

**RANCANG BANGUN SISTEM INFORMASI
LAPORAN HASIL PRODUKSI PADA SEKSI *ASSEMBLING SEAT*
DI PT SUZUKI INDOMOBIL MOTOR CIKARANG PLANT**

TUGAS AKHIR

Untuk Memenuhi Syarat Program Sarjana Terapan

Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta

**OLEH
SITI MASHLUHAH
1314005**



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2018**

LEMBAR PERSETUJUAN DOSEN PEMBIMBING
POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN R.I

TANDA PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : **RANCANG BANGUN SISTEM INFORMASI
LAPORAN HASIL PRODUKSI PADA SEKSI
ASSEMBLING SEAT DI PT SUZUKI INDOMOBIL
CIKARANG PLANT**

Disusun Oleh

Nama : Siti Mashluhah
Nim : 1314005
Program Studi : Sistem Informasi Industri Otomotif

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

Jakarta, 08 September 2018

Dosen Pembimbing

Noveriza Yuliasari, S.Si, M.T
NIP: 197811212009012003

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Siti Mashlulah

Nim : 1314005

Program Studi : Sistem Informasi

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

**“RANCANG BANGUN SISTEM INFORMASI LAPORAN HASIL PRODUKSI PADA
SEKSI ASSEMBLING SEAT DI PT SUZUKI INDOMOBIL CIKARANG PLANT ”**

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

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

Jakarta, Agustus 2018
Yang Membuat Pernyataan,

Siti Mashlulah

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Siti Mashluhah
NIM : 1314005
Judul TA : Rancang Bangun Sistem Informasi Laporan Hasil Produksi Pada Seksi Assembling Seat Di PT Suzuki Indomobil Motor Cikarang Plant
Pembimbing : Noveriza Yuliasari, S.Si, M.T

No	Tanggal	Keterangan	Paraf
1.	17 April 2018	Proposal	
2.	26 April 2018	Bab I	
3.	02 Mei 2018	Revisi Bab I	
4.	16 Mei 2018	Bab II	
5.	18 Mei 2018	Revisi Bab II dan Bab III	
6.	22 Mei 2018	Revisi Bab III	
7.	30 Mei 2018	Bab IV	
8.	10 Juli 2018	Bab V	
9.	20 Juli 2018	Revisi Bab V (<i>Flowmap</i> dan <i>Use case</i>)	
10.	24 Juli 2018	Revisi Bab V (<i>Activity</i> dan <i>Sequence</i>)	
11.	27 Juli 2018	Revisi Bab V (<i>Class Diagram</i> dan <i>Deployment Diagram</i>)	
12.	01 Agustus 2018	Bab V dan Bab VI	
13.	03 Agustus 2018	Revisi Bab V ERD dan Bab VI	
14.	06 Agustus 2018	Revisi ERD CDM dan Kamus Data	
15.	08 Agustus 2018	Cek Kelengkapan Dokumen	

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif

Dosen Pembimbing

Drs. Jacob Saragih, M.M.

NIP : 195404281986031002

Noveriza Yuliasari, S.Si, M.T

NIP : 197811212009012003



ABSTRAK

PT Suzuki Indomobil Motor Cikarang Plant merupakan perusahaan yang bergerak di bidang industri otomotif manufaktur yang berfokus dalam produksi *seat* mobil suzuki ertiga. PT Suzuki Indomobil Motor Cikarang Plant membutuhkan pengembangan sebuah sistem informasi laporan hasil produksi khususnya pada seksi *assembling seat*. Laporan hasil produksi pada PT Suzuki Indomobil Motor Cikarang Plant masih kurang maksimal karena menggunakan *form* manual sehingga banyak sekali resiko kehilangan, rusak atau rangkapnya data serta lamanya proses pelaporan yang dibuat. Rancang bangun sistem informasi laporan hasil produksi sangat diperlukan dalam perusahaan untuk melakukan proses pengolahan data hasil produksi yang sesuai dengan perencanaan. Sistem informasi yang diusulkan akan mempermudah dalam melakukan pengolahan data laporan harian hasil produksi sehingga proses pengintegrasian data antara *Leader Assy* dan *Head Assy* akan lebih maksimal. Pengembangan sistem informasi laporan hasil produksi menggunakan metode *prototype evolutioner*. Analisis dan perancangan yang digunakan adalah analisis dan perancangan berorientasi objek dengan *tools* pemodelan sistem yaitu menggunakan *unified modeling language* (UML), *windows navigation diagram* dan *flowmap*. Perancangan dan pembuatan sistem informasi monitoring produksi ini menggunakan Java NetBeans 8.2 dan MySQL 5.6.12. Sistem informasi laporan hasil produksi ini dapat mempermudah dalam melakukan pengolahan data hasil produksi untuk membuat perencanaan *overtime* apabila data belum sesuai dengan perencanaan, serta pembuatan laporan hasil produksi. Sistem informasi ini juga memudahkan *user* dalam menyimpan data serta keamanan penyimpanan data yang terjamin karena telah menggunakan basis data. Untuk penerapan sistem yang baru, disarankan untuk melakukan sosialisasi kepada bagian terkait dan pemeliharaan aplikasi agar dapat berjalan dengan baik.

Kata Kunci : Sistem informasi, laporan produksi, perencanaan *overtime* pelaporan, *prototype evolutioner*.

KATA PENGANTAR

Puji syukur penulis panjatkan bagi Tuhan Yang Maha Esa, yang atas berkat dan kasih-Nya sehingga penulis mampu menyelesaikan Tugas Akhir dengan judul **“Rancang Bangun Sistem Informasi Laporan Hasil Produksi Pada Seksi *Assembling Seat* Di PT Suzuki Indomobil Motor Cikarang Plant”**.

Tugas akhir ini disusun guna memenuhi sebagian syarat yang harus dipenuhi dalam menempuh program sarjana terapan jurusan Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

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

1. ALLAH Subhanahu Wa Ta'ala Tuhan yang Maha Esa.
2. Bapak Drs. Mustofa, MT. selaku Direktur Politeknik STMI Jakarta.
3. Bapak Drs. Jacob Saragih, MM. selaku Ketua Prodi Sistem Informasi Industri Otomotif.
4. Ibu Noveriza Yuliasari, S.Si,M.T. selaku dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
5. Kedua orang tua saya. Yang selalu mendoakan, memberikan motivasi dan pengorbanannya baik segi moril, materi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
6. Diana Ratna Sari, Rully, Rizal Fahmi yang selalu mendukung, memberikan saran dan mau membagi pengalamannya dalam pemahaman pembuatan program untuk penulis.
7. Bapak Sriyana, Bapak Rohman, dan Bapak Agus Syah Ramdhani selaku pembimbing di PT Suzuki Indomobil Motor Cikarang Plant. Serta seluruh pekerja khususnya di Bagian Produksi yang telah membimbing dan

memberikan informasi yang berguna kepada penulis dalam penyusunan Tugas Akhir.

8. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
9. Teman-teman PKL di PT Suzuki Indomobil Motor Cikarang Plant. yang telah membantu dalam memberikan masukan dalam pengerjaan laporan kerja lapangan 1, 2 dan Tugas Akhir.
10. Seluruh teman-teman prodi Sistem Informasi Industri Otomotif angkatan 2014 yang telah memberikan semangat dan motivasi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
11. Serta semua pihak yang telah membantu penulis dan yang tidak dapat disebutkan satu per satu.

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

Jakarta, Agustus 2018

Penulis,

Siti Mashlulah

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN DENGAN DOSEN PEMBIMBING	
LEMBAR PERNYATAAN KEASLIAN	
ABSTRAK	v
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	viii
DAFTAR TABEL	ix
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Pokok Permasalahan	2
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah	4
1.5. Manfaat Tugas Akhir	4
1.6. Sistematika Penulisan	4
BAB II LANDASAN TEORI	7
2.1. Pengertian Rancang Bangun	7
2.2. Pengertian Sistem	7
2.2.1. Karakteristik Sistem	9
2.2.2. Jenis-Jenis Sistem	11
2.2.3. Tujuan Sistem.....	13
2.3. Pengertian Dan Konsep Dasar Informasi	13
2.4. Pengertian Sistem Informasi	14
2.4.1 Fungsi Sistem Informasi	15
2.4.2 Komponen Sistem Informasi.....	15
2.4.3 Klasifikasi dan tujuan informasi	17

2.5.	Konsep Dasar Laporan	17
2.5.1	Jenis-Jenis Laporan	18
2.6.	Perencanaan.....	18
2.6.1	Pengertian Produksi	19
2.6.2	Pengertian <i>Overtime</i>	20
2.6.3	Perencanaan Produksi	21
2.7.	Pengertian <i>Seat</i>	21
2.8.	<i>System Development Life Cycle</i> (SDLC)	23
2.9.	Model Prototype.....	28
2.9.1	<i>Evolutioner Prototype</i>	28
2.9.2	<i>Prototype Requirement</i>	29
2.10.	<i>Flowchart</i>	30
2.10.1	Simbol-simbol Flowchart.....	31
2.10.2	Jenis-jenis Flowchart.....	33
2.11.	<i>Unified Modeling language</i> (UML).....	34
2.11.1	<i>Use Case Diagram</i>	35
2.11.2	<i>Activity Diagram</i>	37
2.11.3	<i>Sequence Diagram</i>	38
2.11.4	<i>Class Diagram</i>	40
2.12.	<i>Entity Relationship Diagram</i> (ERD)	41
2.12.1	Kosenp ERD	42
2.12.2	Metodologi membangun ERD	43
2.13.	<i>Conceptual Data Model</i>	44
2.14.	Kamus Data	47
2.15.	<i>Windows Navigation Diagram</i>	48
2.16.	Definisi Java.....	49
2.16.1	NetBeans	49
2.16.2	Fitur-Fitur NetBeans	50
2.17.	MySQL.....	50
2.17.1	Keunggulan MySQL	51
2.17.2	Tipe Data MySQL	52

2.18. <i>Functional Testing</i>	53
BAB III METODOLOGI PENELITIAN	54
3.1. Jenis Dan Sumber Data	54
3.2. Metode Pengumpulan Data	55
3.3. Metode Pengembangan Sistem	56
3.4. Kerangka Penelitian	57
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA	60
4.1. Sejarah Perusahaan.....	60
4.2. Profil Perusahaan	61
4.3. Logo Perusahaan	62
4.4. Visi dan Misi Perusahaan.....	62
4.5. Struktur Organisasi Perusahaan	64
4.6. Struktur Organisasi pada seksi <i>Assembling seat</i>	65
4.7. Produk	66
4.8. Sistem Laporan Harian Produksi	68
4.9. Dokumen pada sistem laporan hasil produksi.....	68
4.10. Prosedur pencatatan laporan harian hasil produksi	72
4.11. Penambahan waktu kerja/ <i>overtime</i>	75
4.12. Pemodelan Sistem Berjalan Dengan UML	75
BAB V ANALISIS DAN PEMBAHASAN	79
5.1. Analisis Kebutuhan <i>Functional</i> Sistem	79
5.2. Alur Sistem Informasi Produksi Usulan	81
5.3. Analisis Sistem Usulan	83
5.3.1. <i>Use Case Diagram</i>	83
5.3.2. <i>Activity Diagram</i>	91
5.3.3. <i>Sequence Diagram</i>	101
5.3.4. <i>Class Diagram</i>	110
5.3.5. <i>Deployment diagram</i>	111
5.4. Permodelan Data Sistem Usulan.....	112
5.4.1 <i>Entity Relationship Diagram (ERD)</i>	112
5.4.2 <i>Conceptual Data Model (CDM)</i>	113

5.4.3	Kamus Data	114
5.5	Perancangan Sistem	118
5.5.1	<i>Windows Navigation Diagram (WND)</i>	118
5.5.2	Perancangan <i>Interface</i> Sistem	120
5.6	Implementasi Sistem	127
5.7	<i>Functional Testing</i>	128
BAB VI	KESIMPULAN DAN SARAN	134
6.1.	Kesimpulan	134
6.2.	Saran	134
DAFTAR PUSTAKA		135
LAMPIRAN		

DAFTAR GAMBAR

	Halaman
Gambar II.1 Karakteristik Dari Suatu Sistem	11
Gambar II.2 Pengembangan <i>Prototype Evolusioner</i>	29
Gambar II.3 Pengembangan <i>Prototype Requirement</i>	30
Gambar II.4 UML <i>Diagram</i>	35
Gambar II.5 Contoh ERD	42
Gambar II.6 Contoh <i>Windows Navigation Diagram</i>	49
Gambar III.1 Kerangka Penelitian	59
Gambar IV.1 Logo Perusahaan PT Suzuki Indomobil Motor	62
Gambar IV.2 Struktur Organisasi PT Suzuki Indomobil Motor	64
Gambar IV.3 Struktur Organisasi <i>section Assembling seat</i>	65
Gambar IV.4 Dokumen Rencana Produksi	69
Gambar IV.5 Data Harian Hasil Produksi	70
Gambar IV.6 Laporan Hasil Produksi	71
Gambar IV.7 Aliran Data Sistem Informasi Produksi	73
Gambar IV.8 <i>Use Case Diagram</i> Sistem Informasi Produksi	78
Gambar V.1 <i>Flowmap</i> Sistem Informasi Laporan Hasil Produksi Usulan ..	82
Gambar V.2 <i>Use Case Diagram</i> Sistem Informasi Laporan Produksi	84
Gambar V.3 <i>Activity Diagram Login</i>	93
Gambar V.4 <i>Activity Diagram</i> Proses Mengelola Data Master	94
Gambar V.5 <i>Activity Diagram Input</i> Rencana Produksi Harian	95
Gambar V.6 <i>Activity Diagram</i> Mencetak Rencana Produksi Harian	96
Gambar V.7 <i>Activity Diagram Input</i> Hasil Produksi	97
Gambar V.8 <i>Activity Diagram</i> Validasi Laporan Hasil Produksi	98
Gambar V.9 <i>Activity Diagram</i> Membuat Rencana <i>Overtime</i>	99
Gambar V.10 <i>Activity Diagram</i> mencetak <i>overtime</i>	100
Gambar V.11 <i>Activity Diagram</i> mencetak laporan hasil produksi	101
Gambar V.12 <i>Sequence Diagram Login</i>	102

Gambar V.13	<i>Sequence Diagram</i> Mengelola Data Master.....	103
Gambar V.14	<i>Sequence Diagram</i> Menginput Rencana Produksi.....	104
Gambar V.15	<i>Sequence Diagram</i> Mencetak Rencana Produksi	105
Gambar V.16	<i>Sequence Diagram</i> Data Menginput Hasil Produksi	106
Gambar V.17	<i>Sequence Diagram</i> Data Validasi Hasil Produksi	107
Gambar V.18	<i>Sequence Diagram</i> Data Membuat Rencana Overtime.....	108
Gambar V.19	<i>Sequence Diagram</i> Mencetak Overtime.....	109
Gambar V.20	<i>Sequence Diagram</i> mencetak Laporan Hasil Produksi	110
Gambar V.21	<i>Class Diagram</i> Laporan Hasil Produksi.....	111
Gambar V.22	<i>Deployment Diagram</i>	112
Gambar V.23	<i>Entity Relationship Diagram</i> usulan	113
Gambar V.24	<i>Conceptual Data Model</i>	114
Gambar V.25	<i>Windows Navigation Diagram</i>	119
Gambar V.26	<i>Interface Form</i> Login	120
Gambar V.27	<i>Interface</i> Halaman Utama	121
Gambar V.28	<i>Interface Form</i> Data user	121
Gambar V.29	<i>Interface Form</i> Data Produk.....	122
Gambar V.30	<i>Interface Form</i> Input Rencana Produksi	123
Gambar V.31	<i>Interface Form</i> Input Detail Produksi	124
Gambar V.32	<i>Interface Form</i> Input Hasil Produksi.....	124
Gambar V.33	<i>Interface Form</i> Membuat Rencana overtime	125
Gambar V.34	<i>Interface Form</i> Validasi	126
Gambar V.35	<i>Interface Form</i> Laporan Hasil Produksi.....	126

DAFTAR TABEL

	Halaman
Tabel II.1 Simbol <i>Flowchart</i>	31
Tabel II.2 Simbol-Simbol <i>Use case Diagram</i>	36
Tabel II.3 Simbol-Simbol <i>Activity Diagram</i>	37
Tabel II.4 Simbol-Simbol <i>Sequence Diagram</i>	38
Tabel II.5 Simbol <i>Sequence Diagram</i> pada jenis objek.....	40
Tabel II.6 Simbol-Simbol <i>Class Diagram</i>	40
Tabel II.7 Notasi <i>Entity Relationship Diagram</i> (ERD).....	42
Tabel II.8 Simbol-simbol ERD	43
Tabel II.9 Simbol-Simbol <i>Conceptual Data Model</i> (CDM)	44
Tabel II.10 Aturan <i>Conceptual Data Model</i> (CDM).....	45
Tabel II.11 Contoh Kamus Data	48
Tabel II.12 Jenis Tipe Data MySQL	53
Tabel IV.1 Produk <i>Seat</i>	66
Tabel IV.2 Tabel Penentuan <i>Overtime</i>	75
Tabel IV.3 Tabel perhitungan <i>tactime</i>	75
Tabel IV.4 Definisi Aktor Sistem Informasi Produksi.....	76
Tabel IV.5 Definisi <i>Use Case</i> Sistem Informasi Laporan Hasil Produksi.....	77
Tabel V.1 Kebutuhan <i>Functional</i> Sistem.....	79
Tabel V.2 Definisi Aktor <i>Use Case Diagram</i> Sistem Usulan.....	85
Tabel V.3 Definisi <i>Use Case Diagram Description</i>	85
Tabel V.4 Deskripsi <i>Use Case Diagram Login</i>	86
Tabel V.5 Deskripsi <i>Use Case</i> Mengelola Data Master	87
Tabel V.6 Definisi <i>Use Case</i> Menginput Data Rencana Produksi.....	87
Tabel V.7 Definisi <i>Use Case</i> Mencetak Rencana Produksi.....	88
Tabel V.8 Definisi <i>Use Case</i> Menginput Data Aktual Hasil Produksi.....	89
Tabel V.9 Definisi <i>Use Case</i> membuat rencana <i>overtime</i>	90
Tabel V.10 Definisi <i>Use Case</i> mencetak rencana <i>overtime</i>	91
Tabel V.11 Definisi <i>Use Case</i> memvalidasi	91
Tabel V.12 Definisi <i>Use Case</i> Mencetak Laporan Harian Hasil Produksi.....	92
Tabel V.13 Tabel <i>User</i>	115

Tabel V.14	Tabel Produk	116
Tabel V.15	Tabel Rencana Produksi.....	116
Tabel V.16	Tabel Detail Rencana Produksi	117
Tabel V.17	Tabel Hasil Produksi	117
Tabel V.18	Tabel <i>Overtime</i>	118
Tabel V.19	<i>Functional Test Use Case Login</i>	127
Tabel V.20	<i>Functional Test Use Case</i> mengelola data master	128
Tabel V.21	<i>Functional Test Use Case</i> membuat rencana produksi	129
Tabel V.22	<i>Functional Test Use Case</i> mencetak rencana produksi.....	130
Tabel V.23	<i>Functional Test Use Case</i> menginput hasil produksi.....	131
Tabel V.24	<i>Functional Test Use Case</i> membuat overtime	132
Tabel V.25	<i>Functional Test Use Case</i> mencetak overtime	132
Tabel V.26	<i>Functional Test Use Case</i> memvalidasi hasil produksi	133
Tabel V.27	<i>Functional Test Use Case</i> mencetak laporan hasil produksi	133

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan sistem informasi sekarang sangat cepat dan pesat, tidak sedikit yang menggunakan sistem informasi untuk membantu kemudahan dalam bekerja. Sistem informasi merupakan suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan laporan-laporan yang diperlukan. Sistem informasi juga digunakan untuk mengambil keputusan atau mengendalikan informasi.

Sistem informasi sangat penting bagi eksistensi perusahaan. Terutama dalam perkembangan usahanya yang membutuhkan banyak waktu dan tenaga. Apalagi nilai dari sebuah informasi menjadi hal yang penting dalam menjalankan operasional perusahaan. Jika kurang mendapatkan informasi dalam waktu tertentu maka perusahaan akan mengalami ketidakmampuan mengontrol sumber daya sehingga ketika mengambil keputusan yang strategis akan terganggu dan akhirnya dapat mengalami kekalahan dalam bersaing. Keuntungan perusahaan jika menerapkan sistem informasi adalah memberikan informasi yang benar kepada bagian tertentu secara cepat pada waktu yang tepat.

PT Suzuki Indomobil Motor Cikarang Plant merupakan salah satu perusahaan yang bergerak dibidang manufaktur otomotif perakitan mobil ertiga. Dalam mendukung kegiatan produksi nya PT Suzuki indomobil Motor Cikarang Plant telah menerapkan komitmen untuk menghasilkan produk yang sesuai dengan persyaratan dan harapan pelanggan, dengan cara memproduksi barang dengan cepat, tepat yang sesuai dengan permintaan pelanggan. Hal ini tentunya berkaitan dengan bagian produksi. Pada PT Suzuki Indomobil Motor Cikarang Plant bagian produksinya adalah khusus untuk memproduksi *seat* mobil suzuki ertiga dan perakitan mobil suzuki ertiga. PT Suzuki Indomobil Motor Cikarang Plant memiliki

divisi *assembling seat*. Divisi *assembling seat* merupakan divisi yang mengatur, memonitoring proses produksi *seat*.

Divisi *assembling seat* memiliki sistem pengolahan data harian hasil produksi yang terdiri dari *daily report*, dan rencana produksi harian. Laporan harian hasil produksi pada PT Suzuki Indomobil Motor Cikarang Plant ini masih dicatat secara manual dan dibantu dengan *microsoft excel* untuk menyimpan data nya secara komputerisasi. Penulisan tangan yang dilakukan oleh operator adalah laporan harian hasil produksi yang dicatat pada *daily report*. Kemudian *daily report* tersebut diberikan kepada *head assembling seat* untuk diinput ke *microsoft excel*, untuk dijadikan arsip atau tempat penyimpanan data laporan harian hasil produksi. Namun *head assembling seat* ini sering tidak langsung menginput ke komputer pada hari tersebut, tetapi dilakukan keesokan harinya. Hal ini sering menimbulkan terjadinya kehilangan dan kerusakan data. Sehingga *section head assembling seat* tidak bisa melihat jumlah hasil produksi *seat* sesuai dengan perencanaan atau tidak, dan untuk mengontrol jumlah produksinya *head assembling seat* harus mengontrol via telepon atau turun langsung ke *genba*. Hal ini juga berdampak pada kecepatan proses penginputan laporan harian hasil produksi pada suatu periode tertentu yang membutuhkan waktu lama, karena harus mencari satu per satu.

Untuk mengatasi masalah tersebut, diperlukan aplikasi untuk membantu perusahaan dalam mengatasi masalah tersebut. Adapun judul Tugas Akhir ini adalah “RANCANG BANGUN SISTEM INFORMASI LAPORAN HASIL PRODUKSI PADA SEKSI *ASSEMBLING SEAT* DI PT SUZUKI INDOMOBIL CIKARANG PLANT”.

1.2 Pokok Permasalahan

Permasalahan yang terjadi pada seksi *assembling seat* pada PT Suzuki Indomobil Motor Cikarang Plant adalah sebagai berikut:

1. Pencatatan data laporan harian hasil produksi sudah terkomputerisasi dengan *microsoft excel*, namun untuk *daily report* masih di catat manual oleh operator. data harian hasil produksi pada *daily report* tidak langsung diinput ke *microsoft*

excel langsung pada hari tersebut, tetapi *diinput* keesokan harinya. Sehingga sering menimbulkan terjadinya kehilangan data dan kesalahan dalam penulisan jumlah hasil produksi.

2. *Head Assy* tidak dapat mengakses berapa jumlah *seat* yang sudah diproduksi karena belum adanya sistem atau aplikasi yang memiliki *database* yang bisa diakses langsung dengan bagian produksi sehingga untuk membuat perencanaan *overtime head assy* harus mengecek ke bagian produksi apakah jumlah hasil sesuai dengan rencana.
3. Pembuatan laporan hasil produksi *seat* belum memiliki pelaporan yang memudahkan *leader* dan *Head Assy* dalam membaca data hasil produksi antara rencana dengan aktual hasil produksi.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah menganalisis, merancang dan membangun suatu sistem informasi laporan hasil produksi *seat* pada PT Suzuki Indomobil Motor Cikarang Plant yang mampu:

1. Mencatat data hasil produksi secara komputerisasi dengan aplikasi laporan hasil produksi, agar mengurangi kesalahan dalam mencatat jumlah hasil produksi harian dan kehilangan data pada waktu tertentu.
2. Memudahkan *leader* dalam menyimpan data harian hasil produksi dengan adanya sistem basis data, dan memudahkan *leader* dalam pembuatan perencanaan *overtime* apabila hasil tidak sesuai dengan perencanaan.
3. Mengolah data hasil produksi untuk dijadikan laporan hasil berdasarkan periode dalam waktu 1 bulan untuk memudahkan *leader* dan *head assy* untuk melihat laporan hasil produksi antara rencana produksi dengan hasil aktual produksi.

1.4 Batasan Masalah

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

1. Tempat penelitian adalah PT Suzuki Indomobil Motor Cikarang Plant.
2. Penelitian dilakukan pada seksi *assembling seat* selama satu bulan, dari bulan Agustus s.d September 2017.
3. Analisis dan penelitian yang dilakukan hanya sebatas menangani masalah pengelolaan data laporan harian hasil produksi pada seksi *assembling seat* yang menangani laporan hasil produksi.
4. *Tools* yang digunakan untuk pemrograman ini adalah menggunakan Java *NetBeans* 8.2 dan *database* MySQL 5.6.12

1.5 Manfaat Tugas Akhir

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

1. Hasil penelitian ini agar dapat diimplementasikan di perusahaan untuk membantu kinerja perusahaan dalam melakukan pengolahan data laporan hasil produksi yang dilakukan.
2. Mendapatkan informasi yang lengkap terhadap setiap laporan proses produksi yang dilakukan setiap harinya.

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada di dalamnya. Agar lebih mempermudah perumusan yang akan dibahas pada Tugas Akhir ini, maka diuraikan tahapan-tahapan dalam penyusunan Tugas Akhir ini sebagai berikut:

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Bab ini menjelaskan tentang berbagai teori yang diperoleh dari buku-buku literatur ataupun berbagai macam referensi. Teori-teori yang dipaparkan pada laporan ini adalah seputar pengertian sistem, jenis sistem, pengertian informasi, jenis-jenis informasi, konsep dasar laporan, jenis laporan, pengertian produksi, pengertian *overtime*, pengertian *seat*, metodologi pengembangan sistem, *windows navigation* diagram, *flowchart*, ERD, *Unified Modelling Language* (UML), Java, *Netbeans*, dan MySQL.

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, pengolahan data diagram alir, dan *use case* sistem informasi laporan harian hasil produksi yang sedang berjalan berdasarkan pengamatan selama melaksanakan penelitian di PT Suzuki Indomobil Motor Cikarang Plant.

BAB V ANALISIS DAN PEMBAHASAN

Pada bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis kebutuhan sistem, memodelkan sistem dengan *Unified Modelling Language* (UML), kamus data, perancangan hierarki menu, perancangan tampilan layar, sampai kebutuhan *software* dan *hardware* yang diperlukan.

BAB VI KESIMPULAN DAN SARAN

Pada bab penutup ini dikemukakan kesimpulan-kesimpulan dari hasil penelitian dan saran-saran untuk pihak perusahaan dalam berbagai hal yang berhubungan dengan sistem informasi laporan hasil produksi pada seksi *assembling seat*.

BAB II

LANDASAN TEORI

2.1 Pengertian Rancang Bangun

Menurut McLeod (2011) rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisis dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan. Rancangan sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru.

Perancangan adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik. Sedangkan pengertian bangun atau pembangunan sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian. Bangun sistem adalah membangun sistem informasi dan komponen yang didasarkan pada spesifikasi desain. Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil analisis ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut ataupun memperbaiki sistem yang sudah ada.

2.2 Pengertian Sistem

Menurut Romney dan Steinbart yang terdapat dalam buku Mulyani (2016) mengatakan sistem adalah kumpulan dari dua atau lebih komponen yang saling bekerja dan berhubungan untuk mencapai tujuan tertentu. Dia juga berpendapat bahwa perusahaan adalah suatu sistem yang terdiri dari beberapa departemen yang bertindak sebagai subsistem yang membentuk sistem perusahaan tersebut. Sedangkan pendapat lain mengatakan sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk

melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu (Kamus Teknologi dan Informasi, 2009).

Definisi secara umum kata sistem berasal dari bahasa latin (*systēma*) dan bahasa Yunani (*sustēma*) yaitu suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Pada dasarnya manusia hidup di dunia penuh dengan sistem. Apa yang terlihat di sekeliling kita dalam kehidupan sehari-hari sebenarnya merupakan kumpulan dari sebuah sistem, dan sistem tersebut tersusun atau terorganisir dengan kompleks. Dikatakan kompleks karena dunia ini memang tersusun dari beberapa subsistem yang berbeda satu sama lain dan mereka berinteraksi pada tingkat tertentu. Sedangkan sistem dalam sudut pandang perusahaan atau organisasi adalah tempat sumber daya manusia yang berinteraksi untuk merealisasikan formulasi tujuan yang telah ditetapkan. Oleh karena itu dengan sendirinya, sebuah sistem dapat dikatakan sangat rumit.

Berikut adalah beberapa definisi sistem menurut beberapa ahli yang dikutip oleh Hutahean (2014), diantaranya adalah:

1. Fat

Sistem adalah suatu himpunan “benda” nyata atau abstrak (*a set of thing*) yang terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan (*unity*) untuk mencapai tujuan tertentu secara efisien dan efektif.

2. Indrajit

Sistem mengandung arti kumpulan-kumpulan dari komponen-komponen yang dimiliki unsur keterkaitan antara satu dengan lainnya.

3. Jogiyanto

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

4. Murdick, R.G

Suatu sistem adalah seperangkat elemen yang membentuk kumpulan atau procedure-procedure/bagan-bagan pengolahan yang mencari suatu tujuan tertentu.

5. Jerry Futz Gerald

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

6. Davis, G.B

Sistem secara fisik adalah kumpulan dari elemen-elemen yang beroperasi bersama-sama untuk menyelesaikan suatu sasaran.

7. Dr. Ir. Harijono Djojodiharjo

Suatu sistem adalah sekumpulan objek yang mencakup hubungan fungsional antara ciri tiap objek, dan yang secara keseluruhan merupakan suatu kesatuan secara fungsional.

8. Lani Sidharta

Sistem adalah himpunan dari bagian-bagian yang saling berhubungan yang secara bersama mencapai tujuan-tujuan yang sama.

2.2.1 Karakteristik Sistem

Menurut Hutahean (2014) supaya sistem itu dikatakan baik, maka sistem yang baik memiliki karakteristik yaitu:

1. Komponen

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen sistem terdiri dari komponen yang berupa subsistem atau bagianbagian dari sistem.

2. Batasan Sistem (*boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini

memungkinkan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan luar sistem (*environment*)

Lingkungan luar sistem (*environment*) adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tetap dijaga dan yang merugikan yang harus dijaga dan dikendalikan, kalau tidak akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung sistem (*interface*)

Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumberdaya mengalir dari subsistem ke subsistem lain. Keluaran (*output*) dari subsistem akan menjadi masukan (*input*) untuk subsistem lain melalui penghubung.

5. Masukan Sistem (*input*)

Masukan adalah energi yang dimasukkan kedalam sistem, yang dapat berupa perawatan (*maintenance input*), dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah energi yang di proses untuk didapatkan keluaran. Contoh dalam sistem komputer program adalah *Maintenance input* sedangkan data adalah signal input untuk diolah menjadi informasi.

6. Keluaran Sistem (*output*)

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh komputer menghasilkan panas yang merupakan sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

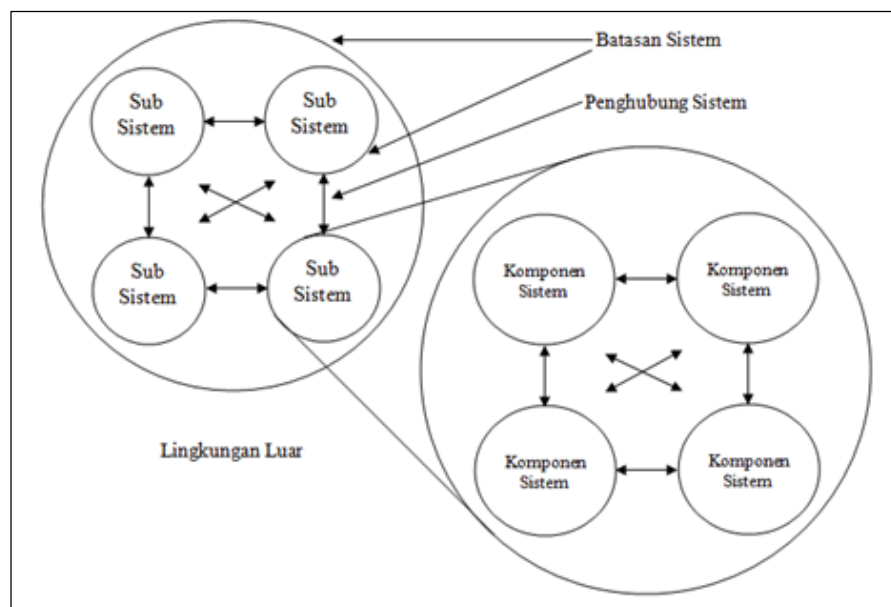
7. Pengolah Sistem

Pengolah sistem menjadi bagian pengolah yang akan merubah masukan menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi, sistem akuntansi akan mengolah data menjadi laporan-laporan keuangan.

8. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan input yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil apabila mengenai sasaran atau tujuannya. Jika suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.

Untuk lebih jelasnya, karakteristik sistem dapat dilihat pada gambar dibawah ini:



Gambar II.1. Karakteristik Dari Suatu Sistem
(Sumber: Mustakini, 2011)

2.2.2 Jenis-Jenis Sistem

Menurut McLeod dan Schell dalam buku Sri Mulyani (2016) berpendapat ada 5 jenis sistem yaitu:

a. *Transaction Processing System* (TPS)

Sebuah perusahaan akan memiliki banyak transaksi dalam kegiatan operasionalnya. Transaksi-transaksi tersebut awalnya di proses secara manual atau yang sedikit lebih canggih biasa kita kenal dengan istilah *punch-card system*. Namun seiring berjalannya waktu, para ahli teknologi terus mengembangkan sebuah sistem yang bisa mengolah semua transaksi-transaksi perusahaan.

sistem komputer yang pertama kali dikembangkan adalah *Elektronik Data Processing* (EDP) kemudian muncullah *Accounting Information System* (AIS) dan sekarang disebut dengan istilah *Transaction Processing System* (TPS).

Transaction Processing System adalah sebuah sistem komputer yang didesain untuk mengolah transaksi yang tidak hanya terbatas pada *database* atau *file system* namun juga melakukan pengolahan beberapa operasi transaksi dimana semua transaksi harus berhasil atau semua transaksi harus dibatalkan. Sebagai diseluruh Indonesia.

b. *Management Information System* (MIS)

Management Information System adalah sebuah sistem yang sudah terkomputerisasi yang melakukan pengolahan data agar bisa digunakan oleh orang yang membutuhkannya.

c. *Virtual Office System*

Virtual Office System merupakan pengembangan dari *Office Automation System* yaitu mesin komputer (*hardware*) dan *software* yang digunakan untuk membuat, mengumpulkan, menyimpan, memanipulasi dan menyebarkan informasi untuk kebutuhan perkantoran/perusahaan secara *digital* untuk mengerjakan tugas-tugas perusahaan guna mencapai tujuan yang telah ditetapkan.

mereka berpendapat bahwa cikal bakal *office automation* yaitu berawal dari sebuah aplikasi yang disebut *word processing* yang menggunakan jaringan-jaringan komunikasi elektronik. Sebagai contoh aplikasi tersebut misalnya *e-mail*, *voice mail*, *electronic calendaring*, *fax (facsimile)*, namun sekarang *office automation* lebih mengarah kepada *personal productivity system*, dimana seseorang bisa menggunakan alat-alat elektronik untuk melakukan manajemen dan membantu mengerjakan tugas-tugas pribadi/bisnis mereka. Sebagai contoh *Personal Data Assistant* (PDA) adalah alat elektronik yang digunakan untuk melakukan manajemen *email*, *address book*, data-data bisnis lain dan sebagainya. Kemampuan *office automation* dalam melakukan segala sesuatu melahirkan sebuah konsep yang disebut *office automation system*.

d. *Decision Support System (DSS)*

Decision Support System merupakan sebuah sistem yang membantu seorang manajer atau sekelompok kecil manajer untuk memecahkan sebuah permasalahan.

e. *Enterprise Resource Planning (ERP)*

ERP system merupakan sistem yang terkomputerisasi yang melibatkan seluruh *resource* manajemen dalam sebuah perusahaan.

2.2.3 Tujuan Sistem

Tujuan sistem merupakan target atau sasaran akhir yang ingin dicapai oleh suatu sistem. Agar supaya target tersebut bisa tercapai, maka target atau sasaran tersebut harus diketahui terlebih dahulu ciri-ciri atau kriterianya. Upaya mencapai suatu sasaran tanpa mengetahui ciri-ciri atau kriteria dari sasaran tersebut kemungkinan besar sasaran tersebut tidak akan pernah tercapai. Ciri-ciri atau kriteria dapat juga digunakan sebagai tolak ukur dalam menilai suatu keberhasilan suatu sistem dan menjadi dasar dilakukannya suatu pengendalian (Susanto, 2013).

2.3 Pengertian Dan Konsep Dasar Informasi

Menurut Kusriani dan Andri (2007) Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi. Data belum memiliki nilai, sedangkan informasi sudah memiliki nilai. Informasi dikatakan bernilai bila manfaatnya lebih besar dibanding untuk mendapatkannya.

Sedangkan menurut Hutahean (2014) informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Dan kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.

Informasi yang baik memiliki beberapa kriteria, dan Menurut Kusriani dan Andri (2007) Informasi yang berkualitas adalah informasi yang memiliki tiga kriteria yaitu:

1. Akurat (*Accurate*)

Informasi yang harus bebas dari kesalahan, tidak bisa ataupun menyesatkan. Akurat juga berarti bahwa informasi tersebut harus dapat dengan jelas mencerminkan maksudnya.

2. Tepat pada waktunya (*timeliness*)

Informasi yang datang pada penerima tidak boleh terlambat. Di dalam pengambilan keputusan, informasi yang sudah usang tidak lagi bernilai. Bila informasi datang terlambat sehingga pengambilan keputusan terlambat dilakukan. Hal ini dapat bernilai fatal bagi perusahaan.

3. Relevan (*relevance*)

Informasi yang disampaikan harus mempunyai keterkaitan dengan masalah yang akan dibahas dengan informasi tersebut. Informasi harus bermanfaat bagi pemakainya. Disamping karakteristik, nilai informasi (*value of information*) ditentukan oleh dua hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih besar dibanding biaya untuk mendapatkannya.

2.4 Pengertian Sistem Informasi

Menurut Anggraeni dan Rita (2017) sistem adalah kumpulan orang yang saling bekerja sama dengan ketentuan-ketentuan aturan yang sistematis dan terstruktur untuk membentuk suatu kesatuan yang melaksanakan suatu fungsi untuk mencapai suatu tujuan. Sistem memiliki beberapa karakteristik atau sifat yang tersiri dari komponen sistem, masukan sistem, lingkungan luar sistem, penghubung sistem, masukan sistem, keluaran sistem, pengolahan sistem dan sasaran sistem. Sedangkan informasi adalah data yang diolah menjadi lebih berguna dan berarti bagi penerimanya, serta untuk mengurangi ketidakpastian dalam proses pengambilan keputusan mengenai suatu keadaan. Sistem informasi merupakan suatu kombinasi teratur dari orang-orang, *hardware*, *software*, jaringan komunikasi dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi.

2.4.1 Fungsi Sistem Informasi

Berikut merupakan fungsi dari sistem informasi menurut Anggraeni dan Rita (2017), yaitu:

1. Untuk meningkatkan aksesibilitas data yang ada secara efektif dan efisien kepada pengguna, tanpa dengan perantara sistem informasi.
2. Memperbaiki produktivitas aplikasi pengembangan dan pemeliharaan sistem.
3. Menjamin tersedianya kualitas dan keterampilan dalam memanfaatkan sistem informasi secara kritis.
4. Mengidentifikasi kebutuhan mengenai keterampilan pendukung sistem informasi.
5. Mengantisipasi dan memahami akan konsekuensi ekonomi.
6. Menetapkan investasi yang akan diarahkan pada sistem informasi.
7. Mengembangkan proses perencanaan yang efektif.

2.4.2 Komponen Sistem Informasi

Menurut Anggraeni dan Rita (2017) Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari komponen *input*, komponen model, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran. Berikut ini adalah penjelasan dari masing-masing komponen yang terlibat di dalamnya, yaitu:

1. Komponen *input*

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

2. Komponen model

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

3. Komponen *output*

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

4. Komponen teknologi

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

5. Komponen *hardware*

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

6. Komponen *software*

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

7. Komponen basis data

Merupakan blok yang berisi definisi basis data yang disediakan untuk menyimpan data-data yang akan disimpan dalam media penyimpan. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

8. Komponen kontrol

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

2.4.3 Klasifikasi Dan Tujuan Sistem Informasi

Menurut Anggraeni dan Rita (2017) Ada beberapa cara untuk mengelompokkan sistem informasi. Klasifikasi yang umum dipakai antara lain didasarkan pada:

1. Level Organisasi, berdasarkan level organisasi sistem informasi dikelompokkan menjadi sistem informasi departemen, sistem informasi perusahaan, dan sistem informasi antar organisasi.
2. Area Fungsional, sebagaimana diketahui bahwa dalam sebuah organisasi memiliki sejumlah bidang fungsional seperti akuntansi, pemasaran, produksi, dan sebagainya.

Tujuan dari sistem informasi adalah:

1. Menyediakan informasi untuk membantu pengambilan keputusan manajemen.
2. Membantu petugas di dalam melaksanakan operasi perusahaan dari hari ke hari.
3. Menyediakan informasi yang layak untuk pemakai pihak luar perusahaan.

2.5 Konsep Dasar Laporan

Menurut Kusmayadi (2008) Laporan adalah segala sesuatu hal yang dilaporkan. Dalam laporan, penulis laporan hanya menyampaikan hal-hal yang esensial (intisari), hal-hal yang pokok yang bertalian dengan tugasnya. Dengan demikian, orang yang menerima laporan itu segera mengetahui masalahnya atau hasil.

Pada hakikatnya, penulisan laporan adalah penyampaian informasi yang bersifat fakta tentang sesuatu dari satu pihak kepada pihak lain. Dengan kata lain, penulisan laporan menyangkut tiga hal, yaitu:

1. Apa yang dilaporkan
2. Siapa yang melaporkan
3. Kepada siapa laporan itu disampaikan

Menurut pengertian umum laporan adalah suatu dokumen sebagai hasil serangkaian kegiatan mencari dan menyajikan informasi mengenai suatu hal tertentu. Dalam arti yang paling dasar, dokumen adalah sebuah makalah tertulis

yang memuat fakta-fakta sebagai kesaksian mengenai kenyataan suatu hal. dengan demikian, menurut intinya laporan merupakan keterangan atau informasi yang dikumpulkan, diolah, dan disajikan secara tertulis. Syarat laporan yang baik adalah sebagai berikut:

1. Ditulis dalam bahasa yang baik dan jelas agar memberikan pengertian yang tepat (tidak bisa/ambigu).
2. Isi laporan harus diurutkan secara logis dan sistematis.
3. Isi laporan yang disajikan harus lengkap.
4. Isi laporan yang disajikan harus dapat dipertanggungjawabkan kebenarannya.
5. Disajikan secara menarik, yaitu menarik perhatian pembaca baik dari segi kebutuhan pembaca maupun dari segi nilainya.

2.5.1 Jenis-Jenis Laporan

Menurut Kusmayadi (2008) laporan digolongkan dalam beberapa jenis, yaitu berdasarkan tujuan, waktu, dan gaya tulis, yang masing-masing mencakup dua kelompok sebagai berikut:

1. Berdasarkan Tujuan
 - a. Laporan Perencanaan
 - b. Laporan Pengontrolan
2. Berdasarkan waktu
 - a. Laporan berkala
 - b. Laporan khusus
3. Menurut gaya tulis
4. Laporan resmi (informatif atau persuasif)
5. Laporan tak resmi (informatif atau persuasif)

2.6 Perencanaan

Menurut Semara (2018) perencanaan adalah rangkaian kegiatan menetapkan hal-hal yang akan dikerjakan pada waktu yang akan datang berdasarkan fakta-fakta dan pemikiran yang matang dalam rangka pencapaian tujuan yang diinginkan. Perencanaan juga merupakan pedoman dan acuan bagi para

pelaksana kegiatan, agar kegiatan yang ada dapat berjalan sesuai dengan rencana dan tujuan yang telah ditetapkan bersama.

Pengertian perencanaan menurut para ahli diantaranya antara lain sebagai berikut (Semara, 2018):

1. Garth N. Jone

Perencanaan sebagai suatu proses pemilihan dan pengembangan dari tindakan yang paling baik atau menguntungkan untuk mencapai tujuan.

2. Mc. Farland

Perencanaan sebagai fungsi manajemen dimana pimpinan kemungkinan menggunakan pengaruh dari pada kewenangannya, yang dapat mengubah tujuan dan kegiatan di organisasi.

2.7 Pengertian Produksi

Menurut Fahmi (2014) produksi adalah sesuatu yang dihasilkan oleh suatu perusahaan baik berbentuk barang (*goods*) maupun jasa (*services*) dalam suatu periode waktu yang selanjutnya dihitung sebagai nilai tambah bagi perusahaan. Bentuk hasil produksi dengan kategori barang (*goods*) dan jasa (*services*) sangat tergantung pada kategori aktivitas bisnis yang dimiliki perusahaan yang bersangkutan. Jika perusahaan *manufacture* (pabrik) sudah jelas produksi yang dihasilkan dalam bentuk barang, sedangkan untuk bisnis perhotelan, travel, pendidikan adalah berbentuk jasa. Barang bersifat *tangible asset* dan jasa bersifat *intangible asset*.

Jika ditelaah lebih lanjut, pengertian produksi dapat ditinjau dari dua sudut, yaitu:

1. Pengertian produksi dalam arti sempit adalah mengubah bentuk barang menjadi barang baru, ini menimbulkan *form utility*.
2. Pengertian produksi dalam arti luas adalah usaha yang menimbulkan kegunaan karena *place*, *time* dan *possession*.

Pada saat suatu organisasi dituntut untuk memiliki produksi yang continue, maka artinya organisasi tersebut harus memiliki daya saing pasar. Jika tidak, maka organisasi tersebut tidak menempatkan konsep produksi secara sesungguhnya.

Karena organisasi produksi memiliki konsep yang berhubungan dengan pencarian bahan baku, pengolahan bahan baku dan akhirnya pada pencapaian nilai-nilai ekonomis yang dimaksud. Pada dasarnya produksi bisa di bagi menjadi dua bagian yaitu:

1. Produksi Langsung

Produksi Langsung adalah produksi yang menggunakan faktor-faktor produksi alam dan tenaga kerja. Produksi langsung terbagi menjadi dua bagian yaitu:

a. Produksi Primer

Suatu usaha aktivitas produksi yang bisa menghasilkan suatu produk dengan menggunakan bahan langsung dari alam. Misalnya: pertanian, perkebunan dan perikanan.

b. Produksi Sekunder

Usaha dengan menggunakan bahan yang sudah diolah untuk kembali diolah lagi menjadi barang yang lebih bermanfaat. Misalnya: pembuatan mobil, sepeda, baju dan sebagainya.

2. Produksi Tak Langsung

Produksi Tak Langsung adalah produksi yang tidak menaikkan nilai penggunaan dan bukan dari alam tetapi memberikan sumbangan jasa yang sangat bermanfaat bagi perusahaan. Misalnya: akuntan, ilmuwan, satpam, dan sebagainya.

2.8 Perencanaan Produksi

Menurut Gasperz (2005) Perencanaan produksi merupakan suatu proses penetapan tingkat output manufakturing secara keseluruhan guna memenuhi tingkat penjualan yang direncanakan dan inventori yang diinginkan. Rencana produksi mendefinisikan tingkat manufakturing, biasanya dinyatakan sebagai tingkat bulanan untuk periode satu tahun atau lebih, untuk setiap kelompok produk.. Perencanaan produksi merupakan tanggung jawab manajemen puncak (*top management*) yang membutuhkan konsensus dari semua departemen fungsional, terutama dari departemen pemasaran, keuangan, PPIC, dan Produksi.

2.9 **Pengertian *Overtime***

Overtime menurut Wikipedia (2010) adalah jumlah waktu seseorang bekerja di luar jam kerja normal. Kerja lembur atau *Overtime* adalah pekerjaan yang dilakukan oleh karyawan, atas dasar perintah atasan, yang melebihi jam kerja biasa pada hari-hari kerja, atau pekerjaan yang dilakukan pada hari istirahat mingguan karyawan atau hari libur resmi. Perusahaan menerapkan jam lembur dengan memperhatikan target produksi yang telah ditetapkan. Penerapan jam lembur untuk semua proses adalah sama meskipun kapasitas produksi masing-masing proses berbeda. Hal ini menyebabkan adanya penerapan lembur pada setiap proses, meskipun proses tertentu tidak membutuhkan lembur.

2.10 ***Assembling***

Menurut Yamit (2013) perakitan adalah gerakan yang menggabungkan satu objek dengan objek yang lain sehingga menjadi satu kesatuan. Perakitan juga diartikan sebagai suatu proses penyusunan dan penyatuan beberapa bagian komponen menjadi suatu alat atau mesin yang mempunyai fungsi tertentu. Pekerjaan perakitan dimulai bila objek sudah siap dipasang (biasanya setelah diarahkan) dan berakhir bila objek sudah tergabung secara sempurna. Pada prinsipnya perakitan dalam proses manufaktur terdiri dari pasangan semua bagian-bagian komponen menjadi suatu produk, proses pengencangan, proses inspeksi dan pengujian fungsional, pemberian nama atau label, pemisahan hasil perakitan yang baik dan hasil perakitan yang buruk, serta pengepakan dan penyiapan untuk pemakaian akhir. Perakitan merupakan proses khusus bila dibandingkan dengan proses manufaktur lainnya, misalnya proses permesinan (freis, bubut, bor, dan gerinda) dan pengelasan yang sebagian pelaksanaannya hanya meliputi satu proses saja. Sementara dalam perakitan bisa meliputi berbagai proses manufaktur.

2.11 **Pengertian *Seat***

Seat mempunyai peran yang sangat penting. Seat dibuat dari busa lembut yang dapat memberi kenyamanan untuk diduduki. Beragam bentuk dan modelnya tergantung dari tipe masing-masing kendaraan tetapi memiliki fungsi yang sama.

Fungsi seat adalah untuk tempat duduk pada mobil yang terdiri dari kursi depan, kursi kedua, dan kursi ketiga. Kursi jok harus mampu menopang berat penumpang. Disatu sisi, kursi jok harus mampu meredam setiap kejutan yang diakibatkan oleh permukaan jalan yang tidak rata. Dengan cara ini kenyamanan penumpang atau pengemudi dapat terjaga selama berkendara. Saat ini telah dikembangkan teknologi pada kursi jok untuk meningkatkan keamanan dan kenyamanan saat berkendara. Dalam www.otomotifinfo.com (2014) terdapat beberapa macam fitur pada kursi jok mobil antara lain:

1. *Aktif Headrest System*

Headrest dipasangkan sebagai sandaran kepala. Beberapa *headrest* dilengkapi dengan penyetel ketinggian untuk menyesuaikan dengan ketinggian pengemudi. Ada teknologi aktif *headrest system* untuk melindungi pengemudi dari cedera leher saat mobil ditabrak dari belakang. Sistem ini sedikit menaikkan *headrest* dan menahan kepala dari gaya ke belakang. Dengan cara ini cedera leher dan kepala pengemudi atau penumpang dapat diminimalisir.

2. *Auto Adjuster Seat*

Pada beberapa jenis mobil terdapat *power seat control*. Fitur ini mempermudah dalam mengatur posisi tempat duduk sesuai dengan postur duduk pengemudi/penumpang. Dengan pengaturan *auto adjuster seat*, pengemudi atau penumpang depan dapat duduk dengan nyaman saat berkendara. Cukup tekan tombol pada *power seat control*, maka motor *power seat* akan bekerja. Terdapat beberapa pilihan *auto adjuster seat* yaitu:

- a. *Seat Slide*

Fitur *seat slide* untuk mengatur posisi kursi jok ke arah maju atau mundur. Penyetelan ini berguna untuk mengatur jarak antara kursi jok dengan posisi injakan pedal gas, pedal rem dan juga posisi roda kemudi.

- b. *Reclining* Fitur

Reclining digunakan untuk mengatur posisi sandaran kursi jok ke arah depan atau belakang.

c. *Front Vertical*

Fitur *front vertical* digunakan untuk mengatur ketinggian sisi depan dari kursi jok.

d. *Seat Lifter (Rear Vertical)*

Berfungsi untuk mengatur ketinggian kursi jok secara keseluruhan menyesuaikan dengan postur ketinggian pengemudi.

3. *Ottoman Function*

Pada beberapa mobil premium, pada kursi jok baris kedua atau ke tiga dilengkapi dengan *ottoman function*. Bedanya dengan kursi jok biasa adalah dilengkapinya sandaran betis, untuk menambah kenyamanan, yang dapat diatur ketinggiannya. Pada beberapa model juga ditambahkan dengan fitur pijat elektrik. Tersedia berbagai pilihan mode pijatan, area dan kecepatan pijatan. Fitur ini dapat meningkatkan kenyamanan penumpang saat berpergian.

4. *Seat Temperature*

Fitur ini berfungsi untuk mengatur temperatur kursi jok mobil. Tersedia pilihan untuk menghangatkan atau mendinginkan kursi jok mobil. Hal ini tentunya menambah kenyamanan saat berkendara.

5. *Isofix* Fitur

Isofix sangat berguna bagi anda yang membawa anak kecil/balita pada saat perjalanan. Kursi jok dengan fitur *isofix* dapat digunakan untuk menaruh dan mengikat kursi bayi. Anda cukup mengaitkan kursi bayi ke bagian isofix, dan mengikatkan *safety belt* melewati kursi bayi tersebut, dan bayi anda akan lebih aman selama perjalanan. Biasanya kursi isofix tersedia di kursi baris ke dua, karena memang posisi ini lah yang paling tepat dan paling aman untuk anak kecil.

6. *Captain Seat*

Captain seat satu tempat duduk untuk satu orang. *Captain seat* biasanya diletakkan pada kursi baris ke dua atau ke tiga. Pada umumnya kursi baris ke dua dapat memuat 3 penumpang dengan *captain seat* ini hanya dapat digunakan untuk dua penumpang. Penggunaan *captain seat* dapat meningkatkan kenyamanan saat duduk, karena posisi duduk dapat diatur sedemikian rupa

menyesuaikan dengan keinginan penumpang. Beberapa tipe dilengkapi dengan *arm rest*, pengaturan *side support* untuk bagian *headrest* dan lain-lain.

2.12 *System Development Life Cycle (SDLC)*

Menurut Dennis, dkk (2010) *System Development Life Cycle (SDLC)* memiliki 4 perangkat fase dasar yaitu *planning*, *analysis*, *design*, dan *implementation*. Beberapa proyek dalam membangun sebuah sistem informasi mungkin dapat berbeda-beda sesuai dengan caranya masing-masing, tetapi hampir semua proyek memiliki elemen dari 4 fase tersebut. Masing-masing fase tersebut tersusun dari beberapa langkah-langkah yang menghasilkan *deliverable* atau hasil kegiatan contohnya beberapa dokumen spesifik dan file yang menjelaskan pemahaman tentang proyek.

Dalam beberapa proyek, tahapan dan langkah-langkah SDLC diproses dalam suatu alur tahapan, dimulai dari awal hingga akhir. Dalam proyek lain, tim proyek terus berpindah dari fase ke fase tersebut secara teratur, bertahap, iteratif, atau dalam bentuk pola lainnya. Dalam SDLC, dijelaskan fase-fase, tindakan, serta beberapa teknik yang digunakan untuk menyempurnakan langkah-langkah yang dijelaskan secara umum.

Ada dua hal penting untuk dipahami tentang SDLC:

1. Mendapatkan dan memahami pengertian umum dari tahapan dan langkah-langkah yang digunakan dalam proyek Sistem Informasi serta beberapa teknik yang menghasilkan dokumen yang pasti.
2. Kedua hal penting untuk dipahami bahwa SDLC merupakan proses penyempurnaan yang dilakukan secara bertahap.

Dokumen yang dihasilkan dalam tahap analisis, memberikan ide umum dari suatu bagian dari sistem baru. Dokumen dari *deliverable* ini digunakan sebagai *input* pada tahap *design*, yang kemudian disempurnakan untuk menghasilkan dokumen yang menjelaskan secara detail dari sistem yang dibuat. Dokumen ini, akan digunakan dalam tahap implementasi untuk menghasilkan sistem yang sebenarnya. Berikut dibawah ini merupakan penjabaran dari setiap fase dalam SDLC:

1. *Planning* (Perencanaan)

Tahap *planning*/perencanaan adalah proses dasar yang menjelaskan mengapa sebuah sistem informasi harus dibangun dan menentukan bagaimana sebuah tim proyek akan membangunnya. Terdapat dua langkah dalam *planning* (perencanaan) yaitu:

- a. Inisiasi proyek, pada tahap ini nilai bisnis suatu sistem terhadap organisasi diidentifikasi: Apakah sistem tersebut akan menurunkan biaya atau meningkatkan pendapatan? Sebagian besar gagasan untuk sistem baru berasal dari luar area Departemen IT (misalnya dari Departemen Pemasaran, Departemen Akuntansi) dalam suatu bentuk *system request* (permintaan sistem). Permintaan sistem menyajikan ringkasan singkat tentang kebutuhan sistem dalam suatu bisnis dan menjelaskan bagaimana suatu sistem akan menciptakan nilai bisnis. Departemen IT bekerja sama dengan departemen lain yang menghasilkan permintaan (disebut sponsor proyek) untuk melakukan analisis kelayakan sistem informasi.

System request (permintaan sistem) dan analisis kelayakan disajikan dan dipresentasikan kepada pihak komite persetujuan *Information System* (komite pengarah), yang menentukan apakah proyek tersebut harus diambil atau tidak.

- b. Setelah proyek disetujui, proyek memasuki manajemen proyek. Selama manajemen proyek, manajer proyek membuat sebuah rencana kerja, membentuk anggota staff proyek, dan anggota teknisi lainnya di tempat, untuk membantu tim proyek mengendalikan dan mengarahkan proyek pada keseluruhan tahap SDLC.

2. *Analysis* (Analisis)

Analisis menurut Muslihudin (2016) merupakan tahapan awal dalam pengembangan sistem dan merupakan tahap fundamental yang sangat menentukan kualitas sistem yang akan dikembangkan. Tahap analisis dalam SDLC menjelaskan siapa yang akan memakai sistem, apa yang sistem kerjakan serta kapan dan dimana sistem akan digunakan. Selama pada tahap ini, tim proyek menyelidiki sistem yang ada saat ini, mengidentifikasi peluang untuk

melakukan perbaikan, dan mengembangkan sistem baru. Tahap ini memiliki tiga langkah:

- a. Langkah yang pertama adalah Strategi Analisis. Langkah ini dikembangkan untuk mengarahkan usaha dari tim proyek untuk menganalisis sistem yang telah ada (disebut sebagai *as-is system*) beserta masalah-masalah yang ada dan untuk merancang suatu sistem yang baru (disebut sebagai *to-be system*).
- b. Langkah selanjutnya adalah mengumpulkan kebutuhan-kebutuhan sistem (melalui wawancara atau kuesioner). Pada tahap ini terdapat juga masukan analisis informasi dari sponsor proyek dan pihak lainnya yang mengarah pada pengembangan konsep untuk sistem baru. Konsep sistem ini kemudian digunakan sebagai dasar untuk mengembangkan seperangkat model analisis bisnis, yang menggambarkan bagaimana bisnis akan beroperasi jika sistem baru dikembangkan.
- c. Kemudian Analisis, konsep sistem, dan model digabungkan menjadi dokumen yang disebut proposal sistem, yang dipresentasikan ke sponsor proyek dan pengambil keputusan utama lainnya (misalnya, anggota komite persetujuan) yang memutuskan apakah proyek tetap terus dilanjutkan.

3. Design (Perancangan)

Tahap *design* atau perancangan memutuskan bagaimana sistem akan beroperasi, mulai dari hal yang berkaitan dengan perangkat keras (*hardware*), perangkat lunak (*software*) dan infrastruktur jaringan seperti:

- a. Antarmuka pengguna (*interface*),
- b. Formulir,
- c. Laporan,
- d. Program spesifik: *database*, dan *file-file* lainnya yang akan dibutuhkan.

Meskipun sebagian besar keputusan strategis mengenai sistem dibuat dalam pengembangan konsep sistem selama tahap analisis, langkah-langkah dalam tahap desain menentukan secara tepat bagaimana sistem akan beroperasi. Fase desain ini memiliki empat langkah:

- a. Strategi perancangan (*Design Strategy*) adalah hal yang pertama kali yang dikembangkan. Strategi perancangan menjelaskan apakah sistem akan

dikembangkan oleh programmer perusahaan sendiri, atau akan di-*outsourcing* ke perusahaan lain (biasanya perusahaan konsultan), atau apakah perusahaan akan membeli paket perangkat lunak yang sudah ada.

- b. Pengembangan desain arsitektur (*architecture design*) yang mengarah pada dasar untuk sistem, yang menggambarkan perangkat keras, perangkat lunak, dan infrastruktur jaringan yang akan digunakan. Dalam banyak kasus, sistem akan menambah atau mengubah infrastruktur yang sudah ada dalam suatu organisasi. Desain antarmuka menentukan bagaimana pengguna akan berpindah ke sistem yang baru (misalnya, metode navigasi seperti menu dan tombol di layar) serta laporan yang akan digunakan oleh sistem.
- c. Pengembangan *database* dan spesifikasi file. Pengembangan ini menentukan dengan tepat data apa yang akan disimpan dan di mana data tersebut akan disimpan.
- d. Tim analis mengembangkan desain program (*Program Design*), yang mendefinisikan program yang perlu dikembangkan dan apa yang akan dilakukan masing-masing program.

Kumpulan dari hasil kegiatan ini seperti desain arsitektur, desain antarmuka, spesifikasi *database* dan *file*, dan perancangan program merupakan spesifikasi sistem yang diserahkan ke tim pemrograman untuk diimplementasi. Pada akhir tahap desain, analisis kelayakan dan rencana proyek diperiksa ulang dan diperbaiki, dan keputusan lain dibuat oleh sponsor proyek dan komite persetujuan mengenai apakah proyek dihentikan atau dilanjutkan.

4. *Implementation* (Implementasi)

Tahap akhir dalam SDLC adalah tahap implementasi, di mana sistem benar-benar dibangun (atau dibeli, dalam kasus paket *design software*). Fase ini sangat penting, karena kebanyakan sistem merupakan bagian terlama dan paling mahal dari proses pengembangan. Fase implementasi ini memiliki tiga langkah:

- a. Langkah awal adalah Konstruksi Sistem (*System Construction*). Sistem ini dibangun dan diuji untuk memastikan kinerja dari sistem, bekerja sebagaimana yang telah dirancang. Karena biaya *bugs* bisa menjadi sangat besar, pengujian merupakan salah satu langkah paling kritis dalam tahap

implementasi. Sebagian besar organisasi memberi lebih banyak waktu dan perhatian untuk menguji daripada mengubah program di tempat penerapan pertama.

- b. Instalasi Sistem, instalasi adalah proses dimana sistem lama dimatikan dan yang baru dinyalakan atau dihidupkan. Salah satu aspek terpenting dari konversi (perubahan dari suatu sistem ke sistem yang baru) adalah pengembangan rencana pelatihan (*training plan*) untuk mengajarkan pengguna cara menggunakan sistem baru dan membantu mengelola perubahan-perubahan yang disebabkan oleh sistem yang baru.
- c. Tim analis menetapkan sebuah rencana dukungan (*support plan*) untuk sistem. Rencana ini biasanya mencakup tinjauan pasca-pelaksanaan formal atau informal serta cara sistematis untuk mengidentifikasi perubahan besar dan kecil yang diperlukan untuk sistem tersebut.

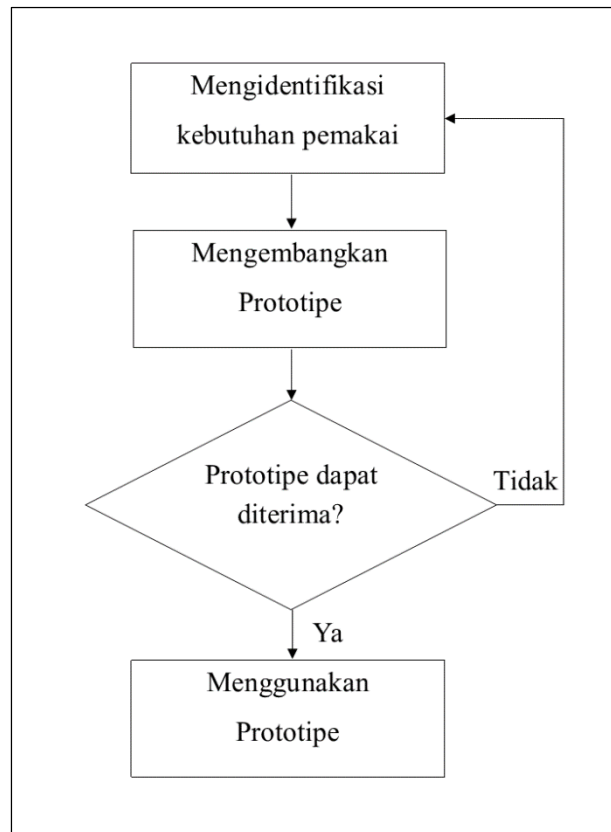
2.13 Model *Prototype*

Menurut Dennis, dkk (2010) model *prototype* atau disebut *prototyping* melakukan fase analis, desain, dan implementasi secara bersamaan, dan ketiga fase tersebut dilakukan berulang kali dalam sebuah siklus sampai sistem yang dibangun selesai. Dengan metodologi ini, dasar-dasar analisis dan desain dilakukan dan segera bekerja pada sistem prototipe. Prototipe pertama memiliki fitur yang sedikit dan biasanya merupakan bagian paling awal dari sistem yang akan digunakan, prototipe ini akan ditampilkan kepada *user* yang akan memberikan komentarnya atas pengalaman dari pemakaian prototipe tersebut, komentar ini selanjutnya akan dimanfaatkan untuk melakukan fase re-analisis, re-desain, dan re-implementasi untuk prototipe kedua, proses ini akan terus berulang sampai analis, *user*, dan sponsor setuju dengan prototipe yang dibuat telah menyediakan fungsionalitas yang cukup untuk dilakukan instalasi pada sebuah organisasi.

2.13.1 *Prototype Evolusioner*

Menurut Dennis, dkk (2010) *prototype evolusioner* adalah *prototype* yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi. Ada

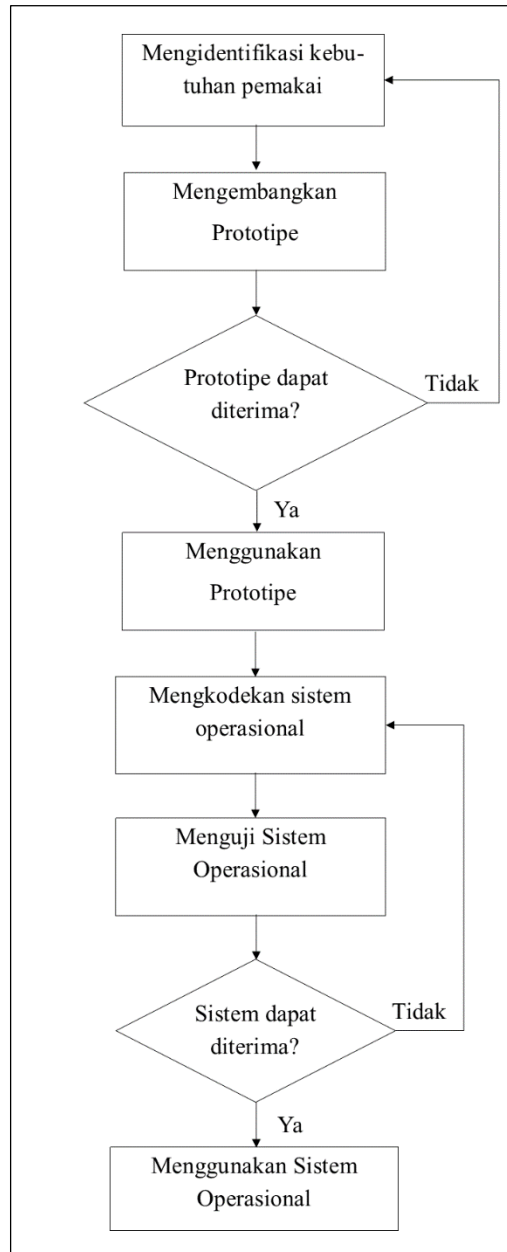
empat langkah yang diambil dalam mengembangkan suatu prototipe *evolusioner* yaitu identifikasi kebutuhan pengguna, mengembangkan prototipe, menentukan prototipe dapat diterima atau tidak, dan penggunaan prototipe.



Gambar II.2. Pengembangan Prototipe *Evolusioner*
(Sumber: McLeod, 2011)

2.13.2 *Prototype Requirement*

Menurut Dennis, dkk (2010) prototipe *requirement* adalah suatu pengembangan untuk menentukan kebutuhan fungsional dari sistem baru pada saat para pengguna tidak mampu mengungkapkan dengan tepat apa yang mereka butuhkan. Saat kebutuhan telah ditentukan prototipe *requirement* dapat mulai dikerjakan dan proyek siap untuk mengembangkan suatu sistem yang baru.



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

2.14 Flowchart

Menurut Sitorus (2015) *Flowchart* merupakan sebuah algoritma yang terstruktur dan mudah dipahami oleh orang lain (khususnya *programmer* yang bertugas mengimplementasikan program), maka dibutuhkan alat bantu yang berbentuk diagram alir (*flowchart*). *Flowchart* menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga *flowchart* merupakan langkah-

langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu. Diagram alir ini akan menunjukkan alur di dalam program secara logika. Diagram alir ini selain dibutuhkan sebagai alat komunikasi, juga diperlukan sebagai alat dokumentasi. Berikut komponen-komponen diagram alir yaitu:

1. Diagram alir digambarkan dengan orientasi dari atas ke bawah dan dari kiri ke kanan.
2. Setiap kegiatan/proses dalam diagram alir harus dinyatakan secara eksplisit.
3. Setiap diagram alir harus dimulai dari satu state dan berakhir pada satu atau lebih terminal akhir/*terminator/halt state*.
4. Gunakan *connector* dan *off-page connector state* dengan label yang sama untuk menunjukkan keterhubungan antar *path* algoritma yang terputus atau terpotong. Misalnya sebagai akibat pindah atau ganti halaman.

Tujuan dari *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas menggunakan simbol-simbol yang standar.

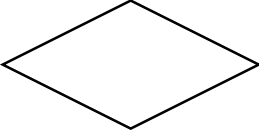
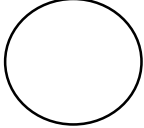
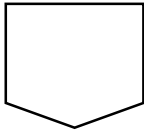
2.14.1 Simbol-simbol *Flowchart*

Menurut Sitorus (2015) tabel berikut merupakan simbol-simbol yang digunakan untuk menggambarkan algoritma dalam bentuk diagram alir dan kegunaan dari simbol-simbol yang bersangkutan.

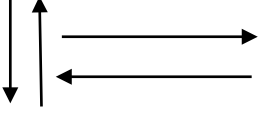
Tabel II.1 Simbol-simbol *Flowchart*

No	Simbol	Nama	Fungsi
1		<i>Terminal</i>	Menyediakan permulaan atau akhir suatu program
2		<i>Input/Output</i>	Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya

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

No	Simbol	Nama	Fungsi
3		<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
4		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban ya atau tidak
5		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
6		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
7		<i>Predefined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
8		<i>Punched Card</i>	Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu
9		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>)

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

No	Simbol	Nama	Fungsi
10		<i>Flow</i>	Menyatakan jalannya arus suatu proses
11		<i>Punched Tape</i>	Menyatakan <i>input</i> atau <i>output</i> yang menggunakan pita kertas berlubang

(Sumber: Sitorus, 2015)

2.14.2 Jenis-jenis *Flowchart*

Menurut Sitorus (2015) *flowchart* dibedakan menjadi 5 Jenis, berikut merupakan penjelasan masing-masing jenis *flowchart*:

1. *System Flowchart*

System Flowchart dapat didefinisikan sebagai bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem ini menunjukkan apa yang dikerjakan di sistem.

2. *Document Flowchart*

Bagan alir dokumen (*document flowchart*) atau disebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

3. *Schematic Flowchart*

Bagan alir (*schematic flowchart*) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Perbedaannya adalah, bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem, juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan dari gambar-gambar ini adalah untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol bagan alir. Penggunaan gambar-gambar ini memudahkan untuk dipahami, tetapi sulit dan lama menggambarinya.

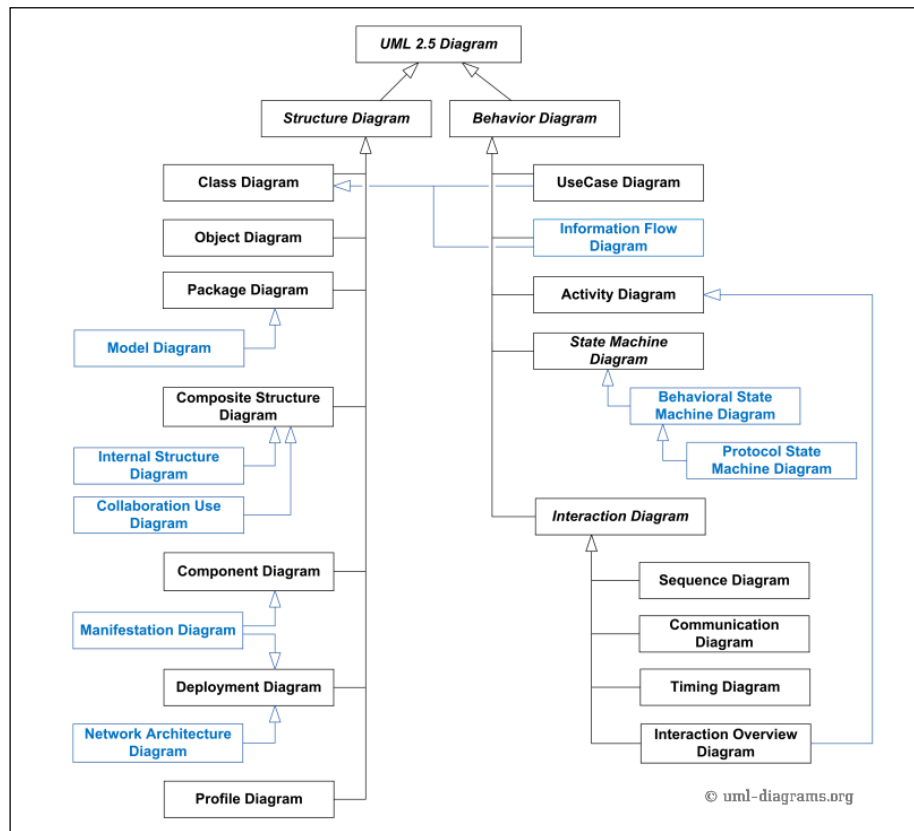
4. *Program Flowchart*

Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dapat terdiri dari dua macam, yaitu bagan alir logika program (*program logic flowchart*) dan bagan alir program komputer terinci (*detailed computer program flowchart*). Bagan alir logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Bagan alir logika program ini dipersiapkan oleh analisis sistem. Proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri. Bagan alir ini juga berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur.

2.15 ***Unified Modelling Language (UML)***

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. Tujuan dari UML adalah untuk menyediakan kosa kata yang umum dari istilah-istilah berbasis objek dan teknik yang cukup banyak untuk memodelkan proyek pengembangan sistem dari analisis ke desain.

Menurut Dennis (2010), diagram-diagram yang ada dalam UML terbagi menjadi dua bagian utama yaitu *Structure diagram* dan *Behavior diagram*. *Structure diagram* biasanya digunakan untuk mempresentasikan data dan hubungan statik yang ada di dalam sebuah sistem informasi. Sedangkan *Behavior diagram* menyediakan para analis dengan sebuah gambaran hubungan yang dinamis antara instansi atau objek yang merepresentasikan sistem informasi bisnis.



Gambar II.4 UML Diagram
(Sumber: uml-diagrams.org, 2017)

Berikut ini penjelasan singkat dari pembagian kategori UML menurut website uml-diagrams.org:

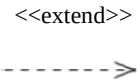
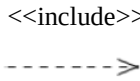
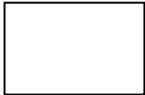
1. *Structure diagrams*, yaitu kumpulan diagram yang menunjukkan struktur statis sistem dan bagian-bagiannya pada tingkat abstraksi dan implementasi yang berbeda dan bagaimana bagian-bagian itu saling terkait satu sama lain. Elemen dalam diagram struktur mewakili konsep sistem yang berarti, dan mungkin mencakup konsep abstrak, dunia nyata dan implementasi.
2. *Behavior diagrams*, yaitu kumpulan diagram yang menunjukkan perilaku dinamis objek dalam suatu sistem, yang dapat digambarkan sebagai rangkaian perubahan sistem dari waktu ke waktu.

2.15.1 Use Case Diagram

Diagram *use case* dibuat untuk menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Suatu landasan pola pikir yang ditekankan dalam

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

Tabel II.2 Simbol-Simbol *Use case Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Use case</i>	Merepresentasikan bagian utama dari sistem secara fungsional.
2		<i>Actor</i>	Seseorang atau sistem yang mendapatkann keuntungan dari sistem
3		<i>Association</i>	Menghubungkan suatu actor dengan <i>use case</i> ..
4		<i>Extend</i>	Merepresentasikan eksistensi dari use case untuk menyertakan perilaku optional
5		<i>Include</i>	Mewakili dimasukkannya satu <i>use case</i> ke <i>use case</i> yang lain.
6		<i>Subject</i>	Menyertakan nama subjek didalam maupun diatas
7		<i>Generalization</i>	Merepresentasikan <i>use case</i> khusus ke yang lebih umum.

(Sumber: Dennis, 2010)

2.15.2 Activity Diagram

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

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

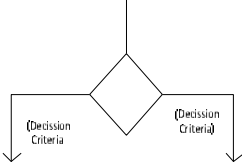
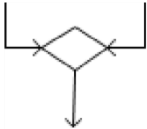
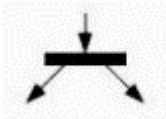
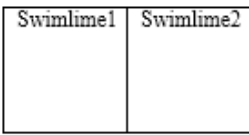
1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
4. Rancangan menu yang akan ditampilkan pada perangkat lunak

Berikut adalah simbol-simbol yang ada pada *Activity Diagram*:

Tabel II.3 Simbol-Simbol *Activity Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Action</i>	Merupakan notasi yang simple, dan tidak untuk dianalisis perilakunya. Dinamakan sesuai aksi yg dilakukan.
2		<i>Activity</i>	Digunakan untuk mewakili satu set aksi yang dilakukan.
3		<i>Object Node</i>	Digunakan untuk mewakili objek yang terhubung ke satu set arus objek.

Tabel II.3 Simbol-Simbol *Activity Diagram* (lanjutan)

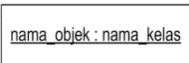
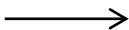
No	Simbol	Nama	Deskripsi
4		<i>Control Flow</i>	Menampilkan urutan eksekusi.
5		<i>Object Flow</i>	Menunjukkan aliran dari sebuah objek dari satu aktivitas ke aktivitas lainnya.
6		<i>Initial Node</i>	Menggambarkan permulaan dari suatu set aktivitas atau aksi.
7		<i>Final-Flow Node</i>	Menggambarkan akhir dari setiap aktivitas.
8.		<i>Decision Node</i>	Untuk mewakili suatu kondisi pengujian, yang bertujuan untuk memastikan bahwa <i>control flow</i> atau <i>object flow</i> hanya menuju ke satu arah.
9.		<i>Merge Node</i>	Untuk menyatukan kembali <i>decision path</i> yang dibuat dengan menggunakan <i>decision node</i> .
10.		<i>Fork node</i>	Untuk memisahkan perilaku menjadi serangkaian aktivitas yang berjalan secara parallel atau bersamaan.
11.		<i>Join Node</i>	Untuk menyatukan kembali serangkaian aktivitas yang berjalan secara parallel atau bersamaan.
12.		<i>Swimlane</i>	Untuk membagi sebuah <i>activity diagram</i> menjadi beberapa kolom, dengan tujuan menempatkan aktivitas tertentu pada individu yang bertanggung jawab atas terlaksananya aktivitas tersebut.

(Sumber: Dennis, 2010)

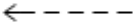
2.15.3 Sequence Diagram

Menurut Denis (2010) *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 *usecase*. 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:

Tabel II.4 Simbol-Simbol *Sequence* Diagram

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

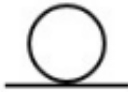
Tabel II.4 Simbol-Simbol *Sequence Diagram* (lanjutan)

6.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri secara langsung.
7.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri.

(Sumber: Dennis, 2010)

Untuk menggambarkan simbol objek pada *sequence diagram* diperlukan simbol tambahan. Adapun simbol *sequence diagram* yang dikutip dari Windu Gata (2013) adalah sebagai berikut.

Tabel II.5 Simbol *Sequence Diagram* pada jenis objek

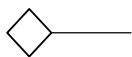
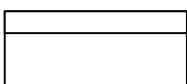
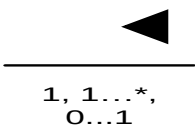
No	Simbol	Nama	Deskripsi
1		<i>Boundary Class</i>	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi interface atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan form cetak.
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilaksanakan
3		<i>Control Class</i>	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.

(Sumber: Windu Gata, 2013)

2.15.4 Class Diagram

Sebuah diagram kelas adalah model statis yang menunjukkan kelas dan hubungan antar kelas yang tetap konstan dalam sistem dari waktu ke waktu. Sebuah *class* merupakan deskripsi dari sekumpulan objek yang memiliki properti (*attribute*), operasi (*method*), relasi (*association*), dan tingkah laku () yang sama. Sebuah class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda /fungsi). (Alan Dennis, 2010). Berikut simbol-simbol yang ada pada diagram kelas:

Tabel II.6 Simbol-Simbol *Class Diagram*

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

(Sumber: Dennis, 2010)

2.16 ERD (*Entity Relationship Diagram*)

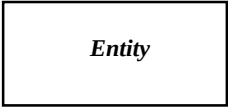
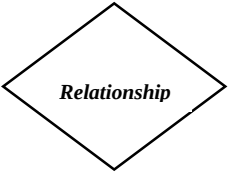
Menurut Mulyani (2016) *Entity Relationship Diagram* adalah salah satu *tools* diagram yang digunakan untuk memodelkan konseptual (abstraksi) data. Diagram ini sangat populer dan banyak digunakan oleh para pengembang sistem dalam memodelkan data. *ERD* adalah *tools* yang digunakan untuk melakukan permodelan data secara abstrak dengan tujuan untuk mendeskripsikan atau menggambarkan struktur dari data yang akan digunakan (Bagui dan Erap). Menurut Wikipedia (2007) mengatakan bahwa *ERD* adalah *tools* yang digunakan untuk

memodelkan data dengan tujuan untuk menghasilkan penggambaran struktur *database* secara konseptual dengan metode *top-down*.

2.16.1 Konsep ERD

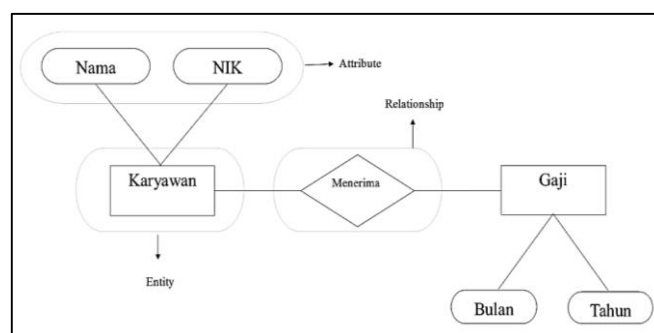
Menurut Suryantara (2015) pada dasarnya ERD dibentuk oleh beberapa notasi, yaitu:

Tabel II.7 Notasi ERD

No	Notasi	Deskripsi
1.		Orang, tempat, atau benda yang memiliki nama tunggal.
2.		Properti dari entitas yang digunakan oleh minimal 1 proses bisnis dipecah dalam detail.
3.		Menunjukkan hubungan antar 2 entitas dideskripsikan dengan kata kerja, memiliki modalitas (null, not null). Memiliki kardinalitas.

(Sumber: Suryantara, 2015)

Berikut merupakan gambar yang menggambarkan ERD yang dibentuk dari ketiga notasi tersebut.

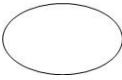
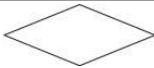
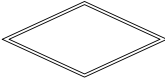
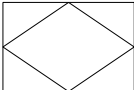


Gambar II.5 contoh ERD
sumber: Mulyani (2016)

Sedangkan menurut Rosa dan Shalahuddin (2016) ERD adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang

matematika. ERD digunakan untuk pemodelan basis data relasional, simbol-simbol yang digunakan dalam ERD, yaitu:

Tabel II.8 Simbol-simbol *Entity Relationship Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Strong Entity</i>	Karakteristik atau tipe entitas yang tidak bergantung pada entitas lain.
2		<i>Attribute</i>	Properti atau karakteristik tipe entitas.
3		<i>Multivalued Attribute</i>	Karakteristik tipe entitas yang dapat memiliki nilai lebih dari satu.
4		<i>Relationship</i>	Hubungan antar entitas.
5		<i>Weak Entity</i>	Entitas yang bergantung pada entitas lain.
6		<i>Identifying Entity</i>	Entitas yang merupakan induk dari <i>weak entity</i> .
7		<i>Associative Entity</i>	Hanya bisa terdapat antara 2 entitas dan menggambarkan hubungan diantara 2 entitas tersebut.
8		<i>Derived Attribute</i>	Suatu atribut yang dihasilkan dari atribut yang lain.

(Sumber: Hoffer, 2010)

Di dalam ERD terdapat *multiplicity*, *multiplicity* merupakan suatu ketentuan (*constraint*) yang mengatur hubungan antara objek-objek entitas. Oleh karena itu, *multiplicity* merupakan kebijakan atau aturan bisnis (*business rule*) yang ditetapkan oleh pemakai atau organisasi, yang dapat berbeda dari satu organisasi dengan

organisasi lain. Lebih lanjut, *multiplicity* dapat dibagi menjadi dua batasan atau aturan, yaitu:

1. Batasan kardinalitas

Batasan kardinalitas (*cardinality constraint*) yang menentukan jumlah maksimum objek entitas lain yang dapat berasosiasi dengan suatu objek entitas tertentu.

2. Batasan partisipasi

Batasan partisipasi (*participation constraint*) yang menentukan apakah semua atau sebagian objek entitas berpartisipasi pada suatu jenis hubungan entitas.

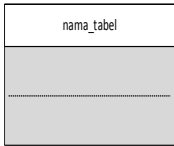
Perbandingan kardinalitas yang dilihat dari dua arah hubungan disebut dengan rasio kardinalitas atau derajat hubungan, terdapat tiga jenis rasio kardinalitas sebagai berikut: (Pahlevi, 2013)

1. *One-to-one* (1:1), jika nilai kardinalitas pada kedua arah 1.
2. *One-to-many* (1:N) atau *many-to-one* (N:1), jika nilai kardinalitas pada salah satu arah satu (1) dan arah lainnya banyak (N).
3. *Many-to-many* (N:N), jika nilai kardinalitas pada kedua arah banyak (N).

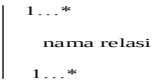
2.17 Conceptual Data Model

Menurut Rosa dan Shalahuddin (2016) CDM (*Conceptual Data Model*) atau model konsep data adalah konsep yang berkaitan dengan pandangan pemakai terhadap data yang disimpan dalam basis data. CDM dibuat sudah dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel untuk keperluan implementasi ke basis data. CDM merupakan hasil penjabaran lebih lanjut dari ERD. Berikut adalah simbol-simbol yang ada pada CDM:

Tabel II.9 Simbol-Simbol *Conceptual Data Model* (CDM)

Simbol	Deskripsi
Entitas/table 	Entitas atau tabel yang menyimpan data dalam basis data

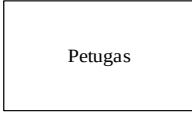
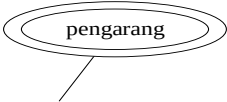
Tabel II.9 Simbol-Simbol *Conceptual Data Model* (CDM) lanjutan

Simbol	Deskripsi
	Relasi antar tabel yang terdiri atas nama relasi dan <i>multiplicity</i>

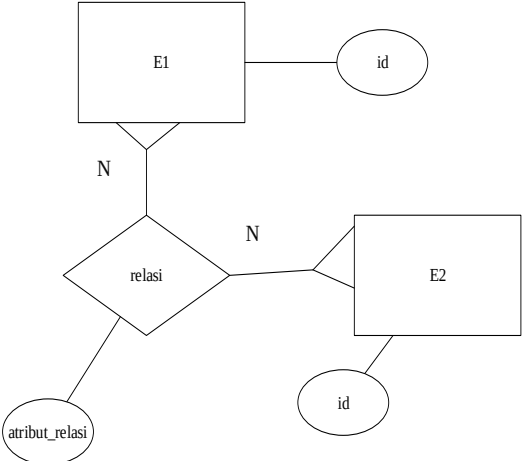
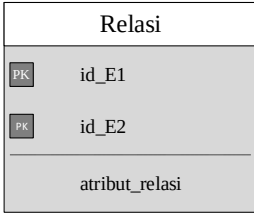
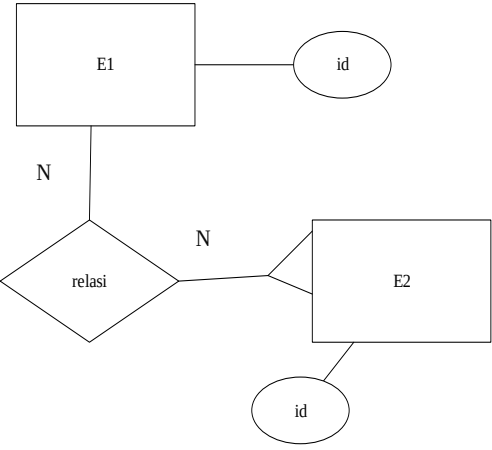
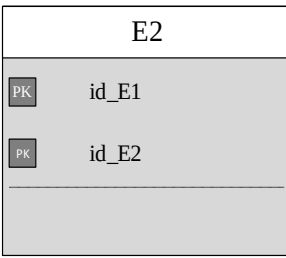
(Sumber: Rosa dan Shalahuddin, 2016)

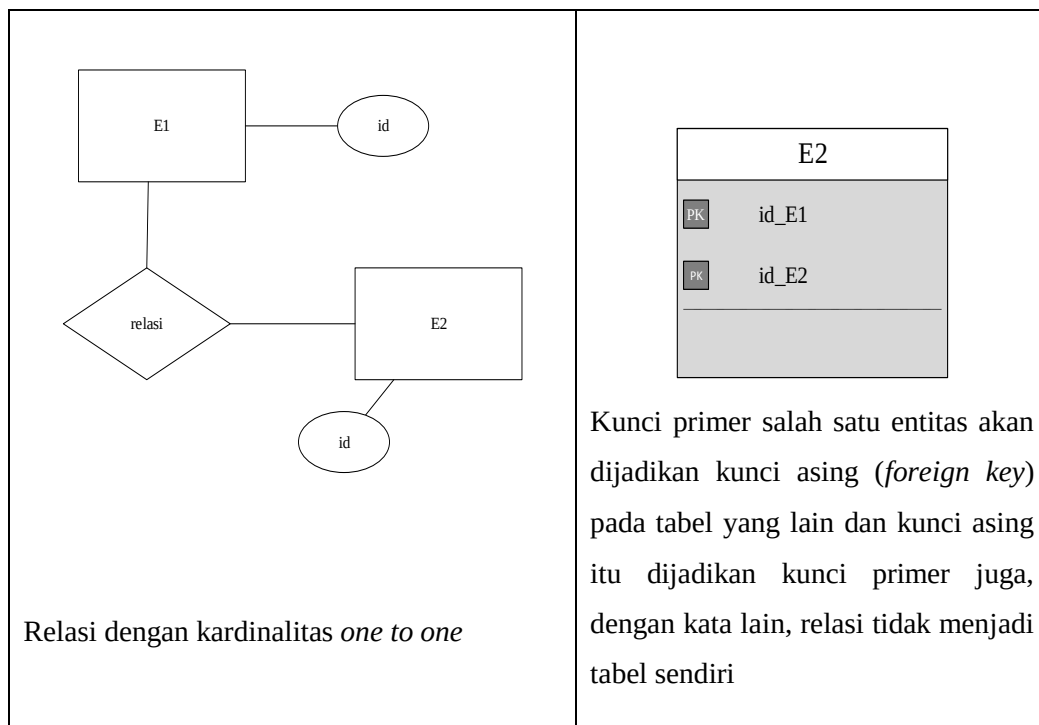
Aturan untuk mengubah ERD menjadi CDM secara umum adalah sebagai berikut:

Tabel II.10 Simbol-Simbol Aturan *Conceptual Data Model* (CDM)

ERD	CDM
 Entitas	 Menjadi sebuah tabel tersendiri
 Atribut <i>multivalue</i>	 Menjadi sebuah tabel tersendiri dengan kunci primer (primary key) adalah kunci primer pada entitas dan memiliki atribut dengan nama seperti pada atribut entitas

Tabel II.10 Simbol-Symbol Aturan *Conceptual Data Model* (CDM) (Lanjutan)

ERD	CDM
 Relasi dengan kardinalitas <i>many to many</i>	 Menjadi sebuah tabel tersendiri dengan kunci primer adalah atribut yang menjadi kunci primer di kedua entitas yang direlasikannya.
 Relasi dengan kardinalitas <i>one to many</i>	 Kunci primer entitas yang memiliki hubungan <i>one</i> akan dijadikan kunci primer di entitas yang memiliki hubungan <i>many</i> dengan kata lain, relasi tidak menjadi tabel sendiri

Tabel II.10 Simbol-Simbol Aturan *Conceptual Data Model* (CDM) (Lanjutan)

(Sumber: Rosa dan Shalahuddin, 2016)

2.18 Kamus Data

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

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

Keuntungan penggunaan kamus data adalah (Sommerville, 2003):

1. Kamus data merupakan mekanisme untuk manajemen nama. Banyak orang yang harus menciptakan nama untuk entitas dan relasi ketika mengembangkan model sistem yang besar. Nama-nama ini harus dipakai secara konsisten dan

tidak boleh bentrok. Kamus data dapat memeriksa keunikan nama dan memberitahu analis persyaratan sekiranya terjadi duplikasi nama.

2. Kamus data sebagai tempat penyimpanan informasi yang dapat menghubungkan analis, desain, implementasi dan evolusi. Sementara sistem dikembangkan, informasi diambil untuk memberitahu perkembangan informasi baru ditambahkan pada sistem. Semua informasi mengenai entitas berada pada satu tempat. Berikut adalah contoh penulisan kamus data:

Spesifikasi Tabel pemasok

Nama Tabel : Pemasok

Tipe : File master

Tabel II.11 Contoh Kamus Data Untuk Tabel Pemasok

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

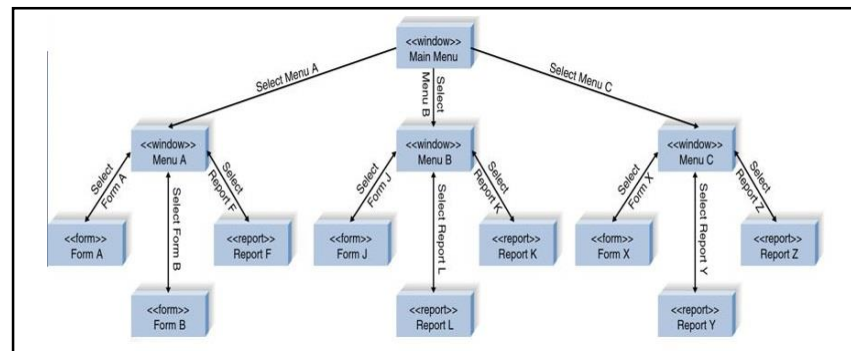
(Sumber: Jogiyanto, 2005)

2.19 Windows Navigation Diagram

Windows Navigation Diagram (WND) menurut Dennis (2010):

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

Contoh WND menurut Dennis (2011) dapat dilihat pada Gambar II.6.



Gambar II.6 Contoh *Windows Navigation Diagram*
(Sumber: Dennis, 2011)

2.20 Java

Menurut Jubilee Enterprise (2015) Java merupakan pemrograman yang menanjak popularitasnya dewasa ini. Selain gratis, java mudah didapatkan dan juga tangguh. Java sendiri lahir pada tahun 1991 yang diprakasai oleh tim Sun melalui proyek bernama Green yang dipimpin oleh James Gosling.

Java menerapkan fitur-fitur dari bahasa pemrograman yang lain, di mana fitur itu dianggap merupakan suatu kelebihan oleh tim Sun. Misalnya JVM/JRE yang merupakan mesin maya pada bahasa *Pascal*, *Sintaks* dan *Exception Handling* diambil dari C/C++ dan lain sebagainya.

Java adalah bahasa pemrograman yang dapat membuat seluruh bentuk aplikasi, desktop, web, mobile dan lainnya, sebagaimana dibuat dengan menggunakan bahasa pemrograman konvensional yang lain. Bahasa Pemrograman Java ini berorientasi objek (*OOP-Object Oriented Programming*), dan dapat dijalankan pada berbagai platform sistem operasi. Perkembangan Java tidak hanya terfokus pada satu sistem operasi, tetapi dikembangkan untuk berbagai sistem operasi dan bersifat open source.

2.20.1 Netbeans

Menurut Jubilee Enterprise (2015) Netbeans merupakan sebuah aplikasi *Integrated Development Environment* (IDE) yang berbasiskan Java dari Sun Microsystems yang berjalan di atas swing. Swing merupakan sebuah teknologi Java

untuk pengembangan aplikasi desktop yang dapat berjalan pada berbagai macam *platform* seperti windows, linux, Mac OS dan Solaris. Sebuah IDE merupakan lingkup pemrograman yang di integrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan *Graphic User Interface* (GUI), suatu kode editor atau teks, suatu *compiler* dan suatu *debugger*.

Netbeans juga digunakan oleh sang programmer untuk menulis, meng-*compile*, mencari kesalahan dan menyebarkan program netbeans yang ditulis dalam bahasa pemrograman java namun selain itu dapat juga mendukung bahasa pemrograman lainnya dan program ini pun bebas untuk digunakan dan untuk membuat *professional desktop*, *enterprise*, *web*, and *mobile applications* dengan *Java language*, *C/C++*, dan bahkan *dynamic languages* seperti *PHP*, *JavaScript*, *Groovy*, dan *Ruby*.

2.20.2 Fitur-Fitur Netbeans

Menurut Jubilee Enterprise (2015) Netbeans memiliki banyak kelebihan dari segi *code* sampai pada penggunaan aplikasinya. Hal tersebut tidak terlepas dari fitur yang dimiliki oleh Netbeans. Fitur-fitur yang terdapat di dalamnya antara lain:

1. *Smart Code Completion*: untuk mengusulkan nama variabel dari suatu tipe, melengkapi *keyword* dan mengusulkan tipe parameter dari sebuah *method*.
2. *Bookmarking*: fitur yang digunakan untuk menandai baris yang suatu saat hendak kita modifikasi.
3. *Go to Commands*: fitur yang digunakan untuk *jump* ke deklarasi variabel, *source code* atau file yang ada pada project yang sama.
4. *Code Generator*: jika kita menggunakan fitur ini kita dapat meng-*generate constructor*, *setter* dan *getter method* dan yang lainnya.

Error Stripe: fitur yang akan menandai baris yang eror dengan memberi *highlight* merah.

2.21 MySQL

Menurut Welling dan Thomson (2003) *My Structured Query Language* (MySQL) adalah *Relational Database Management System* (RDBMS) yang sangat

cepat dan kuat. Sebuah basis data memungkinkan *user* untuk melakukan penyimpanan yang efisien, pencarian dan pengurutan data. MySQL server memiliki kemampuan melakukan kontrol akses terhadap data untuk memastikan bahwa setiap *user* dapat bekerja dengan sesuai, menyediakan akses yang cepat, dan meyakinkan bahwa hanya *user* yang mempunyai hak akses yang dapat mengaksesnya. Oleh karena itu, MySQL merupakan *database server* yang dapat digunakan banyak *user* dan banyak pekerjaan.

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

2.21.1 Keunggulan MySQL

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

1. *Portability*

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

2. *Open source*

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

3. *Multi User*

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

4. *Performance tuning*

Mempunyai kecepatan yang tinggi dalam menangani *query*.

5. *Column types*

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

6. *Command* dan *function*

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

7. *Security*

Memiliki beberapa lapisan sekuritas seperti tingkat *subnet mask*, *hostname*, *privilege user* dengan sistem perijinan yang mendetail serta *password* yang terenkripsi.

8. *Scalability* dan *limits*

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

9. *Localization*

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

10. *Connectivity*

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

11. *Interface*

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

12. *Client* dan *tools*

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

13. Struktur tabel

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

2.21.2 Tipe Data MySQL

Menurut Sutaji (2012) MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam

merancang *table*. Beberapa jenis data yang tersedia pada MySQL dapat dilihat pada Tabel II.8 (Sutaji, 2012).

Tabel II.11 Jenis Tipe Data pada MySQL

Jenis Data	Keterangan
CHAR	Sebuah <i>string</i> dengan panjang tetap. Sisa Jumlah karakter yang belum terisi akan diisi dengan spasi, akan tetapi spasi ini dibuang jika data dipanggil. Jangkauan nilai M adalah 1-255 karakter
VARCHAR	<i>String</i> dengan panjang berupa variabel. M bisa mencapai 65535
DATE	Data berupa tanggal. Format tanggal dalam bentuk ‘YYYY-MM-DD’
TIME	Data berupa waktu. Format waktu dalam bentuk ‘HH:MM:SS’
TINYINT	Bilangan antara -128 sampai dengan +127
SMALLINT	Bilangan antara -32768 sampai dengan +32767
INT	Bilangan antara -2147683648 sampai dengan +2147683647
FLOAT	Bilangan <i>floating point</i> yang kecil (presisi tunggal). Jangkauan nilai yang diizinkan adalah -3.402823466E+38 s/d -1.175494351E-38,0 dan 1.175494351E-38 s/d 3.402823466E+38
DOUBLE	Bilangan <i>floating point</i> dengan ukuran normal (presisi ganda). Jangkauan nilai yang diizinkan adalah -1.7976931348623157E+308 s/d -2.225073858507201E-308,0 dan 2.225073858507201E-308 s/d 1.7976931348623157E+308
ENUM	Sebuah <i>enumeration</i> . Sebuah obyek string yang hanya boleh memiliki satu nilai, yang terambil dari ‘value1’, ‘value2’, ..., NULL atau nilai spesial “” <i>error</i> ”. Sebuah <i>enum</i> dapat menampung 65535 pilihan nilai
TEXT, BLOB	Sebuah TEXT atau BLOB dengan panjang karakter maksimum 65535 karakter

Sumber: Sutaji (2012)

2.22 s Functional Testing

Functional testing merupakan bagian dari black box testing, menurut Simarmata (2010) functional testing adalah pengujian yang dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan. Pengujian fungsional meliputi seberapa baik sistem melaksanakan fungsinya, termasuk perintah-perintah pengguna, manipulasi data, pencarian dan proses bisnis, pengguna layar, dan integrasi.

BAB III

METODOLOGI PENELITIAN

Metodologi penelitian atau *methodology of research* berasal dari kata “*metoda*” yang berarti cara atau teknik dan “*logos*” yang berarti ilmu. Untuk mendapatkan gambaran yang lebih jelas tentang arti metodologi penelitian, maka beberapa pengertian atau definisi metodologi penelitian adalah sebagai berikut:

1. Menurut Siyoto dan Sodik (2015) metodologi penelitian merupakan langkah-langkah yang kita gunakan dalam melakukan suatu penelitian dan melakukan analisis kritikal dari metode penelitian.
2. Metodologi penelitian menurut Hidayat dan Sedarmayanti (2011) adalah pembahasan mengenai konsep teoritik berbagai metode, kelebihan dan kekurangan, yang dalam karya ilmiah dilanjutkan dengan pemilihan metode yang digunakan.
3. Metodologi penelitian menurut Narbuko dan Achmadi (2012) adalah ilmu/pengetahuan yang mempelajari tentang usaha untuk menemukan atau mencari kembali, mengembangkan, dan menguji kebenaran suatu pengetahuan dengan cara ilmiah yang berjalan secara terus-menerus untuk mendapatkan data dengan tujuan dan kegunaan tertentu.

Dari beberapa pengertian dan definisi diatas maka dapat diperoleh kesimpulan pengertian metodologi penelitian adalah sebagai suatu usaha atau proses mencari jawaban atas suatu pertanyaan atau masalah dengan cara sabar, hati-hati, terencana, sistematis atas fakta atau prinsip-prinsip, mengembangkan dan menguji kebenaran ilmiah suatu pengetahuan.

3.1 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 Suzuki Indomobil Motor Cikarang Plant.

1. Data Primer

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

2. Data Sekunder

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

3.2 Metode Pengumpulan Data

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

1. Studi Lapangan

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

- a. Pengamatan, tahap ini dilakukan secara langsung di Seksi *Asssembling seat* PT Suzuki Indomobil Motor Cikarang Plant dengan mengamati pencatatan laporan harian hasil produksi yang sedang berjalan mulai dari proses produksi yang dilakukan oleh bagian produksi sampai dengan pencatatan data laporan harian hasil produksi pada *daily report*. Melalui teknik ini data yang dibutuhkan untuk diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.
- b. Wawancara, yaitu mencari data yang dibutuhkan secara langsung melalui memberikan pertanyaan spontan terhadap segala hal yang diperlukan pada

penyusunan Tugas Akhir ini. Wawancara ini dilakukan kepada karyawan pada seksi *Assembling Seat* di PT Suzuki Indomobil Motor Cikarang Plant.

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.

3.3 Metode Pengembangan Sistem

Masalah yang ada pada sistem saat ini telah teridentifikasi pada tahap sebelumnya, dan akan menjadi gambaran untuk mengetahui tindakan yang harus dilakukan untuk mengatasi masalah tersebut. Pada penelitian ini untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat pengembangan sistem. Sistem dikembangkan menjadi lebih terkomputerisasi dan berbasis objek. Pengembangan sistem dilakukan dengan menggunakan metode *prototype evolutioner*, Metode prototipe ini terdiri dari tahap identifikasi kebutuhan pengguna, membuat prototipe, menentukan prototipe diterima atau tidak dan menggunakan prototipe.

Berikut ini akan dibahas secara singkat mengenai tahapan-tahapan yang dilakukan dalam pengembangan sistem.

1. Mengidentifikasi kebutuhan pengguna

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

2. Membuat satu prototipe

Pengembangan mempergunakan satu alat *prototyping* atau lebih untuk membuat prototipe. Contoh dari alat-alat *prototyping* tersebut adalah generator aplikasi terintegrasi dan *toolkit prototyping*. Generator aplikasi terintegrasi adalah sistem piranti lunak siap pakai yang mampu membuat seluruh fitur yang diinginkan dari sistem baru. Sedangkan *toolkit prototyping* meliputi sistem-

sistem piranti lunak terpisah, seperti sistem manajemen basis data, yang masing-masing mampu membuat sebagian dari fitur-fitur sistem yang diinginkan.

3. Menentukan apakah prototipe dapat diterima

Pengembangan mendemonstrasikan prototipe kepada pengguna untuk mengetahui apakah telah memberikan hasil yang memuaskan. Jika iya, akan dilakukan langkah selanjutnya, dan jika tidak, prototipe akan direvisi dari tahap awal dengan pemahaman yang lebih baik mengenai kebutuhan pengguna.

4. Menggunakan prototipe

Menjadikan prototipe sebagai sistem yang dijalankan.

3.4 Kerangka Penelitian

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

1. Pendahuluan

Pendahuluan merupakan langkah awal yang dilakukan untuk memulai penelitian. Dengan melakukan studi pustaka, observasi, dan wawancara.

2. Identifikasi Masalah

Pada tahap ini bertujuan untuk mengetahui sistem yang sedang berjalan. Serta mengetahui masalah yang ada pada sistem tersebut.

3. Rumusan Masalah

Tahap ini bertujuan untuk mencari jalan keluar dari masalah yang ada pada sistem.

4. Tujuan Penelitian

Tujuan penelitian dimaksudkan agar dapat mencapai tujuan yang hendak dicapai dari penelitian.

5. Batasan Masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya.

6. Penerapan Metode *Prototype Evolusioner*

a. Mengidentifikasi kebutuhan pelanggan

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

b. Membuat prototipe

Membuat sebuah prototipe sesuai dengan analisis yang dilakukan. Pada tahap ini akan dirancang aplikasi tentang sistem pengendalian kualitas bahan baku. Membuat prototipe nantinya akan memuat tentang:

- Memodelkan sistem menggunakan *Unified Modeling Language (UML)*.
- Memodelkan basis data dengan menggunakan *class diagram* dan kamus data.
- Merancang sistem dengan menggunakan *Windows Navigation Diagram*, *Flowmap*, dan *Interface*.
- Membuat prototipe dengan menggunakan NetBeans sebagai *software* dan MySQL 5.6.12 sebagai *databasenya*.

c. Mengevaluasi prototipe

Memutuskan untuk menggunakan prototipe yang telah divalidasi menggunakan metode *prototype evolusioner*. Apabila prototipe tidak sesuai dengan kebutuhan *user*, maka akan kembali ke tahap identifikasi.

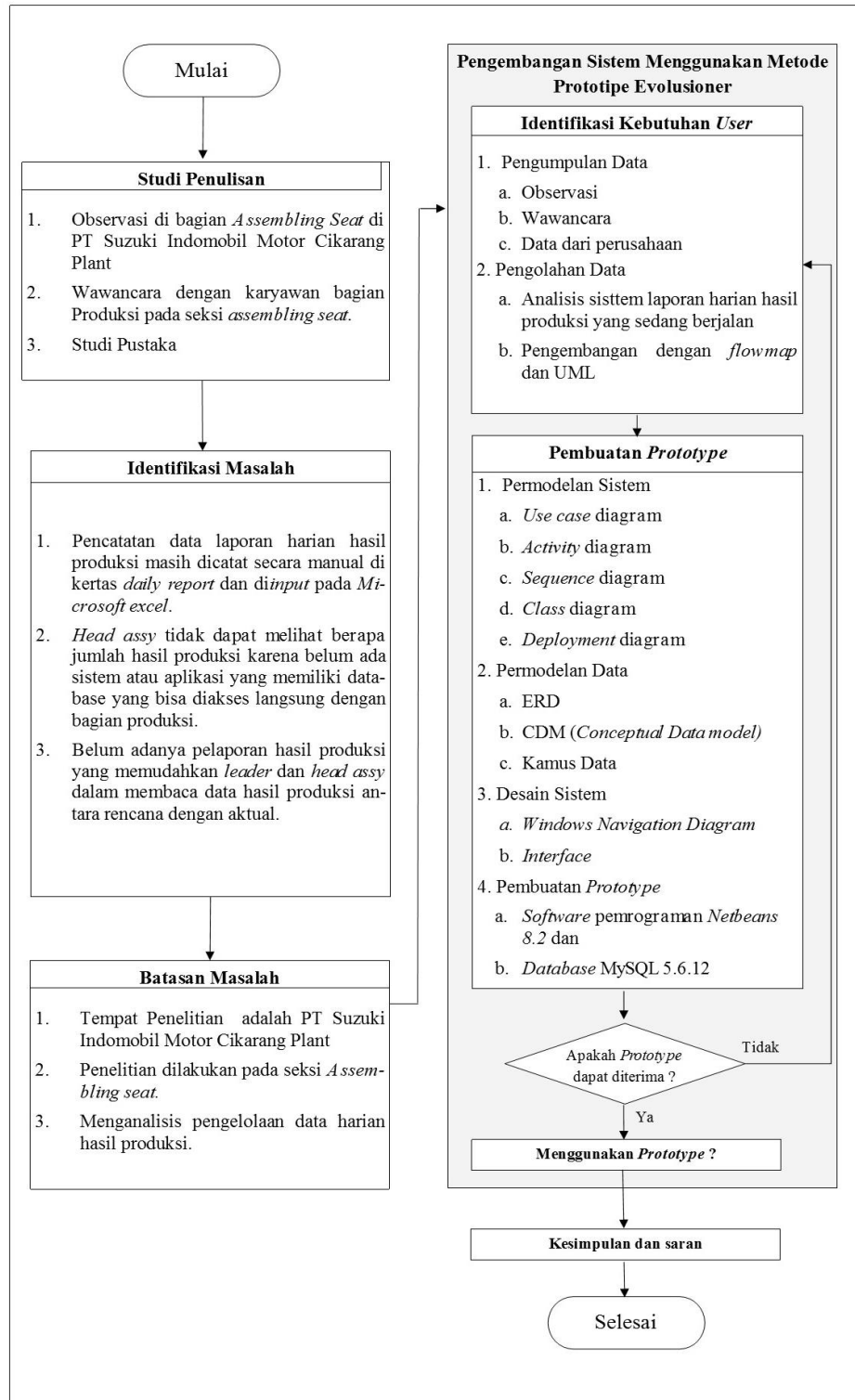
d. Menggunakan prototipe

Tahap ini merupakan tahap akhir dari pembuatan sistem dengan metode prototipe evolusioner. Pada tahap ini perangkat lunak yang sudah jadi dan sudah lulus uji, siap untuk digunakan oleh *user*.

7. Kesimpulan dan Saran

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

Berikut adalah Gambar III.1 *Flowchart* kerangka penelitian untuk menyelesaikan masalah dalam Tugas Akhir ini:



Gambar III.1 Kerangka Penelitian
(Sumber: Pengolahan Data, 2018)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Tentang Perusahaan

PT Suzuki Indomobil Motor merupakan salah satu perusahaan otomotif terkemuka di Indonesia. Dimana perusahaan ini memproduksi kendaraan roda empat dan kendaraan roda dua. PT Suzuki Indomobil Motor merupakan anak cabang perusahaan dari Suzuki Group yang berpusat di Jepang sehingga masih dikontrol secara terpusat oleh Suzuki Group.

PT Suzuki Indomobil Motor meresmikan pabrik baru nya dengan fasilitas perakitan mobil, produksi mesin dan transmisi mobil di Kawasan Industri Greendland International Industrial Center (GIIC) Blok AC No.1 Kota Deltamas, Cikarang Pusat-Bekasi, Jawa Barat. Pabrik PT Suzuki Indomobil Motor yang terletak di Cikarang merupakan pabrik Suzuki Indonesia yang ke-4, dimana pabrik ini merupakan pabrik perakitan mobil ke-2 setelah Tambun Plant 2, Jl. Raya Diponegoro Km, 38,2 Tambun-Bekasi, Jawa Barat, serta pabrik produksi mesin dan transmisi ke-2 setelah Cakung Plant, Jl. Raya Penggilingan, Jakarta Timur. PT Suzuki Indomobil Motor Cikarang Plant diresmikan pada tanggal 29 Mei 2015 sebagai salah satu produsen produk mobil Ertiga seluruh tipe untuk pasar domestik (Indonesia) dan ekspor (Thailand dan Brunei), dan mesin Ertiga (K14B/1.400cc) dan Karimun Wagon R (K10B/1.000cc), serta transmisi kendaraan roda empat (untuk pasar nasional & ekspor ke Thailand, Pakistan). Berdiri diatas luas tanah 130,7 Hektar dan luas bangunan 11 Hektar, pabrik baru PT Suzuki Indomobil Motor Cikarang Plant ini merupakan pabrik terluas dan terbesar diantara pabrik-pabrik yang dimiliki PT Suzuki Indomobil Motor di Indonesia saat ini. Pabrik yang memiliki berbagai fasilitas dengan teknologi terkini dan modern dengan mengikuti standar global prinsipal Suzuki Motor Corporation Japan.

4.2 Profil Perusahaan

Profil Perusahaan merupakan salah satu aspek penting dalam publikasi sebuah perusahaan. Fungsi sebuah profil perusahaan adalah sebagai media komunikasi dan penyampaian informasi tentang perusahaan terhadap pihak-pihak terkait, baik internal maupun eksternal perusahaan. Berikut adalah profil dari perusahaan PT Suzuki Indomobil Cikarang Plant:

Nama Perusahaan	: PT Suzuki Indomobil Motor Cikarang Plant
Status Perusahaan	: Perseroan Terbatas
Jenis Usaha	: Komponen Otomotif Mobil
Berdiri	: 29 Mei 2015
Lokasi	: Kawasan Industri Greendland International Industrial Center (GIIC) Blok AC No.1 Kota Deltamas, Cikarang Pusat–Bekasi, Jawa Barat.
Telp	: 021-29601900
Fax	: 021-29601901/021-29601919
Luas Tanah	: 130,7 Hektar
Luas Bangunan	: 11 Hektar
Jumlah Karyawan	: ± 1.152 orang
Aktivitas	: Perakitan mobil, Produksi mesin, dan transmisi mobil
Produksi	: Mobil Ertiga seluruh tipe
Kapasitas Produksi	: ± 60.000 Mobil per tahun
Status Kepemilikan	: 99% Suzuki Motor Corporation Japan 1% PT Suzuki Indomobil Motor
Jam Kerja	
Kantor	: 07.30 - 16.30 WIB
Pabrik Shift I	: 07.30 - 16.30 WIB
Pabrik Shift II	: 17.00 - 23.00 WIB
Pabrik Shift III	: 23.30 - 07.00 WIB
Website	: https://www.suzuki.co.id/

4.3 Logo Perusahaan

PT Suzuki Indomobil Motor memiliki logo perusahaan sebagai berikut:



Gambar IV.1 Logo Perusahaan PT Suzuki Indomobil Motor
(Sumber: PT Suzuki Indomobil Motor, 2017)

4.4 Visi, Misi, *Strategy* dan *Values* Perusahaan

PT Suzuki Indomobil Motor Cikarang Plant adalah perusahaan otomotif yang bergerak dibidang produksi kendaraan roda empat (R4). Perusahaan ini mempunyai visi dan misi perusahaan sebagai berikut:

Visi:

Menjadi perusahaan yang terkemuka di dalam Suzuki Global Operation (*challenge*) yang dihargai dan dikagumi di Indonesia (*customer satisfaction*).

Misi:

Menginginkan pertumbuhan dan pengembangan perusahaan yang berimbang berdasarkan azaz kerja keras integritas dan kebersamaan untuk selalu mencapai hasil yang lebih baik.

***Strategy*:**

1. *Network*

Membangun jaringan strategi yang menguntungkan.

2. *Market*

Membangun “*Brand equity*” produk yang kuat dan penetrasi segmentasi pasar.

3. *Product*

Mengembangkan dan memproduksi produk berkualitas yang memberikan nilai tambah bagi pengembangan usaha dan pertumbuhan perusahaan.

4. *People*

Membangun SDM terampil dan terpercaya.

Values:

1. *Visionary*

a. *Creative dan Innovative*

Banyak ide atau gagasan dalam menyelesaikan masalah-masalah pekerjaan.

b. *Inspiratif*

Mengilhami orang atau pihak lain dalam mengembangkan ide-ide dan memberikan inspirasi bagi rekan kerja, kelompok, atas gagasan yang diajukan.

c. *Aspiratif*

Memiliki keinginan untuk maju dan berkembang.

2. *Integrity*

a. *Mendahului Kepentingan Perusahaan*

Menghindari adanya benturan kepentingan (*conflict of interest*) dan tidak menggunakan fasilitas perusahaan untuk urusan pribadi.

b. *Komitmen*

Tidak menyalahgunakan jabatan untuk kepentingan diri sendiri, bertanggung jawab terhadap tugas atau pekerjaannya, dan satu kata dengan perbuatan (*walk the talk*).

c. *Ketulusan Hati*

Bertindak demi untuk tujuan bersama, tanpa ada maksud tersembunyi.

d. *Kejujuran*

Bicara sesuai fakta dan tidak ada upaya memanipulasi data.

e. *Disiplin*

Memenuhi janji yang sudah dibuat, komit terhadap waktu, dan mentaati *policy* dan prosedur yang berlaku.

3. *Perseverance*

a. Gigih/Ulet/Keteguhan Hati

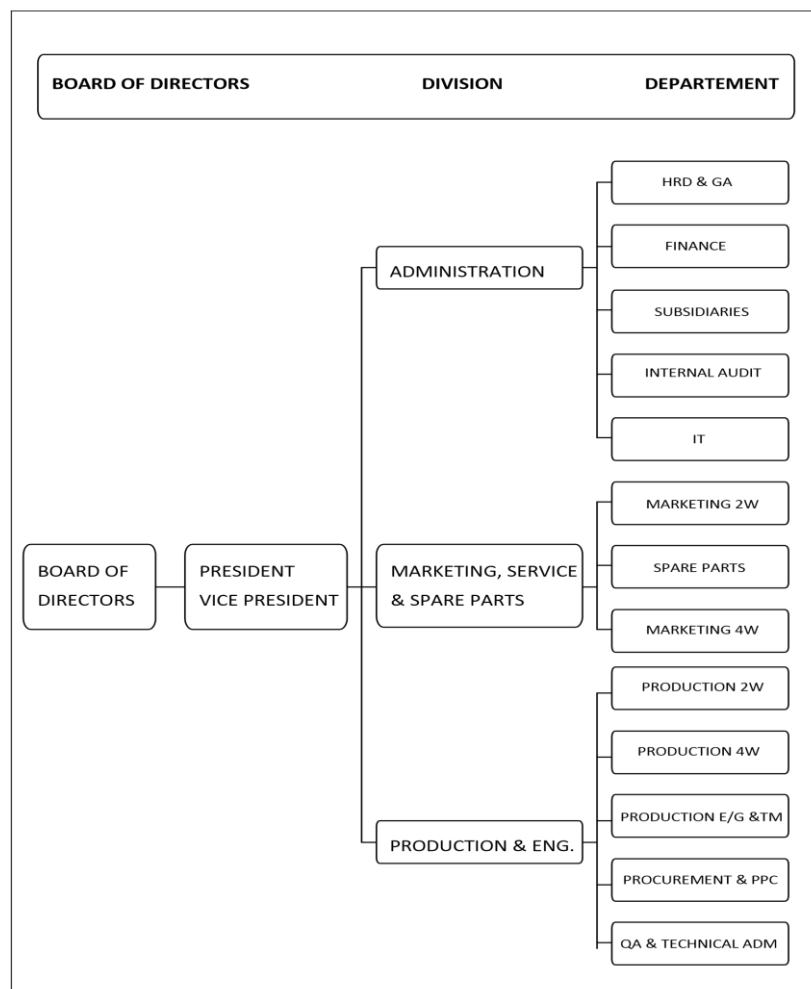
Keinginan kuat untuk berhasil, tidak mudah menyerah, dan memiliki daya juang yang kuat untuk menyelesaikan pekerjaan.

b. Bekerja Keras

Kesediaan berbuat lebih untuk kemajuan perusahaan dan kemajuan untuk mencapai hasil kerja lebih baik.

4.5 Struktur Organisasi pada PT Suzuki Indomobil Cikarang Plant

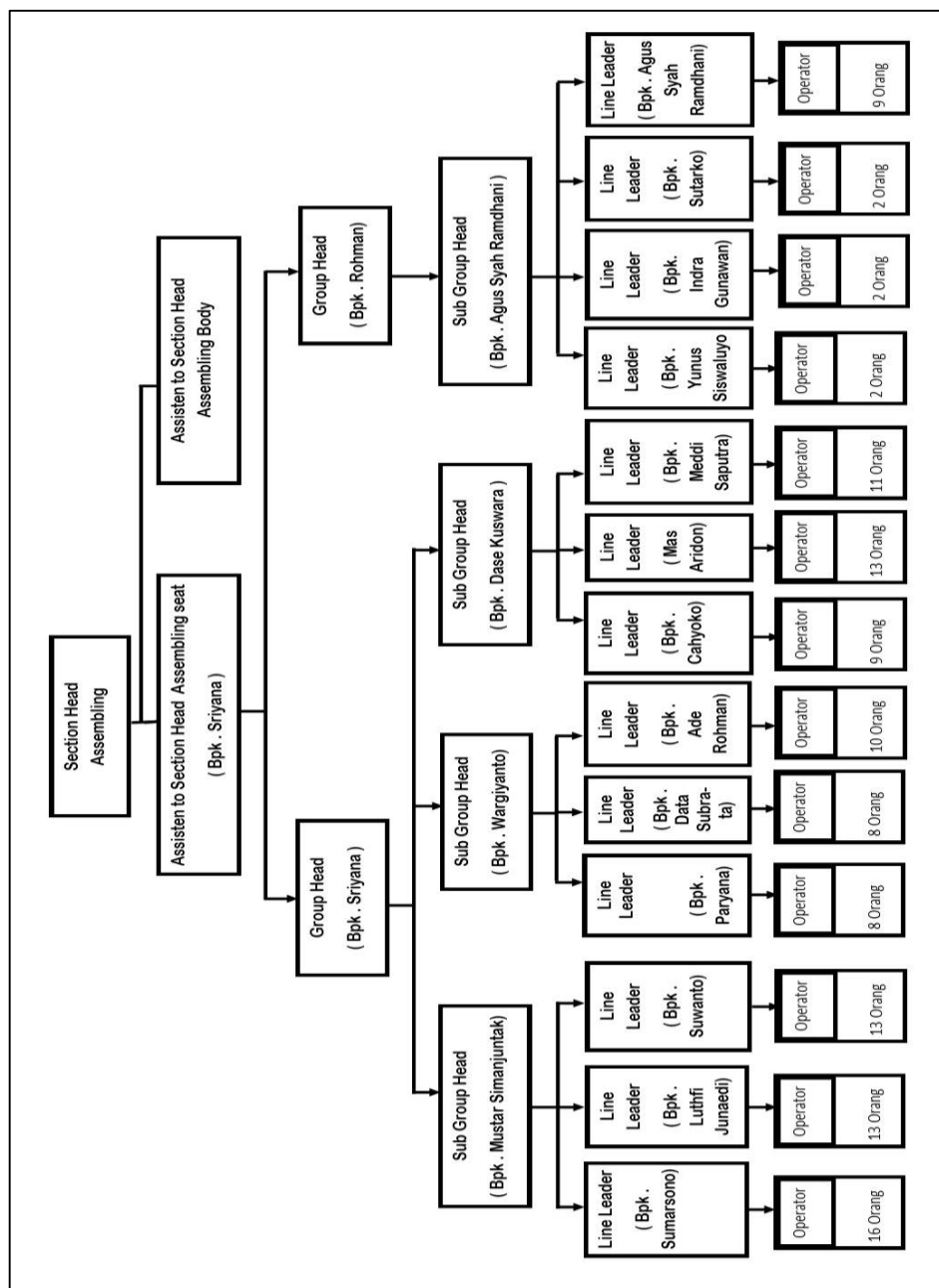
Setiap perusahaan baik perusahaan kecil atau besar memiliki struktur organisasi. Adapun Struktur Organisasi keseluruhan yang ada pada PT Suzuki Indomobil Motor Cikarang Plant adalah:



Gambar IV.2 Struktur Organisasi PT Suzuki Indomobil Motor Cikarang Plant
(Sumber: PT Suzuki Indomobil Motor Cikarang Plant, 2017)

4.6 Struktur Organisasi pada *Section Assembling seat*

Struktur organisasi yang baik akan memudahkan dalam pencapaian tujuan perusahaan. Demikian halnya pada PT Suzuki Indomobil Motor Cikarang Plant khususnya pada *section Assembling seat* dalam mencapai tujuannya diperlukan sistem manajemen organisasi yang baik. Berikut ini adalah struktur organisasi *section Assembling Seat* yang dapat dilihat pada Gambar IV.3:



Gambar IV.3 Struktur Organisasi *section Assembling seat*
(Sumber: PT Suzuki Indomobil Motor Cikarang Plant, 2017)

4.7 Produk

Seiring dengan meningkatnya ilmu teknologi serta meningkatnya kebutuhan masyarakat akan suatu alat transportasi yang sangat penting bagi masyarakat, maka dibutuhkan suatu alat transportasi yang sesuai dengan kriteria yang diinginkan konsumen. Pada dasarnya masyarakat menginginkan produk kendaraan roda empat dengan kualitas yang baik, model produk yang menarik serta harga yang terjangkau oleh masyarakat baik kelas ekonomi menengah maupun ekonomi atas, maka Suzuki Group mendirikan PT Suzuki Indomobil Motor Cikarang Plant. Dimana produk yang dihasilkan adalah Mobil Ertiga Type GA, GL, GX, dan Dreza. Selain itu PT Suzuki Indomobil Motor Cikarang Plant juga memproduksi *Seat* Model GA, GL, GX dan Dreza.

Berikut merupakan hasil produksi pada PT Suzuki Indomobil Motor Cikarang Plant:

Tabel IV.1 Produk *Seat*

No	Type Seat	Gambar Seat
1.	GA (STD)	

Tabel IV.1 Produk *Seat* (Lanjutan)

No	Type <i>Seat</i>	Gambar <i>Seat</i>
2.	GL	
3.	GX	
4.	Dreza	

(Sumber: Hasil Analisis Data, 2017)

4.8 Sistem Laporan Harian Produksi

Laporan hasil produksi merupakan hal penting bagi setiap perusahaan kecil atau besar. Hal ini berlaku pada PT Suzuki Indomobil Motor Cikarang Plant. Pada perusahaan tersebut memiliki laporan hasil produksi yang berbeda pada masing-masing *line* produksi. Seperti pada *line Assembling seat, urethane, sewing, cutting, dan welding*.

Proses pencatatan laporan harian hasil produksi, terutama pada *section Assembling seat* dicatat menggunakan kertas untuk jumlah produksi hariannya. Proses pencatatan dilakukan oleh masing-masing *leader* di setiap *line*. Kemudian laporan tersebut di serahkan ke bagian *head assembling seat* untuk di input ke *microsoft excel*.

4.9 Dokumen pada Sistem Laporan Harian Hasil Produksi

Sistem Informasi Laporan harian produksi pada PT Suzuki Indomobil Motor dilakukan secara manual yaitu pencatatan laporan harian hasil produksi ditulis pada kertas. Dokumen yang terlibat dalam sistem informasi laporan harian hasil produksi adalah:

1. Rencana Produksi

Rencana produksi pada PT Suzuki Indomobil Motor Cikarang Plant berisikan jadwal dan jumlah suatu produk harus selesai di produksi sesuai jadwal yang di tentukan. Rencana produksi ini dibuat oleh bagian PPC dan akan diberikan ke pada bagian produksi sebagai bahan acuan untuk melakukan kegiatan proses produksi.

Daily Production Planning

Seat Production

Aug-17

Note

holidays 17-Aug Hari Kemerdekaan

Issued Date : 18 July 2017

PREPARED	CHECKED	APPROVED
Agus SR	Rohman	Srivana T.Oishi

		Jul																																																																																																																																																																																																																																																																																																																																																																																																																																								
--	--	-----	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Gambar IV.4 Dokumen Rencana Produksi
(Sumber: PT Suzuki Indomobil Motor Cikarang Plant, 2018)

Keterangan:

- Daily Production Planning* : Rencana Harian Produksi
- Note : Catatan
- Holidays* : Tanggal Merah/hari besar
- Prepared* : Dibuat oleh *Leader Assy*
- Checked* : Di cek dan di validasi oleh *Head Assy*
- Approved* : Di setuju oleh *Assisten to section assembling seat* dan *section head assembling seat*
- Model : Model *Seat*
- Type Product : Tipe *Seat*

2. Data Harian hasil Produksi

Data harian hasil produksi merupakan dokumen yang digunakan bagian produksi untuk mencatat hasil produksi harian. Data harian hasil produksi berisikan jumlah produk yang dihasilkan setiap hari. Setiap hari data harian

hasil produksi dicatat oleh operator dan diserahkan kepada kepala bagian produksi.

Daily Report Assy Seat		
Hari/Tanggal :	:	:
Waktu Normal (menit) :	:	:
(renda-kamis 430 menit ; joni'at 430 menit)	:	:
Jml MP : _____	Tidak Hadir : _____	Nama : _____

PREPARED	APPROVED
Leader	Head Assy

PLAN PRODUCTION

	P12	P31	P08	DRV
Total				

FR

	QTY (pcs)	
	PLAN	PROD
P12		
P31		
P08		
DRV		
SUBTOTAL		

ZND

	QTY (pcs)	
	PLAN	PROD
P12		
P31		
P08		
DRV		
SUBTOTAL		

3RD

	QTY (pcs)	
	PLAN	PROD
P12		
P31		
P08		
DRV		
SUBTOTAL		

Deffect

	FR	ZND	3RD	QTY
TOTAL				

QTY Reject

	FR	ZND	3RD	QTY
TOTAL				

NOTES :

Gambar IV.5 Data Harian Hasil Produksi
(Sumber: PT Suzuki Indomobil Motor Cikarang Plant, 2018)

Keterangan:

Hari/Tanggal	: Waktu Produksi
Waktu Normal	: Waktu Proses Produksi dalam 1 hari
Jumlah MP	: Jumlah <i>Manpower</i> yang hadir
Tidak Hadir	: Jumlah <i>Manpower</i> tidak hadir
Nama	: Nama operator yang mengisi <i>daily report</i>
<i>Plan production</i>	: Jumlah rencana Produksi
<i>Deffect</i>	: Jumlah seat yang <i>deffect</i>

Reject : Jumlah seat *reject*
Fr : Kursi baris pertama
2ND : Kursi baris kedua
3RD : Kursi baris terakhir

3. Laporan Hasil Produksi seksi *Assembling seat*

Data harian hasil produksi merupakan dokumen yang digunakan bagian produksi untuk mencatat hasil produksi harian. Data harian hasil produksi berisikan jumlah produk yang dihasilkan setiap hari. Setiap hari data harian hasil produksi dicatat oleh operator dan diserahkan kepada kepala bagian produksi.

PT Suzuki Indomobil Motor Cikarang Plant

REPORT ASSY SEAT

ISSUE DATE: 31 Agst 2017 NOTE: _____ PROD PERIOD: Agst 2017
 NAME: Rohman HOLIDAY: 17 - Kemerdekaan

Process: *Assembling seat*

Day	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Total
Plan Prod			202	208	210	208	0	0	205	209	210	211	204	0	0	206	197	209	203	195	0	0	204	192	207	209	206	0	0	197	190	208	205	4493
Actual Prod			190	208	194	208	0	0	205	189	205	193	190	0	0	185	197	209	198	195	0	0	190	185	207	209	195	0	0	182	190	190	180	4304
±			-12	0	-16	0	0	0	0	-20	-5	-18	-14	0	0	-21	0	0	-5	0	0	0	-14	-7	0	0	-11	0	0	-15	0	-18	-25	-201
OT			1	0	1	0	0	0	0	1	1	1	0	0	0	1	0	0	1	0	0	0	1	1	0	0	1	0	0	1	0	1	2	14

APPROVED BY: Head Assy Seat

Gambar IV.6 Laporan Hasil Produksi
(Sumber: PT Suzuki Indomobil Motor Cikarang Plant, 2018)

Keterangan:

Report Assy Seat : Laporan Produksi *Assembling Seat*
Note : Catatan
Issue Date : Tanggal Pembuatan
Name : Nama Pembuat Laporan
Prod Period : Periode produksi (bulan)

<i>Holiday</i>	: Hari libur nasional
<i>Process</i>	: <i>Line Proses</i>
<i>Approved</i>	: Di setuju oleh <i>Assisten to section assembling seat</i> dan <i>section head assembling seat</i>
<i>Day</i>	: Tanggal
<i>Plan Product</i>	: Rencana Produksi
<i>Actual Prod</i>	: Hasil aktual Produksi
<i>OT</i>	: <i>Overtime</i>

4.10 Prosedur Pencatatan Laporan Harian Hasil Produksi

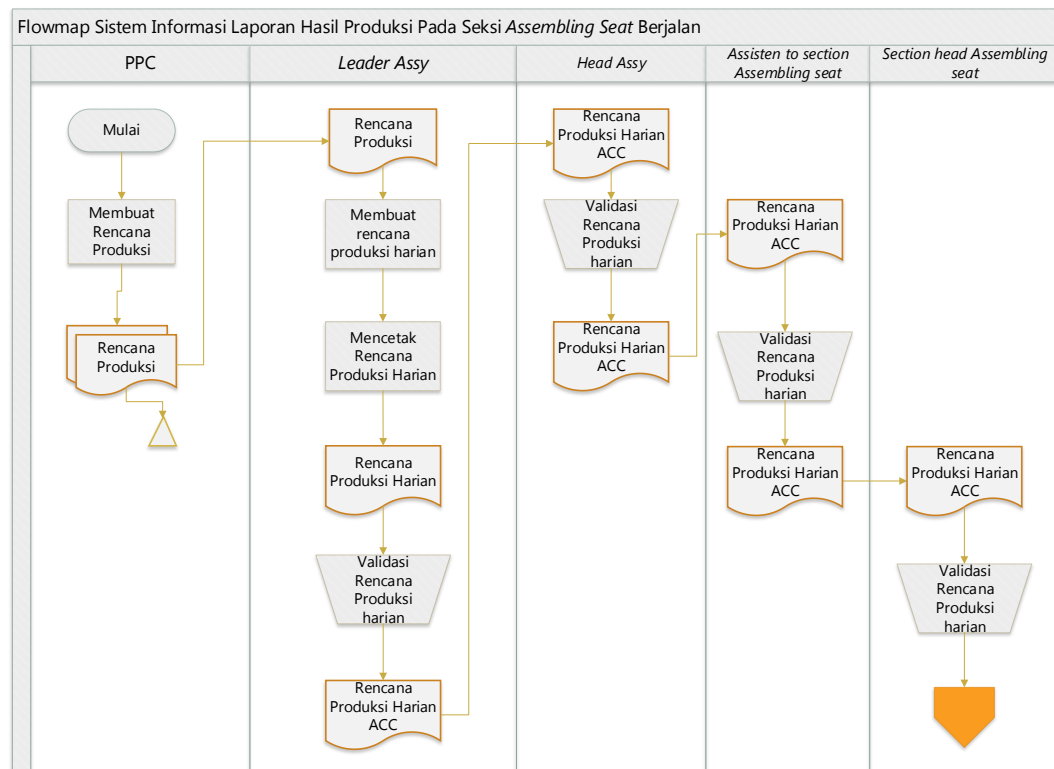
Prosedur sistem informasi produksi pada PT Suzuki Indomobil Motor Cikarang Plant terdiri dari beberapa tahapan yaitu:

1. Bagian PPC Membuat Rencana Produksi sebanyak 2 rangkap, 1 rangkap untuk diberikan ke *Leader Assembling Seat* dan 1 rangkap disimpan menjadi arsip PPC.
2. Setelah menerima rencana produksi, *Leader Assembling seat* kemudian membuat rencana jumlah produksi harian dengan *Microsoft Excel*.
3. Kemudian rencana produksi tersebut di cetak dan di validasi oleh *Leader Assembling seat*.
4. Rencana yang sudah dibuat kemudian di validasi oleh *head assy, Assisten to section assembling seat, Section head assembling seat* sebagai acuan jumlah produksi.
5. Kemudian operator produksi melakukan persiapan sebelum melakukan produksi.
6. Proses produksi dilakukan sesuai dengan jumlah produksi yang sudah direncanakan.
7. Setelah proses produksi selesai, operator mengisi *form* data harian hasil produksi pada dokumen *daily report*.
8. Data harian tersebut kemudian diberikan kepada *Leader Assembling Seat* untuk kemudian di cek dan dilakukan validasi apakah hasil rencana tersebut sesuai

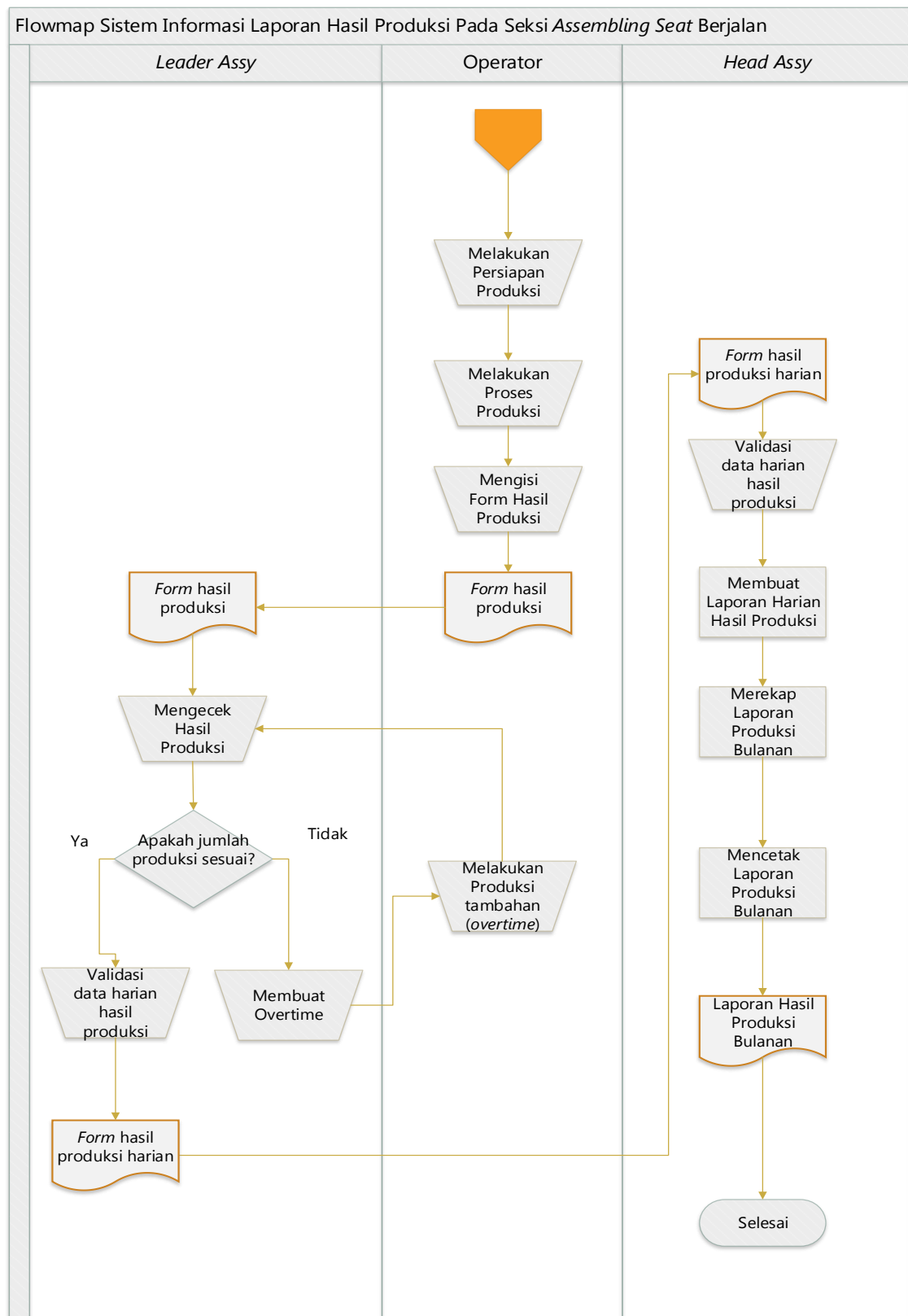
dengan rencana atau tidak. Jika tidak maka ketika diakhir proses produksi membuat rencana *overtime*.

9. Operator kemudian melakukan produksi tambahan pada waktu *overtime*.
10. Jika data sudah divalidasi oleh *leader* kemudian di berikan kepada *Head Assy* untuk di validasi dan diinput ke *microsoft excel*.
11. *Head Assy* memvalidasi data yang sudah sesuai, dilanjutkan dengan membuat rekap laporan harian hasil produksi tersebut untuk nantinya menjadi laporan bulanan.
12. Setelah membuat laporan produksi, *Head Assembling seat* menecetak laporan hasil produksi.
13. Kemudian laporan tersebut disimpan sebagai laporan produksi *seat* dan arsip yang ada di PT Suzuki Indomobil Motor Cikarang Plant.

Berikut merupakan aliran data sistem informasi laporan harian produksi pada PT Suzuki Indomobil Motor Cikarang Plant yang berjalan:



Gambar IV.7 Aliran Data Sistem Informasi laporan hasil Produksi
(Sumber: Hasil analisis, 2018)



Gambar IV.7 Aliran Data Sistem Informasi laporan hasil Produksi (Lanjutan)
(Sumber: Hasil analisis, 2018)

4.11 Penambahan Waktu Kerja/*Overtime* Pada PT Suzuki Indomobil Motor Cikarang Plant

Berdasarkan hasil pengamatan terhadap sistem yang berjalan untuk mengetahui bagaimana cara menentukan jumlah *overtime* yang harus dilakukan oleh operator produksi, untuk memenuhi jumlah hasil produksi yang sesuai dengan perencanaan adalah sebagai berikut:

Tabel IV.2 Tabel Penentuan *Overtime*

No	Jumlah barang yang harus di produksi	Keterangan
1.	$\leq 22 \text{ seat}$	OT 1 Jam
2.	23 – 45 <i>seat</i>	OT 2 Jam
3.	46 – 68 <i>seat</i>	OT 3 Jam

Sumber : Hasil pengamatan dengan operator produksi (2017)

Dimana penentuan *overtime* tersebut berdasarkan hasil perhitungan *tactime* sebagai berikut:

Tabel IV.3 Tabel perhitungan *tactime*

No	Keterangan	Jumlah
1	Total Jam kerja	8 jam
2	Total waktu proses kerja (<i>Standar time</i> : $1255 / 60 = 20,91$)	1255 s
3	Total <i>manpower</i> dalam seksi <i>assembling seat</i>	9 orang
4	<i>Tactime</i> dengan Efisiensi 94% (<i>standar time</i> / mp/ 94%)	2,472419
5	Qty per hari ((60/EFF94%) x (jam kerja))	194,17 ~ 194
6	PPH (<i>part per hour</i>) (60/EFF)	24,27 ~ 24

(Sumber : Hasil pengamatan dengan *leader* produksi 2017)

4.12 Permodelan Sistem Pencatatan Laporan Harian Hasil Produksi dengan UML

Analisis terhadap sistem yang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan kelemahan sistem serta solusi atas

masalah tersebut sehingga dapat dijadikan rancang bangun sistem yang baru. Kegiatan analisis sistem yang berjalan dengan menggunakan analisis sistem yang berorientasi pada objek-objek yang sangat diperlukan oleh sistem yang akan dirancang. Dengan maksud untuk menitik beratkan pada alur proses dari sistem.

Kemudian dari hasil analisis ini direpresentasikan dengan UML melalui diagram *use case*. Pertimbangan dari diagram ini dapat mewakili secara keseluruhan sistem yang berjalan yang dapat dimengerti oleh *user*. Penjelasan *use case diagram* sistem informasi produksi yang sedang berjalan pada PT Suzuki Indomobil Motor Cikarang Plant adalah sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* berdasarkan proses sistem informasi produksi pada PT Suzuki Indomobil Motor Cikarang Plant dapat dilihat pada table berikut:

Tabel IV.4. Definisi Aktor Sistem Informasi Produksi

No.	Aktor	Definisi
1.	PPC	Membuat Rencana Produksi dan memberikan rencana produksi tersebut kepada <i>leader assembling seat</i> .
2.	<i>Leader Assembling Seat</i>	Menerima rencana produksi, kemudian membuat rencana produksi harian, membuat rencana overtime
3.	<i>Assisten To section assembling seat</i>	Memvalidasi rencana produksi yang sudah dibuat oleh <i>Leader Assembling seat</i>
4.	<i>Section Assembling Seat</i>	Memvalidasi rencana produksi yang sudah dibuat oleh <i>Leader Assembling seat</i>
5.	<i>Operator Assembling Seat</i>	Melakukan persiapan sebelum melakukan produksi, Melakukan Proses Produksi, dan mengisi <i>form</i> hasil produksi
6.	<i>Head Assembling seat</i>	Menerima laporan harian hasil produksi, merekap laporan menjadi laporan bulanan

(Sumber: Hasil Analisis, 2018)

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* berdasarkan proses sistem informasi produksi pada PT Suzuki Indomobil Motor Cikarang Plant dapat dilihat pada tabel IV.5 berikut:

Tabel IV.5. Definisi *Use Case* Sistem Informasi Laporan Hasil Produksi

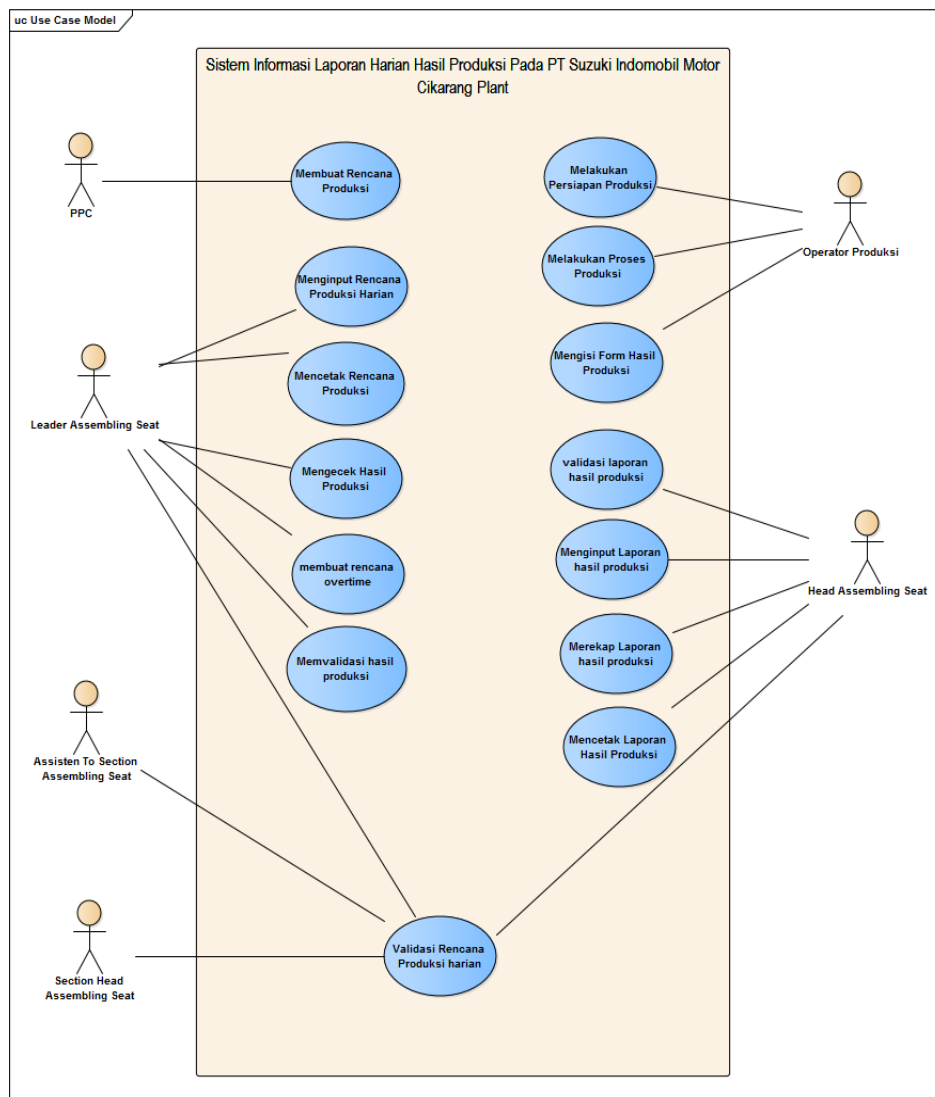
No.	<i>Use Case</i>	Definisi
1.	Membuat rencana produksi	Proses membuat rencana produksi untuk menghasilkan suatu produk oleh bagian PPC
2.	Menginput rencana produksi harian	Proses menginput rencana produksi harian oleh <i>Leader Assembling seat</i> .
3.	Mencetak Rencana Produksi	Proses mencetak rencana produksi untuk acuan melakukan proses produksi
4.	Memvalidasi Rencana Produksi	Proses memvalidasi rencana produksi oleh <i>Leader Assy, head assy, Assisten To Section Assembling seat, Section head assembling seat</i>
5.	Mempersiapkan proses Produksi	Proses untuk mempersiapkan proses produksi
6.	Melakukan proses produksi	Proses untuk menghasilkan suatu produk yang dilakukan oleh bagian produksi.
7.	Mengisi jumlah hasil produksi harian	Proses pencatatan hasil produksi perhari pada <i>Daily report</i> oleh <i>Operator Assembling seat</i> .
8.	Mengecek Hasil produksi	Proses untuk mengecek rencana produksi dan aktual hasil produksi
9.	Membuat rencana <i>overtime</i>	Proses untuk membuat rencana <i>overtime</i>
10.	Memvalidasi Laporan Hasil Produksi	Proses memvalidasi hasil produksi
11.	Menginput Hasil Produksi	Proses Menginput hasil produksi yang sudah di validasi ke <i>microsoft excel</i>

Tabel IV.5. Definisi *Use Case* Sistem Informasi Produksi (lanjutan)

No.	Use Case	Definisi
12.	Membuat laporan harian hasil produksi	Proses membuat laporan harian hasil produksi.
13.	Membuat rekap data bulanan hasil produksi	Proses membuat laporan produksi bulanan oleh <i>Head Assembling seat</i> .

(Sumber: Hasil Analisis, 2018)

Use Case Diagram sistem informasi produksi yang sedang berjalan pada PT Suzuki Indomobil Motor Cikarang Plant terdiri dari enam aktor sebagaimana digambarkan pada gambar IV.8 berikut ini:

Gambar IV.8 *Use Case Diagram* Sistem Informasi Produksi
(Sumber: Hasil analisis, 2018)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan *Functional* Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, maka dibutuhkan perancangan aplikasi sistem informasi laporan harian hasil produksi *seat* pada seksi *Assembling Seat*. Berikut adalah daftar kebutuhan fungsional sistem untuk aplikasi sistem informasi laporan harian hasil produksi *seat* pada seksi *Assembling Seat*.

Tabel V.1 Kebutuhan *Functional* Sistem

Masalah	Kebutuhan User	Kebutuhan Fungsional Sistem	Uraian
Belum ada sistem yang melibatkan hak akses	Pembatasan hak user dalam mengakses program	Login Pengguna	Proses untuk masuk ke sistem dan yang dapat masuk ke sistem, yaitu: - <i>Leader Assy</i> - <i>Head Assy</i>
Belum ada media pengolahan data yang terhubung ke <i>database</i>	Pengolahan data yang lebih mudah	- Pengelolaan Data Master	Data Master Diantaranya Data <i>user</i> dan data Produk. Proses yang dapat dilakukan pada pengelolaan data tersebut adalah: - Dapat menambah dan mengubah data tersebut - Dapat mencari data tersebut. - Dapat menghapus data.

Tabel V.1 Kebutuhan Fungsional Sistem (lanjutan)

Masalah	Kebutuhan User	Kebutuhan Fungsional Sistem	Uraian
Belum ada media pengolahan data hasil produksi yang terintegrasi dengan setiap data yang ada	Mengolah data hasil produksi secara tepat dan cepat	Pengelolaan Data Transaksi Hasil Produksi	Data transaksi diantaranya adalah data <i>input</i> hasil produksi, data bulanan hasil produksi. Proses yang dapat dilakukan pada pengelolaan data tersebut adalah: - Dapat menambah dan mengubah data tersebut. - Dapat mencari dan menghapus data tersebut
Belum ada perencanaan penambahan waktu kerja/ <i>overtime</i> jika ada kekurangan dalam jumlah hasil produksi.	Perencanaan penambahan waktu kerja/ <i>overtime</i> jika ada kekurangan dalam jumlah hasil produksi.	Perencanaan <i>Overtime</i>	Proses yang dapat dilakukan pada pengelolaan Data Hasil Produksi yaitu: - Dapat mengetahui Pencapaian jumlah produksi yang di rencanakan dengan hasil produksi yang aktual. - Dapat mengetahui keputusan untuk memenuhi hasil produksi. - Dapat mencari dan menghapus data tersebut.

Tabel V.1 Kebutuhan Fungsional Sistem (lanjutan)

Masalah	Kebutuhan User	Kebutuhan Fungsional Sistem	Uraian
Antar bagian belum terintegrasi	Memvalidasi data harian hasil produksi tanpa harus datang langsung ke bagian produksi	Validasi data harian hasil produksi	Proses yang dapat dilakukan pada validasi data harian hasil produksi yaitu: - Dapat memvalidasi data harian hasil produksi - Dapat mencari data harian hasil produksi

Sumber: Pengolahan Data (2018)

5.1.1 Kebutuhan Non-Functional

1. Usability

- Memudahkan dalam mengelola data laporan hasil produksi.
- Memudahkan dalam melihat dan mencetak laporan hasil produksi.
- Memudahkan dalam mengetahui jumlah barang yang produksi.
- Memudahkan dalam mengetahui rencana *overtime*

2. Portability

- Aplikasi laporan hasil produksi digunakan hanya untuk *leader* dan *Head Assembling seat*.
- Komputer harus memiliki *database server* MySQL.
- Komputer harus menginstall aplikasi *Netbeans* untuk menggunakan aplikasi laporan hasil produksi.

3. Reliability

- Sistem aplikasi laporan hasil produksi memiliki hak akses (*login*) dengan kode karyawan dan *password* yang telah ditentukan.

4. Supportability

- Menggunakan sistem operasi *Windows*.

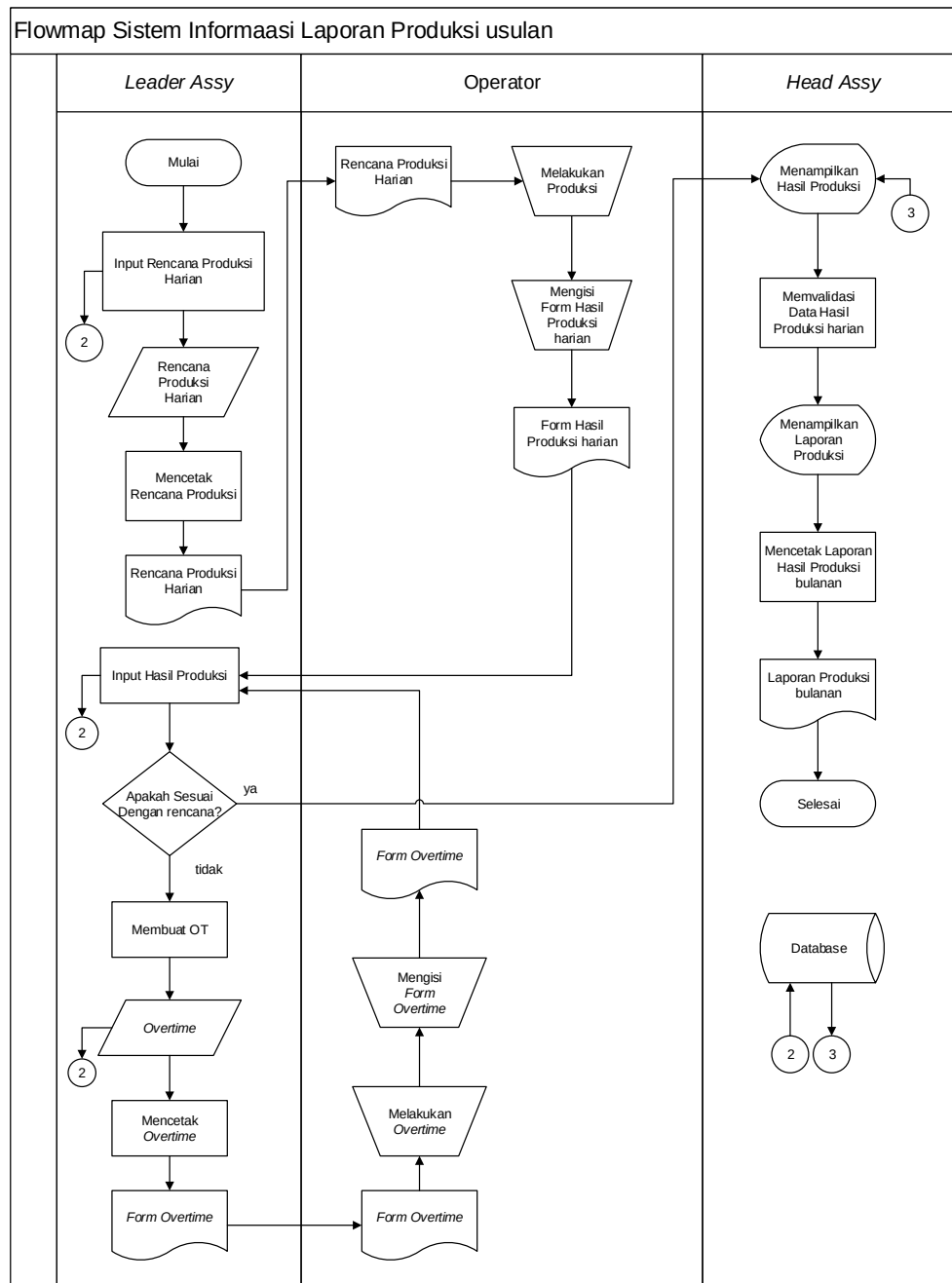
- Spesifikasi komputer minimal Core i3.
- Minimal RAM 4GB.
- Minimal *hardisk* 500 GB.
- Menggunakan *mouse* dan *keyboard*.
- Menggunakan *printer* untuk mencetak laporan data stok barang jadi.

5.2 Perancangan *Flowmap* Sistem Usulan

Urutan prosedur dan *Flowmap* sistem informasi laporan harian hasil produksi yang diusulkan sebagai berikut:

1. *Leader Assy* menginput rencana produksi harian kemudian rencana produksi yang sudah dibuat oleh *Leader Assy* lalu dicetak dan diberikan kepada operator untuk melakukan proses produksi.
2. Setelah operator melakukan proses produksi kemudian operator mengisi *form* hasil produksi dan *form* hasil produksi diberikan ke *Leader Assy* untuk diinput ke sistem.
3. Jika hasil produksi sesuai dengan rencana maka *Head Assy* dapat memvalidasi kemudian *Head Assy* dapat melihat grafik hasil produksi lalu mencetak laporan hasil produksi yang telah divalidasi dan jika hasilnya tidak sesuai dengan perencanaan maka *Leader Assy* membuat rencana *overtime*.
4. Rencana *overtime* yang sudah dibuat kemudian di cetak dan diberikan kepada operator untuk melakukan produksi kembali setelah jam kerja berakhir untuk memenuhi jumlah produksi lalu operator mengisi *form overtime* dan diberikan ke *Leader Assy* untuk di validasi *head assy*. Kemudian laporan yang sudah di validasi di cetak sebagai arsip laporan hasil produksi *assembling seat*.

Flowmap sistem informasi laporan harian produksi yang diusulkan sebagai berikut:



Gambar V.1 Flowmap Sistem Informasi Laporan Harian Hasil Produksi (usulan)
Sumber: Hasil Analisis (2018)

5.3 Analisis Sistem Informasi Usulan

Analisis proses sistem informasi laporan harian hasil produksi pada *Assembling seat* menggunakan *tools* permodelan sistem UML (*Unified Modeling Language*) yang

meliputi pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*, *deployment diagram* dan pembuatan kamus data. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun.

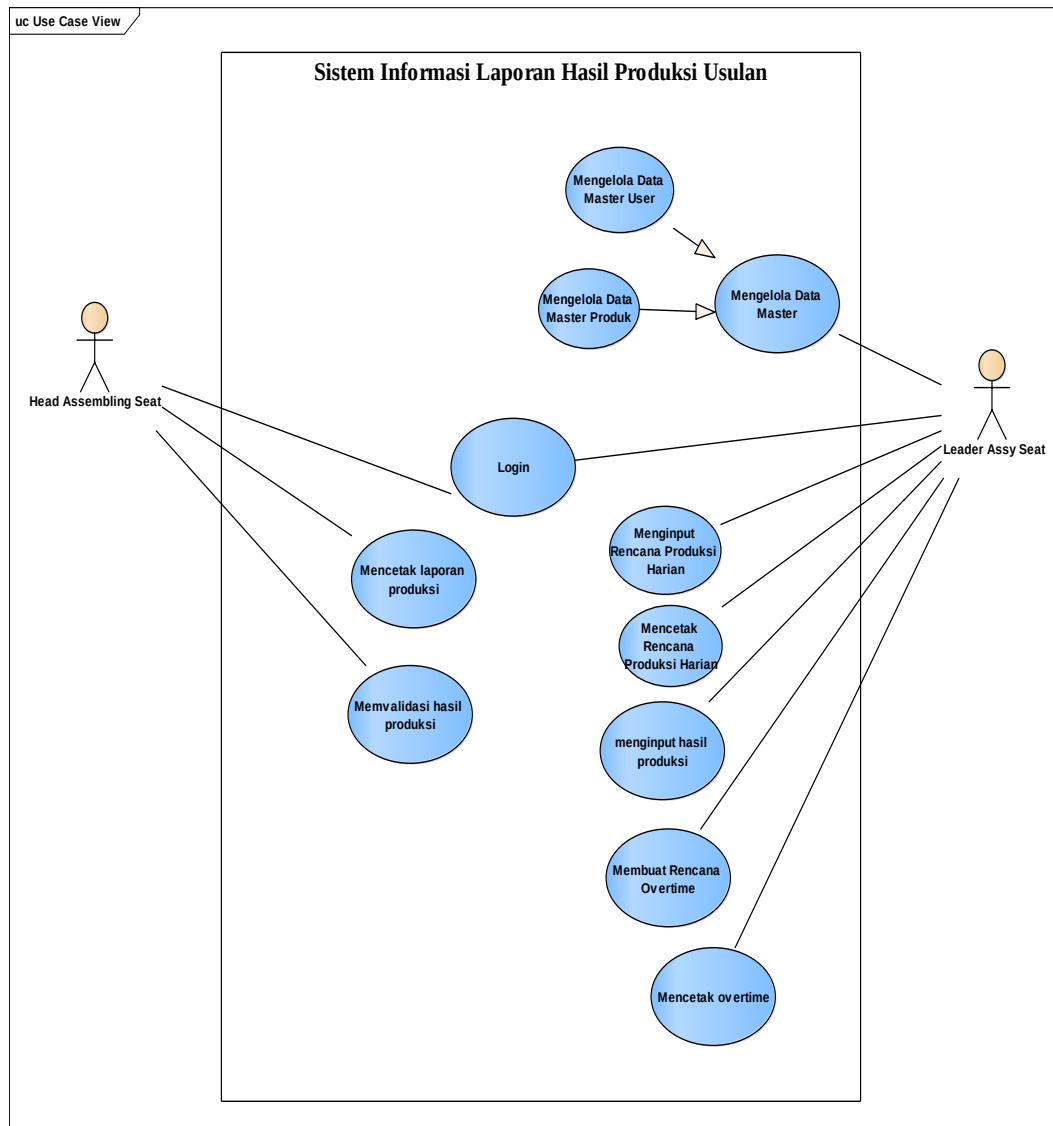
Overtime menurut Wikipedia (2010) adalah jumlah waktu seseorang bekerja di luar jam kerja normal. Kerja lembur atau *Overtime* adalah pekerjaan yang dilakukan oleh karyawan, atas dasar perintah atasan, yang melebihi jam kerja biasa pada hari-hari kerja, atau pekerjaan yang dilakukan pada hari istirahat mingguan karyawan atau hari libur resmi. Adapun Cara perhitungan *overtime* pada PT Suzuki Indomobil Motor Cikarang Plant adalah sebagai berikut:

$$\text{Overtime} = \text{Jumlah rencana} - \text{jumlah hasil produksi} / \text{PPH}$$

5.3.1 Use Case Diagram

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

Berdasarkan analisis yang telah dilakukan, maka *use case diagram* sistem informasi laporan harian produksi usulan dapat dilihat pada Gambar V.2 sebagai berikut ini:



Gambar V.2 Use Case Diagram Sistem Informasi Laporan Harian Produksi Seat Usulan

Sumber: Hasil Analisis (2018)

Penjelasan *use case diagram* sistem informasi laporan harian hasil produksi *seat* usulan dijelaskan sebagai berikut ini:

1. Definisi Aktor

Pendefinisian aktor pada *diagram* sistem informasi laporan harian hasil produksi *seat* usulan dijelaskan pada Tabel V.2

Tabel V.2 Definisi Aktor *Use Case*

No	Aktor	Deskripsi
1	<i>leader Assembling seat</i>	<i>Leader Assembling seat</i> merupakan aktor yang menerima laporan harian hasil produksi, dan membuat rencana <i>overtime</i> .
2	<i>Head Assembling seat</i>	Kepala bagian seksi <i>Assembling seat</i> yang dapat memvalidasi dan melihat laporan hasil produksi bulanan dan laporan Pencapaian hasil produksi.

Sumber: Hasil Analisis (2018)

2. Definisi *Use Case*

Berdasarkan Gambar V.2 *use case diagram* usulan di atas maka dibuatlah definisi *use case* yang merupakan gambaran detail dari *use case* yang berisi nama *use case*, deskripsi *use case*, aktor, dan *normal flow events*. Pendefinisian *use case* pada *use case diagram* proses produksi *seat* pada seksi *assembling seat* di PT Suzuki Indomobil Motor Cikarang Plant adalah:

Tabel V.3 Definisi *Use Case Diagram* Usulan

No.	<i>Use Case</i>	Deskripsi
1	Mengelola data master	Proses mengelola data master
2	<i>Input Rencana Produksi</i>	Proses <i>menginput</i> rencana produksi harian yang telah dibuat oleh <i>Leader Assy Seat</i>
3	Mencetak Rencana Harian Produksi	Proses mencetak rencana produksi harian yang kemudian akan diberikan kepada operator
4	<i>Input Hasil Produksi</i>	Proses <i>penginputan</i> hasil produksi yang dilakukan oleh <i>Leader Assembling Seat</i>
5	Membuat Rencana <i>Overtime</i>	Proses untuk membuat rencana <i>overtime</i>

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

No.	<i>Use Case</i>	Deskripsi
6	Mencetak Rencana <i>Overtime</i>	Proses mencetak <i>overtime</i>
7	Memvalidasi Hasil Produksi	Proses menampilkan <i>Form Validasi</i> untuk di Validasi dilakukan oleh <i>Head Assy Seat</i>
8	Mencetak Laporan Produksi	Proses mencetak laporan produksi.
9	<i>Login</i>	Proses untuk melakukan <i>login</i> atau masuk kedalam aplikasi

(Sumber: Hasil Analisa Data, 2018)

3. Skenario *Use Case*

Skenario jalannya masing-masing *use case* pada sistem informasi pelaporan produksi usulan dapat dilihat pada poin berikut:

Tabel V.4 di bawah ini berisi deskripsi untuk *use case login*.

Tabel V.4 Definisi *Use Case Login*

Nama <i>Use case</i>	<i>Login</i>
Aktor	<i>Leader assembling seat , Head Assembling seat,</i>
Deskripsi <i>Use case</i>	<i>Use case</i> ini menggambarkan proses pengguna masuk ke sistem informasi laporan hasil produksi <i>seat</i> .
<i>Normal Flow of Events</i>	1. <i>User</i> membuka aplikasi 2. Sistem menampilkan menu <i>login</i>.
<i>Normal Flow of Events</i>	3. <i>User</i> mengisi <i>username</i> dan <i>password</i> di menu <i>login</i>. 4. Sistem memvalidasi data yang di-input. Apabila data tidak <i>valid</i>, maka muncul pesan gagal <i>login</i> dan <i>user</i> tersebut tidak dapat menggunakan sistem. Apabila data <i>valid</i>, maka sistem akan menampilkan menu utama sesuai hak akses.

Sumber: Hasil Analisis (2018)

Tabel V.5 di bawah ini berisi deskripsi *use case* mengelola data *master*

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

Nama <i>Use case</i>	Mengelola data <i>master</i>
Aktor	<i>Head Assembling seat</i>
Deskripsi <i>Use case</i>	<i>Use case</i> ini menggambarkan pengelolaan data master perusahaan yaitu data master produk dan data master <i>user</i>
<i>Normal Flow of Events</i>	1. <i>Head assembling seat</i> telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. <i>Head assembling seat</i> memilih menu Data Master. 4. <i>Head assembling seat</i> memilih Submenu Data Master 5. <i>Head assembling seat</i> akan mengubah data dan klik simpan. 6. Jika ingin menghapus maka <i>leader</i> memilih data lalu klik pada tabel kemudian klik hapus, sistem akan menghapus data. 7. Sistem akan memperbaharui basis data. 8. Sistem akan menampilkan halaman data master. 9. Sistem menampilkan halaman data master yang dipilih.<i>Sub head assembling seat</i> dapat memilih aksi cari, simpan, edit dan hapus.

Sumber: Hasil Analisis (2018)

Tabel V.6 di bawah ini berisi deskripsi untuk *use case* memasukkan data Rencana Produksi

Tabel V.6 Definisi *Use Case* Menginput Data Rencana Produksi

Nama <i>Use case</i>	Menginput data Rencana Produksi
Aktor	<i>Leader Assembling Seat</i>
Deskripsi <i>Use case</i>	Proses memasukkan data Rencana Produksi dilakukan apabila <i>Leader Assembling Seat</i> telah menerima Rencana Produksi dari PPC.
<i>Normal Flow of Events</i>	1. Sistem menampilkan menu utama 2. <i>Leader Assembling seat</i> memilih Menu Data Transaksi.

Tabel V.6 Definisi *Use Case* Menginput Data Rencana Produksi (lanjutan)

	3. <i>Leader Assembling seat</i> telah sukses melakukan login 4. Sistem menampilkan menu utama 5. <i>Leader Assembling seat</i> memilih Menu Data Transaksi. 6. <i>Leader Assembling seat</i> memilih Submenu Data Rencana Produksi 7. Sistem menampilkan halaman submenu Data Rencana Produksi. 8. Sistem menampilkan Form Rencana Produksi. 9. Admin <i>Sub Head Assembling seat</i> memasukkan Data Rencana Produksi, klik simpan. 10. <i>Leader</i> mengklik detail rencana. 11. Sistem menampilkan Form Detail Rencana produksi harian <i>Seat</i> 12. <i>Leader Assembling seat</i> dapat memilih aksi Simpan, Edit, Hapus. Sistem memperbarui data
--	---

Sumber: Hasil Analisis (2018)

Tabel V.7 di bawah ini berisi deskripsi untuk *use case* mencetak rencana produksi.Tabel V.7 Definisi *Use Case* mencetak rencana produksi

Nama <i>Use case</i>	Mencetak Rencana Produksi
Aktor	<i>Leader Assembling Seat</i>
Deskripsi <i>Use case</i>	Proses mencetak rencana produksi
<i>Normal Flow of Events</i>	1. <i>Leader Assembling Seat</i> telah sukses melakukan <i>login</i> . 2. Sistem menampilkan menu utama. 3. <i>Leader Assembling Seat</i> memilih menu Data Transaksi. 4. <i>Leader Assembling Seat</i> memilih Submenu transaksi. 5. Sistem menampilkan submenu data transaksi, lalu <i>Leader</i> memilih rencana produksi.

Tabel V.7 Definisi *Use Case* mencetak rencana produksi (lanjutan)

	6. Sistem menampilkan <i>form</i> tambah data rencana produksi. 7. <i>Leader Assembling seat</i> memilih data rencana produksi, klik cetak. 8. Sistem menampilkan rencana yang akan di cetak 9. Sistem memperbaharui basis data. 10. Sistem menampilkan Rencana <i>Overtime</i> .
--	---

Sumber: Hasil Analisis (2018)

Tabel V.8 di bawah ini berisi deskripsi untuk *use case* menginput data aktual hasil produksi

Tabel V.8 Definisi *Use Case* Menginput Data Aktual Hasil Produksi

Nama <i>Use case</i>	Menginput Data Aktual Hasil Produksi
Aktor	<i>Leader Assembling Seat</i> Produksi
Deskripsi <i>Use case</i>	Proses memasukkan data aktual hasil produksi dilakukan oleh <i>Leader Assembling Seat</i>
<i>Normal Flow of Events</i>	1. <i>Leader Assembling seat</i> telah sukses melakukan <i>login</i> . 2. Sistem menampilkan menu utama. 3. <i>Leader Assembling seat</i> memilih menu data transaksi. 4. <i>Leader Assembling seat</i> memilih Submenu Data hasil produksi 5. Sistem menampilkan <i>form</i> data <i>input</i> hasil produksi. 6. <i>Leader Assembling seat</i> memasukkan data aktual hasil produksi. 7. <i>Leader Assembling seat</i> dapat memilih aksi Simpan, Edit, Hapus. 8. Jika memilih Simpan, sistem akan menyimpan ke <i>database</i> . 9. Jika ingin mengubah maka klik data yang akan diedit, sistem akan menampilkan data yang akan diubah langsung di <i>form</i> . Jika memilih ubah maka sistem akan menyimpan data yang sudah diubah.

Tabel V.8 Definisi *Use Case* Menginput Data Aktual Hasil Produksi (lanjutan)

	10. Jika memilih Hapus, sistem akan menghapus data. 11. Sistem memperbaharui basis data. 12. Sistem menampilkan halaman Data aktual hasil produksi.
--	---

Sumber: Hasil Analisis (2018)

Tabel V.9 di bawah ini berisi deskripsi untuk *use case* membuat rencana penambahan jam kerja/*overtime*.

Tabel V.9 Definisi *Use Case* membuat rencana *overtime*

Nama <i>Use case</i>	Membuat Rencana <i>Overtime</i>
Aktor	<i>Leader Assembling Seat</i>
Deskripsi <i>Use case</i>	Proses Perhitungan untuk penambahan jam kerja operator produksi
<i>Normal Flow of Events</i>	1. <i>Leader Assembling Seat</i> telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. <i>Leader Assembling Seat</i> memilih menu Data Transaksi. 4. <i>Leader Assembling Seat</i> memilih Submenu transaksi. 5. Sistem menampilkan submenu data transaksi, lalu <i>Leader</i> memilih rencana <i>overtime</i> . 6. Sistem menampilkan <i>form</i> tambah data rencana <i>overtime</i> 7. <i>Leader Assembling seat</i> memasukkan data rencana <i>overtime</i> , klik simpan. 8. <i>Leader Assembling seat</i> dapat memilih aksi Simpan, Edit, Hapus. 9. Jika <i>Leader Assembling seat Seat</i> memilih aksi Hapus, Sistem menghapus data rencana <i>overtime</i> . 10. Sistem memperbaharui basis data. 11. Sistem menampilkan Rencana <i>Overtime</i> .

Sumber: Hasil Analisis (2018)

Tabel V.10 di bawah ini berisi deskripsi untuk *use case* mencetak rencana overtime.

Tabel V.10 Definisi *Use Case* mencetak rencana overtime

Nama <i>Use case</i>	Mencetak <i>Overtime</i>
Aktor	<i>Leader Assembling Seat</i>
Deskripsi <i>Use case</i>	Proses mencetak rencana overtime
<i>Normal Flow of Events</i>	1. <i>Leader Assembling Seat</i> telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. <i>Leader Assembling Seat</i> memilih menu Data Transaksi. 4. <i>Leader Assembling Seat</i> memilih Submenu transaksi. 5. Sistem menampilkan submenu data transaksi, lalu <i>Leader</i> memilih rencana overtime. 6. Sistem menampilkan form tambah data rencana overtime 7. <i>Leader Assembling seat</i> memilih data rencana overtime, klik cetak. 8. Sistem menampilkan rencana yang akan di cetak 9. Sistem memperbaharui basis data. 10. Sistem menampilkan Rencana <i>Overtime</i>.

Sumber: Hasil Analisis (2018)

Tabel V.11 di bawah ini berisi deskripsi untuk *use case* memvalidasi.

Tabel V.11 Definisi *Use Case* memvalidasi hasil produksi

Nama <i>Use case</i>	Memvalidasi hasil produksi
Aktor	<i>Head Assembling Seat</i>
Deskripsi <i>Use case</i>	Proses memvalidasi hasil produksi
<i>Normal Flow of Events</i>	1. <i>Head Assembling Seat</i> telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. <i>Head Assembling Seat</i> memilih menu Validasi. 4. Sistem menampilkan data rencana produksi 5. <i>Head Assembling Seat</i> memilih validasi atau pending 6. Sistem memperbarui data hasil produksi

Sumber: Hasil Analisis (2018)

Tabel V.12 di bawah ini berisi deskripsi untuk *use case* mencetak laporan Harian Hasil Produksi

Tabel V.12 Definisi *Use Case* Mencetak Laporan Hasil Produksi

Nama <i>Use case</i>	Mencetak Laporan Hasil Produksi
Aktor	<i>Head Assembling Seat</i>
Deskripsi <i>Use case</i>	Proses mencetak laporan harian hasil produksi
<i>Normal Flow of Events</i>	1. <i>Head Assembling Seat</i> telah sukses melakukan login. 2. Sistem menampilkan menu utama. 3. <i>Head Assembling Seat</i> memilih menu data laporan. 4. <i>Head Assembling Seat</i> memilih Submenu data laporan hasil produksi <i>seat</i>. 5. <i>Head Assembling Seat</i> memasukkan data yang ingin ditampilkan, klik cari. 6. Sistem menampilkan laporan hasil produksi <i>seat</i>. 7. <i>Head Assembling Seat</i> klik cetak untuk mencetak laporan hasil produksi <i>seat</i>. 8. Sistem menampilkan data laporan hasil produksi <i>seat</i>.

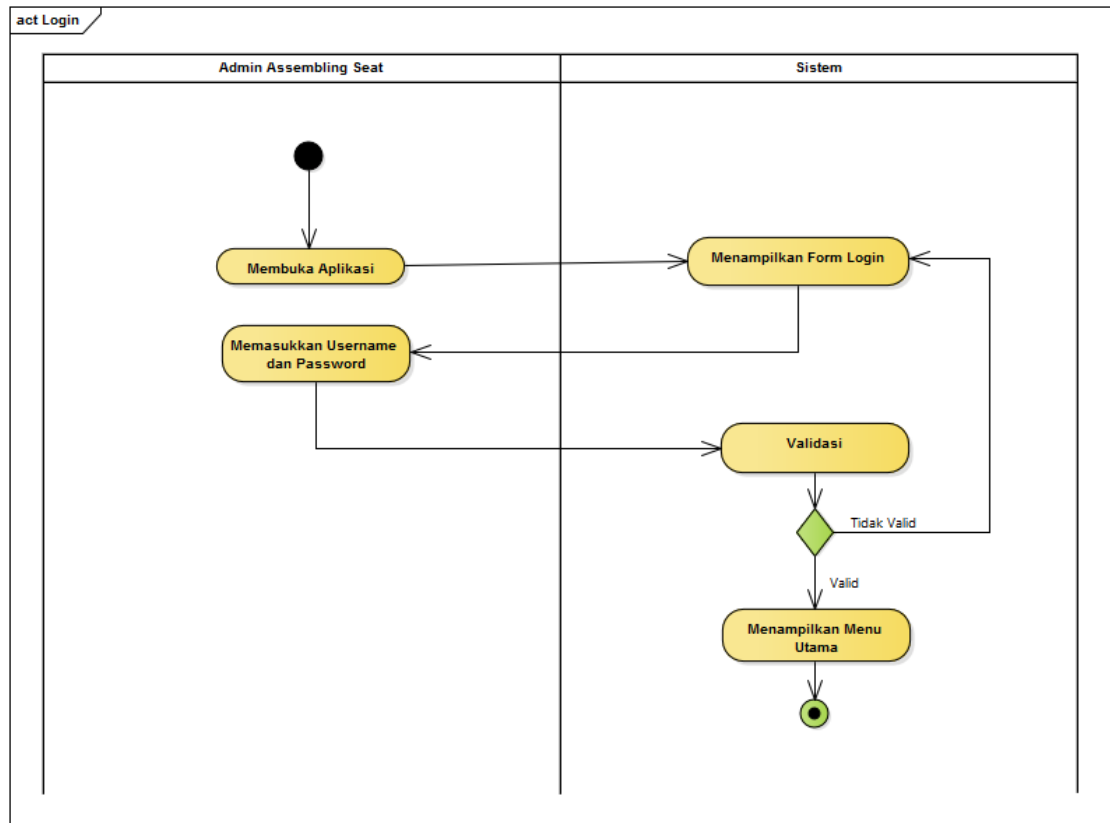
Sumber: Hasil Analisis (2018)

5.3.2 Activity Diagram

Activity diagram digunakan untuk menggambarkan aliran kerja tiap *use case* pada Sistem Informasi laporan hasil produksi *seat*. *Activity diagram* masing-masing *use case* digambarkan pada gambar-gambar berikut ini.

1. Activity Diagram Proses Login

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

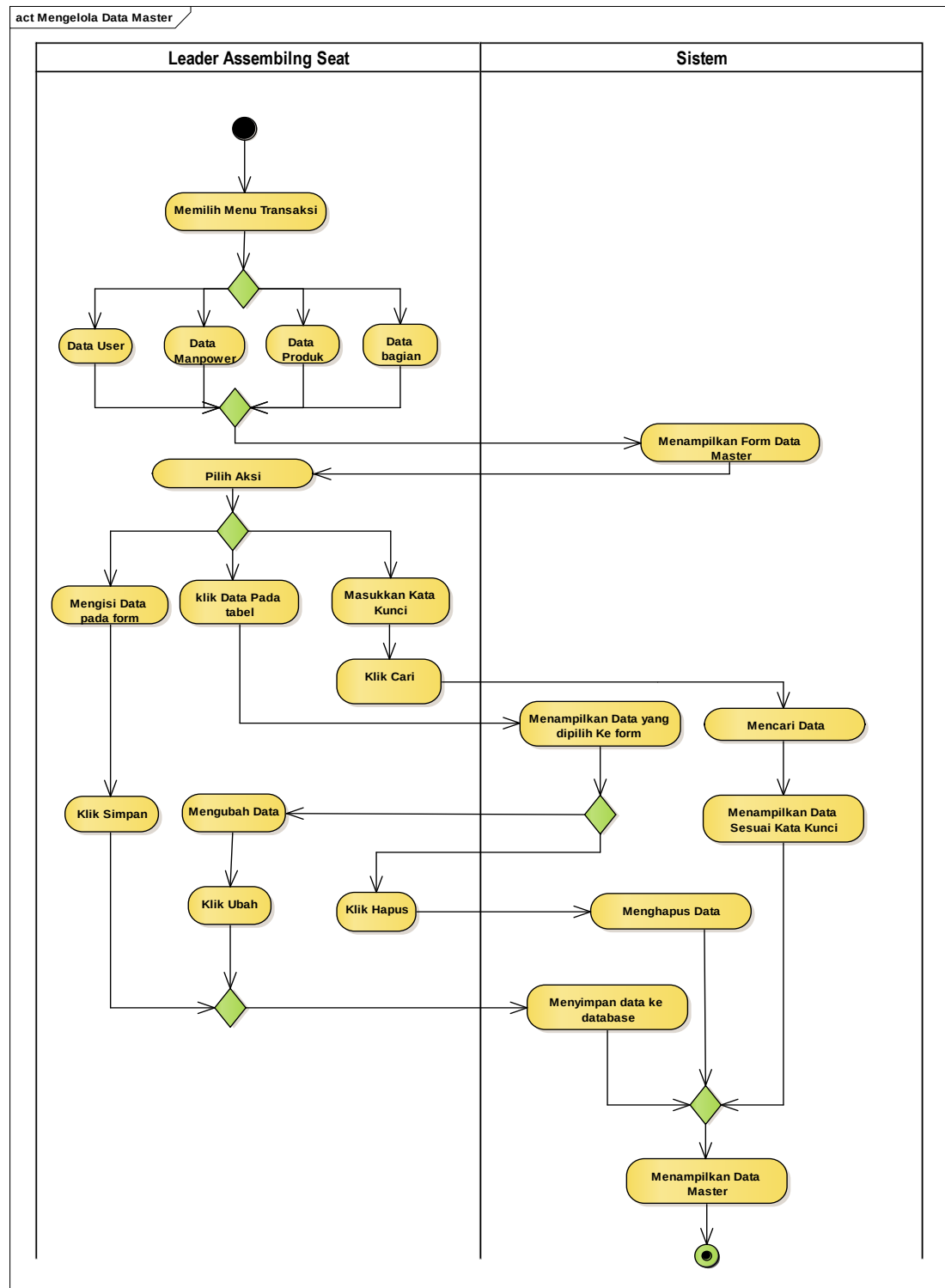


Gambar V.3 Activity Diagram Proses Login

Sumber: Hasil Analisis (2018)

2. Activity Diagram Mengelola Data Master

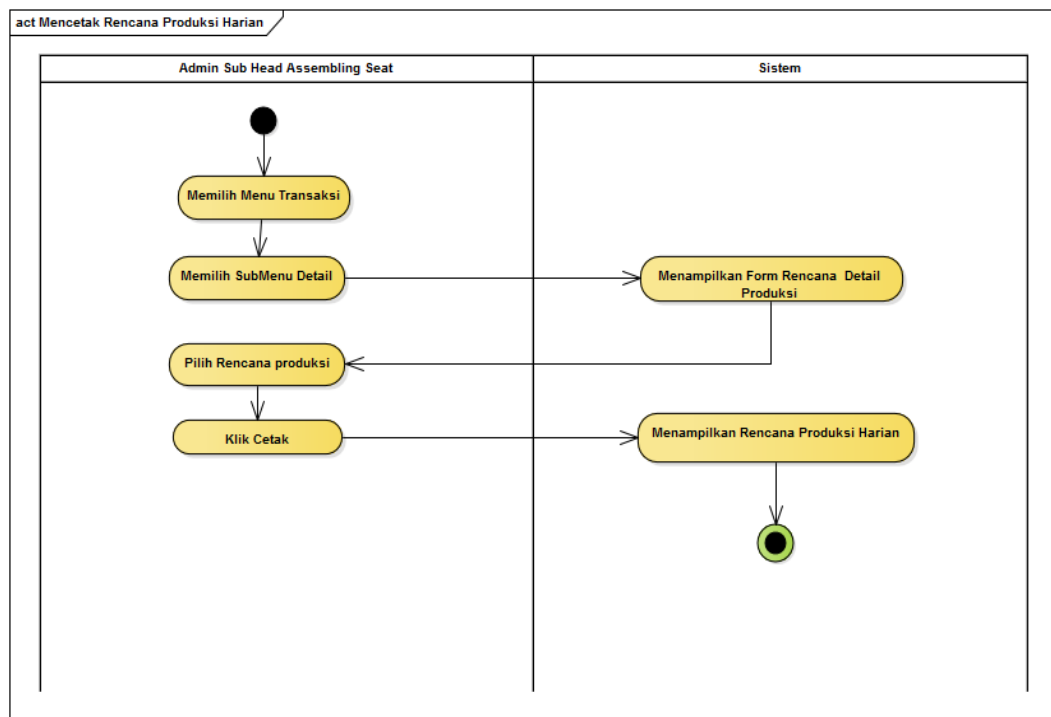
Activity diagram mengelola data master berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi, yang dilakukan oleh *user* yaitu *Leader Assembling Seat* untuk mengelola data master dalam sistem usulan yaitu data master produk, dan data master *user*. Activity diagram Mengelola Data Master yang diusulkan dapat dilihat pada Gambar V.4 sebagai berikut:



Gambar V.4 Activity Diagram Proses Mengelola Data Master
Sumber: Hasil Analisis (2018)

4. Activity Diagram Mencetak Rencana Produksi Harian

Activity diagram mencetak rencana produksi harian berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram mencetak rencana harian yang diusulkan dapat dilihat pada Gambar V.6 sebagai berikut:

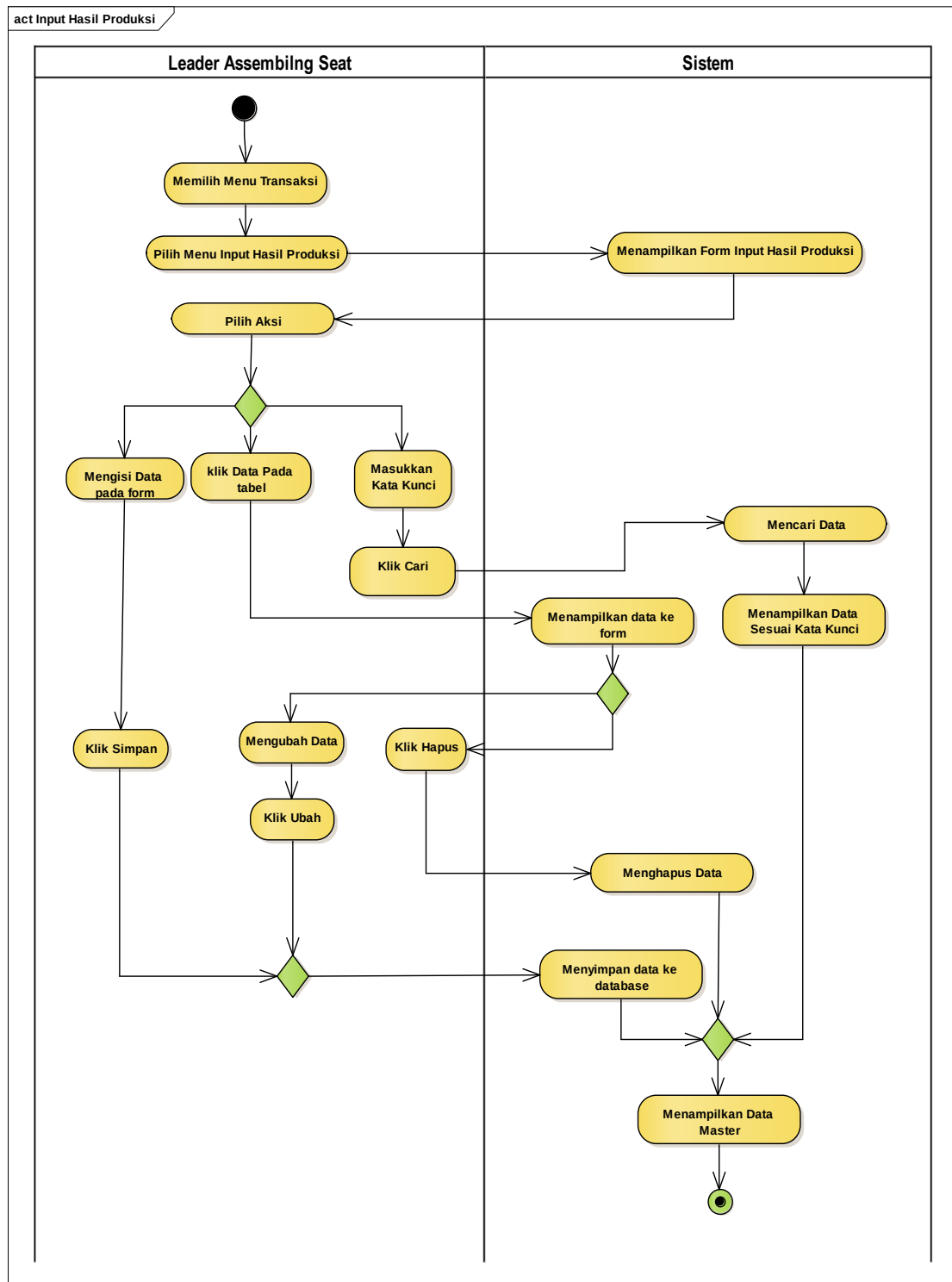


Gambar V.6 Activity Diagram Mencetak Rencana Produksi Harian

Sumber: Hasil Analisis (2018)

5. Activity Diagram Input Hasil Produksi

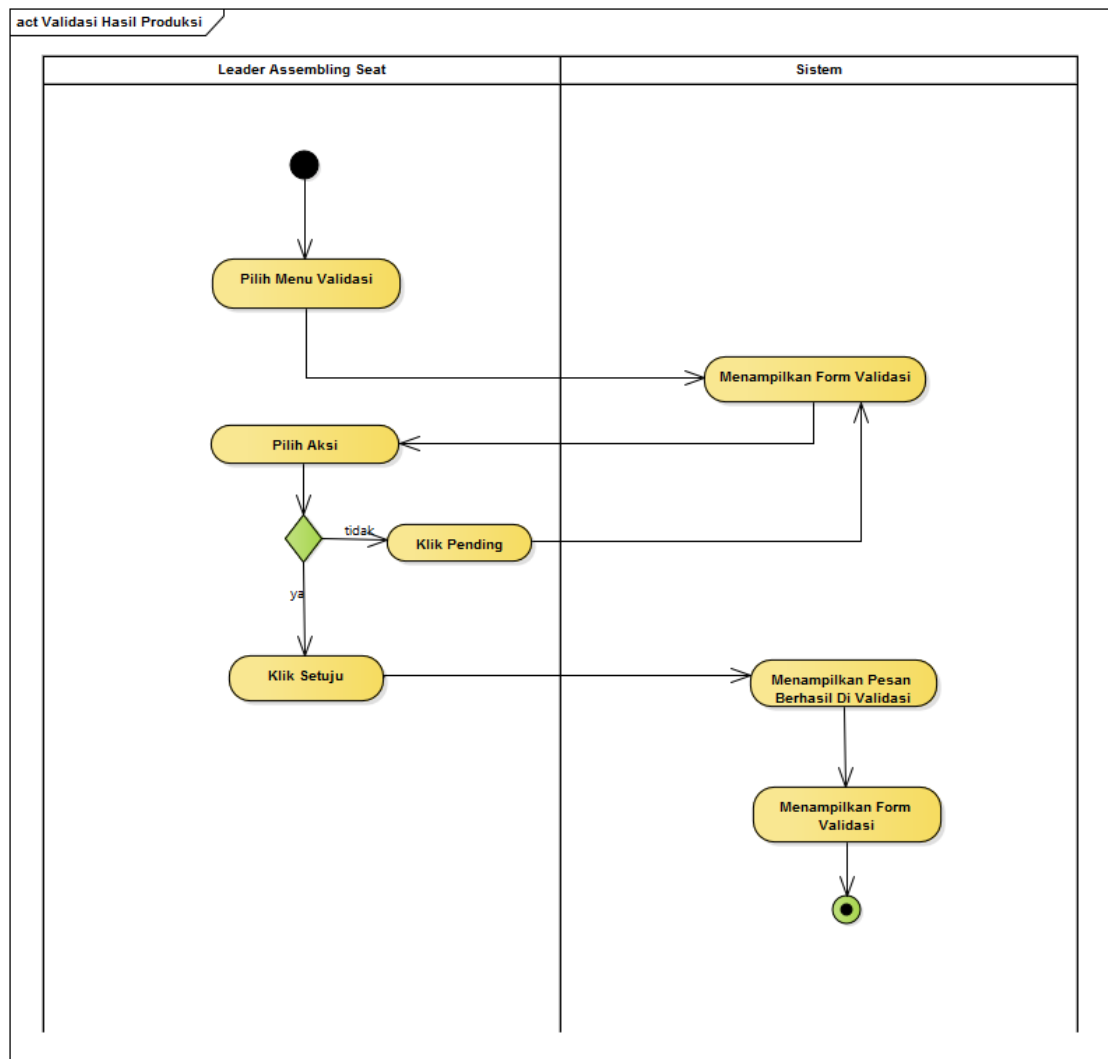
Activity diagram input hasil produksi berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram input rencana produksi harian yang diusulkan dapat dilihat pada Gambar V.7 sebagai berikut:



Gambar V.7 Activity Diagram Input Hasil Produksi
Sumber: Hasil Analisis (2018)

6. Activity Diagram Validasi Hasil Produksi

Activity diagram validasi hasil produksi berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram validasi hasil produksi yang diusulkan dapat dilihat pada Gambar V.8 sebagai berikut:



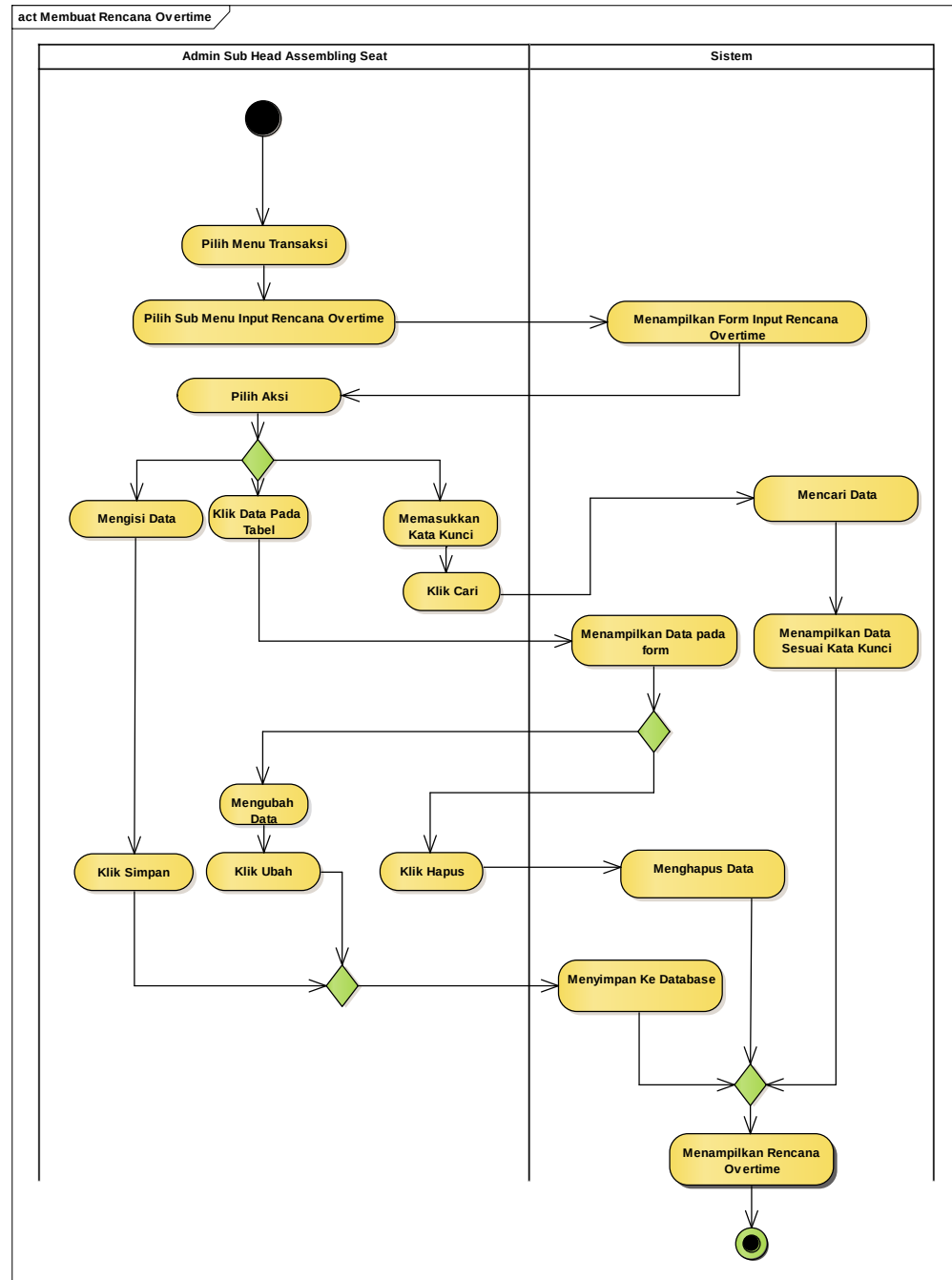
Gambar V.8 Activity Diagram Validasi Laporan Hasil Produksi

Sumber: Hasil Analisis (2018)

7. Activity Diagram Membuat Rencana Overtime

Activity diagram membuat rencana overtime berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram

membuat rencana *overtime* yang diusulkan dapat dilihat pada Gambar V.9 sebagai berikut:

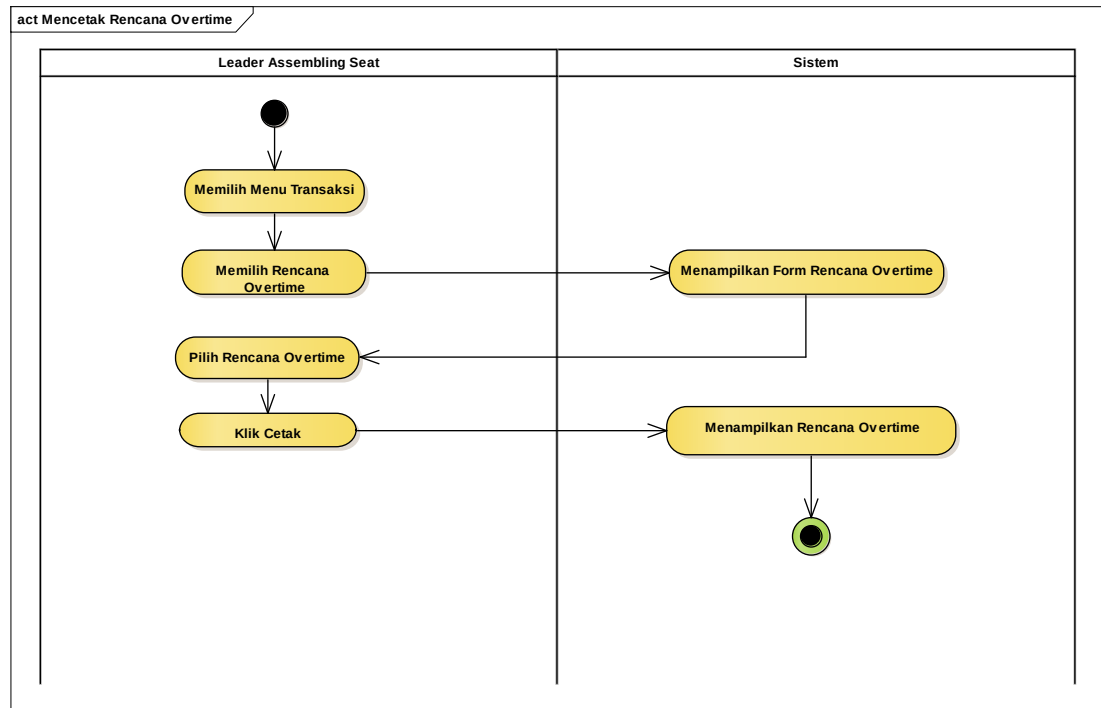


Gambar V.9 Activity Diagram Membuat Rencana Overtime

Sumber: Hasil Analisis (2018)

8. Activity Diagram Mencetak Overtime

Activity diagram mencetak rencana overtime berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram mencetak rencana overtime yang diusulkan dapat dilihat pada Gambar V.10 sebagai berikut:

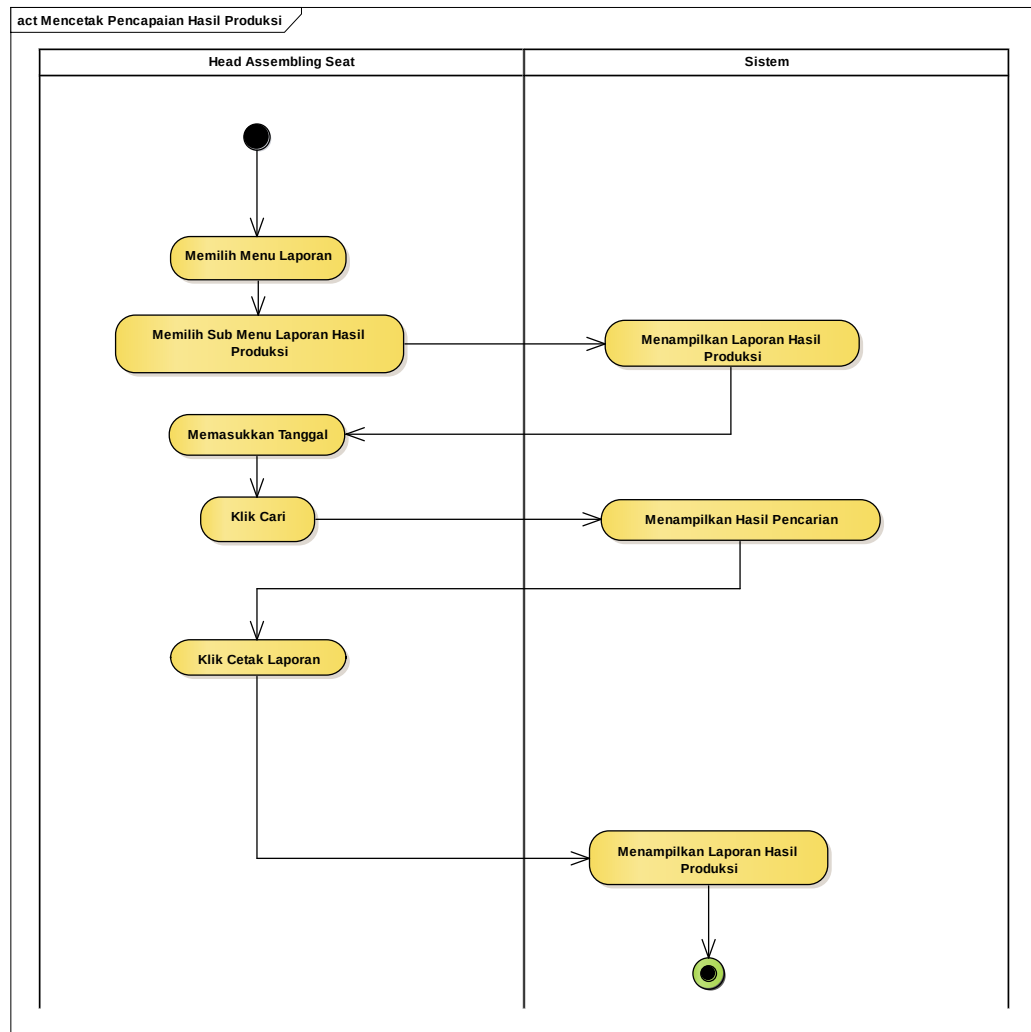


Gambar V.10 Activity Diagram Mencetak Rencana Overtime

Sumber: Hasil Analisis (2018)

9. Activity Diagram Mencetak Laporan Hasil Produksi

Activity diagram mencetak laporan hasil produksi berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. Activity diagram mencetak laporan hasil produksi yang diusulkan dapat dilihat pada Gambar V.11 sebagai berikut:



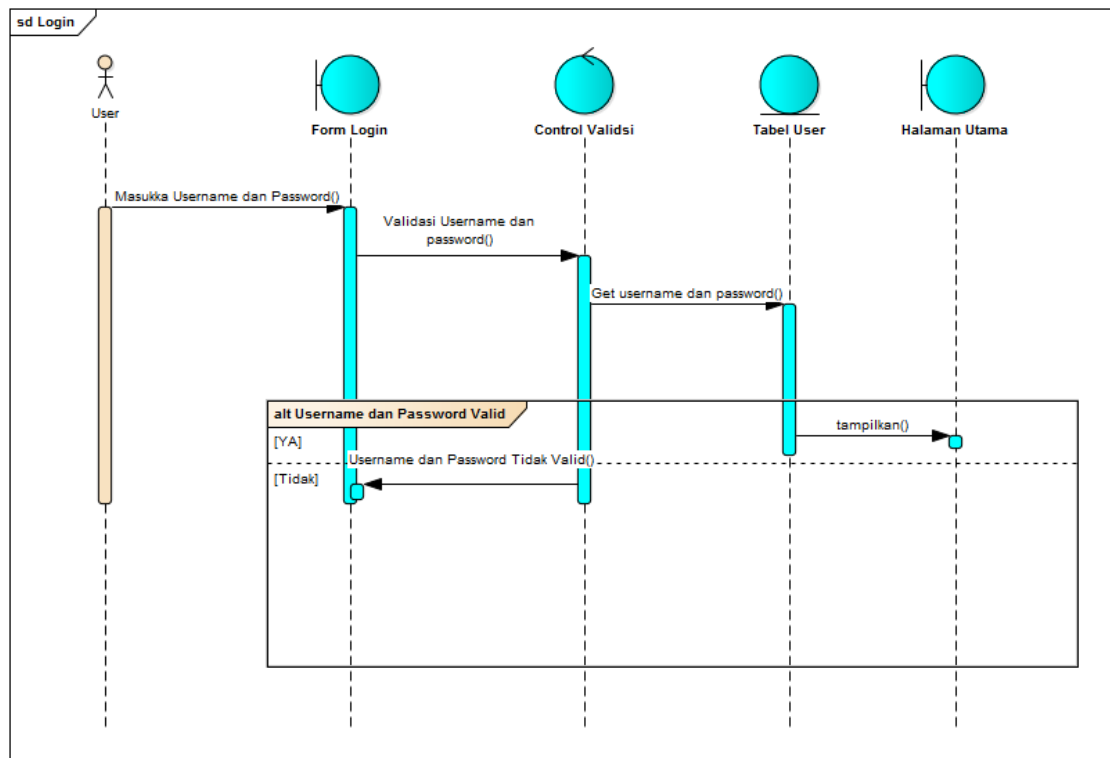
Gambar V.11 *Activity Diagram* mencetak laporan hasil produksi
 Sumber: Hasil Analisis (2018)

5.3.3 *Sequence Diagram*

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

1. Sequence Diagram Proses Login

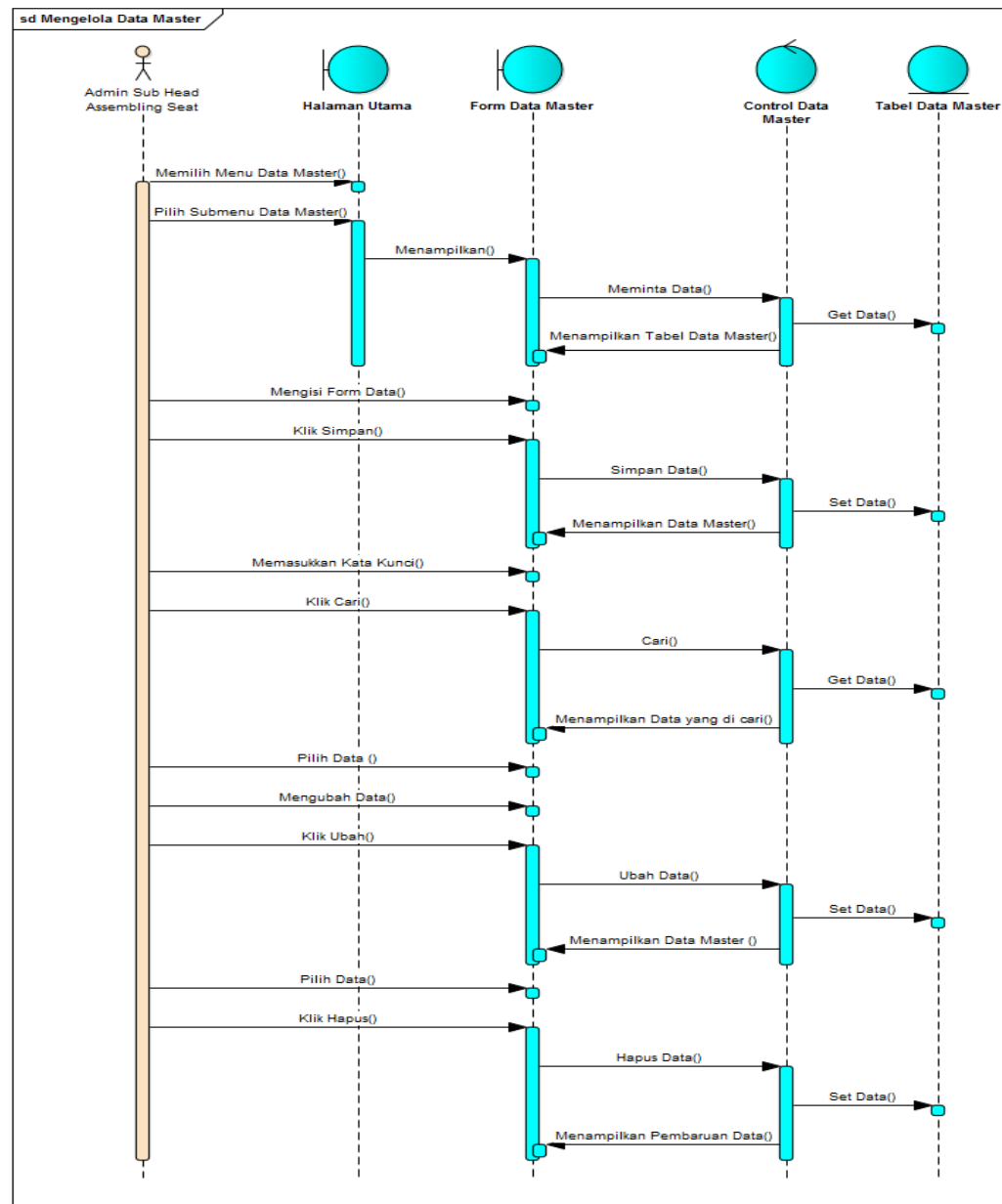
Sequence diagram login menjelaskan sebuah *sequence diagram* dalam proses login. Proses ini dilakukan oleh *user* sebelum masuk sistem. Adapun *sequence diagram* dari *use case login* dapat dilihat pada Gambar V.12 sebagai berikut:



Gambar V.12 *Sequence Diagram login*
(Sumber: Hasil Analisis Data, 2018)

2. Sequence Diagram Proses Mengelola Data Master

Sequence diagram Mengelola Data Master menjelaskan sebuah *sequence diagram* dalam proses mengelola data master yaitu menambah, menghapus dan mengubah data peminjaman. Adapun *sequence diagram* dari *use case mengelola data master* dapat dilihat pada Gambar V.13 sebagai berikut:

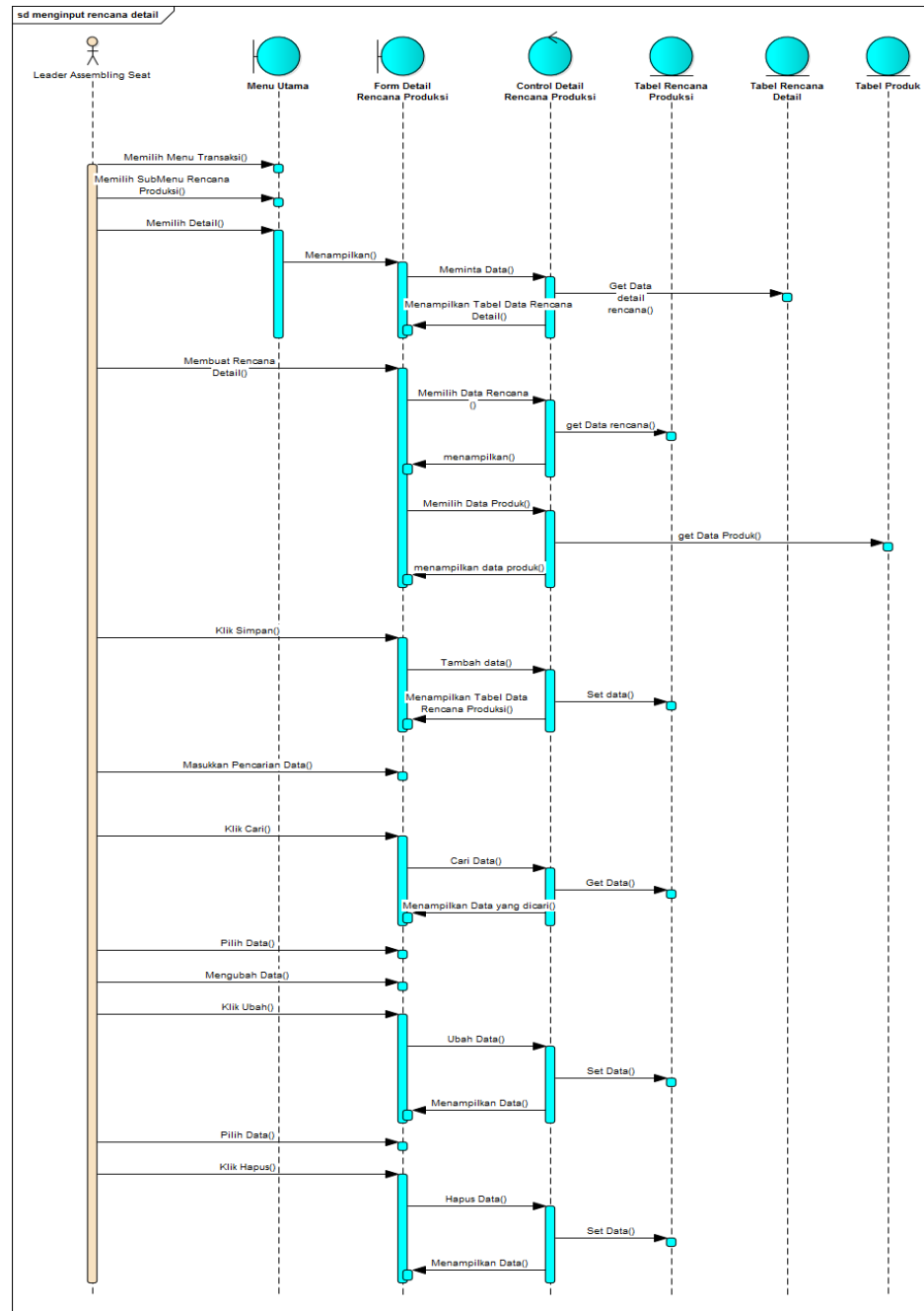


Gambar V.13 Sequence Diagram Mengelola Data Master
(Sumber: Hasil Analisis Data, 2018)

3. Sequence Diagram Input Rencana Produksi Harian

Sequence diagram input rencana produksi harian menjelaskan sebuah sequence diagram dalam proses transaksi menambah, menghapus dan mengubah data

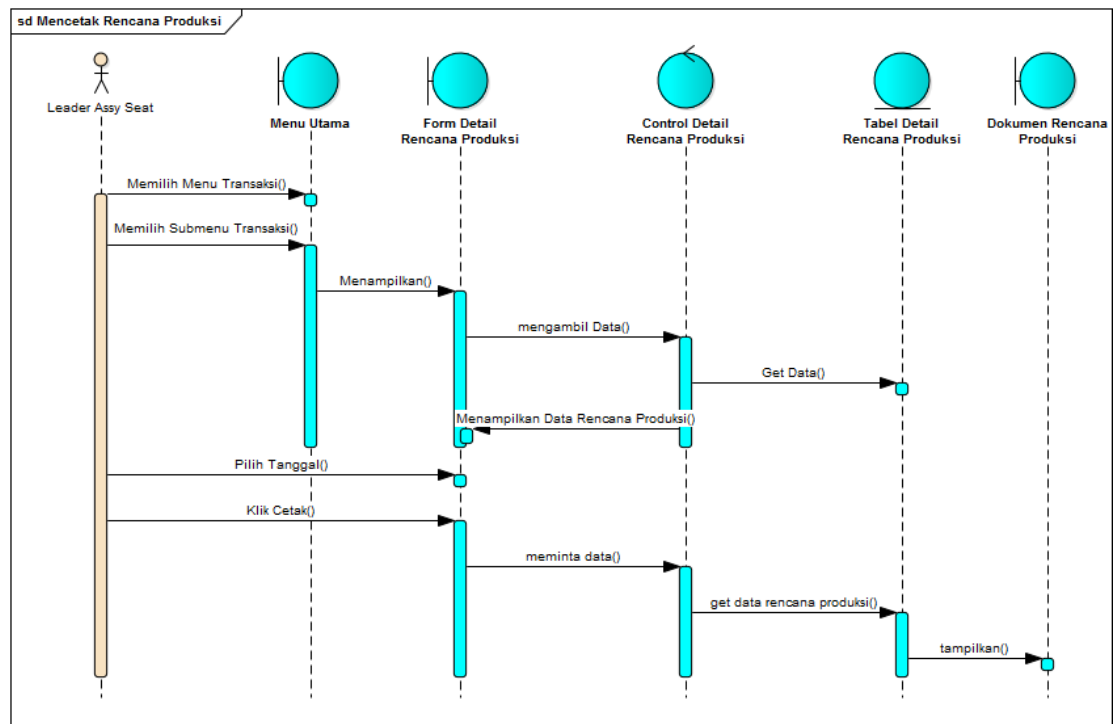
rencana produksi harian. Adapun *sequence diagram* dari *use case input* rencana produksi harian dapat dilihat pada Gambar V.14 sebagai berikut



Gambar V.14 *Sequence Diagram* Menginput Rencana Produksi
(Sumber: Hasil Analisis Data, 2018)

4. *Sequence Diagram* Mencetak Rencana Produksi Harian

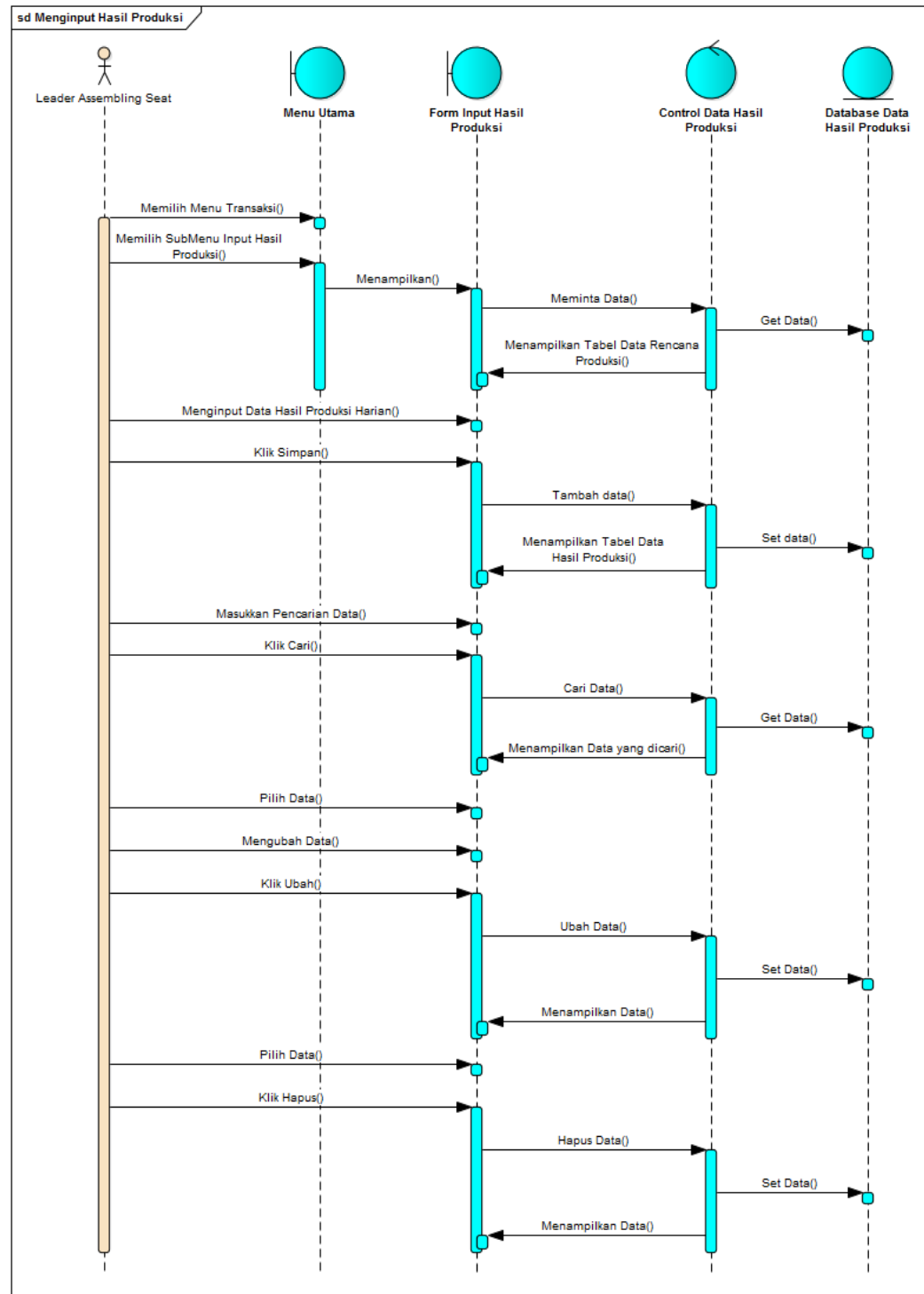
Sequence diagram mencetak rencana produksi harian menjelaskan sebuah *sequence diagram* dalam proses mencetak rencana produksi harian berdasarkan rencana produksi *assembling body*. Adapun *sequence diagram* dari *use case* mencetak jadwal harian produksi operator dapat dilihat pada Gambar V.15 sebagai berikut:



Gambar V.15 *Sequence Diagram* Mencetak Rencana Produksi
(Sumber: Hasil Analisis Data, 2018)

5. *Sequence Diagram* Input Hasil Produksi

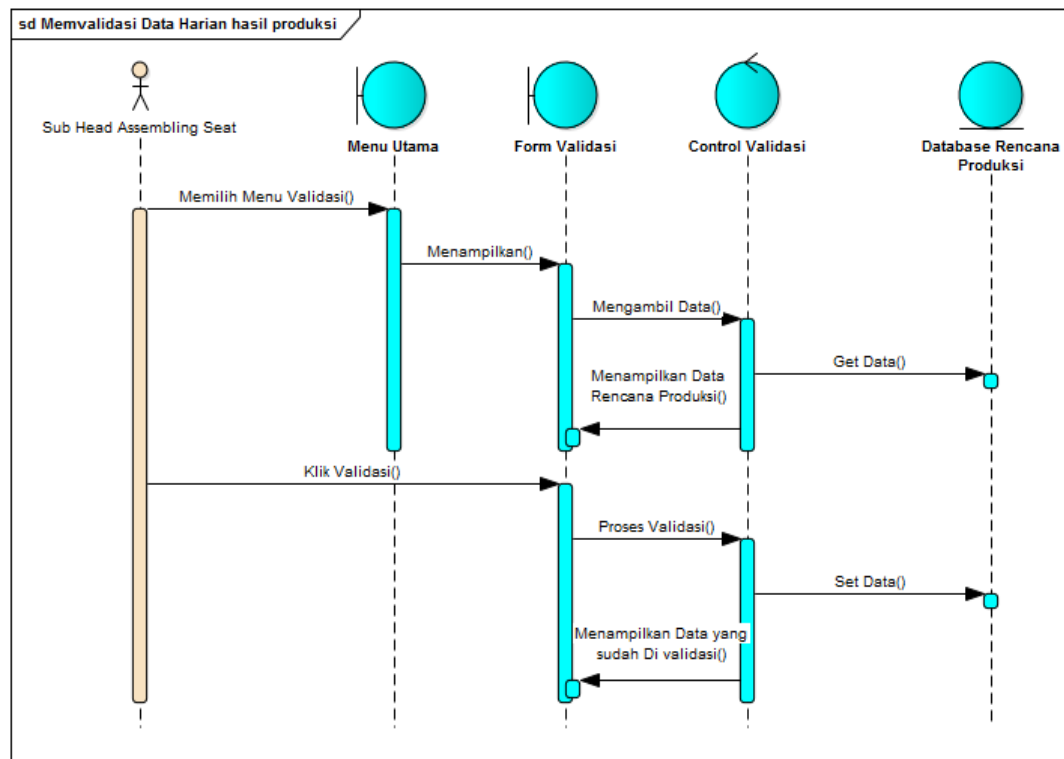
Sequence diagram input hasil produksi menjelaskan sebuah *sequence diagram* dalam proses mengisi data-data pada *form input* hasil produksi. Adapun *sequence diagram* dari *use case* input hasil produksi dapat dilihat pada Gambar V.16 sebagai berikut:



Gambar V.16 Sequence Diagram Menginput Hasil Produksi
(Sumber: Hasil Analisis Data, 2018)

6. *Sequence Diagram* Validasi Hasil Produksi

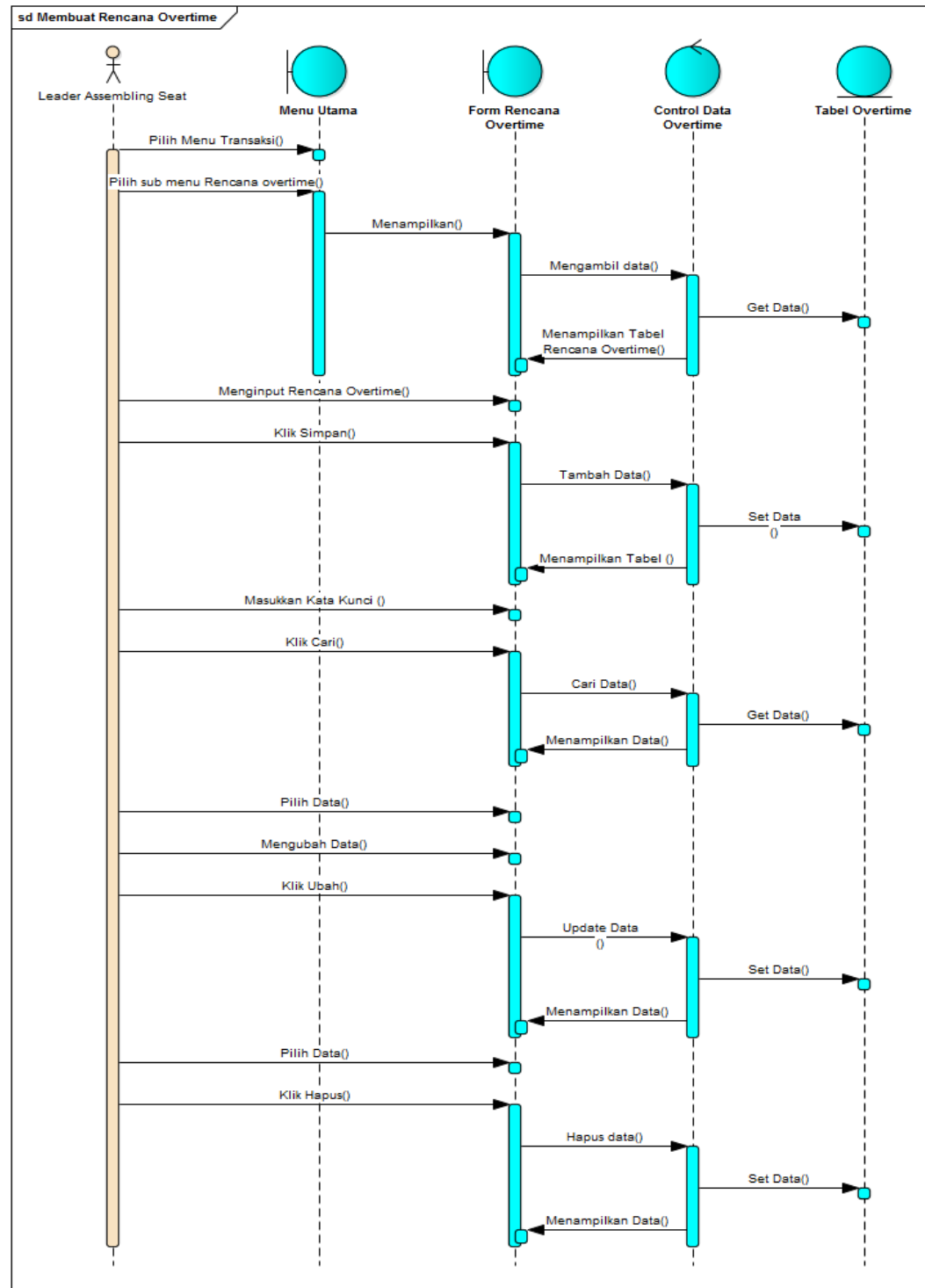
Sequence diagram menampilkan *form* validasi menjelaskan sebuah *sequence diagram* dalam proses menampilkan data validasi. Adapun *sequence diagram* dari *use case* menampilkan data *form* validasi dapat dilihat pada Gambar V.17 sebagai berikut:



Gambar V.17 *Sequence Diagram* Validasi Hasil Produksi
(Sumber: Hasil Analisis Data, 2018)

7. *Sequence Diagram* Membuat Rencana Overtime

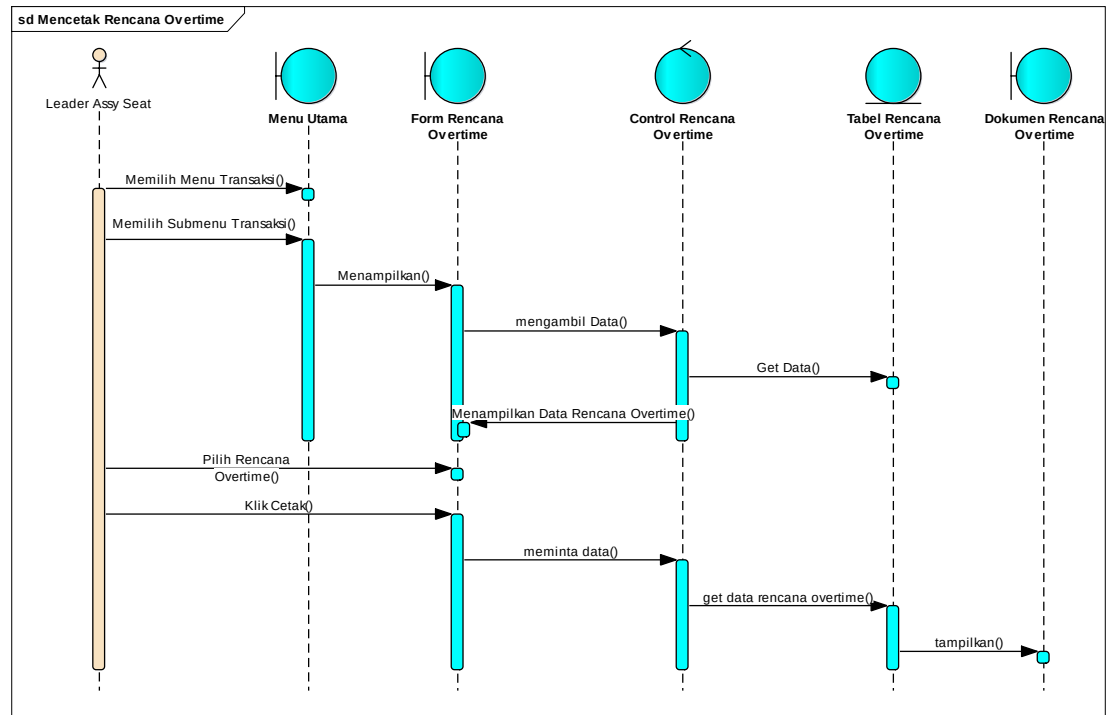
Sequence diagram membuat rencana *overtime* menjelaskan sebuah *sequence diagram* dalam proses menampilkan data rencana *overtime*. Adapun *sequence diagram* dari *use case* membuat *overtime* dapat dilihat pada Gambar V.18 sebagai berikut:



Gambar V.18 Sequence Diagram Membuat Rencana Overtime
(Sumber: Hasil Analisis Data, 2018)

8. *Sequence Diagram Mencetak Rencana Overtime*

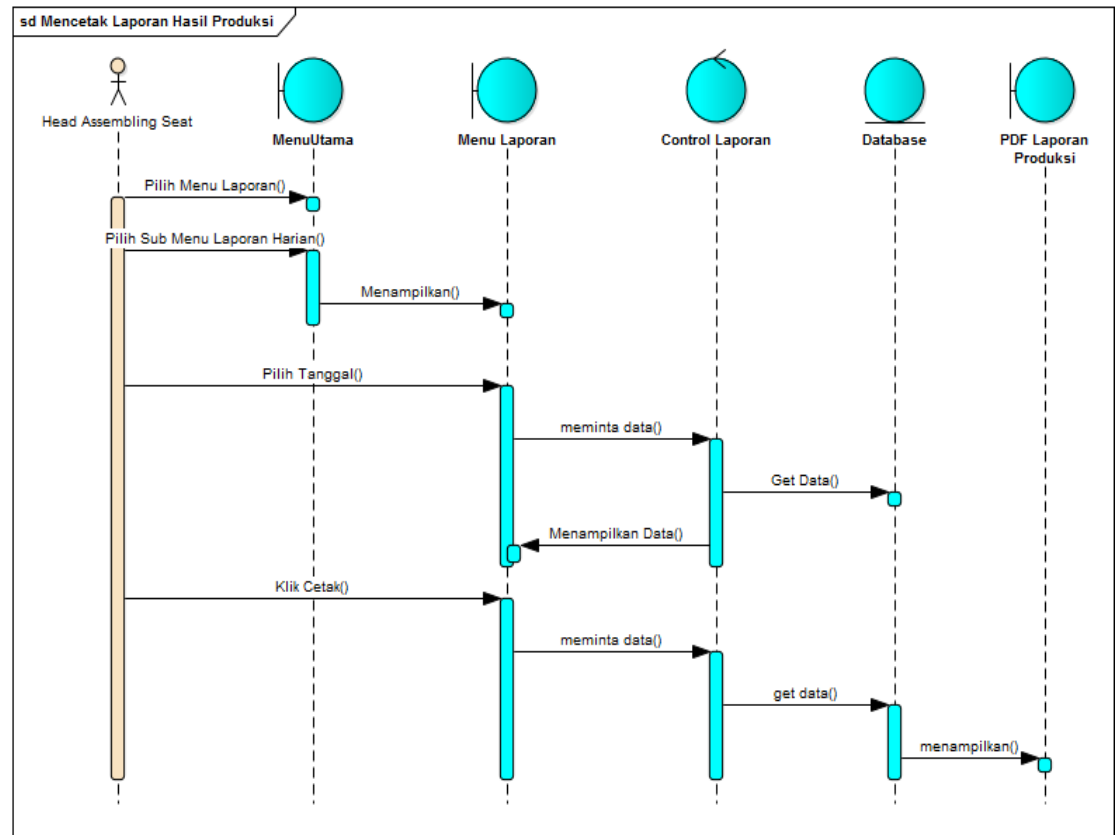
Sequence diagram mencetak rencana overtime menjelaskan sebuah *sequence diagram* dalam proses menampilkan data rencana overtime yang akan di cetak. Adapun *sequence diagram* dari *use case* mencetak overtime dapat dilihat pada Gambar V.19 sebagai berikut:



Gambar V.19 *Sequence Diagram* Mencetak Laporan Hasil Produksi
(Sumber: Hasil Analisis Data, 2018)

9. *Sequence Diagram Mencetak Laporan Hasil Produksi*

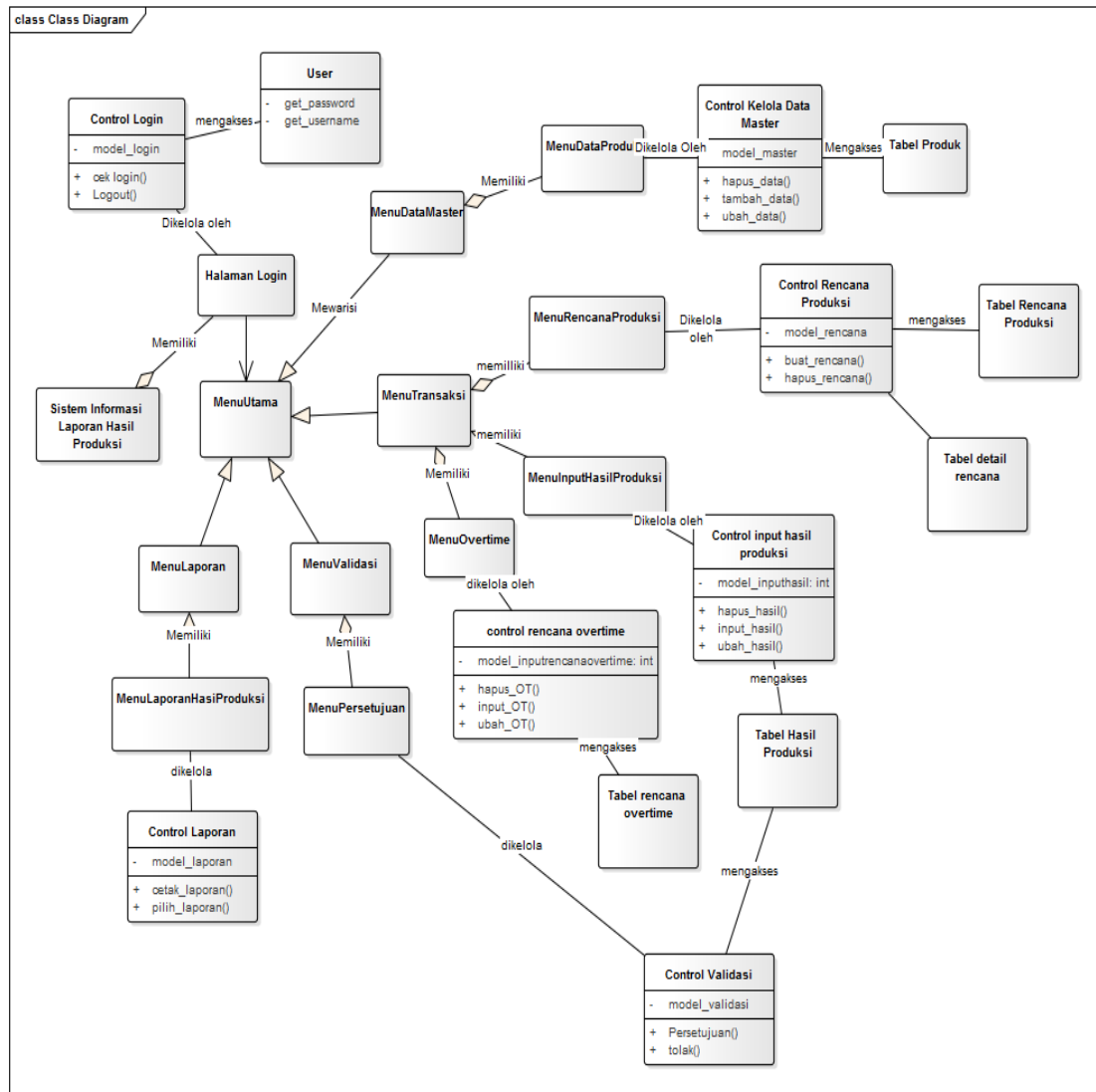
Sequence diagram mencetak laporan produksi menjelaskan sebuah *sequence diagram* dalam proses mencetak laporan produksi berdasarkan hasil aktual proses produksi. Adapun *sequence diagram* dari *use case* Mencetak laporan produksi dapat dilihat pada Gambar V.20 sebagai berikut:



Gambar V.20 *Sequence Diagram* Mencetak Laporan Hasil Produksi
(Sumber: Hasil Analisis Data, 2018)

5.3.4 *Class Diagram*

Class diagram pada usulan sistem informasi laporan hasil produksi *seat* digunakan untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat, untuk membangun sistem *class diagram* memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem.

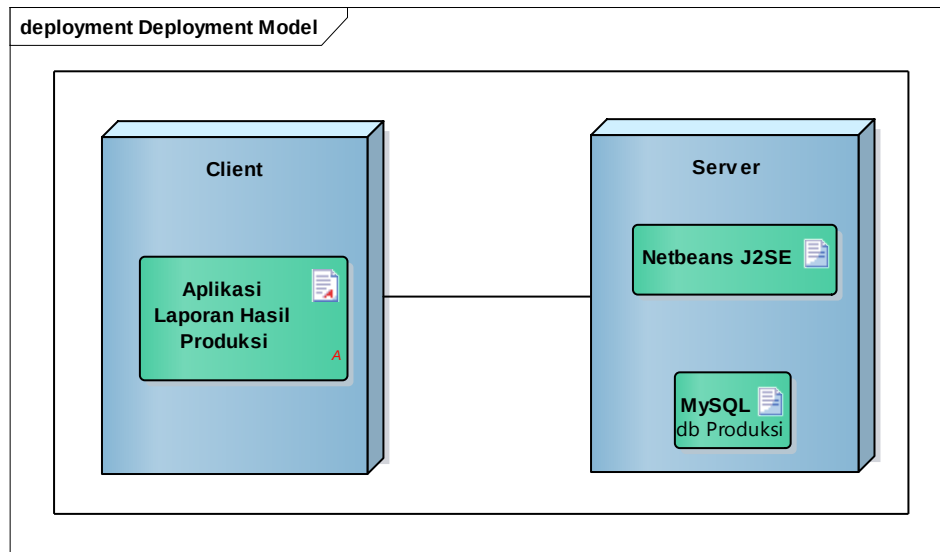


Gambar V.21 Class Diagram Laporan Hasil Produksi

(Sumber: Hasil Analisis Data, 2018)

5.3.5 Deployment Diagram

Deployment diagram pada usulan sistem informasi laporan hasil produksi *seat* digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. Berikut merupakan *deployment diagram* sistem informasi laporan hasil produksi *seat*



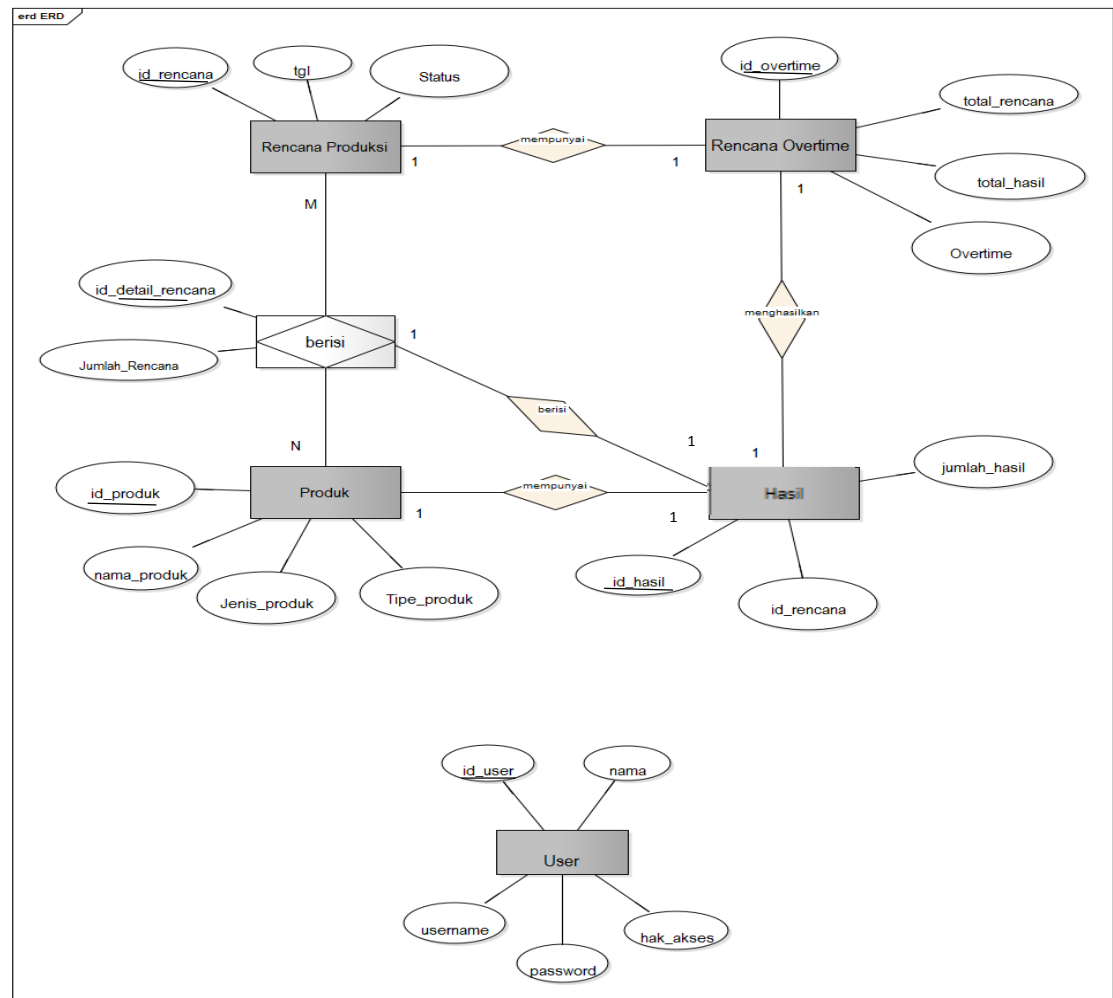
Gambar V.22 *Deployment Diagram* Laporan Hasil Produksi
(Sumber: Hasil Analisis Data, 2018)

5.4 Permodelan Data Sistem Usulan

Pemodelan data pada sistem informasi laporan hasil produksi *seat* di PT Suzuki Indomobil Motor Cikarang Plant menggunakan dua cara yaitu *Entity Relationship Diagram* (ERD) yang digunakan untuk menjelaskan hubungan antar data dalam basis data dan kamus data yang digunakan untuk menjelaskan isi dari *database* yang digunakan dalam sistem usulan.

5.4.1 *Entity Relationship Diagram* (ERD)

ERD digunakan sebagai suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi dalam sistem usulan. ERD digambarkan dengan simbol dan notasi yang berbeda, berikut merupakan ERD dari sistem informasi laporan hasil produksi *seat*:

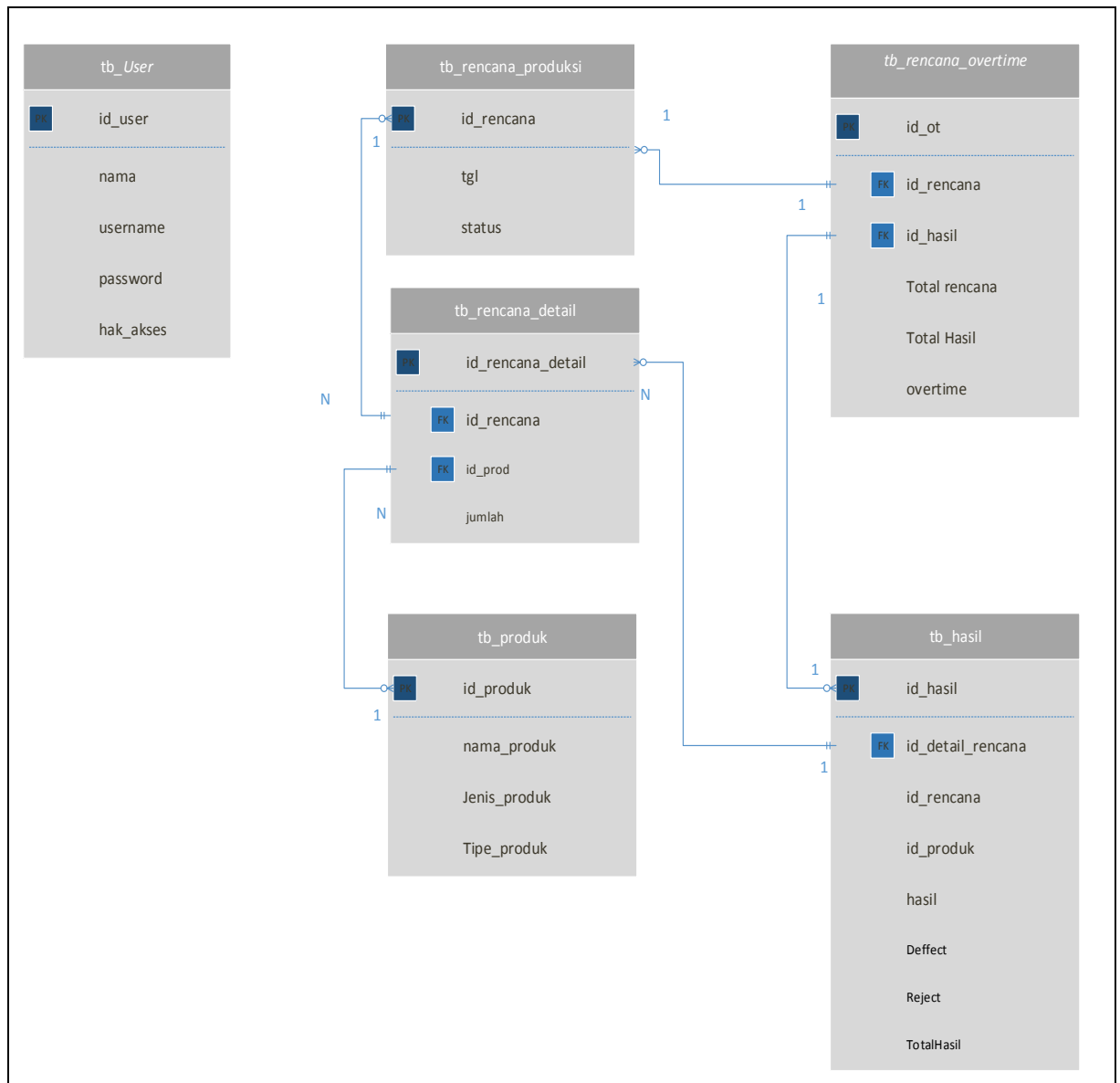


Gambar V.23 Entity Relationship Diagram

(Sumber: Hasil Analisis Data, 2018)

5.4.2 Conceptual Data Model (CDM)

Conceptual Data Model digunakan untuk menggambarkan atau merepresentasikan seluruh muatan informasi yang dikandung oleh basis data, berikut merupakan *Conceptual Data Model* dari sistem usulan:



Gambar V.24 *Conceptual Data Model*
(Sumber: Hasil Analisis Data, 2018)

5.4.3 Kamus Data

Kamus menggambarkan sebuah katalog data yang terdapat didalam sistem dengan maksud untuk mendefinisikan data yang mengalir di dalam sistem dengan lengkap sehingga user dan analis sistem mempunyai pengertian yang sama tentang

input, *output*, dan komponen data store. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan dan basis data. Berikut adalah kamus data sistem informasi laporan hasil produksi:

1. Spesifikasi tabel *user*

- a. Nama tabel : *tb_user*
- b. Fungsi : Untuk menyimpan data *user*/pengguna
- c. Tipe : File data *master*

Tabel V.13 Spesifikasi Tabel *tb_user*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id <i>user</i>	<i>id_user</i>	Varchar	10	<i>Primary key</i>
2	Nama	<i>nama</i>	Varchar	30	
3	<i>Username</i>	<i>username</i>	Varchar	30	
4	<i>Password</i>	<i>password</i>	Varchar	20	
5	Hak akses	<i>Hak_akses</i>	Varchar	10	

(Sumber: Hasil Analisis, 2018)

2. Spesifikasi tabel Produk

- a. Nama tabel : *tb_produk*
- b. Fungsi : Untuk menyimpan data produk
- c. Tipe : File data *master*

Tabel V.14 Spesifikasi Tabel produk

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	id Produk	id_ptoduk	Varchar	10	<i>Primary key</i>
2	Nama produk	Nama_Produk	Varchar	30	
3	Jenis_produk	Jenis_Produk	Varchar	20	

(Sumber: Hasil Analisis, 2018)

3. Spesifikasi tabel rencana produksi

- a. Nama tabel : tb_rencana
- b. Fungsi : Untuk menyimpan data rencana produksi
- c. Tipe : File data transaksi

Tabel V.15 Spesifikasi Tabel rencana produksi

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	id rencana	id_rencana	Varchar	10	<i>Primary key</i>
2	Tanggal rencana	Tanggal	Date		
3	Status	status	varchar	10	

(Sumber: Hasil Analisis, 2018)

4. Spesifikasi tabel detail rencana

- a. Nama tabel : tb_detail_rencana
- b. Fungsi : Untuk menyimpan data detail rencana produksi
- c. Tipe : File data transaksi

Tabel V.16 Spesifikasi Tabel Detail Rencana

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id detail Rencana	id_detilhasil	Varchar	10	Primary key
2	Id rencana	id_spare part	Varchar	10	Foreign key
3	Id Produk	jumlah	Varchar	10	Foreign key
4	Jumlah Rencana	keterangan	int		

(Sumber: Hasil Analisis, 2018)

5. Spesifikasi tabel hasil produksi

- a. Nama tabel : tb_hasil
- b. Fungsi : Untuk menyimpan data hasil produksi
- c. Tipe : File data transaksi

Tabel V.17 Spesifikasi Tabel Hasil Produksi

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id hasil	id_hasil	Varchar	10	Primary key
2	Id detail rencana	Id_detail_rencana	Varchar	10	Foreign key
3	Id rencana	Id_rencana	Varchar	10	Foreign key
4	Id Produk	Id_Prod	Varchar	10	Foreign key
5	Hasil	hasil	Int		
6	Defect	Defect	Int		
7	Reject	reject	Int		
8	Jumlah Hasil	total_hasil	Int		

(Sumber: Hasil Analisis, 2018)

6. Spesifikasi tabel *Overtime*

- a. Nama tabel : tbot
- b. Fungsi : Untuk menyimpan data *overtime*
- c. Tipe : File data transaksi

Tabel V.18 Spesifikasi Tabel rencana *overtime*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id <i>Overtime</i>	id_OT	Varchar	10	<i>Primary key</i>
2	Id Rencana	Id_rencana	Varchar	10	<i>Foreign key</i>
3	Id hasil	Id_hasil	Varchar	10	<i>Foreign key</i>
4	Jumlah rencana	Jumlah_hasil	Int		
5	Jumlah hasil	Jumlah_rencana	Int		
6	<i>Overtime</i>	ot	Varchar	20	

(Sumber: Hasil Analisis, 2018)

5.5 Perancangan Sistem

Perancangan sistem pada sistem usulan digunakan sebagai ilustrasi yang menggambarkan bagaimana aplikasi yang dibuat berjalan dan juga menggambarkan navigasi dari aplikasi tersebut dengan *tools Windows Navigation Diagram (WND)* serta menggambarkan tampilan antar muka dari aplikasi.

5.5.1 *Windows Navigation Diagram (WND)*

Windows Navigation Diagram pada sistem usulan digunakan untuk menunjukan bagaimana navigasi dari halaman-halaman yang terdapat di dalam aplikasi, berikut merupakan *Windows Navigation Diagram* sistem informasi laporan hasil produksi *seat*:

(Sumber: Hasil Analisis Data, 2018)

5.5.2 Perancangan Antarmuka Pengguna

Perancangan antarmuka dalam sistem usulan merupakan penggambaran antarmuka aplikasi yang hendak dibuat, antarmuka merupakan bagian yang berhubungan langsung dengan pengguna, berikut merupakan rancangan antarmuka sistem informasi laporan hasil produksi:

1. Form Login

Form login merupakan *form* yang digunakan pengguna sebagai jalan masuk ke dalam aplikasi dengan cara memasukkan *username* dan *password* yang terdapat didalam *form login* tersebut, rancangan *form login* dapat dilihat pada gambar berikut:

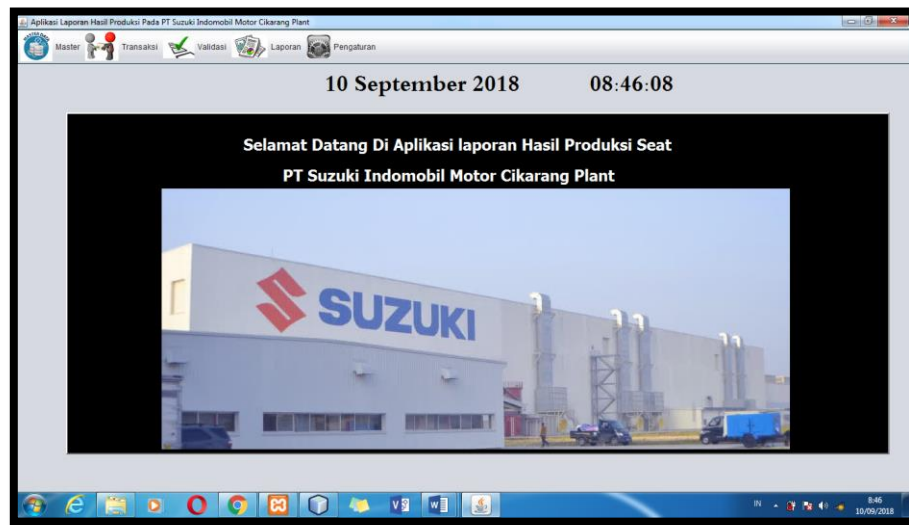


The image shows a login form for an application titled "APLIKASI SISTEM INFORMASI LAPORAN HASIL PRODUKSI SEAT". The form is centered on a light gray background. It has a dark green header bar with the application title in white. Below the header, there is a dark gray box containing the login form. The form has a blue header bar with the text "Masukkan id pengguna dan kata sandi Anda :)". Below this, there are two white input fields. The first field is labeled "Id Pengguna" and the second field is labeled "Kata Sandi". At the bottom of the form, there are two buttons: "Batal" (Cancel) and "Masuk" (Login).

Gambar V.26 *Form login* Aplikasi
(Sumber: Hasil Analisis Data, 2018)

2. Tampilan Halaman Utama

Tampilan halaman utama merupakan tampilan yang muncul pertama kali saat user berhasil melakukan *login* ke dalam sistem, tampilan halaman utama dapat dilihat pada gambar berikut:



Gambar V.27 Halaman Utama
(Sumber: Hasil Analisis Data, 2018)

3. Tampilan *Form Data Master User*

Tampilan halaman data master produk merupakan tampilan yang muncul ketika user memilih sub menu data master *user*, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data master *user*, berikut tampilan halaman data master *user*:

No	Id Pengguna	Nama	Username	Password	Hak Akses

Gambar V.28 *Form Data User*
(Sumber: Hasil Analisis Data, 2018)

4. Tampilan *Form* Data Master Produk

Tampilan halaman data master produk merupakan tampilan yang muncul ketika user memilih sub menu data master produk, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data master produk, berikut tampilan halaman data master produk:

Gambar V.29 *Form* Data Produk
(Sumber: Hasil Analisis Data, 2018)

5. Tampilan *Form* Rencana Produksi

Tampilan *Form* Rencana Produksi merupakan tampilan yang muncul ketika user memilih sub menu data transaksi, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data rencana Produksi, berikut tampilan *form* rencana produksi:

Gambar V.30 *Form Rencana Produksi*
(Sumber: Hasil Analisis Data, 2018)

6. Tampilan *Form* Detail Rencana Produksi

Tampilan *form* detail rencana produksi merupakan tampilan yang muncul ketika user memilih *button* detail rencana produksi, halaman ini menyediakan *form* untuk menampilkan data detail rencana produksi, berikut tampilan *form* detail rencana produksi:

Gambar V.31 *Form Detail Rencana Produksi*
(Sumber: Hasil Analisis Data, 2018)

7. Tampilan *Form Input* Hasil Produksi

Tampilan *form input* hasil produk merupakan tampilan yang muncul ketika user memilih sub menu data transaksi, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data hasil produksi, berikut tampilan *form input* hasil produk:

Form Input Hasil Produksi

Id Hasil: Hasil Produksi:

ID Detail: Defect:

ID Rencana: Reject:

Id Produk: Total Hasil:

Jumlah rencana: Minus:

Id Hasil	IdRencana	Id Produk	Jumlah Rencana	Hasil Produksi

TOTAL

Gambar V.32 *Form input* hasil produksi
(Sumber: Hasil Analisis Data, 2018)

8. Tampilan *Form Rencana Overtime*

Tampilan *form* rencana *overtime* merupakan tampilan yang muncul ketika user memilih sub menu transaksi, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data rencana *overtime*, berikut tampilan *form* rencana *overtime*:

Gambar V.33 *Form rencana overtime*
(Sumber: Hasil Analisis Data, 2018)

9. Tampilan *Form* validasi

Tampilan *form* validasi merupakan tampilan yang muncul ketika user memilih menu validasi, halaman ini menyediakan tombol untuk memvalidasi hasil rencana produksi, berikut tampilan *form* validasi:

Gambar V.34 *Form rencana overtime*
(Sumber: Hasil Analisis Data, 2018)

10. *Form Laporan Hasil Produksi*

Tampilan laporan hasil produksi merupakan tampilan yang muncul ketika user memilih sub menu laporan, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mencari data berdasarkan tanggal, berikut tampilan untuk cari laporan hasil produksi:

Id Rencana	Id Produk	Nama Pro...	Tipe Produk	Jenis Produk	Rencana	Aktual	Minus	OT

Gambar V.35 Laporan hasil produksi
(Sumber: Hasil Analisis Data, 2018)

5.6 Implementasi Usulan

Tahap ini adalah tahap pengkodean program menggunakan bahasa pemrograman Java dengan *NetBeans 8.2* aplikasi editor. Setiap *interface* berisikan kode program agar program dapat dijalankan sesuai dengan fungsinya. Untuk mendukung kebutuhan implementasi sistem diperlukan suatu spesifikasi perangkat lunak (*software*) maupun perangkat keras (*hardware*). Adapun spesifikasinya sebagai berikut:

1. Analisis Kebutuhan *Software*
 - a. Sistem Operasi : *Microsoft Windows 7*
 - b. Database Server : *MYSQL 5.6.12*
 - c. Bahasa Pemrograman : *NetBeans 8.2*

2. Analisis Kebutuhan *Hardware*

- a. *Processor* : Minimal *Processor Core i3*
- b. *RAM* : Minimal *RAM 4 GB*
- c. *Harddisk* : Minimal *Harddisk 500 GB*
- d. *Mouse, Keyboard, Monitor* sebagai peralatan antar muka.
- e. *Printer* sebagai media pencetakan dokumen

5.7 *Functional Testing*

Functional testing merupakan pengujian yang dilakukan untuk memeriksa apakah fungsionalitas dari aplikasi yang dibangun sesuai dengan apa yang diharapkan, *functional testing* menguji sistem berdasarkan *use case* yang telah dibuat berikut merupakan pengujian yang dilakukan terhadap aplikasi sistem informasi pembelian laporan hasil Produksi:

1. *Functional Test Use Case Login*

Berikut merupakan *functional test* dari *use case* login:

Tabel V.19 *Functional Test Use Case Login*

No	<i>Test Case Name</i>	<i>Test Case</i>	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Validasi Login	Masuk kedalam sistem menggunakan <i>username</i> dan <i>password</i>	<i>Username</i> = "AgusSR" <i>Password</i> = "1234"	Berhasil masuk	Berhasil masuk	<i>Pass</i>

Tabel V.19 *Functional Test Use Case Login* (lanjutan)

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
2	Validasi Login	Masuk kedalam sistem menggunakan <i>username</i> dan <i>password</i> yang tidak sesuai	<i>Username</i> = "Agus" <i>Password</i> = "produksi"	Tidak berhasil masuk kedalam sistem	Tidak berhasil masuk kedalam sistem	Pass
3	Validasi Login	Tidak mengisi kolom <i>username</i> dan <i>password</i>	<i>Username</i> = " " <i>Password</i> = " "	Menampilkan pesan error kolom harus diisi	Menampilkan pesan error	Pass

(Sumber: Hasil Analisis, 2018)

2. *Functional Test Use Case* mengelola data master

Berikut merupakan *functional test* dari *use case* mengelola data master:

Tabel V.20 *Functional Test Use Case Mengelola Data Master (Produk)*

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Menambah data master produk	Menambah data master dengan mengisi semua kolom pada form	Nama Produk="ErtigaGA" Tipe="P12" Jenis="FR"	Berhasil tambah data	berhasil tambah data	Pass

Tabel V.20 Functional Test Use Case Mengelola Data Master Produk(lanjutan)

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
2	Menambah data master produk	Menambah data master dengan mengosongkan salah satu kolom	Nama Produk="ErtigaG A" Tipe ="" Jenis="FR"	Muncul pesan kolom harus terisi	Muncul pesan kolom harus terisi	Pass
3	Mengubah data master produk	Melakukan perubahan pada atribut master produk	Mengganti Nama Produk dan Tipe Produk	Data Master produk terubah	Data supplier terubah	Pass
4	Menghapus data master produk yang dipilih	Hapus data master produk dengan Id Produk="AS001"	Klik hapus	Menghapus data dari tabel Master produk	Menghapus data dari tabel produk	pass

(Sumber: Hasil Analisis, 2018)

3. *Functional Test Use Case* membuat Rencana produksi harian

Berikut merupakan *functional test* dari *use case* membuat rencana produksi harian:

Tabel V.21 *Functional Test Use Case* Membuat Rencana Produksi Harian

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Membuat rencana produksi	Membuat rencana produksi harian baru dengan	Memasukan data yang sesuai pada kolom Id	Berhasil tambah data	Berhasil tambah data	Pass

	harian baru	mengisi semua kolom pada form rencana	Rencana dan tanggal	rencana produksi	rencana produksi	
2	Membuat rencana produksi harian baru	Mengedit tanggal Rencana Produksi	Mengklik data pada tabel dan ubah tanggal	Berhasil mengubah rencana	Berhasil mengubah tanggal produksi	Pass
3	Membuat rencana produksi harian baru	Menghapus data rencana produksi harian dengan id rencana	Memilih hapus pada <i>button</i>	Permintaan data yang dipilih terhapus dari tabel	Permintaan data yang dipilih terhapus dari tabel	Pass

(Sumber: Hasil Analisis, 2018)

4. *Functional Test Use Case* mencetak Rencana Produksi

Berikut merupakan *functional test* dari *use case* mencetak rencana produksi:

Tabel V.22 *Functional Test Use Case* Mencetak rencana produksi

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Mencetak rencana produksi berdasarkan id_rencana	Melakukan cetak Rencana dengan tombol cetak	Memilih tombol cetak Rencana produksi	Menampilkan Rencana Produksi berformat XPS	Menampilkan Rencana Produksi berformat XPS mengenerate tabel dari database	Pass

(Sumber: Hasil Analisis, 2018)

5. *Functional Test Use Case input hasil produksi*

Berikut merupakan *functional test* dari *use case input* hasil produksi produksi harian:

Tabel V.23 *Functional Test Use Case input* Hasil Produksi

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Menginput hasil produksi harian baru	Menginput hasil produksi dengan mengisi semua kolom pada form hasil produksi	Memasukan data yang sesuai pada kolom hasil produksi	Berhasil tambah data hasil produksi	Berhasil tambah data hasil produksi	Pass
2	Menginput hasil produksi harian baru	Mengedit hasil produksi	Mengklik data pada tabel dan ubah pada <i>form</i> detail	Berhasil mengubah data hasil produksi	Berhasil mengubah data hasil produksi	Pass
3	Menginput hasil produksi harian baru	Menghapus data hasil produksi harian dengan id hasil	Memilih hapus pada <i>button</i>	Permintaan data yang dipilih terhapus dari tabel	Permintaan data yang dipilih terhapus dari tabel	Pass

(Sumber: Hasil Analisis, 2018)

6. *Functional Test Use Case membuat Rencana Overtime*

Berikut merupakan *functional test* dari *use case* membuat rencana *Overtime*:

Tabel V.24 *Functional Test Use Case* Membuat Rencana Overtime

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Membuat rencana Overtime baru	Membuat rencana Overtime baru dengan mengisi semua kolom pada form rencana OT	Memasukan data yang sesuai pada kolom overtime	Berhasil tambah data rencana overtime	Berhasil tambah data rencana overtime	Pass
2	Membuat rencana Overtime baru	Mengedit Rencana Overtime	Mengklik data pada tabel dan ubah pada form	Berhasil mengubah rencana overtime	Berhasil mengubah tanggal produksi	Pass
3	Membuat rencana Overtime baru	Menghapus data rencana overtime dengan id overtime	Memilih hapus pada button	Permintaan data yang dipilih terhapus dari tabel	Permintaan data yang dipilih terhapus dari tabel	Pass

(Sumber: Hasil Analisis, 2018)

7. *Functional Test Use Case* mencetak Rencana overtime

Berikut merupakan *functional test* dari *use case* mencetak rencana produksi:

Tabel V.25 *Functional Test Use Case* Mencetak rencana overtime

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Mencetak rencana OT berdasarkan id OT	Melakukan cetak Rencana OT dengan tombol cetak	Memilih tombol cetak Rencana OT	Menampilkan Rencana Produksi berformat XPS	Menampilkan Rencana Produksi berformat XPS mengenerate tabel dari database	Pass

(Sumber: Hasil Analisis, 2018)

8. *Functional Test Use Case* memvalidasi hasil produksi

Berikut merupakan *functional test* dari *use case* mencetak rencana produksi:

Tabel V.26 *Functional Test Use Case* Mencetak rencana produksi

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Memvalidasi Hasil produksi sesuai berdasar Id rencana	Melakukan validasi dengan pilihan pada combo box validasi	Memilih tombol simpan validasi	Menampilkan hasil produksi yang sudah di validasi	Menampilkan hasil Produksi yang sudah di validasi	Pass

(Sumber: Hasil Analisis, 2018)

9. *Functional Test Use Case* mencetak Laporan Hasil Produksi

Berikut merupakan *functional test* dari *use case* mencetak laporan hasil produksi:

Tabel V.27 *Functional Test Use Case* Mencetak laporan hasil produksi

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Mencetak hasil produksi berdasar kan periode 1 bulan	Melakukan cetak laporan hasil produksi	Memilih tombol cetak laporan hasil produksi	Menampilkan laporan hasil Produksi berformat XPS	Menampilkan laporan hasil Produksi berformat XPS mengenerate tabel dari database	Pass

(Sumber: Hasil Analisis, 2018)

BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Berdasarkan hasil penelitian selama proses pengumpulan dan pengolahan data, analisis sistem, dan implementasi sistem usulan maka dapat diambil kesimpulan sebagai berikut:

1. Dengan membuat sistem informasi laporan hasil produksi yang terkomputerisasi berbasis dekstop, diharapkan dapat mengurangi kehilangan dan kesalahan *penginputan* data hasil produksi.
2. Dengan adanya sistem yang sudah terintegrasi dengan *database* diharapkan akan memudahkan *leader* dalam penyimpanan data hasil produksi, perencanaan *overtime*, dan cepat dalam menginformasikan data hasil produksi yang sesuai dengan rencana produksi yang diberikan.
3. Dengan memberikan fasilitas laporan produksi bulanan ini, diharapkan memudahkan *leader Assembling seat* dalam membaca laporan jumlah rencana, jumlah aktual produksi, jumlah *defect ,reject* produksi.

6.2. Saran

Saran yang diberikan untuk pengembangan sistem informasi perencanaan produksi dan kebutuhan bahan baku ini selanjutnya adalah sebagai berikut:

1. Untuk penerapan sistem baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi penggunaan sistem ini kepada bagian yang terkait.
2. Sebaiknya dilakukan pemeliharaan aplikasi secara berkala, sehingga aplikasi dapat berjalan dengan baik.
3. Selalu melakukan pengembangan pada aplikasi, sehingga aplikasi dapat terus diperbaharui dari segi fitur dan tampilan untuk meningkatkan kinerja kerja bagian terkait.
4. Untuk penelitian selanjutnya sistem informasi ini bisa diintegrasikan dengan bagian PPC.

DAFTAR PUSTAKA

- Anggraeni, Elisabet Yunaeti dan Rita Irviani. 2017. *Pengantar sistem informasi*. Yogyakarta: CV Andi Offset
- Dennis, Alan. 2010. *System Analysis and Design with UML 2.0*. New York: John Wiley and Sons, Inc.
- Eliza. 2012. *Tip dan Trik Merawat Ponsel Berbasis Android*. Jakarta: PT Elex Media Komputindo
- Fahmi, Irham. 2014. *Manajemen Produksi Dan Operasi. Cetakan Kedua*. Bandung: Alfabeta.
- Gaspersz, Vincent. 2005. *Production planning and inventory control* berdasarkan pendekatan sistem terintegrasi MRP II dan JIT menuju manufakturing 21. Cetakan kelima. Jakarta: Gramedia Pusaka Utama
- Hoffer, Jeffrey A., Prescott Mary B., Mcfadden, Fred R. 2010. *Modern Database Management*. New Jersey: Pearson Prentice Hall.
- Hutahean, Jeperson. 2014. *Konsep Sistem Informasi*. Yogyakarta: Deepublish.
- Jogiyanto. 2005. *Sistem Informasi Teknologi Edisi II. Pendekatan Terintegrasi Konsep Dasar Teknologi, Aplikasi Pengembangan dan Pengelolaannya*. Yogyakarta. Kanisius.
- Jubilee Enterprise. 2015. *Mengenal Java dan Database Dengan Netbeans*. Jakarta: PT Elex Media Komputindo.
- Kusmayadi, Ismail. 2008. *Be smart bahasa indonesia kelas VIII SMP/MTs*. Cetakan pertama. Jakarta. PT Multi Printindo Persada
- Kusrini dan Andri. 2007. *Tuntunan praktis membangun sistem informasi akuntansi dengan visual basic dan microsoft sql server*. Edisi I Yogyakarta: Andi

- McLeod, Raymond. 2011. *Sistem Informasi Manajemen*. Jakarta. PT Prenhallindo
- Mulyani, Sri. 2016. *Metode Analisis dan perancangan sistem*. Edisi Kedua Cetakan Ke-1. Bandung: Abdi Sistematika.
- Muslihudin, Muhammad. 2016. *Analisis dan perancangan sistem informasi menggunakan model terstruktur dan UML*. Yogyakarta: Andi
- Narbuko, Cholid dan Achmadi, Abu. 2012. *Metodologi Penelitian*. Jakarta: Bumi Aksara
- Rosa, A.S, Shalahuddin, M. 2016. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Salbino, Sherief. 2014. *Buku Pintar Gadget Android untuk pemula*. Cetakan pertama. Jakarta: Kunci Komunikasi
- Satya Putra, Alfa dan Aritonang, Eva Maulina. 2014. *Beginning Android Programming with ADT Bundle*. Jakarta: PT Elex Media Komputindo
- Sedarmayanti dan Hidayat, Syarifudin. 2011. *Metodologi Penelitian*. Bandung: Mandar Maju
- Semara, I Made Trisna. 2018. *Perencanaan dan perancangan hotel*. Yogyakarta: Deepublish.
- Simarmata, Janner. 2010. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- Sitorus, Lamhot. 2015. *Algoritma dan pemrograman*. Cetakan Pertama. Yogyakarta: Andi.
- Siyoto, Sandu dan sodik, ali. 2015. *Metodologi penelitian*. Cetakan pertama. Yogyakarta: Literasi Media Publishing.
- Suryantara, I Gusti Ngurah. 2015. *Merancang Aplikasi Distributor dengan VB.Net*. Jakarta: PT Elex Media Komputindo

- Sutaji, Deni. 2012. “*Sistem Inventory Mini Market dengan PHP dan JQuery*”. Yogyakarta: Lokomedia
- Welling, L. dan Thomson. 2003. *PHP and MySql Web Development, Second Edition*. Indianapolis : Sams Publishing
2014. *Fitur yang Bermanfaat pada Kursi Jok Mobil*. <http://www.otomotifinfo.com/fitur-kursi-jok-mobil.html> (Tanggal Akses: 18 Mei 2018)
2006. Daftar simbol pada use case diagram. <http://www.repository.widyatama.ac.id/xmlui/bitstream/handle/123456789/4686/Daftar%20Simbol.pdf> (Tanggal Akses: 18 Mei 2018).
2010. *Overtime*. <https://en.wikipedia.org/wiki/Overtime> (Tanggal Akses: 07 Agustus 2018)

LAMPIRAN A

Antar Muka

1. Form Login

Login Aplikasi

**APLIKASI SISTEM INFORMASI
LAPORAN HASIL PRODUKSI SEAT**

Masukkan id pengguna dan kata sandi Anda :)

Id Pengguna

Kata Sandi

Batal Masuk

S

2. Menu Utama

Aplikasi Laporan Hasil Produksi Pada PT Suzuki Indomobil Motor Cikarang Plant

10 September 2018 08:46:08

Selamat Datang Di Aplikasi laporan Hasil Produksi Seat
PT Suzuki Indomobil Motor Cikarang Plant

SUZUKI

3. Form Data Master User

Aplikasi Laporan Hasil Produksi Pada PT Suzuki Indomobil Motor Cikarang Plant

10 September 2018 08:47:03

Data User

Id Pengguna

Nama

Username

Password

Hak Akses

Tambah Ubah Utang Hapus

No	Id Pengguna	Nama	Username	Password	Hak Akses
1	AS-001	Agus Syah R	AgusSR	1234	Leader
2	AS-002	Rohman	HeadRohm	1234	Head

4. Form Data Master Produk

No	ID Produk	Nama Produk	Tipe Produk	Jenis Produk
1	AS001	Eriga GA	YLB	Front (FR)
2	AS002	Eriga GA	YLB	Second (2ND)
3	AS003	Eriga GA	YLB	Thrid (3RD)
4	AS004	Eriga GL	P31	Front (FR)
5	AS005	Eriga GL	P31	Second (2ND)
6	AS006	Eriga GL	P31	Thrid (3RD)
7	AS007	Eriga GX	P08	Front (FR)
8	AS008	Eriga GX	P08	Second (2ND)
9	AS009	Eriga GX	P08	Thrid (3RD)
10	AS010	Eriga Dreza	DRV	Front (FR)
11	AS012	Eriga Dreza	DRV	Second (2ND)
12	AS013	Eriga Dreza	DRV	Thrid (3RD)

5. Form Rencana Produksi

No	ID Rencana	Tanggal	Status
1	ASS01	2018-08-01	Validasi
2	ASS02	2018-08-02	Pending
3	ASS03	2018-08-03	Pending
4	ASS04	2018-08-23	Validasi

6. Form Detail Rencana

No	ID Renc.	Tanggal	ID Produk	Nama Pr.	Jenis Pr.	Tipe Pro.	Jumlah
1	ASS01	2018-08-01	AS001	Eriga GA	YLB	Front	15
2	ASS01	2018-08-01	AS002	Eriga GA	YLB	Second	20
3	ASS01	2018-08-01	AS003	Eriga GA	YLB	Thrid	20
4	ASS01	2018-08-01	AS003	Eriga GA	YLB	Thrid	12
5	ASS01	2018-08-01	AS005	Eriga GL	P31	Second	10
6	ASS03	2018-08-01	AS013	Eriga Dr.	DRV	Thrid	12
7	ASS04	2018-08-01	AS001	Eriga GA	YLB	Front	10
8	ASS04	2018-08-01	AS002	Eriga GA	YLB	Second	15

TOTAL 114

7. Form Input Hasil Produksi

Form Input Hasil Produksi

Id Hasil: Hasil Produksi:

ID Detail: Defect:

ID Rencana: Reject:

Id Produk: Total Hasil:

Jumlah rencana: Minus:

Hitung

Simpan Ubah Ulang Hapus

No	ID Ha.	ID Ra.	ID Pr.	Nama	Jenis	Tipe	Renc.	Hasil	Defect	Reject	Total
1	01	ASS01	Eriga	YL8	Front	24	15	10	3	2	5
2	02	ASS01	Eriga	YL9	Seco	25	20	10	1	1	10

8. Laporan Harian Hasil Produksi

Laporan Hasil Produksi Seat			
PT Suzuki Indomobil Motor Cikarang Plant			
ID_Rencana	ASS01	Total_Rencana	250
Tanggal	8/2/18 12:00 AM	Total_Hasil	210
		OT	Overtime 2 Jam
ID_Produk	Rencana	Hasil	Minus
AS006	100	89	10
AS006	100	800	100
AS009	900	89	10
AS009	900	800	100
AS005	80	89	10
AS005	80	800	100
AS001	100	89	10
AS001	100	800	100
AS002	200	89	10
AS002	200	800	100

Monday 13 August 2018

Page 1 of 1

LAMPIRAN B

Source Code

Login

```
package ta_lulu;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.JOptionPane;

/**
 *
 * @author lulu
 */
public class login extends javax.swing.JFrame {
    Connection con;
    Statement stat;
    ResultSet rs;
    String sql;
    String hakakses;

    public login() {
        initComponents();
        //pemanggilan fungsi koneksi database yang sudah kita buat pada class
        koneksi.java
        Koneksi DB = new Koneksi();
        DB.config();
        con = DB.con;
        stat = DB.stm;
    }
    private void cekuser(){
        Koneksi DB = new Koneksi();
        DB.config();
        con = DB.con;
        stat = DB.stm;
        try{sql = "SELECT hak_akses FROM tb_user WHERE
username='"+t_username.getText()+"' AND password='"+t_pass.getText()+"'";
        rs = stat.executeQuery(sql);
        while (rs.next()){
            hakakses = rs.getString(1);}

            if(hakakses.equals("")){
                JOptionPane.showMessageDialog(this, "ID atau password salah");
            }

            else {
                setmenulogin();
            }

        } catch (Exception e) {
            if(t_username.getText().equals("")){
                JOptionPane.showMessageDialog(this, "Username Tidak Boleh Kosong");
            }
            else if (t_pass.getText().equals("")){

```

```

        JOptionPane.showMessageDialog(this, "password tidak boleh kosong");
    }

    JOptionPane.showMessageDialog(this, "Username atau Password Salah");

}    // TODO add your handling code here:
}
private void setmenulogin(){
    if (hakakses.equals("Admin")){
        MenuUtama a = new MenuUtama(); // memanggil form lain
        a.setVisible(true);
        a.setLocationRelativeTo(null);
        a.MenuDataMaster.setEnabled(true);
        a.MenuTransaksi.setEnabled(true);
        a.MenuValidasi.setEnabled(true);
        a.laporan.setEnabled(true);
        a.setting.setEnabled(true);

        dispose();//menghilangkan form

    }
    else if(hakakses.equals("Leader")){
        MenuUtama a = new MenuUtama(); // memanggil form lain
        a.setVisible(true);
        a.setLocationRelativeTo(null);//lokasi form
        a.MenuDataMaster.setEnabled(false);
        a.MenuTransaksi.setEnabled(true);
        a.MenuValidasi.setEnabled(false);
        a.laporan.setEnabled(true);
        a.setting.setEnabled(true);

        dispose();//menghilangkan form

    }
}

```

CRUD

```

package ta_lulu;
import java.sql.Connection;
import java.sql.SQLException;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

/**
 *
 * @author lulu
 */
public class Formdatauser extends javax.swing.JPanel {
    private DefaultTableModel model;
    public static int statusSearching=0;

    /**
     * Creates new form datauser
     */
    public Formdatauser() {
        initComponents();
        load_table();
        kosong();
    }

    /** CREATE */
    private void btnsimpan_userActionPerformed(java.awt.event.ActionEvent evt) {
        try {
            String sql = "INSERT INTO tb_user VALUES ('"+t_id.getText()+"', '"+t_nama.getText()+"', '"+t_username.getText()+"', '"+t_pass.getText()+"', '"+cmb_akses.getSelectedItem()+"')";
            java.sql.Connection conn;
            conn = (Connection)Koneksi.koneksiDB();
            java.sql.PreparedStatement pst=conn.prepareStatement(sql);
            pst.execute();
            JOptionPane.showMessageDialog(null, "Penyimpanan Data Berhasil");
        } catch (Exception e) {
            JOptionPane.showMessageDialog(this, e.getMessage());
        }
        load_table();
        kosong();
    }

    private void btn_ulanguserActionPerformed(java.awt.event.ActionEvent evt) {
        kosong();
    }

    /** READ */
    private void TableuserMouseClicked(java.awt.event.MouseEvent evt) {
        //menampilkan data kedalam form pengisian:
        int baris = Tableuser.rowAtPoint(evt.getPoint());
        String id_user =Tableuser.getValueAt(baris , 1).toString();
        t_id.setText(id_user);
        String nama =Tableuser.getValueAt(baris , 2).toString();
        t_nama.setText(nama);
        String username =Tableuser.getValueAt(baris , 3).toString();
        t_username.setText(username);
        String password = Tableuser.getValueAt(baris,4).toString();
    }

```

```

        t_pass.setText(password);
        String akses = Tableuser.getValueAt (baris, 5).toString();
        cmb_akses.setSelectedItem(akses);
    }
private void load_table(){
    // membuat tampilan model tabel
    model = new DefaultTableModel();
    model.addColumn("No");
    model.addColumn("Id Pengguna");
    model.addColumn("Nama");
    model.addColumn("Username");
    model.addColumn("Password");
    model.addColumn("Hak Akses");

    //menampilkan data database kedalam tabel
    try {
        int no=1;
        String sql = "select * from tb_user";
        java.sql.Connection conn=(Connection)Koneksi.koneksiDB();
        java.sql.Statement stm=conn.createStatement();
        java.sql.ResultSet res=stm.executeQuery(sql);
        while(res.next()){
            model.addRow(new
Object[]{no++,res.getString(1),res.getString(2),res.getString(3),res.ge
tString(4),res.ge
tString(5)});
        }
        Tableuser.setModel(model);
    } catch (Exception e) {} }

private void kosong(){
    t_id.setText(null);
    t_nama.setText(null);
    t_username.setText(null);
    t_pass.setText(null);

    cmb_akses.setSelectedItem(null);
}

private void btn_ubahuserActionPerformed(java.awt.event.ActionEvent evt) {

/** UPDATE */
    try {
        String sql ="UPDATE tb_user SET id_user = '"+t_id.getText()+"',nama =
 '"+t_nama.getText()+"',username = '"+t_username.getText()+"', password =
 '"+t_pass.getText()+"', hak_akses = '"+ cmb_akses.getSelectedItem()+"' WHERE id_use
='"+t_id.getText()+"'";
        java.sql.Connection conn=(Connection)Koneksi.koneksiDB();
        java.sql.PreparedStatement pst=conn.prepareStatement(sql);
        pst.execute();
        JOptionPane.showMessageDialog(null, "data berhasil di edit");
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, "Perubahan Data Gagal"+e.getMessage());
    }
    load_table();
    kosong();
}

```

```

/** DELETE */
private void btn_hapususerActionPerformed(java.awt.event.ActionEvent evt) {
    //hapus data
    try {
        String sql="delete from tb_user where id_user='"+t_id.getText()+"'";
        java.sql.Connection conn=(Connection)Koneksi.koneksiDB();
        java.sql.PreparedStatement pst=conn.prepareStatement(sql);
        pst.execute();
        JOptionPane.showMessageDialog(this," Data Berhasil Di hapus");
    } catch (Exception e){
        JOptionPane.showMessageDialog(this, e.getMessage());
    }
    load_table();
    kosong(); }

private void textSearchingActionPerformed(java.awt.event.ActionEvent evt) {
    statusSearching=1;
    if(textSearching.getText().isEmpty())
    { statusSearching = 0;}
    else if(statusSearching==1){
        DefaultTableModel Tabelbrg = new DefaultTableModel();
        Tabelbrg.addColumn("ID Pengguna");
        Tabelbrg.addColumn("Nama ");
        Tabelbrg.addColumn("Username");
        Tabelbrg.addColumn("Password");
        Tabelbrg.addColumn("Hak Akses");
        String cari = textSearching.getText();
        try{
            String sql = "Select * From tbuser Where id_user LIKE '"+cari+"'
            + "OR nama LIKE '"+cari+"' OR username LIKE '"+cari+"'
            + "OR password LIKE '"+cari+"'OR hak_akses LIKE '"+cari+"'ORDER BY
id_user";

            java.sql.Connection conn=(Connection)Koneksi.koneksiDB();
            java.sql.Statement stm=conn.createStatement();
            java.sql.ResultSet res=stm.executeQuery(sql);
            while(res.next()){
                Tabelbrg.addRow(new Object[]{
                    res.getString(1),
                    res.getString(2),
                    res.getString(3),
                    res.getString(4),
                    res.getString(5),
                });
                //nomortabel++;
            }
            Tableuser.setModel(Tabelbrg);
        }catch(SQLException ex){
            JOptionPane.showMessageDialog(this, "Data yang dicari tidak ada !!!!");
        }
    } } }

```

Form Source code

```
private DefaultTableModel model1;

    JComboBox jc = new JComboBox();
    JPanel panel = new JPanel();
    Statement st;
    public Formdetail()
    {

        initComponents();

        load_table();
        kosong();
        this.setSize(1200, 600);
        try{
            con = DriverManager.getConnection("jdbc:mysql://localhost/dbproduksi","root","");
            st = con.createStatement();
            String s = "select * from tb_rencana";
            rs = st.executeQuery(s);
            while(rs.next())
            {
                cmb_rencana.addItem(rs.getString(1)+" | "+rs.getString(2));
            }
        }catch(Exception e){
            JOptionPane.showMessageDialog(null, "ERROR");
        }finally{
            try{
                st.close();
                rs.close();
                con.close();
            }catch(Exception e){
                JOptionPane.showMessageDialog(null, "ERROR CLOSE");
            }
        }
        panel.add(jc);
        this.setVisible(true);
    }

    public static void main(String[] args){
        new Formdetail();
    }

    public Formdetail fAB=null;

    public String detail, idprod, namaprod, jenisprod, tipe, jml;

    public String getdet() {
        return detail;
    }
    public String getIdprod() {
        return idprod;
    }
    public String getNamaprod() {
        return namaprod;
    }
}
```

```

    public String getJenisprod() {
        return jenisprod;
    }
    public String getTipe() {
        return tipe;
    }
    public String getjmlah() {
        return jml;
    }

    public final void auto_sum(){
        int total = 0;
        for (int i =0; i< tabeldetail.getRowCount(); i++){
            int amount = Integer.parseInt((String)tabeldetail.getValueAt(i, 5));
            total += amount;
        }
        txt_total.setText(Integer.toString(total));
    }
    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        Produk1 fDB = new Produk1();
        fDB.fAB = this;
        fDB.setVisible(true);
        fDB.setResizable(false);
    }
    public void itemTerpilih(){
        Produk1 fDB = new Produk1();
        fDB.fAB = this;
        t_id_detail_rencana.setText(detail);
        idproduk.setText(idprod);
        nm.setText(namaprod);
        t_tipe.setText(jenisprod);
        jenis.setText(tipe);
        t_jumlah.setText(jml);
    }
    //cetak
    private void cetakActionPerformed(java.awt.event.ActionEvent evt) {
        MessageFormat header = new MessageFormat("Daily Planning Production Assy Seat");
        MessageFormat footer = new MessageFormat("Page{0,number,integer}");

        try{

            tabeldetail.print(JTable.PrintMode.NORMAL, header, footer);

        } catch (java.awt.print.PrinterException e) {
            System.err.format("cannot print", e.getMessage());
        }
    }

```

LAMPIRAN C

Wawancara

1. Proses Bisnis

- Saya : Selamat Pagi pak,
- Pak Rohman : Iya Pagi,
- Saya : Mohon maaf sebelumnya pak, saya ingin wawancara dengan bapak perihal alur proses bisnis yang ada di bagian produksi ini pak, apakah bisa pak?
- Pak Rohman : Oh ya bisa, silahkan
- Saya : Saya ingin tanya pak untuk alur proses produksi ini bagaimana ya pak ?
- Pak Rohman : Oh, ya untuk prosesnya itu ada proses *cutting*, *sewing*, *headrest*, *urethane*, *welding*, *Assembling seat*.
- Saya : Boleh dijelaskan pak untuk masing masing proses tersebut?
- Pak Rohman : Ya,
Cutting adalah proses untuk pemotongan bahan baku materialnya yaitu trim atau biasa juga disebut dengan kulit untuk *seat*,
Sewing adalah proses menjahit dimana potongan bahan tersebut dijahit untuk jadi sarung kursinya tersebut.
Welding yaitu proses penggabungan komponen kursinya atau kamu bisa lihat di *welding* ada besi besi yang belum digabung jadi satu masih berupa potongan, nah di proses *welding* itu nantinya semua besi atau komponen tadi digabung
Headrest itu pembuatan senderan kepala pada kursi, kulitnya dari proses yang sudah dijahit pada proses *sewing*.
Urethane, yaitu proses pembuatan pad/ busa untuk kursinya, ini adalah campuran bahan kimia *polymer* dan *ISO*, gitu
- Saya : Lalu untuk penggabungannya sampai jadi kursi ada di proses apa ya pak?
- Pak Rohman : Ada di *line* proses *Assembling Seat*
- Saya : Maaf pak, saya ingin tau lebih lanjut pada line proses tersebut pak untuk alur proses bisnisnya bisa pak?

- Pak Rohman : Oh jadi disini masing masing ada *leadernya*, jadi untuk keterangan lebih lanjut apa yang kamu mau tau kamu bisa tanya langsung ke *leadernya* aja ya. Untuk *Assembling seat* itu dengan bapak Agus Syah. Untuk *leader* yang lain jika ingin kamu tanyakan bisa tanya dengan pak Agus juga karna dia juga yang mengawasi seluruhnya di bagian bawah. Kamu langsung turun ke *genba*. Nanti saya yang konfirmasi
- Saya : Oh begitu pak, baik pak terimakasih pak atas informasinya. Selamat pagi pak
- Pak Rohman : Iya sama sama, pagi juga .

2. Penjelasan lebih lanjut tentang pendataan laporan hasil produksi

- Saya : Selamat pagi pak,
- Pak Agus : Iya pagi juga
- Saya : Mohon maaf sebelumnya pak, Saya ingin wawancara dengan bapak perihal alur proses produksi dan alur proses bisnis produksi pada divisi ini apakah bisa pak?
- Pak Agus : Oh ya silahkan..
- Saya : Baik pak terimakasih, saya ingin bertanya pak, tugas bapak untuk divisi *Assembling seat* ini apa saja ya pak?
- Pak Agus : Untuk tugas saya di *Assembling seat* ini adalah sebagai *leader* untuk proses *Assembling seat*, selain itu saya juga bertugas untuk mengurus data laporan hasil produksi yang isinya rencana produksi, hasil produksi, data *reject*, dan *defect*.
- Saya : Mohon maaf pak, kalo boleh tau alur proses bisnis nya bagaimana ya pak ?
- Pak Agus : Boleh, alur proses bisnis apa yang ingin kamu tanyakan ?
- Saya : Untuk alur proses bisnis, tentang laporan hasil produksi pak.
- Pak Agus : Baik, saya akan jelaskan
Yang pertama, Saya menerima Rencana Produksi untuk sebulan dari PPC, kemudian rencana produksi harian tersebut saya *input* ke *microsoft excel* lalu di cetak dan di validasi oleh saya, Pak Rohman, pak Sriyana, dan Mr. Oishi.

- Saya : Validasi nya untuk apa yah pak dan boleh saya tau pak dokumennya seperti apa?
- Pak Agus : Validasi untuk meng ACC bahwa rencana nya sudah sesuai dengan target per harinya. Untuk dokumen boleh namun ini bersifat rahasia jadi jika untuk kamu masukkan ke dalam skripsi kamu nanti akan saya berikan format dalam bentuk kertas dan bisa kamu foto.
- Saya : Baik pak terimakasih.
Untuk rencana kalau tidak sesuai bagaimana pak ? Apakah bapak membuat lagi ?
- Pak Agus : Oh, tidak . Kalau disini kan sudah ada perhitungan untuk Efisiensi jumlah produksi dalam sehari, jadi saya buat rencana produksi nya juga sesuai, begitu.
- Saya : Baik pak, tapi bagaimana pak ketika jumlah produksinya meningkat ? pastinya akan mempengaruhi pak .
- Pak Agus : Oh ya, walaupun rencana nya lebih banyak akan ada *overtime*
- Saya : *Overtime* nya bagaimana pak mohon penjelasannya?
- Pak Agus : *Overtime* kita lihat dari hasil produksinya, kan kalau waktu udh mau jam pulang kamu lihat kan kita ada *briefing*, nah diwaktu tersebut akan di bahas jumlah hasil produksi hari ini, jika belum mencapai target maka baru di rencanakan *overtime*.
- Saya : Oh begitu pak, Selanjutnya pak boleh dijelaskan kembali.
- Pak Agus : Ya, setelah saya menginput Rencana Produksi Harian tersebut ke dalam *Microsoft Excel*, kemudian saya cetak untuk bagian *Assembling seat*
Lalu rencana tersebut diberikan kepada operator sebagai acuan untuk melakukan proses produksi.
Nah, setelah operator selesai melakukan proses produksi, selanjutnya mereka akan mengisi tuh lembar kertas *Daily report*nya. Lalu diberikan ke saya untuk saya cek dan validasi.
- Saya : Oh begitu pak, lalu *daily report* tersebut sebagai laporan harian atau di inputkan ke sistem pak?
- Pak Agus : Laporan tersebut saya bawa ke *office* untuk nantinya di validasi oleh bapak Rohman selaku *Head Assy* apabila sudah sesuai dengan

perencanaan. Namun jika belum sesuai maka belum di validasi dan akan menyuruh *leader* ketika *briefing* untuk merencanakan *overtime*.

- Saya : Selanjutnya setelah melakukan *overtime* bagaimana pak?
- Pak Agus : *Head Assy* memvalidasi dengan menandatangani *daily report* tersebut. Kemudian *daily report* tersebut di *input* oleh *Head Assy* untuk di rekap menjadi laporan bulanan.

3. Mempertanyakan permasalahan

- Saya : Oh begitu pak, sebelumnya terimakasih pak untuk penjelasannya. Untuk pencatatan dokumennya bapak menggunakan apa ya pak?
- Pak Agus : Menggunakan Daily Report, lalu di berikan ke *head assy* untuk di rekap
- Saya : Untuk penyimpanan data nya itu bagaimana pak ?
- Pak Agus : Dengan microsoft excel, dan kalau sudah selesai nanti di cetak dan di simpan sebagai arsip
- Saya : Sekarang saya ingin tanyakan pak untuk permasalahan atau kendala dalam proses pembuatan laporan hasil produksi nya pak, apakah ada kendala atau permasalahan yang mempengaruhi dalam pelaporan hasil produksi pak?
- Pak Agus : Tentunya ada.
- Saya : Boleh dijelaskan pak Apa saja?
- Pak Agus : Baik, data yang sudah diisi oleh operator tadi setelah saya validasi kemudian saya bawa ke *Head Assy*, untuk di validasi atau terkadang Pak Rohman turun sebelum jam *briefing* untuk melihat hasil produksinya. Atau dia telepon ke saya nanya gimana hasil produksinya ?gitu. Jadi itu memakan waktu lama ketika saya harus membawa dokumen tersebut ke atas.
- Saya : Selain itu pak?
- Pak Agus : Nah selain itu data *daily report* nya ga langsung diinput pada hari tersebut,kadang besoknya baru diinput. Jadi kadang data nya hilang dan *backup*annya ke saya. Karena saya juga mendata perhari nya. Jadi kadang agak lama buat rekap laporannya.

- Saya : Oh begitu yah pak, baik pak terimakasih banyak atas informasinya yah pak.
- Pak Agus : Oh iya sama sama, kalau ada yang ingin ditanyakan lebih lanjut lagi bisa hubungi saya ya nanti saya arahkan kamu.
- Saya : Baik pak terimakasih banyak pak. Selamat pagi
- Pak Agus : Iya pagi juga

LAMPIRAN D

Dokumen

1. Dokumen Rencana Produksi Harian

Daily Production Planning		Note		holidays		17-Aug		Hari Kemerdekaan		PREPARED		CHECKED		APPROVED																					
Sat Production		Aug-17								Ayus SR		Rohman		Sivana		T.Oishi																			
Issued Date : 18 July 2017																																			
Product	Type Product	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Total
GA (P12)	FR			15	20	11	16	0	0	21	10	14	18	15	0	0	20	12	23	15	11	0	0	14	23	16	21	17	0	0	10	12	23	20	377
	2ND			20	12	20	12	0	0	20	12	20	12	20	0	0	20	12	20	12	20	0	0	0	12	20	12	20	0	0	20	12	20	12	380
	3RD			14	17	14	17	0	0	14	17	14	17	14	0	0	14	17	14	17	14	0	0	14	17	14	17	14	0	0	14	17	14	17	352
GL (P31)	FR			15	10	17	10	0	0	12	15	15	10	15	0	0	15	10	15	10	15	0	0	15	19	15	10	15	0	0	15	10	15	10	417
	2ND			16	23	16	21	0	0	16	23	16	23	17	0	0	16	20	16	23	16	0	0	12	13	16	23	16	0	0	16	23	16	20	428
	3RD			20	16	20	16	0	0	20	16	20	16	20	0	0	20	16	20	16	21	0	0	20	16	20	16	21	0	0	20	16	20	16	350
GX (P08)	FR			12	20	12	17	0	0	12	20	12	20	12	0	0	11	20	12	20	12	0	0	17	20	15	20	12	0	0	12	10	12	20	288
	2ND			10	11	20	20	0	0	10	11	10	16	20	0	0	10	12	10	11	19	0	0	12	11	11	11	11	0	0	10	11	10	11	399
	3RD			16	18	16	18	0	0	16	18	25	18	16	0	0	16	11	15	18	16	0	0	16	18	16	18	16	0	0	16	18	15	18	445
DREZA (DRV)	FR			18	21	18	21	0	0	18	21	18	21	18	0	0	18	27	18	21	15	0	0	18	21	18	21	18	0	0	18	21	18	21	445
	2ND			25	28	25	28	0	0	25	28	25	28	16	0	0	25	28	25	28	25	0	0	25	10	25	28	25	0	0	25	28	25	28	578
	3RD			21	12	21	12	0	0	21	12	21	12	21	0	0	21	12	21	12	11	0	0	21	12	21	12	21	0	0	21	12	21	12	383
Total			202	208	210	208	0	0	205	209	210	211	204	0	0	206	197	209	203	195	0	0	204	192	207	209	206	0	0	197	190	208	205	4451	

2. Dokumen *Daily Report*

Daily Report Assy Seat

Hari/Tanggal : _____

Waktu Normal (menit) : _____
(senin-kamis 480 menit ; jum'at 430 menit)

Jml MP : _____ Tidak Hadir : _____ Nama : _____

PREPARED	APPROVED
Leader	Head Assy

PLAN PRODUCTION

	P12	P31	P08	DRV
total				

FR	QTY (pcs)	
	PLAN	PROD
P12		
P31		
P08		
DRV		
SUBTOTAL		

2ND	QTY (pcs)	
	PLAN	PROD
P12		
P31		
P08		
DRV		
SUBTOTAL		

3RD	QTY (pcs)	
	PLAN	PROD
P12		
P31		
P08		
DRV		
SUBTOTAL		

Deffect	FR	2ND	3RD	QTY
TOTAL				

QTY Reject	FR	2ND	3RD	QTY
TOTAL				

NOTES :

3. Contoh Laporan Hasil Produksi

REPORT ASSY SEAT

PT Suzuki Indomobil Motor Cikarang Plant

ISSUE DATE31 Agst 2017

NAMERohman

ProcessAssembling seat

NOTE

PROD PERIOD:Agst 2017

HOLIDAY:17 - Kemerdekaan

Day	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Total
Plan Prod			202	208	210	208	0	0	205	209	210	211	204	0	0	206	197	209	203	195	0	0	204	192	207	209	206	0	0	197	190	208	205	4493
Actual Prod			190	208	194	208	0	0	205	189	205	193	190	0	0	185	197	209	198	195	0	0	190	185	207	209	195	0	0	182	190	190	180	4304
±			-12	0	-16	0	0	0	0	-20	-5	-18	-14	0	0	-21	0	0	-5	0	0	-14	-7	0	0	0	-11	0	-15	0	-18	-25	-201	
OT			1	0	1	0	0	0	0	1	1	1	0	0	0	1	0	0	1	0	0	1	1	0	0	1	0	1	0	1	0	1	2	14

APPROVED BY:

Head Assy Seat