains a table with two columns: 'No.' and 'Item'. Overlaid on this is a modal window titled 'Registrasi Item Check Baru'. The modal contains the following fields: 'Item Check', 'Spesifikasi', 'Alat Ukur', 'Interval', 'Operator', and 'Model'. At the bottom of the modal are two buttons: 'Batal' and 'Simpan'.

Gambar V.44 Tampilan Halaman *Item Check*  
 Sumber: Hasil Analisis (2020)

## 20. Halaman *Edit Item Check*

Halaman *edit item check* adalah halaman yang dapat diakses dengan hak akses sebagai admin. Pada halaman *edit item check*, admin dapat mengubah data *item check* dengan mengisi *form* yang ingin diubah. Berikut halaman *edit item check* yang dapat dilihat pada Gambar V.45.

The image shows a web application interface with a sidebar menu on the left and a main content area. The sidebar menu includes the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, DATA MASTER, SKILL, FOREMAN, ASSISTANT FOREMAN, and OPERATOR. The main content area displays a table with columns 'No' and 'Ber'. Overlaid on this is a form titled 'Edit Data Item Check'. The form contains the following fields: Id Check, Rem Check, Spesifikasi, Alat Ukur, Interval, Operator, and Model. At the bottom of the form are two buttons: 'Batal' and 'Update'.

Gambar V.45 Tampilan Halaman *Edit Item Check*  
Sumber: Hasil Analisis (2020)

## 21. Halaman *Checksheet*

Halaman *checksheet* adalah halaman yang dapat diakses dengan hak akses sebagai *operator*. Pada halaman *checksheet*, *operator* dapat melihat data *item check* pada *checksheet* dan menginputnya. Berikut halaman *checksheet* yang dapat dilihat pada Gambar V.46.

CLUTCH HOUSING	USER												
USER	<p>DATA ITEM CHECK</p> <table border="1"> <thead> <tr> <th>No.</th> <th>Item Check</th> <th>Spec</th> <th>Alat Ukur</th> <th>Interval</th> <th>Ops</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td> <input type="button" value="Edit"/> <input type="button" value="Complete"/> </td> </tr> </tbody> </table>	No.	Item Check	Spec	Alat Ukur	Interval	Ops						<input type="button" value="Edit"/> <input type="button" value="Complete"/>
No.		Item Check	Spec	Alat Ukur	Interval	Ops							
						<input type="button" value="Edit"/> <input type="button" value="Complete"/>							
MAIN NAVIGATION													
DASHBOARD													
ITEM CHECK													
CHECKSHEET													
MODEL M025 <													
MODEL M035 <													
LAPORAN CHECKSHEET													
MODEL M025 <													
MODEL M035 <													
SHIFT OPERATOR	<input type="button" value="Day"/> <input type="button" value="Night"/>												

Gambar V.46 Tampilan Halaman *Checksheet*  
 Sumber: Hasil Analisis (2020)

## 22. Halaman *Edit Checksheet*

Halaman *edit checksheet* adalah halaman yang dapat diakses dengan hak akses sebagai *operator*. Pada halaman *edit checksheet*, *operator* dapat menginput data *checksheet* sesuai dengan *item check* yang tersedia. Berikut halaman *edit checksheet* yang dapat dilihat pada Gambar V.47.

Gambar V.47 Tampilan Halaman *Edit Checksheet*  
Sumber: Hasil Analisis (2020)

### 23. Halaman Laporan *Checksheet*

Halaman laporan *checksheet* adalah halaman yang dapat diakses dengan hak akses sebagai *operator*, *As. Foreman*, dan *Foreman*. Pada halaman laporan *checksheet*, *user* dapat melihat data *checksheet* yang telah diinput. Berikut halaman laporan *checksheet* yang dapat dilihat pada Gambar V.48.

CLUTCH HOUSING	USER
USER	CHECKSHEET ITEM CHECK
MAIN NAVIGATION	
DASHBOARD	
ITEM CHECK	
CHECKSHEET SHIFT	
SUDAH DISETUJUI <	
BELUM DISETUJUI <	
LAPORAN CHECKSHEET	
MODEL M025 <	
MODEL M035 <	

No.	Item Check	First	1	2	dit

Gambar V.48 Tampilan Halaman Laporan *Checksheet*  
 Sumber: Hasil Analisis (2020)

#### 24. Halaman *Checksheet Shift* Belum Disetujui

Halaman *checksheet shift* belum disetujui adalah halaman yang dapat diakses dengan hak akses sebagai *As. Foreman*, dan *Foreman*. Pada halaman tersebut, *user* dapat mengaktifkan *checksheet* agar dapat melihat *checksheet* yang telah *diinput*. Berikut halaman *checksheet shift* belum disetujui yang dapat dilihat pada Gambar V.49.

CLUTCH HOUSING	USER													
USER	DATA CHECKSHEET <table border="1"> <thead> <tr> <th>No.</th> <th>Id Sheet</th> <th>Uaa</th> <th>Tanggal</th> <th>Opai</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td> <input type="button" value="Detail"/> <input type="button" value="Approved"/> </td> </tr> </tbody> </table>				No.	Id Sheet	Uaa	Tanggal	Opai					<input type="button" value="Detail"/> <input type="button" value="Approved"/>
No.					Id Sheet	Uaa	Tanggal	Opai						
								<input type="button" value="Detail"/> <input type="button" value="Approved"/>						
MAIN NAVIGATION														
DASHBOARD														
ITEM CHECK														
CHECKSHEET SHIFT														
SUDAH DISETUJUI <														
BELUM DISETUJUI <														
LAPORAN CHECKSHEET														
MODEL M025 <														
MODEL M035 <														

Gambar V.49 Tampilan Halaman *Checksheet Shift* Belum Disetujui  
 Sumber: Hasil Analisis (2020)

## 25. Halaman *Checksheet* Aktif

Halaman *checksheet* aktif adalah halaman yang dapat diakses dengan hak akses sebagai *As. Foreman*, dan *Foreman*. Pada halaman tersebut, *user* akan mendapatkan informasi bahwa *checksheet* tersebut sudah aktif. Berikut halaman *checksheet* aktif yang dapat dilihat pada Gambar V.50.

No	Id Sheet	User	Tanggal	Opai
				Detail Approved

Gambar V.50 Tampilan Halaman *Checksheet* Aktif  
Sumber: Hasil Analisis (2020)

## 26. Halaman *Checksheet Shift* Sudah Disetujui

Halaman *checksheet shift* sudah disetujui adalah halaman yang dapat diakses dengan hak akses sebagai *As. Foreman*, dan *Foreman*. Pada halaman tersebut, *user* akan memvalidasi *checksheet* tersebut. Berikut halaman *checksheet shift* sudah disetujui yang dapat dilihat pada Gambar V.51.

CLUTCH HOUSING		USER																																									
USER		DATA CHECK SHEET <table border="1"> <thead> <tr> <th>No.</th> <th>Id Sheet</th> <th>User</th> <th>Tanggal</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table>		No.	Id Sheet	User	Tanggal																																				
No.	Id Sheet			User	Tanggal																																						
MAIN NAVIGATION																																											
DASHBOARD																																											
ITEM CHECK																																											
CHECKSHEET SHIFT																																											
SUDAH DISETUJUI	<																																										
BELUM DISETUJUI	<																																										
LAPORAN CHECKSHEET																																											
MODEL M025	<																																										
MODEL M035	<																																										

Gambar V.51 Tampilan Halaman *Checksheet Shift* Sudah Disetujui  
 Sumber: Hasil Analisis (2020)

27. Halaman *Data List Reject* dan *Pending* M025 dan M035

Halaman *data list reject* dan *pending* adalah halaman yang dapat diakses dengan hak akses sebagai *As. Foreman*. Pada halaman tersebut, *user* akan menginput *form* tersebut. Berikut halaman *data list reject* dan *pending* yang dapat dilihat pada Gambar V.52.

No.	Operator	Kods	Jumlah	Keterangan	Opa
					Edit Delete

Gambar V.52 Tampilan Halaman *Data List Reject dan Pending* M025 dan M035

Sumber: Hasil Analisis (2020)

## 28. Halaman *Add Reject* M025 dan M035

Halaman *add reject* dan *pending* adalah halaman yang dapat diakses dengan hak akses sebagai *As. Foreman*. Pada halaman tersebut, *user* akan menginput *form* tersebut. Berikut halaman *add reject* dan *pending* yang dapat dilihat pada Gambar V.53.

The image shows a web application interface with a sidebar on the left and a main content area. The sidebar contains the following items:

- CLUTCH HOUSING
- USER
- MAIN NAVIGATION
- DASHBOARD
- ITEM CHECK
- CHECKSHEET SHIFT
- SUDAH DISETUJUI <
- BELUM DISETUJUI <
- LIST REJECT DAN PENDING
- M025 <
- M035 <
- LAPORAN CHECKSHEET
- MODEL M025 <
- MODEL M035 <

The main content area displays a form titled "Registrasi List Reject dan Pending Baru". The form has the following fields:

- Model
- List
- Operator
- Tanggal
- Kode
- Jumlah
- Keterangan
- Penyebab
- Tindakan

At the bottom of the form are two buttons: "Batal" and "Simpan". On the right side of the form, there are buttons for "Reject" and "Delete".

Gambar V.53 Tampilan Halaman *Add Reject* M025 dan M035  
Sumber: Hasil Analisis (2020)

29. Halaman *Edit Reject* dan *Pending* M025 dan M035

Halaman *edit reject* dan *pending* adalah halaman yang dapat diakses dengan hak akses sebagai *As. Foreman*. Pada halaman tersebut, *user* akan mengedit *form* tersebut. Berikut halaman *edit reject* dan *pending* yang dapat dilihat pada Gambar V.54.

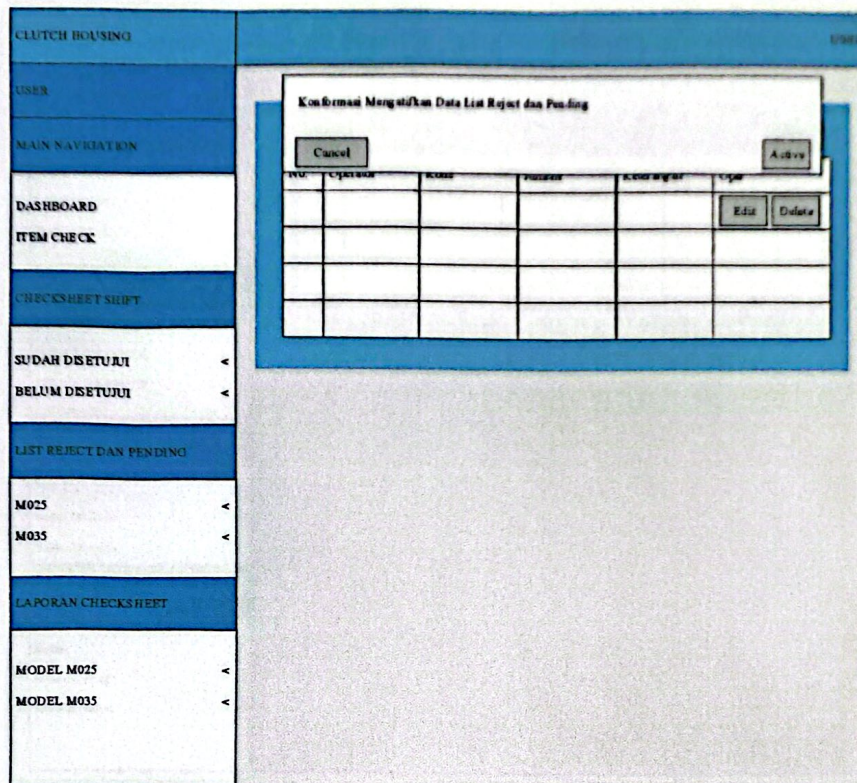
The image shows a web application interface. On the left is a sidebar menu with the following items: CLUTCH HOUSING, USER, MAIN NAVIGATION, DASHBOARD, ITEM CHECK, CHECKSHEET SHIFT, SUDAH DISETUJUI, BELUM DISETUJUI, LIST REJECT DAN PENDING, and LAPORAN CHECKSHEET. The 'LIST REJECT DAN PENDING' section is expanded, showing sub-items M025 and M035. In the center, a modal window titled 'Edit Data List Reject dan Pending' is open. It contains a form with the following fields: Model, List, Operator, Tanggal, Kode, Jumlah, Keterangan, Penyebab, and Tindakan. At the bottom of the modal are two buttons: 'Batal' and 'Update'.

Gambar V.54 Tampilan Halaman *Edit Reject dan Pending* M025 dan M035

Sumber: Hasil analisis (2020)

### 30. Halaman *List Reject dan Pending* Aktif

Halaman *list reject dan pending* aktif adalah halaman yang dapat diakses dengan hak akses sebagai *Foreman*. Pada halaman tersebut, *user* akan mendapatkan informasi bahwa *list* tersebut sudah aktif. Berikut halaman *list reject dan pending* aktif yang dapat dilihat pada Gambar V.55.



Gambar V.46 Tampilan Halaman *List Reject dan Pending* Aktif  
Sumber: Hasil Analisis (2020)

### 31. Halaman Rekap *List Reject dan Pending*

Halaman rekap *reject dan pending* adalah halaman yang dapat diakses dengan hak akses sebagai *Foreman*. Pada halaman tersebut, *user* akan mendapatkan informasi rekap *list reject dan pending*. Berikut halaman rekap *list reject dan pending* yang dapat dilihat pada Gambar V.55.

CLUSTER HOUSING		USER	
USER			
MAIN NAVIGATION			
DASHBOARD			
ITEM CHECK			
CHECKSHEET SHIFT			
SUDAH DISETUIJI <			
BELUM DISETUIJI <			
LIST REJECT DAN PENDING			
Pilih Data Reject <			
Sudah Disediakan <			
Belum Disediakan <			
LAPORAN CHECKSHEET			
Rakap			
MODEL M025 <			
MODEL M035 <			

Data Checksheet			
No.	Tanggal	Quantity	Keterangan
1			
2			
3			

Gambar V.55 Tampilan Halaman Rekapian *List Reject* dan *Pending*  
 Sumber: Hasil Analisis (2020)

### 5.6 Pengujian *Black Box Testing*

Pengujian *black box testing* digunakan untuk menguji perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. *Black box testing* menguji sistem berdasarkan *use case* yang telah dibuat, berikut pengujian yang dilakukan terhadap sistem informasi *monitoring* hasil produksi:

#### 1. *Functional Test Use Case Login*

Berikut merupakan *functional test use case login* yang dapat dilihat pada Tabel V.28.

*Test Case ID* : Login001

*Function* : Operasi validasi saat melakukan *login*.

- Data Assumption** : Fungsi operasi validasi saat melakukan login berjalan dengan baik. Penggunaan huruf kapital dan huruf kecil tidak mempengaruhi pada saat validasi login.
- Deskripsi** : Melakukan *login* ke dalam sistem dengan menguji kesalahan *username* dan *password*.

Tabel V.28 Test Case Login

Test ID	Test Case	Description	Expected Result	Actual Result
001	Validasi <i>login</i> .	<i>Username</i> dan <i>password</i> tidak diisi dan mengklik tombol <i>login</i> .	Tidak dapat <i>login</i> , lalu program mengasih tau data yang harus diisi.	Sesuai.
002	Validasi <i>login</i> .	<i>Input</i> dengan salah satu antara <i>username</i> dan <i>password</i> yang benar, lalu mengklik tombol <i>login</i> .	Tidak dapat <i>login</i> , lalu program menampilkan pesan bahwa <i>username</i> dan <i>password</i> tidak sesuai.	Sesuai.
003	Validasi <i>login</i> .	<i>Input username</i> dan <i>password</i> dengan data yang benar.	Berhasil <i>login</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

## 2. Functional Test Mengelola Data User

Berikut merupakan *functional test* mengelola data *user* yang dapat dilihat pada Tabel V.29.

**Test Case ID** : Mengelola Data *User* 001

- Function** : Operasi saat menampilkan, mengubah, menambah, dan menghapus data *user*
- Data Assumption** : Fungsi operasi saat menampilkan data *user*, mengubah data *user*, menambah data *user*, dan menghapus data *user* berjalan dengan baik.
- Deskripsi** : Melakukan proses menampilkan data *user*, menambah data *user*, mengubah data *user*, dan menghapus data *user*.

Tabel V.29 Test Case Mengelola Data User

<b>Test ID</b>	<b>Test Case</b>	<b>Description</b>	<b>Expected Result</b>	<b>Actual Result</b>
001	Menampilkan data <i>user</i> .	Klik menu data <i>user</i> lalu menampilkan data <i>user</i> .	Berhasil menampilkan data <i>user</i> .	Sesuai.
002	Menambah data <i>user</i> .	Klik tombol <i>add</i> data <i>user</i> lalu menampilkan <i>form</i> registrasi <i>user</i> baru dan isi <i>form</i> tersebut sesuai data.	Berhasil menambahkan <i>user</i> .	Sesuai.
003	Menghapus data <i>user</i> .	Klik tombol <i>delete</i> pada salah satu data <i>user</i> yang dipilih, dan data yang dipilih terhapus.	Berhasil menghapus <i>user</i> .	Sesuai.

Tabel V.30 *Test Case* Mengelola Data *User* (lanjutan)

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
004	Mengubah data <i>user</i> .	Klik tombol edit pada salah satu data <i>user</i> yang dipilih dan menampilkan data <i>user</i> yang dipilih.	Berhasil mengubah data <i>user</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

### 3. *Functional Test* Mengelola Data *Role*

Berikut merupakan *functional test* mengelola data *role* yang dapat dilihat pada Tabel V.30.

<i>Test Case ID</i>	: Mengelola Data <i>Role</i> 001
<i>Function</i>	: Operasi saat menampilkan, mengubah, menambah, dan menghapus data <i>role</i>
<i>Data Assumption</i>	: Fungsi operasi saat menampilkan data <i>role</i> , mengubah data <i>role</i> , menambah data <i>role</i> , dan menghapus data <i>role</i> berjalan dengan baik.
Deskripsi	: Melakukan proses menampilkan data <i>role</i> , menambah data <i>role</i> , mengubah data <i>role</i> , dan menghapus data <i>role</i> .

Tabel V.30 *Test Case* Mengelola Data *Role*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Menampilkan data <i>role</i> .	Klik menu data <i>role</i> lalu menampilkan data <i>role</i> .	Berhasil menampilkan data <i>role</i> .	Sesuai.
002	Menambah data <i>role</i> .	Klik tombol <i>add</i> data <i>role</i> lalu menampilkan <i>form</i> registrasi <i>role</i> baru dan isi <i>form</i> tersebut sesuai data.	Berhasil menambahkan <i>role</i> .	Sesuai.
003	Menghapus data <i>role</i> .	Klik tombol <i>delete</i> pada salah satu data <i>role</i> yang dipilih, dan data yang dipilih terhapus.	Berhasil menghapus <i>role</i> .	Sesuai.
004	Mengubah data <i>role</i> .	Klik tombol <i>edit</i> pada salah satu data <i>role</i> yang dipilih dan menampilkan data <i>role</i> yang dipilih.	Berhasil mengubah data <i>role</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

#### 4. *Functional Test* Mengelola Data Model

Berikut merupakan *functional test* mengelola data model yang dapat dilihat pada Tabel V.31.

- Test Case ID** : Mengelola Data Model 001
- Function** : Operasi saat menampilkan, mengubah, menambah, dan menghapus data model
- Data Assumption** : Fungsi operasi saat menampilkan data model, mengubah data model, menambah data model, dan menghapus data model berjalan dengan baik.
- Deskripsi** : Melakukan proses menampilkan data model, menambah data model, mengubah data model, dan menghapus data model.

Tabel V.31 Test Case Mengelola Data Model

<b>Test ID</b>	<b>Test Case</b>	<b>Description</b>	<b>Expected Result</b>	<b>Actual Result</b>
001	Menampilkan data model.	Klik menu data model lalu menampilkan data model.	Berhasil menampilkan data model.	Sesuai.
002	Menambah data model.	Klik tombol <i>add</i> data model lalu menampilkan <i>form</i> registrasi model baru dan isi <i>form</i> tersebut sesuai data.	Berhasil menambahkan model.	Sesuai.
003	Menghapus data model.	Klik tombol <i>delete</i> pada salah satu data model yang dipilih, dan data yang dipilih terhapus.	Berhasil menghapus model.	Sesuai.

Tabel V.31 *Test Case* Mengelola Data Model (lanjutan)

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
004	Mengubah data model.	Klik tombol <i>edit</i> pada salah satu data model yang dipilih dan menampilkan data model yang dipilih.	Berhasil mengubah data model.	Sesuai.

Sumber: Hasil Analisis (2020)

#### 5. *Functional Test* Mengelola Data Operator

Berikut merupakan *functional test* mengelola data *operator* yang dapat dilihat pada Tabel V.32.

- Test Case ID* : Mengelola Data Operator 001
- Function* : Operasi saat menampilkan, mengubah, menambah, dan menghapus data *operator*
- Data Assumption* : Fungsi operasi saat menampilkan data *operator*, mengubah data *operator*, menambah data *operator*, dan menghapus data *operator* berjalan dengan baik.
- Deskripsi : Melakukan proses menampilkan data *operator*, menambah data *operator*, mengubah data *operator*, dan menghapus data *operator*.

Tabel V.32 *Test Case* Mengelola Data Operator

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Menampilkan data operator.	Klik menu data operator lalu menampilkan data operator.	Berhasil menampilkan data operator.	Sesuai.
002	Menambah data operator.	Klik tombol <i>add</i> data operator lalu menampilkan form registrasi operator baru dan isi form tersebut sesuai data.	Berhasil menambahkan operator.	Sesuai.
003	Menghapus data operator.	Klik tombol <i>delete</i> pada salah satu data operator yang dipilih, dan data yang dipilih terhapus.	Berhasil menghapus operator.	Sesuai.
004	Mengubah data operator.	Klik tombol <i>edit</i> pada salah satu data operator yang dipilih dan menampilkan data operator yang dipilih.	Berhasil mengubah data operator.	Sesuai.

Sumber: Hasil Analisis (2020)

#### 6. *Functional Test* Mengelola Alat Ukur

Berikut merupakan *functional test* mengelola alat ukur yang dapat dilihat pada Tabel V.33.

- Test Case ID** : Mengelola Alat Ukur 001
- Function** : Operasi saat menampilkan, mengubah, menambah, dan menghapus data alat ukur.
- Data Assumption** : Fungsi operasi saat menampilkan data alat ukur, mengubah data alat ukur, menambah data alat ukur, dan menghapus data alat ukur berjalan dengan baik.
- Deskripsi** : Melakukan proses menampilkan data alat ukur, menambah data alat ukur, mengubah data alat ukur, dan menghapus data alat ukur.

Tabel V.33 Test Case Mengelola Alat Ukur

<b>Test ID</b>	<b>Test Case</b>	<b>Description</b>	<b>Expected Result</b>	<b>Actual Result</b>
001	Menampilkan data alat ukur.	Klik menu data alat ukur lalu menampilkan data alat ukur.	Berhasil menampilkan data alat ukur.	Sesuai.
002	Menambah data alat ukur.	Klik tombol <i>add</i> data alat ukur lalu menampilkan <i>form</i> registrasi alat ukur baru dan isi <i>form</i> tersebut sesuai data.	Berhasil menambahkan alat ukur.	Sesuai.
003	Menghapus data alat ukur.	Klik tombol <i>delete</i> pada salah satu data alat ukur yang dipilih, dan data yang dipilih terhapus.	Berhasil menghapus alat ukur.	Sesuai.

Tabel V.33 *Test Case* Mengelola Alat Ukur (lanjutan)

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
004	Mengubah data alat ukur.	Klik tombol <i>edit</i> pada salah satu data alat ukur yang dipilih dan menampilkan data alat ukur yang dipilih.	Berhasil mengubah data alat ukur.	Sesuai.

Sumber: Hasil Analisis (2020)

#### 7. *Functional Test* Mengelola *Item Check*

Berikut merupakan *functional test* mengelola *item check* yang dapat dilihat pada Tabel V.34.

<i>Test Case ID</i>	: Mengelola Data <i>Item Check</i> 001
<i>Function</i>	: Operasi saat menampilkan, mengubah, menambah, dan menghapus data <i>item check</i> .
<i>Data Assumption</i>	: Fungsi operasi saat menampilkan data <i>item check</i> , mengubah data <i>item check</i> , menambah data <i>item check</i> , dan menghapus data <i>item check</i> berjalan dengan baik.
Deskripsi	: Melakukan proses menampilkan data <i>item check</i> , menambah data <i>item check</i> , mengubah data <i>item check</i> , dan menghapus data alat ukur.

Tabel V.34 *Test Case Mengelola Item Check*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Menampilkan data <i>item check</i> .	Klik menu data <i>item check</i> lalu menampilkan data <i>item check</i> .	Berhasil menampilkan data <i>item check</i> .	Sesuai.
002	Menambah data <i>item check</i> .	Klik tombol <i>add data item check</i> lalu menampilkan <i>form</i> registrasi <i>item check</i> baru dan isi <i>form</i> tersebut sesuai data.	Berhasil menambahkan <i>item check</i> .	Sesuai.
003	Menghapus data <i>item check</i> .	Klik tombol <i>delete</i> pada salah satu data <i>item check</i> yang dipilih, dan data yang dipilih terhapus.	Berhasil menghapus <i>item check</i> .	Sesuai.
004	Mengubah data <i>item check</i> .	Klik tombol <i>edit</i> pada salah satu data <i>item check</i> yang dipilih dan menampilkan data <i>item check</i> yang dipilih.	Berhasil mengubah data <i>item check</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

#### 8. *Functional Test Input Data Checksheet*

Berikut merupakan *functional test input data checksheet* yang dapat dilihat pada Tabel V.35.

*Test Case ID* : *Input Data Checksheet 001*

*Function* : *Operasi saat menginput data checksheet.*

**Data Assumption** : Fungsi operasi saat menginput data *checksheet* berjalan dengan baik.

**Deskripsi** : Melakukan proses penginputan data *checksheet* harian.

Tabel V.35 *Test Case Input Data Checksheet*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Menampilkan <i>item check</i> yang harus diisi.	Klik menu <i>checksheet</i> , pilih model produknya, dan klik <i>checksheet</i> CL mana yang ingin diinput.	Berhasil menampilkan data <i>user</i> .	Sesuai.
002	Menginput <i>checksheet</i> .	Klik tombol <i>edit</i> untuk mengisi keterangan berupa angka, klik tombol <i>complete</i> untuk keterangan berupa "OK"	Berhasil menginput <i>checksheet</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

#### 9. *Functional Test* Melihat Data *Checksheet*

Berikut merupakan *functional test* melihat data *checksheet* yang dapat dilihat pada Tabel V.36.

**Test Case ID** : Melihat Data *Checksheet* 001

**Function** : Operasi saat melihat data *checksheet*.

**Data Assumption** : Fungsi operasi saat melihat data *checksheet* berjalan dengan baik.

**Deskripsi** : Melakukan proses untuk melihat data *checksheet* yang sudah diinput.

Tabel V.36 *Test Case* Melihat Data *Checksheet*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Melihat data <i>checksheet</i> .	Klik menu laporan <i>checksheet</i> , klik model produk, dan klik <i>checksheet</i> CL yang ingin dilihat.	Berhasil menampilkan data <i>checksheet</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

#### 10. *Functional Test* Mengaktifkan Data *Checksheet*

Berikut merupakan *functional test* mengaktifkan data *checksheet* yang dapat dilihat pada Tabel V.37.

- Test Case ID* : Mengaktifkan Data *Checksheet* 001
- Function* : Operasi saat mengaktifkan data *checksheet*.
- Data Assumption* : Fungsi operasi saat mengaktifkan data *checksheet* berjalan dengan baik.
- Deskripsi : Melakukan proses untuk mengaktifkan data *checksheet* agar dapat melihat data *checksheet* yang sudah diinput.

Tabel V.37 *Test Case* Mengaktifkan Data *Checksheet*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Melihat <i>checksheet</i> tanpa mengaktifkan.	Klik menu laporan <i>checksheet</i> , pilih model produknya, dan klik CL <i>checksheet</i> yang ingin dilihat.	Berhasil menampilkan <i>checksheet</i> tanpa hasil <i>inputan</i> .	Sesuai.

Tabel V.37 *Test Case* Mengaktifkan Data *Checksheet* (lanjutan)

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
002	Mengaktifkan <i>checksheet</i> .	Klik menu <i>checksheet shift</i> , pilih sub menu belum disetujui, pilih shiftnya. Klik <i>detail</i> .	Berhasil mengaktifkan <i>checksheet</i> .	Sesuai.
003	Melihat <i>checksheet</i> setelah <i>checksheet</i> aktif	Klik menu laporan <i>checksheet</i> , pilih model produknya, dan klik CL <i>checksheet</i> yang ingin dilihat.	Berhasil menampilkan <i>checksheet</i> yang sudah <i>input</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

#### 11. *Functional Test* Validasi Data *Checksheet*

Berikut merupakan *functional test* validasi data *checksheet* yang dapat dilihat pada Tabel V.38.

<i>Test Case ID</i>	: Validasi Data <i>Checksheet</i> 001
<i>Function</i>	: Operasi saat memvalidai data <i>checksheet</i> .
<i>Data Assumption</i>	: Fungsi operasi saat memvalidasi data <i>checksheet</i> berjalan dengan baik.
Deskripsi	: Melakukan proses untuk memvalidasi data <i>checksheet</i> yang sudah diisi.

Tabel V.38 Test Case Validasi Data *Checksheet*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Memvalidasi data <i>checksheet</i>	Klik menu <i>checksheet shift</i> , pilih sub menu belum disetujui, pilih <i>shiftnya</i> , klik tombol <i>approved</i> .	Berhasil memvalidasi <i>checksheet</i> .	Sesuai.
002	Melihat <i>checksheet</i> tervalidasi.	Pilih menu <i>checksheet shift</i> , pilih sub menu sudah setuju, pilih <i>shiftnya</i> .	Berhasil melihat data <i>checksheet</i> yang sudah divalidasi.	Sesuai.

Sumber: Hasil Analisis (2020)

## 12. Functional Test Input List Reject or Pending

Berikut merupakan *functional test input list reject or pending* yang dapat dilihat pada Tabel V.39.

<i>Test Case ID</i>	: <i>Input List Reject or Pending</i> 001
<i>Function</i>	: Operasi saat menginput <i>List Reject or Pending</i>
<i>Data Assumption</i>	: Fungsi operasi saat menginput <i>List Reject or Pending</i> berjalan dengan baik.
Deskripsi	: Melakukan proses penginputan <i>List Reject or Pending</i> harian.

Tabel V.39 *Test Case Input List Reject or Pending*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Menampilkan <i>list reject</i> dan <i>pending</i> yang harus diisi.	Klik menu <i>list reject dan pending</i> , pilih model produknya.	Berhasil menampilkan <i>list reject dan pending</i> .	Sesuai.
002	Menginput <i>list reject</i> dan <i>pending</i> .	Klik tombol <i>add</i> untuk mengisi <i>form</i> , lalu isi <i>form tersebut</i> .	Berhasil menginput <i>list reject dan pending</i> .	Sesuai.

Sumber: Hasil Analisis (2020)

### 13. *Functional Test Validasi List Reject or Pending*

Berikut merupakan *functional test validasi list reject or pending* yang dapat dilihat pada Tabel V.40.

- Test Case ID* : Validasi *List Reject or Pending* 001
- Function* : Operasi saat memvalidai *list reject or pending*.
- Data Assumption* : Fungsi operasi saat memvalidasi *list reject or pending* berjalan dengan baik.
- Deskripsi : Melakukan proses untuk memvalidasi *list reject or pending* yang sudah diisi.

Tabel V.40 *Test Case Validasi List Reject or Pending*

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Memvalidasi <i>list reject or pending</i> .	Klik menu <i>list reject dan pending</i> , pilih sub menu pilih data reject yang belum disetujui, klik tombol <i>approve</i> .	Berhasil memvalidasi <i>list reject or pending</i> .	Sesuai.
002	Melihat <i>list reject or pending</i> tervalidasi.	Klik menu <i>list reject dan pending</i> , pilih sub menu pilih data reject yang sudah disetujui.	Berhasil melihat <i>list reject or pending</i> yang sudah divalidasi.	Sesuai.

Sumber: Hasil Analisis (2020)

#### 14. *Functional Test* Rekap LPP dan LPR

Berikut merupakan *functional test* rekap LPP dan LPR yang dapat dilihat pada Tabel V.41.

- Test Case ID* : Rekap LPP dan LPR 001
- Function* : Operasi saat merekap LPP dan LPR
- Data Assumption* : Fungsi operasi saat merekap LPP dan LPR berjalan dengan baik.
- Deskripsi : Melakukan proses untuk merekap LPP dan LPR yang sudah diisi.

Tabel V.41 *Test Case* Rekap LPP dan LPR

<i>Test ID</i>	<i>Test Case</i>	<i>Description</i>	<i>Expected Result</i>	<i>Actual Result</i>
001	Melihat <i>list reject or pending</i> tervalidasi.	Klik menu <i>list reject dan pending</i> , pilih sub menu pilih data reject yang sudah disetujui, klik approve	Berhasil me <input/> LPP dan LPR ke dalam rekapan.	Sesuai.
002	Melihat Rekap LPP dan LPR	Klik menu rekap.	Berhasil melihat hasil rekapan.	Sesuai.

Sumber: Hasil Analisis (2020)

## BAB VI

### KESIMPULAN DAN SARAN

#### 6.1 Kesimpulan

Berdasarkan hasil pengumpulan, pengolahan, dan analisis data yang telah dilakukan dalam penelitian Tugas Akhir ini dapat disimpulkan beberapa hal yaitu:

1. Sistem informasi *monitoring* hasil produksi *clutch housing* merupakan proses penyimpanan *checksheet* yang pengisiannya dilakukan oleh *Operator* juga *list part pending*, dan *list part reject* yang pengisiannya dilakukan oleh *As. Foreman* serta *Foreman* sebagai aktor yang memvalidasi ketiga data tersebut.
2. Sistem informasi *monitoring* hasil produksi *clutch housing* juga merupakan sistem yang bertujuan untuk meminimalisasi kerentanan hilangnya dokumen *checksheet*, *list part pending*, dan *list part reject*.
3. Sistem informasi *monitoring* hasil produksi *clutch housing* dalam pengisian *checksheet*, *list part pending*, dan *list part reject* sudah terotomatisasi.

#### 6.2 Saran

Saran yang diberikan untuk pengembangan sistem informasi *monitoring* hasil produksi *clutch housing* selanjutnya adalah berikut:

1. Membuat sebuah fungsi untuk mengisi *checksheet*, *list part pending*, dan *list part reject* menggunakan *handphone*.
2. Mencantumkan tanggal pengisian *checksheet* agar dapat mengisi *checksheet* hari sebelumnya pada hari berikutnya guna menghindari *human error* atas kelupaan dalam mengisi *checksheet* pada hari tersebut.

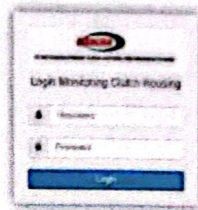
## DAFTAR PUSTAKA

- Anggraeni, E. Y., & Irviani, R. (2017). *Pengantar Sistem Informasi*. Yogyakarta: CV ANDI OFFSET.
- Anhar. (2010). *Panduan Menguasai PHP & MYSQL Secara Otodidak*. Jakarta: Mediakita.
- Assauri, S. (2008). *Manajemen Operasi dan Produksi*. Jakarta: Lembaga penerbit Fakultas Ekonomi Universitas Indonesia.
- Chaerulhaki, R. (n.d.). *Bab II Landasan Teori*. Retrieved from Studylib Mercu Buana: <https://studylibid.com/doc/1087451/bab-ii-landasan-teori>
- Dennis, A., Wixom, B. H., & Roth, M. R. (2012). *System Analysis And Design*. New Jersey: John Willey & Sons, Inc.
- Erizal. (2018). *PEMANTAUAN PELAKSANAAN KEGIATAN*. Retrieved from BPSDM: [https://bpsdm.pu.go.id/center/pelatihan/uploads/edok/2018/10/57f08\\_5c9fb\\_Draft\\_Tayangan\\_Pemantauan\\_Pelaksanaan\\_Kegiatan\\_\\_Diklat\\_Semarang\\_.pdf](https://bpsdm.pu.go.id/center/pelatihan/uploads/edok/2018/10/57f08_5c9fb_Draft_Tayangan_Pemantauan_Pelaksanaan_Kegiatan__Diklat_Semarang_.pdf)
- Fatta, H. A. (2007). *Analisis dan Perancangan Sistem Informasi untuk keunggulan Bersaing Perusahaan dan Organisasi Modern*. Yogyakarta: CV ANDI OFFSET.
- Fitrah, M., & Luthfiah. (2017). *Metodologi Penelitian; Penelitian Kualitatif, Tindakan Kelas & Studi Kasus*. Sukabumi : CV Jejak.
- Hoffer, J. A., Venkataraman, R., & Topi, H. (2015). *Modern Database Management*. New Jersey: Pearson Education.
- Hutahaean, J. (2015). *Konsep Sistem Informasi*. Yogyakarta: CV BUDI UTAMA.

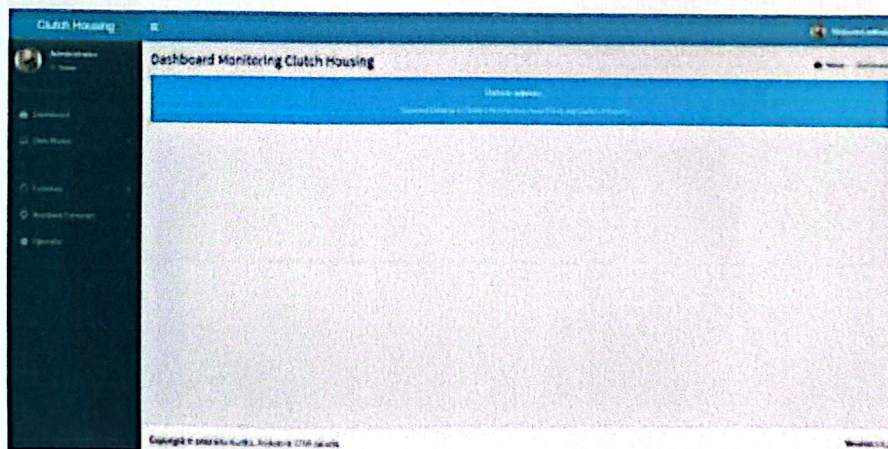
- Jalinus, N., & Ambiyar. (2016). *Media dan Sumber Pembelajaran*. Jakarta: Kencana.
- Jogiyanto, H. (2010). *Analisis dan Desain Sistem Informasi*. Yogyakarta: CV ANDI OFFSET.
- MKM, P. (2017). *MKM Profile*. Jakarta: PT Mitsubishi Krama Yudha Motors and Manufacturing.
- Moerdiyanto. (2004). *Teknik Monitoring dan Evaluasi (MONEV) Dalam Rangka Memperoleh Informasi Untuk Pengambilan Keputusan Manajemen*. Retrieved from Staff UNY: <http://staff.uny.ac.id>
- R, W. (2008). *Monitoring, Evaluasi, dan Pengendalian*. Jakarta: PT Gramedia.
- Rozi, Z. A., & Community, S. (2015). *Bootsrap Design Framework*. Jakarta: PT Elex Media Komputindo.
- S, R. A., & Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak*. Bandung: Informatika Bandung.
- Sukamto, R. A., & Shalahuddin, M. (2013). *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Modula.
- Sutabri, T. (2012). *Analisa Sistem Informasi*. Yogyakarta: CV ANDI OFFSET.
- Sutabri, T. (2012). *Analisis Sistem Informasi*. Yogyakarta: CV ANDI OFFSET.
- Yudhanto, Y., & Prasetyo, H. A. (2018). *Panduan Mudah Belajar Framework Laravel*. Jakarta: PT Elex Media.

## LAMPIRAN A TAMPILAN PROGRAM

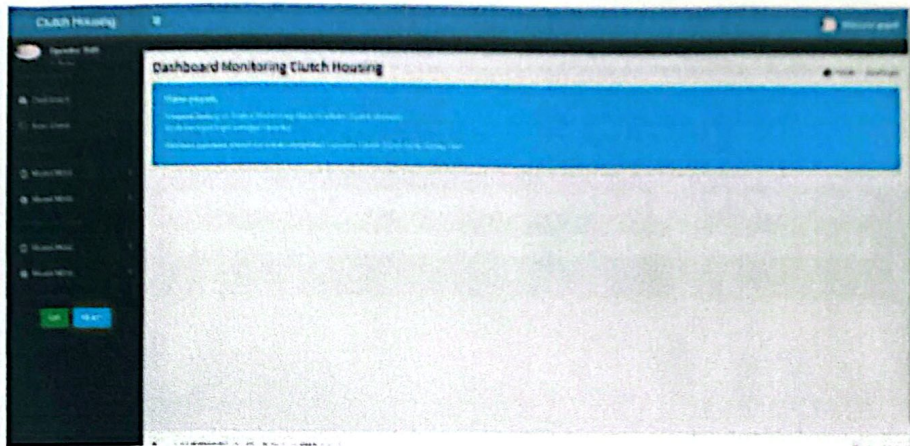
### 1. *Login*



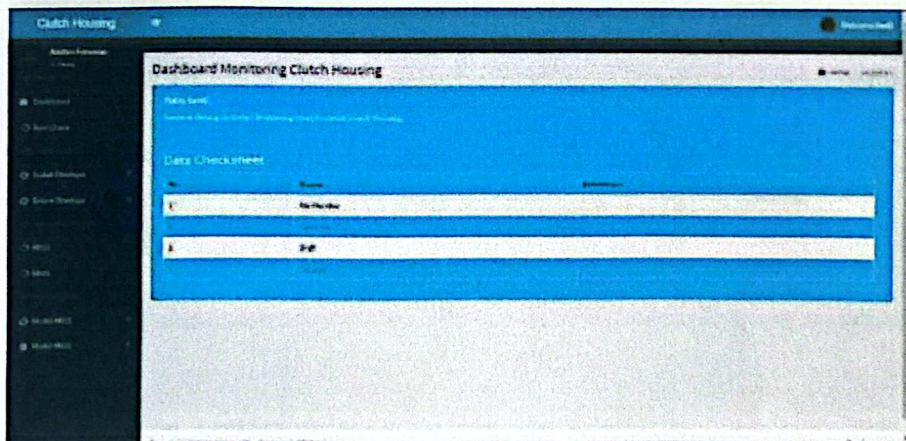
### 2. *Dashboard Admin*



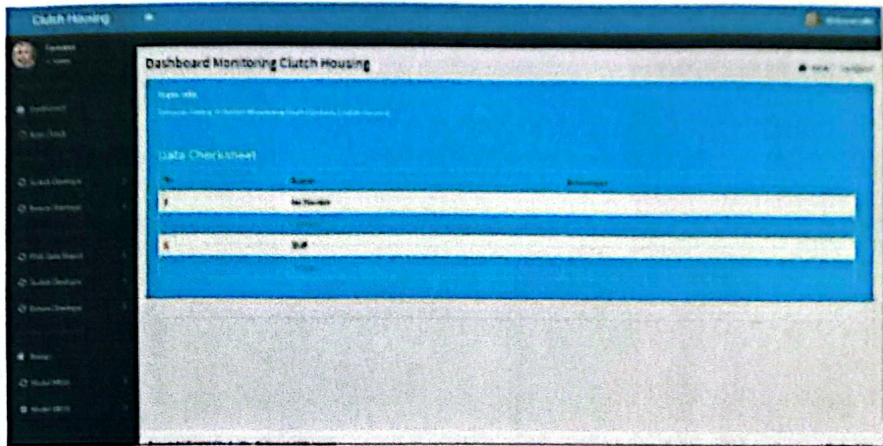
### 3. *Dashboard Operator*



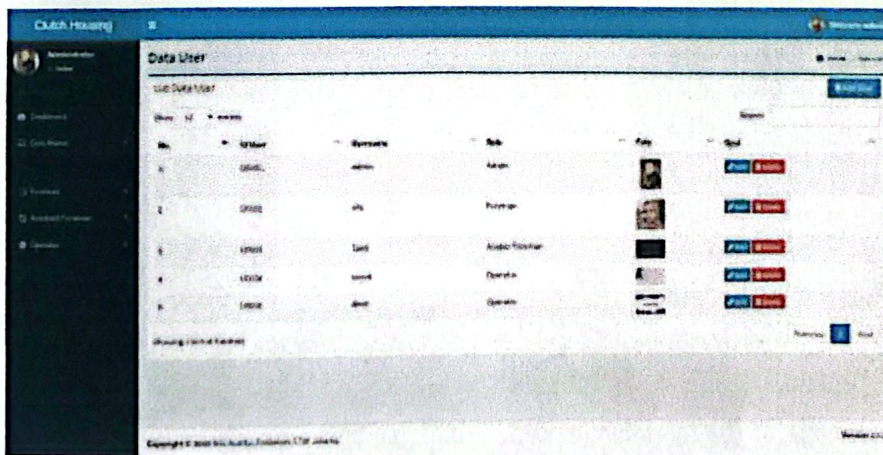
### 4. *Dashboard Assistant Foreman*



5. *Dashboard Foreman*



6. *Form User*



7. *Form Add User*

The screenshot shows a modal window titled "Registrasi User Baru" (New User Registration) overlaid on a dark background. The form contains the following fields and elements:

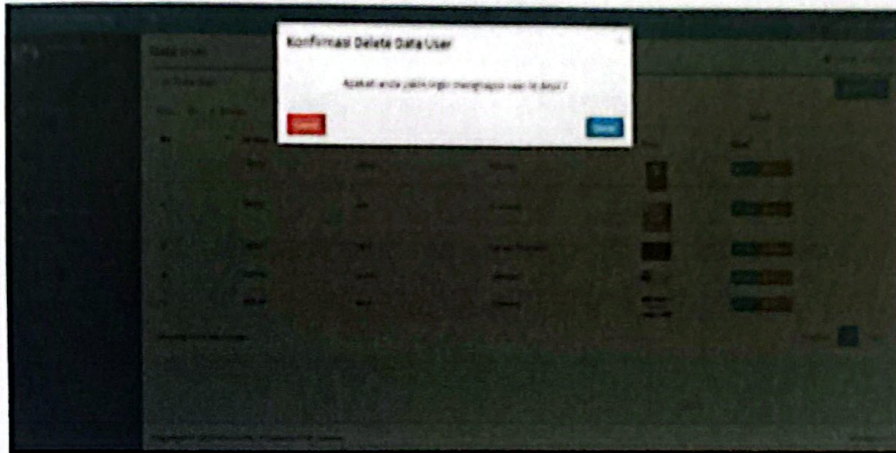
- Username\***: A text input field containing the value "admin".
- Password\***: A password input field containing the value "1234".
- User Role\***: A dropdown menu with "Admin" selected.
- Avatar\***: A field with a small profile picture icon and a "Choose File" button.
- Buttons**: A red "Kembali" (Back) button on the left and a blue "Simpan" (Save) button on the right.

8. *Form Edit User*

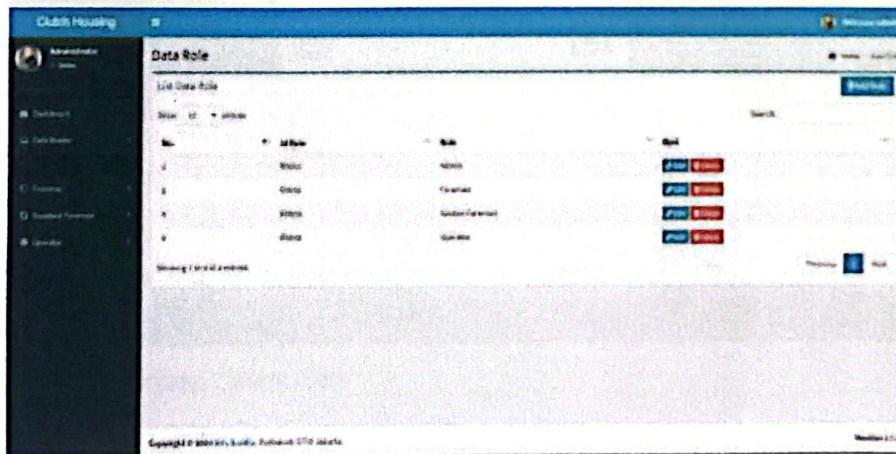
The screenshot shows a modal window titled "Edit Data User" overlaid on a dark background. The form contains the following fields and elements:

- Username\***: A text input field containing the value "admin".
- Password\***: A password input field containing the value "1234".
- User Role\***: A dropdown menu with "Admin" selected.
- Avatar\***: A field with a small profile picture icon and a "Choose File" button.
- Buttons**: A red "Kembali" (Back) button on the left and a blue "Simpan" (Save) button on the right.

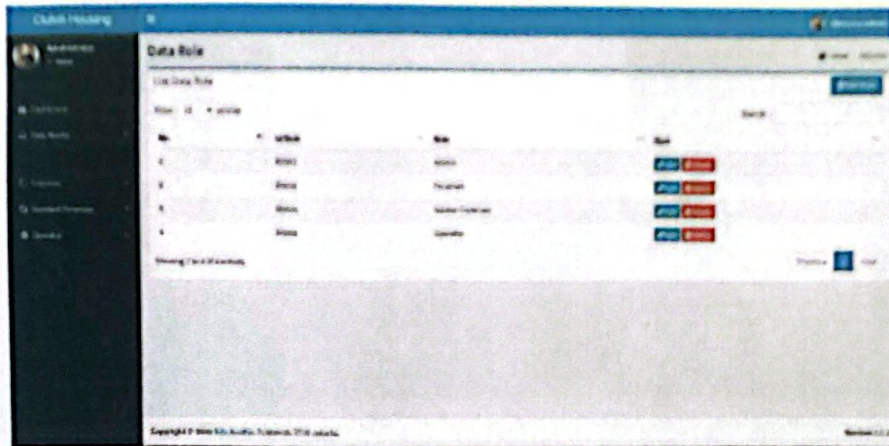
9. *Form Permintaan Delete User*



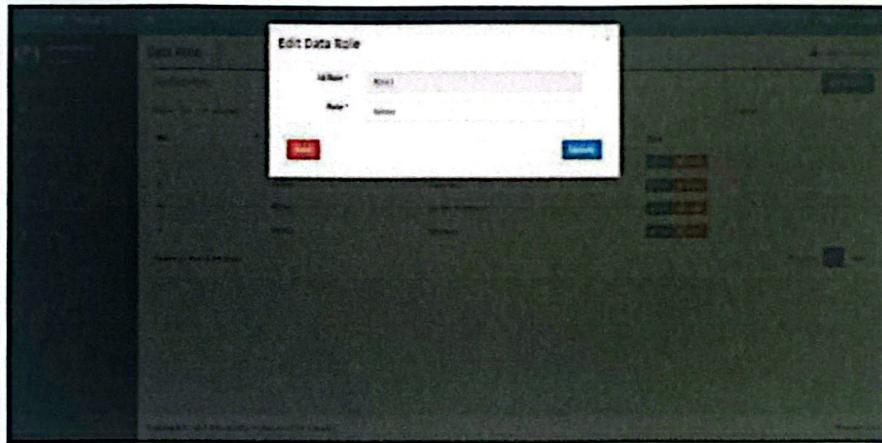
10. *Form Role*



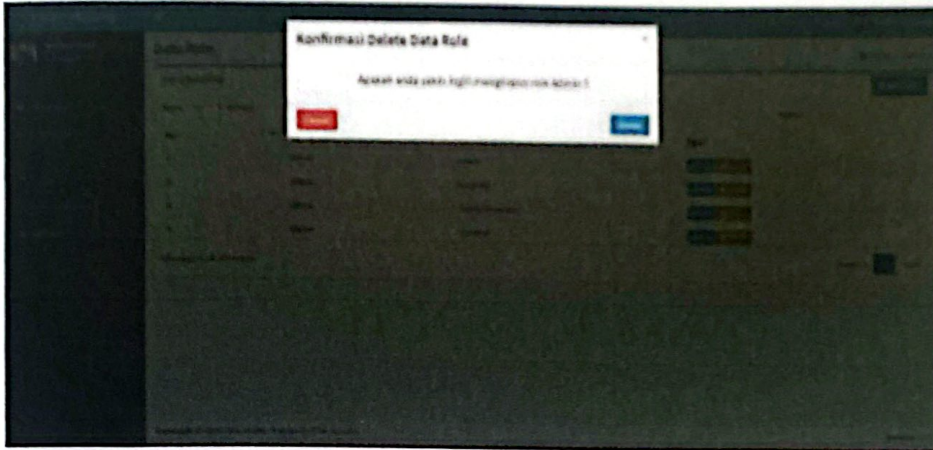
11. *Form Add Role*



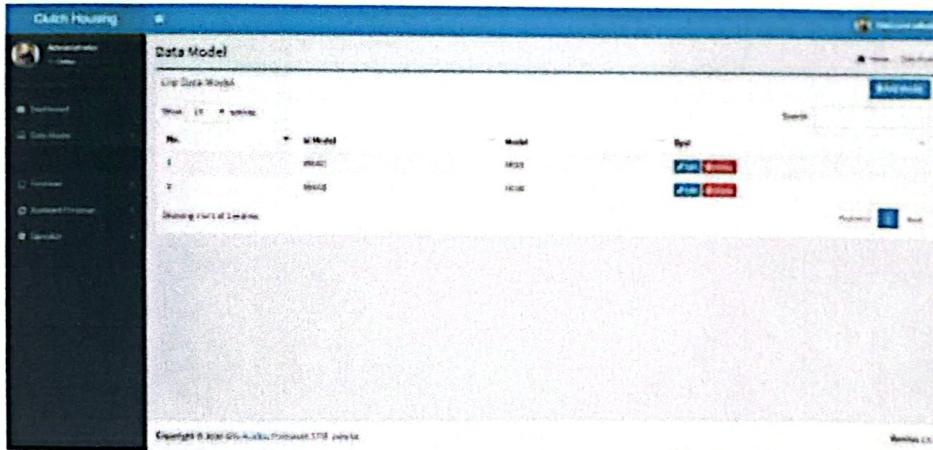
12. *Form Edit Role*



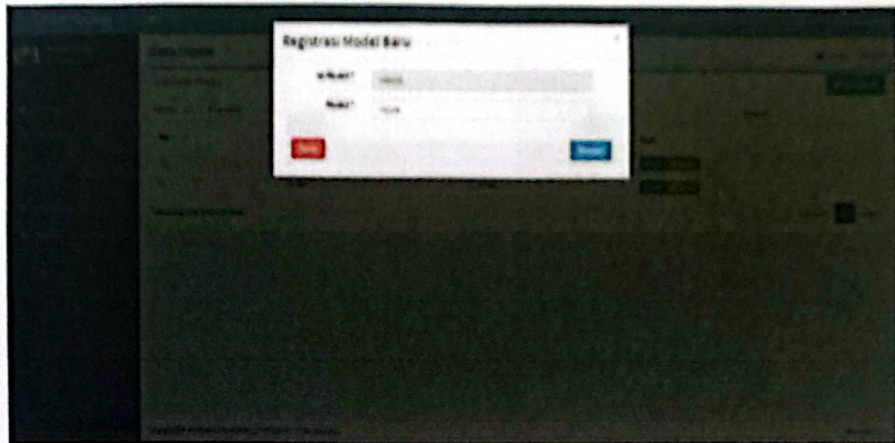
13. *Form Permintaan Delete Role*



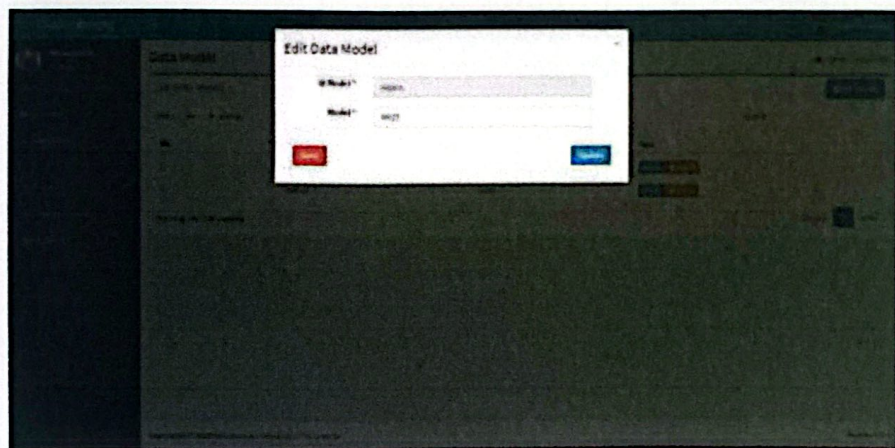
14. *Form Model*



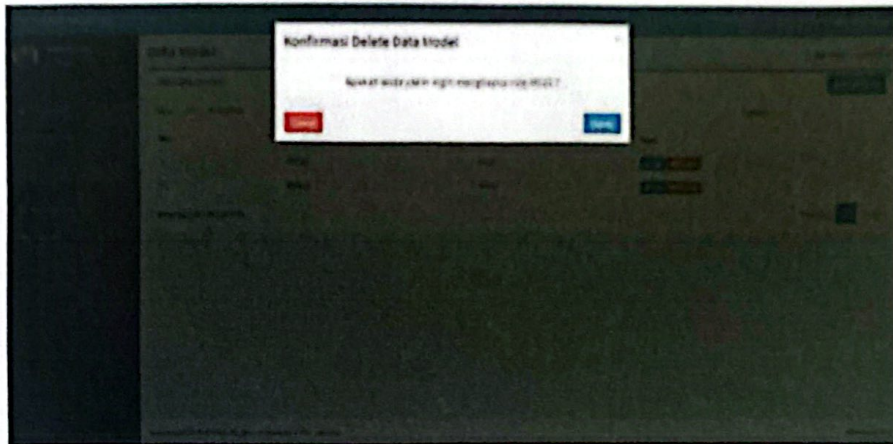
15. *Form Add Model*



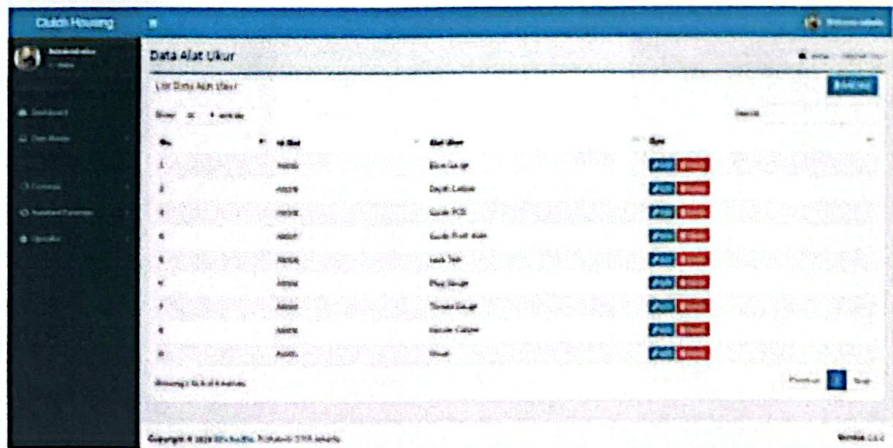
16. *Form Edit Model*



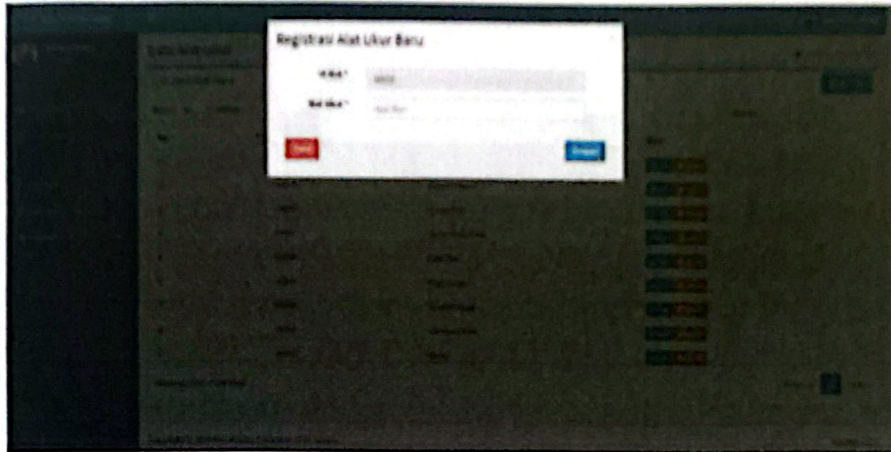
17. *Form Permintaan Delete Model*



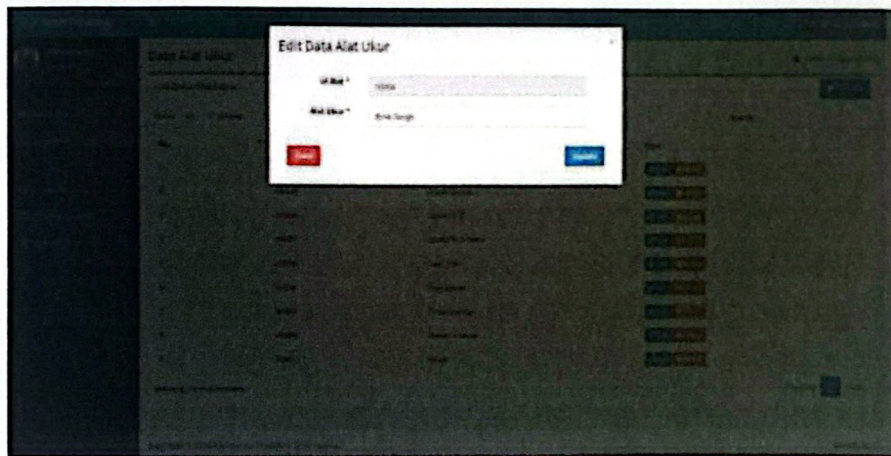
18. *Form Alat Ukur*



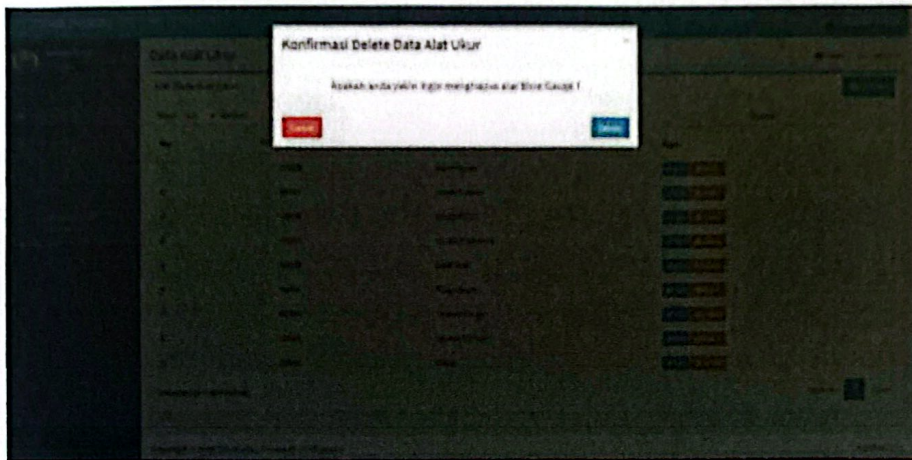
19. *Form Add Alat Ukur*



20. *Form Edit Alat Ukur*



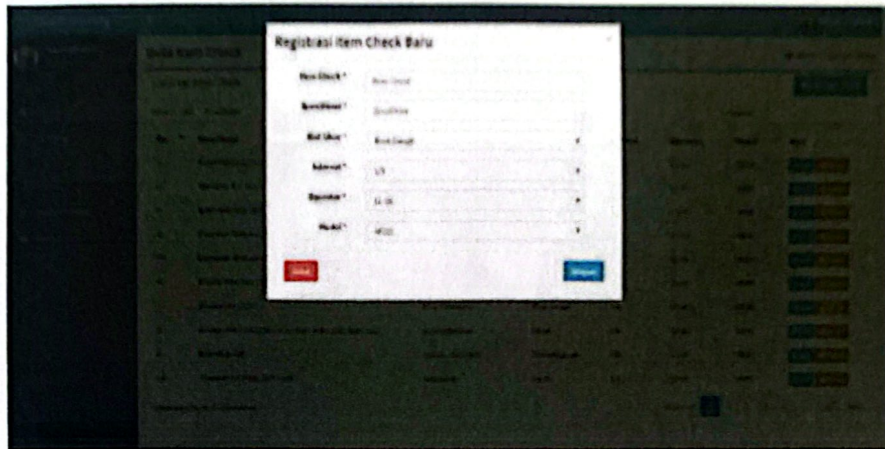
21. *Form Permintaan Delete Alat Ukur*



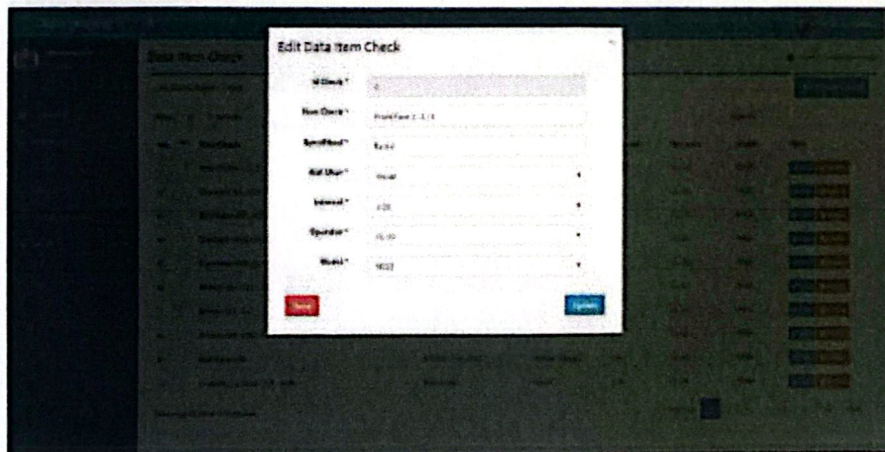
## 22. Form Item Check

No	Item Check	Item	Alat Ukur	Interval	Operator	Hasil	Detail
1	Kalibrasi 1 (20)	Alat	Vial	10	01.01	01.01	Detail
2	Kalibrasi 2 (20)	Alat	Open Range	10	01.01	01.01	Detail
3	Kalibrasi 3 (20, 100, 100)	M 20 + 10 + 10	Timed Range	10	01.01	01.01	Detail
4	Kalibrasi 4 (20, 100)	M 20 + 10, 100 + 100	PS g Gauge	10	01.01	01.01	Detail
5	Kalibrasi 5 (20, 100)	M 10 + 100 + 100	PS g Gauge	10	01.01	01.01	Detail
6	Kalibrasi 6 (10, 100)	M 10 + 100	PS g Gauge	10	01.01	01.01	Detail
7	Kalibrasi 7 (10)	M 10 + 10	PS g Gauge	10	01.01	01.01	Detail
8	Kalibrasi 8 (10, 100, 100, 100, 100)	M 10 + 100 + 100 + 100 + 100	PS g Gauge	10	01.01	01.01	Detail
9	Kalibrasi 9 (10)	M 10 + 10 + 10	Open Range	10	01.01	01.01	Detail
10	Kalibrasi 10 (10)	None	Vial	10	01.01	01.01	Detail

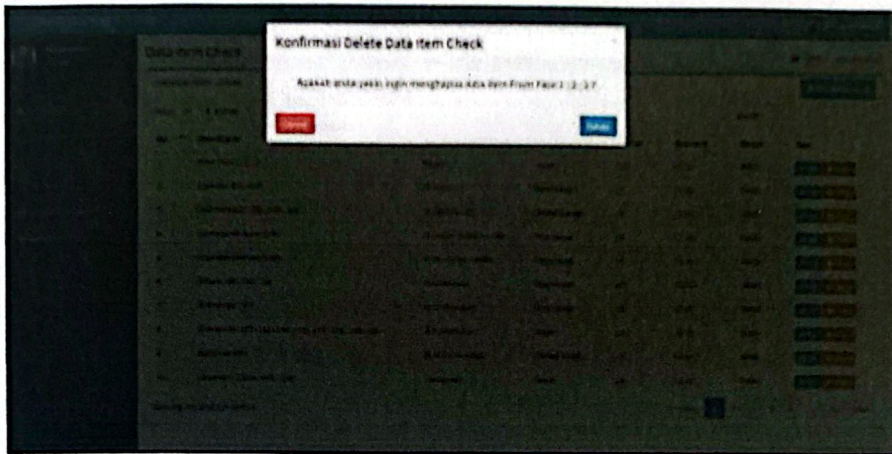
## 23. Form Add Item Check



24. *Form Edit Item Check*



25. *Form Permintaan Item Check*



26. *Form Checksheet*

No.	Nama Check	Spes	Bidang	Kategori	Opis
1	Form Poin 3.2.3.7	Hy 4.0	Hydrol	1.37	[Print] [Refresh]
2	Inspeksi R 1 (RSC)	078.00.01	Revisi Hidrol	1.4	[Print] [Refresh]
3	Buku HSK 101 (001, 004, 109)	0136 x 1.0 x 1.02	Teknik Hidrol	1.0	[Print] [Refresh]
4	Diagram Hidrol (07)	012107 + 0.01 + 0.00	Plan Hidrol	1.9	[Print] [Refresh]
5	Diagram Hidrol (10)	0121 + 0.00 + 0.00	Plan Hidrol	1.0	[Print] [Refresh]
6	Diagram (08) (11) (10)	0113.000.0	Plan Hidrol	1.0	[Print] [Refresh]
7	Diagram (10) (11)	0113.000.0	Plan Hidrol	1.0	[Print] [Refresh]
8	Diagram (10) (11) (11) (12) (13) (14) (15) (16) (17) (18) (19) (20)	0113.000.0	Hydrol	1.0	[Print] [Refresh]
9	Buku HSK 109	0113 x 1.70 x 2.0	Teknik Hidrol	1.0	[Print] [Refresh]
10	Diagram 1.0 (001) (002) (003)	Hydrol	Hydrol	1.0	[Print] [Refresh]

27. *Form Edit Checksheet*

The screenshot shows a web application interface with a table in the background and a modal form titled "Nilai Data Item Check" in the foreground. The form contains the following fields:

- Nilai Check \* (text input)
- Waktu Check \* (text input)
- Nomor Ruang \* (text input)
- Mata Ujian \* (text input)
- Nilai \* (text input)
- Petimbangan \* (text input)

At the bottom of the form are two buttons: a red "Batal" (Cancel) button and a blue "Simpan" (Save) button.

28. *Form Input Hasil Checksheet "OK"*

The screenshot shows a web application interface with a table in the background and a modal dialog box titled "Konfirmasi Data item Check" in the foreground. The dialog box contains the following text:

Konfirmasi Data item Check

Apakah data item Check Form No. 1, 2, 3 sudah lengkap?

At the bottom of the dialog box are two buttons: a red "Batal" (Cancel) button and a blue "Ya" (Yes) button.

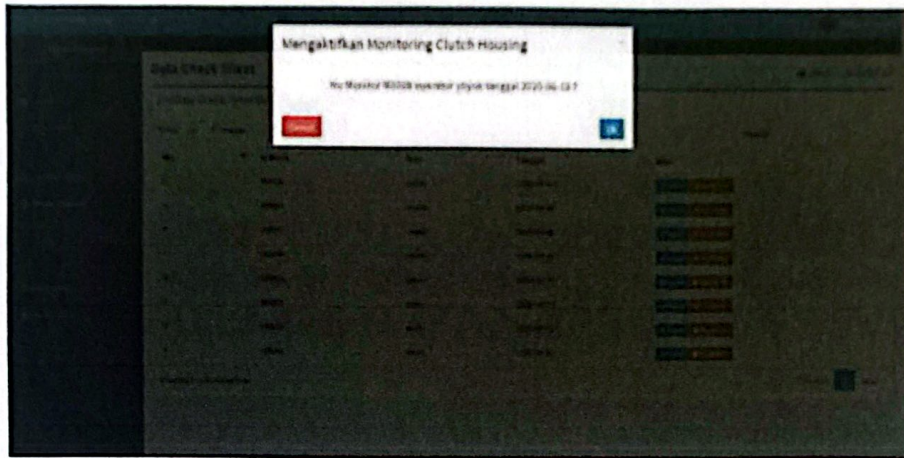
29. Form Laporan Checksheet

No.	Item Check	YTD	Target
1	Account 1 (101)	00	00
2	Account 2 (102)	00	00
3	Account 3 (103)	00	00
4	Account 4 (104)	00	00
5	Account 5 (105)	00	00
6	Account 6 (106)	00	00
7	Account 7 (107)	00	00
8	Account 8 (108)	00	00
9	Account 9 (109)	00	00
10	Account 10 (110)	00	00

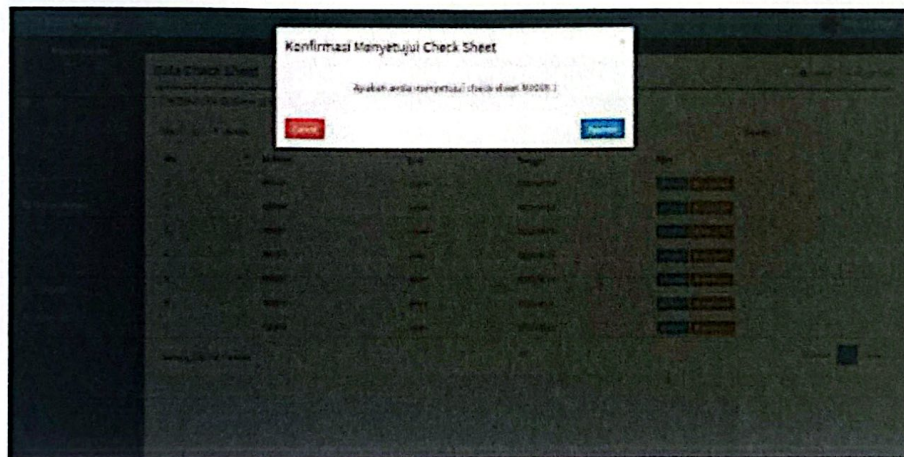
30. Form Checksheet Belum Disetujui

No.	Item	User	Tanggal	Status
1	10001	10001	2020-08-14	Belum Disetujui
2	10002	10002	2020-08-14	Belum Disetujui
3	10003	10003	2020-08-14	Belum Disetujui
4	10004	10004	2020-08-14	Belum Disetujui
5	10005	10005	2020-08-14	Belum Disetujui
6	10006	10006	2020-08-14	Belum Disetujui
7	10007	10007	2020-08-14	Belum Disetujui
8	10008	10008	2020-08-14	Belum Disetujui

31. *Form Permintaan Checksheet* Diaktifkan



32. *Form Permintaan Checksheet* Disetujui



33. *Form Checksheet* Sudah disetujui

**Data Check Sheet**

List Data Check Sheet Shift Day Sabtu-Diizinkan

No	Al Barel	Tanggal
1	1500	2020

Showing 1 of 1 entries

34. *Data List Reject dan Pending* M025 dan M035

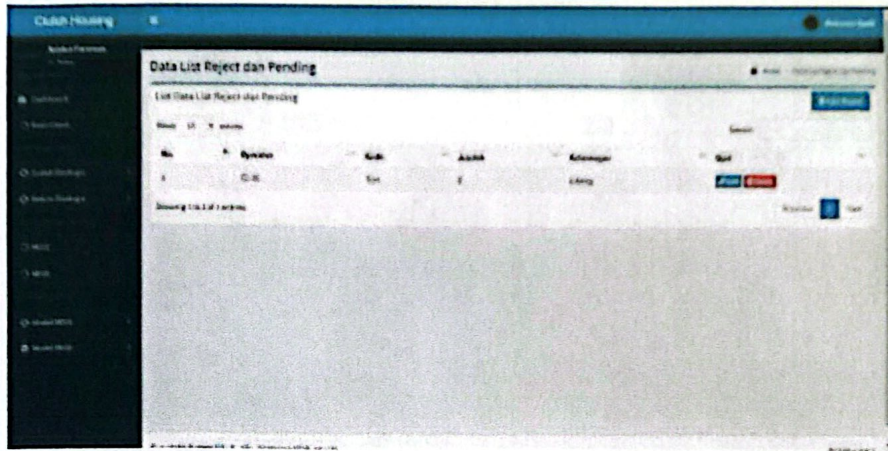
**Data List Reject dan Pending**

List Data List Reject dan Pending

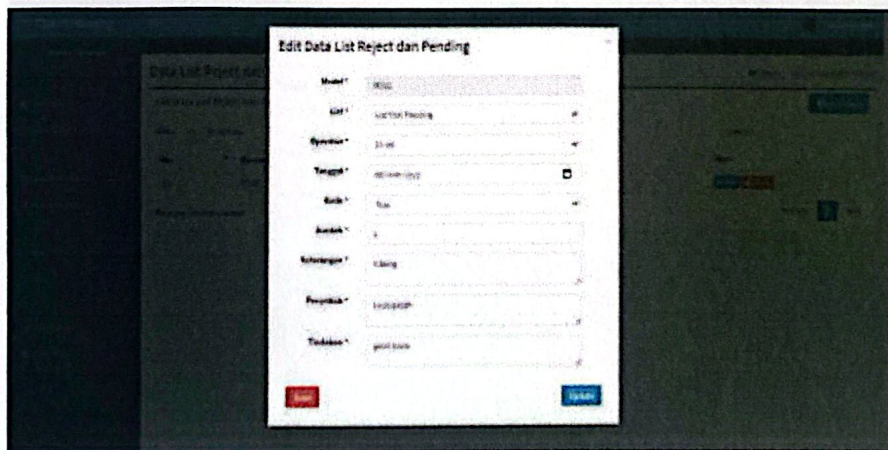
No	Operator	Kuda	Jumlah	Adhimagar	Quit
1	G. B.	100	5	10000	10000

Showing 1 of 1 entries

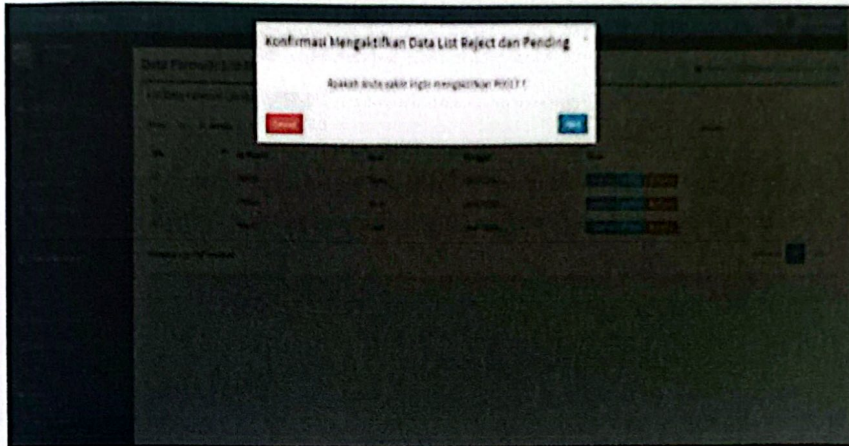
35. Add Reject M025 dan M035



36. Edit Reject dan Pending M025 dan M035



### 37. List Reject dan Pending Aktif



### 38. Rekap List Reject dan Pending

Rekap List Reject dan Pending

Rekap List Reject dan Pending

No.	Tanggal	Quantity	Keterangan
1	21-01-2024	1	List Reject Pending
2	21-01-2024	1	List Reject Final
3	21-01-2024	12	List Reject Proses
4	21-01-2024	2	List Reject Proses
5	20-01-2024	18	List Reject Proses
6	20-01-2024	9	List Reject Proses

Showing 6 list of 6 items

## LAMPIRAN B

### KODE PROGRAM

#### 1. Proses Login

```
<?php
if(isset($_POST['submit']))
{
    include 'koneksidb.php';

    $username = $_POST['username'];
    $pass = $_POST['password'];

    echo $username;
    echo $pass;

    $query = mysqli_query($koneksi, "SELECT * FROM tb_us
er a, tb_role b WHERE username = '". $username. "' AND passwor
d = '". $pass. "' and a.id_role=b.id_role ");

    $data = mysqli_fetch_array($query);
    $user_login = $data['username'];
    $user_role = $data['role'];

    if (mysqli_num_rows($query)>0)
    {
        session_start();
        $_SESSION['username'] = $username;
        $_SESSION['user_role'] = $user_role;

        echo "berhasil login $user_role";
        if ($user_role == 'Admin')
        {
            header('location: admin/admin.php');
        }
        elseif ($user_role == 'Operator')
```

```

        {
            header('location: admin/operator.php');
        }
    }
    else
    {
        echo "Username / password anda salah";
    }
}
?>

```

## 2. Logout

```

<?php
session_start();
session_destroy();
header('location:index.php');
?>

```

## 3. Lembaran *checksheet* untuk di-input

```

<?php
include 'koneksidb.php';
require_once("genid.php");
//$id_operator=$_GET['op'];
//$id_model=$_GET['model'];
$id_operator="CL-10";
$id_model="M025";
$queryitem=mysqli_query($koneksi,"select * from tb_check a,
tb_operator b, tb_model c where a.id_operator=b.id_operator
and a.id_model=c.id_model and operator='$id_operator' and m
odel='$id_model'");
$jumlahitem=mysqli_num_rows($queryitem);
$item=array();
if($jumlahitem>0)
{
    while($rowitem=mysqli_fetch_assoc($queryitem))

```

```

    {
        array_push($item, $rowitem['item']);
    }
}
echo "$item[0] <br>";
echo "$item[1] <br>";
echo "$item[2] <br>";
echo "$item[3] <br>";
echo "$item[4] <br>";
echo "$item[5] <br>";
echo "$item[6] <br>";
echo "$item[7] <br>";
echo "$item[8] <br>";
echo "$item[9] <br>";

$variabel=array("item0","item1","item2","item3","item4","item5","item6","item7","item8","item9");
$tot=array();
$jmlkolom=0;
for($i=0;$i<$jumlahitem;$i++)
{
    $variabel[$i]=array();
    $querycek=mysqli_query($koneksi,"select * from tb_detail_monitor a, tb_check b where a.id_check=b.id_check and b.item='$item[$i]'");
    if(mysqli_num_rows($querycek)>0)
    {
        while($rowcek=mysqli_fetch_assoc($querycek))
        {
            array_push($variabel[$i], $rowcek['keterangan']);
        }
        if(count($variabel[$i])>0)
        {
            array_push($tot,count($variabel[$i]));
        }
        else
        {
            array_push($tot,0);
        }
        if($jmlkolom<$tot[$i])
        {

```

```

    $jmlkolom=$tot[$i];
  }
}
}
echo "$tot[0] <br>";
echo "$tot[1] <br>";
echo "$tot[2] <br>";
echo "$tot[3] <br>";
echo "$tot[4] <br>";
echo "$tot[5] <br>";
echo "$tot[6] <br>";
echo "$tot[7] <br>";
echo "$tot[8] <br>";
echo "$tot[9] <br>";

echo "total kolom $jmlkolom";
?>

```

```

<table>
  <thead>
    <tr>
      <th>No.</th>
      <th>Check Item</th>
      <th>Firt</th>
      <th>1</th>
      <th>2</th>
      <th>3</th>
      <th>4</th>
      <th>5</th>
      <th>6</th>
      <th>7</th>
      <th>8</th>
      <th>9</th>
      <th>10</th>
      <th>11</th>
      <th>12</th>
      <th>13</th>
      <th>14</th>
      <th>15</th>
      <th>16</th>
      <th>17</th>

```

```

        <th>18</th>
        <th>19</th>
        <th>20</th>
        <th>21</th>
        <th>22</th>
        <th>23</th>
        <th>24</th>
        <th>25</th>
        <th>26</th>
        <th>27</th>
        <th>28</th>
        <th>29</th>
        <th>30</th>
        <th>Opsi</th>
    </tr>
</thead>
<tbody>
    <?php
        $no = 1;
        for($i=0;$i<$jumlahitem;$i++)
        {

            ?>
            <tr>
                <td><?php echo $no++;?></td>
                <td><?php echo $item[$i];?></td>
            <?php
                if($tot[$i]==0)
                {

                }
                else
                {
                    for($jml=0;$jml<$tot[$i];$jml++)
                    {
                        ?>
                        <td><?php echo $variabel[$i][$jml];?></t
d>

                    <?php
                        }
                    }
                }
            }
        }
    }

```

```

    }
    ?>
    <td>
    <!--
    <a href="../user/form_edituser.php?id=<?php echo $row['id_us
    er']?>" class="btn btn-primary btn-flat btn-
    xs"><i class="fa fa-pencil"></i> Edit</a-->
    <a href="#" class="btn btn-
    primary btn-flat btn-xs" data-toggle="modal" data-
    target="#updatecheck<?php echo $no; ?>"><i class="fa fa-
    pencil"></i> Edit</a>
    <a href="#" class="btn btn-danger btn-
    flat btn-xs" data-toggle="modal" data-
    target="#deletecheck<?php echo $no; ?>"><i class="fa fa-
    trash"></i> Delete</a>
    <?php
    }
    ?>
    </td>
  </tr>
</tbody>
</table>

```

#### 4. Data *checksheet* yang sudah di-input

```

<?php
error_reporting(0);
session_start();
$tgl=date("Y-m-d");
if($_SESSION['user_role']=="Operator")
{
include '../template/header_operator.php';
}
else if($_SESSION['user_role']=="Admin")
{
include '../template/header_admin.php';
}
else if($_SESSION['user_role']=="Foreman")
{

```

```

include '../template/header_foreman.php';
}
else
{
include '../template/header_asisten.php';
}

include '../koneksidb.php';
require_once("../genid.php");
$id_operator=$_GET['op'];
$id_model=$_GET['model'];
$jmlkolom=0;
$queryitem=mysqli_query($koneksi,"select * from tb_check
a, tb_operator b, tb_model c where a.id_operator=b.id_operat
or and a.id_model=c.id_model and operator='$id_operator' and
model='$id_model'");
$jumlahitem=mysqli_num_rows($queryitem);
$item=array();
if($jumlahitem>0)
{
while($rowitem=mysqli_fetch_assoc($queryitem))
{
array_push($item, $rowitem['item']);
}
}

$variabel=array("item0","item1","item2","item3","item4","ite
m5","item6","item7","item8","item9");
$tot=array();
for($i=0;$i<$jumlahitem;$i++)
{
$variabel[$i]=array();
if($_SESSION['user_role']=="Operator")
{
$querycek=mysqli_query($koneksi,"select * from tb_detail_m
onitor a, tb_check b, tb_monitor c, tb_operator d where a.id
_check=b.id_check and b.item='$item[$i]' and c.tanggal='$tgl
' and c.id_monitor=a.id_monitor and c.id_monitor='$_SESSION[
id_monitor]' and b.id_operator=d.id_operator and d.operator=
'$id_operator'");

```

```

}
else
{
    $querycek=mysqli_query($koneksi,"select * from tb_detail_monitor a, tb_check b, tb_monitor c where a.id_check=b.id_check and b.item='$item[$i]' and c.id_monitor=a.id_monitor and c.id_monitor='$_SESSION[id]'");
}
if(mysqli_num_rows($querycek)>0)
{
    while($rowcek=mysqli_fetch_assoc($querycek))
    {
        array_push($variabel[$i], $rowcek['keterangan']);
    }
    array_push($tot,count($variabel[$i]));
}
else
{
    array_push($tot,0);
}
if($jmlkolom<$tot[$i])
{
    $jmlkolom=$tot[$i];
}
}
?>

```

```

<div class="content-wrapper">
<section class="content-header">
    <h1>Checksheet Item Check
    </h1>
    <ol class="breadcrumb">
    <li><a href="../admin/admin.php"><i class="fa fa-dashboard"></i> Home</a></li>
    <li class="active">Checksheet Item Check</li>
    </ol>
</section>
<!-- Main content -->
<section class="content">

```

```

<div class="row">
  <div class="col-sm-12">
    <div class="box box-primary">
      <div class="box-header with-border">
        <h3 class="box-title">List Data Item Check </h3>
        <h4>Checksheet Tanggal : <?php
          $querytgl=mysqli_query($koneksi,"select * from tb_mon
itor where id_monitor='$_SESSION[id]'");
          if(mysqli_num_rows($querytgl)>0)
          {
            $rowtgl=mysqli_fetch_assoc($querytgl);
            echo date('d-m-Y',strtotime($rowtgl['tanggal']));
          }
        ?></h4>

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

        <div class="table-responsive22">
          <table id="tabel-data" class="table table-
bordered table-striped">
            <thead>
              <tr>
                <th>No.</th>
                <th>Item Check</th>
              </tr>
            </thead>
            <tbody>
              <?php
                for($k=0;$k<$jmlkolom;$k++)
                {
                  echo "<th>";
                  if($k==0)
                  {
                    echo "First";
                  }
                  else
                  {
                    echo "$k";
                  }
                  echo "</th>";
                }
              ?>
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>

```

```

        </tr>
    </thead>
    <tbody>
        <?php
            $no = 1;
            for($i=0;$i<$jumlahitem;$i++)
            {

                ?>
                <tr>
                    <td><?php echo $no++;?></td>
                    <td><?php echo $item[$i];?></td>
                <?php
                    if($tot[$i]==0)
                    {

                    }
                    else
                    {
                        for($jml=0;$jml<$tot[$i];$jml++)
                        {
                            ?>
                            <td><?php echo $variabel[$i][$jml];?></t
d>
                            <?php
                                }
                            }

                            $selisih=$jmlkolom-$tot[$i];
                            for($t=0;$t<$selisih;$t++)
                            {
                                echo "<td>&nbsp;</td>";
                            }
                            ?>
                        </td>
                    </tr>
                <?php
                    }
                ?>
            </tbody>

```

```

        </table>
    </div>
</div>
<script>
$(document).ready(function(){
    $('#tabel-data').DataTable();
});
</script>
</div>
</div>
</div>
</section><!-- /.content -->
</div>

<?php include '../template/footer.php'; ?>

```

##### 5. Tombol untuk *shift*

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Belajar Javascript</title>
<script>
    function Day(){
        alert("Shift Operator DAY");
        tuliscookie("shift","Day");
    }
    function Night(){
        alert("Shift Operator NIGHT");
        tuliscookie("shift","Night");
    }

    function ambilCookie(nama_variabel)
    {
    if (document.cookie.length > 0)
    { if (document.cookie.indexOf(nama_variabel)!=-
    1) //ngecek apakah cookies dgn variabel "namavariabel ada"
    {
    awal=awal + nama_variabel.length+1;
    //memposisikan awal setelah =

```

```

akhir=document.cookie.length;
//memosisikan akhir di akhir string
return unescape(document.cookie.substring(awal,akhir));
//mengembalikan nilai cookies yang merupakan antara awal dan
akhir
}
}
}

//buat tulis cookies
function tuliscookie(nama_variabel,nilai)
{
document.cookie=nama_variabel+ "=" +escape(nilai);
}
</script>
</head>
<body>
<script>

//buat tulis cookies
/*
function tuliscookie(nama_variabel,nilai)
{
document.cookie=nama_variabel+"="+escape(nilai);
}
*/

function Day(){
alert("Shift Operator DAY");
document.cookie=shift="DAY";
}
function Night(){
alert("Shift Operator NIGHT");
//tuliscookie("shift","Night");
document.cookie=shift="NIGHT";
}

</script>

<button onclick="Day()">Day</button>
<button onclick="Night()">Night</button>

```

```
<?php
    echo $_COOKIE['shift'];
?>

</body>
</html>
```