

**RANCANG BANGUN SISTEM INFORMASI PENERIMAAN
KOMPONEN PADA SEKSI ASSEMBLING 4W BERBASIS WEB
MENGUNAKAN PHP 5.5 DAN MYSQL 5.6
PT SUZUKI INDOMOBIL MOTOR**

TUGAS AKHIR

Untuk Memenuhi Sebagian Syarat Penyelesaian Jenjang Diploma
Empat (D-4) Program Studi Sistem Infomasi pada Politeknik STMI Jakarta

**OLEH
APRIAN AZHARI
(1311022)**



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

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PENERIMAAN KOMPONEN PADA SEKSI
ASSEMBLING 4W BERBASIS WEB
MENGUNAKAN PHP 5.5 DAN MYSQL 5.6 PT
SUZUKI INDOMOBIL MOTOR

Disusun Oleh :
Nama : Aprian Azhari
Nim : 1311022
Program Studi : Sistem Informasi

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia pada hari Rabu tanggal 10 November 2016

Jakarta, November 2016

Dosen Pembimbing

Dosen Penguji

Drs. Jacob Saragih, MM
NIP . 195404281986031002

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

Dosen Penguji

Dosen Penguji

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

Dedy Trisanto, S.Kom, MMSI
NIP . 197805052005021002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PENERIMAAN KOMPONEN PADA SEKSI
ASSEMBLING 4W BERBASIS WEB
MENGUNAKAN PHP 5.5 DAN MYSQL 5.6 PT
SUZUKI INDOMOBIL MOTOR

Disusun Oleh :
Nama : Aprian Azhari
Nim : 1311022
Program Studi : Sistem Informasi
Tanggal Seminar : Jumat, 24 Oktober 2016
Tanggal Sidang : Rabu, 10 November 2016
Tanggal Lulus : Rabu, 10 November 2016

Jakarta, 15 November 2016

Menyetujui
Dosen Pembimbing

Drs Jacob Saragih MM
NIP . 195404281986031002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PENERIMAAN KOMPONEN PADA SEKSI
ASSEMBLING 4W BERBASIS WEB
MENGUNAKAN PHP 5.5 DAN MYSQL 5.6 PT
SUZUKI INDOMOBIL MOTOR

Disusun Oleh :
Nama : Aprian Azhari
Nim : 1311022
Program Studi : Sistem Informasi
Tanggal Seminar : Jumat, 24 Oktober 2016
Tanggal Sidang : Rabu, 10 November 2016
Tanggal Lulus : Rabu, 10 November 2016

Jakarta, 15 November 2016

Menyetujui
Asisten Pembimbing

Ahmad Juniar, Skom, MT
NIP . 197906052006041002

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Aprian Azhari
 NIM : 1311022
 Judul TA : Rancang Bangun Sistem Informasi Penerimaan Komponen Pada Seksi
 Assembling 4W Berbasis Web Menggunakan PHP 5.5 dan MYSQL 5.6
 PT Suzuki Indomobil Motor
 Pembimbing : Drs. Jacob Saragih, MM
 Asisten Pembimbing : Ahmad Juniar, Skom, MT

Tanggal	BAB	Keterangan	Paraf
02/05/2016	BAB I	Kesalahan Penulisan Latar Belakang	
23/05/2016	BAB I	Kesalahan Penulisan Sistem Penulisan	
06/06/2016	BAB I	Perbaikan Penulisan BAB I	
20/06/2016	BAB II	Kesalahan Penulisan BAB II	
06/07/2016	BAB II	Kesalahan Penulisan BAB III	
18/07/2016	BAB II Dan III	Perbaikan Penulisan BAB II dan BAB III	
08/08/2016	BAB IV Dan V	Kesalahan BAB V	
09/09/2016	BAB V Dan VI	Perbaikan BAB V	
10/10/2016	BAB VI	Kesalahan BAB VI	
12/10/2016	BAB VI	Perbaikan BAB VI	
17/10/2016	Keseluruhan Laporan	Abstrak, Daftar Pustaka, Daftar isi Daftar Tabel, Daftar Gambar	

Mengetahui,
 Ka Prodi
 Sistem Informasi Industri Otomotif

Dosen Pembimbing

Drs Jacob Saragih MM
 NIP : 195404281986031002

Drs Jacob Saragih MM
 NIP : 195404281986031002

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Aprian Azhari
 NIM : 1311022
 Judul TA : Rancang Bangun Sistem Informasi Penerimaan Komponen Pada Seksi
 Assembling 4W Berbasis Web Menggunakan PHP 5.5 dan MYSQL 5.6
 PT Suzuki Indomobil Motor
 Pembimbing : Drs. Jacob Saragih, MM
 Asisten Pembimbing : Ahmad Juniar, Skom, MT

Tanggal	BAB	Keterangan	Paraf
19/04/2016	BAB I	Kesalahan Penulisan BAB I	
20/04/2016	BAB II dan III	Kesalahan Penulisan BAB II	
28/04/2016	BAB I,II	Perbaikan Penulisan BAB I, II	
29/04/2016	BAB IV	Kesalahan Penulisan BAB IV	
03/05/2016	BAB IV	Perbaikan Penulisan BAB IV	
15/08/2016	BAB V	Kesalahan Penulisan BAB V	
13/09/2016	BAB VI	Kesalahann Penulisan BAB VI	
14/09/2016	BAB V	Perbaikkan Use case dan Activity BAB V	
15/09/2016	BAB VI	Perbaikan Penulisan BAB VI	
17/09/2016	BAB V	Perbaikan Sequence Diagram, Class Diagram BAB V	
18/09/2016	Keseluruhan Laporan	Perbaikan BAB V, Pengajuan BAB VI Abstrak, Daftar Pustaka, Daftar isi Daftar Tabel, Daftar Gambar	

Mengetahui,
 Ka Prodi
 Sistem Informasi Industri Otomotif

Asisten Pembimbing

Drs Jacob Saragih MM
 NIP : 195404281986031002

Ahmad Juniar, Skom, MT
 NIP : 197906052006041002

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Aprian Azhari

Nim : 1311022

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

“Rancang Bangun Sistem Informasi Penerimaan Komponen pada Seksi Assembling 4W Berbasis Web Menggunakan PHP 5.5 dan MYSQL 5.6 PT Suzuki Indomobil Motor”.

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

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

Jakarta, November 2016

Aprian Azhari

ABSTRAK

PT Suzuki Indomobil motor adalah perusahaan manufaktur yang bergerak di bidang otomotif. Salah satu departemen yang menangani proses produksi yaitu departemen produksi, sedangkan seksi *assembling* 4W yang bertugas untuk merakit setiap komponen *engine* mobil. Komponen merupakan faktor terbentuknya sebuah *engine* yang terdiri dari banyak jumlahnya yang memiliki fungsi berbeda untuk setiap komponen. Saat ini sistem penerimaan komponen yang berjalan masih dilakukan secara manual. Sistem pengolahan data komponen masih menggunakan formulir bon permintaan barang serta melakukan penyimpanan dan pengolahan data di dalam *Microsoft Excel*. Serta tidak adanya aplikasi yang terintegrasi dengan database sebagai penyimpanan data mengakibatkan terjadinya kehilangan data komponen *engine* hal ini menyebabkan lamanya penambahan dan penggantian komponen Untuk mengatasi permasalahan tersebut maka perlu dibuat sebuah sistem informasi penerimaan komponen yang dapat memberikan informasi dengan lebih efisien dan aktual. Metodologi pengembangan sistem informasi yang digunakan pada penelitian ini adalah *evolutionary prototype*. Untuk membuat sistem tersebut maka diperlukan analisis dan perancangan sistem dengan pemodelan sistem menggunakan *unified modelling language* (UML). Sistem informasi penerimaan komponen dibangun dengan menggunakan PHP 5.5 sebagai bahasa pemrograman dan MYSQL 5.6 sebagai basis datanya. Sistem informasi penerimaan komponen ini nantinya dapat memudahkan pencatatan data komponen karena sudah terkomputerisasi sehingga laporan penerimaan dapat langsung diolah secara otomatis. Dalam penerapan sistem yang lama ke sistem yang baru sebaiknya dilakukan secara bertahap serta Perlunya pemeliharaan terhadap sistem yang telah dibuat.

Kata kunci: Sistem Informasi, *Assembling Engine* 4W, UML, PHP 5.5, MYSQL 5.6.

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas limpahan nikmat, rahmat serta karunia yang senantiasa diberikan sehingga penulis dapat menyelesaikan Laporan Tugas Akhir dengan judul “RANCANG BANGUN SISTEM PENERIMAAN KOMPONEN *ASSEMBLING* 4W BERBASIS WEB MENGGUNAKAN PHP 5.5 DAN MYSQL 5.6 PT. SUZUKI INDOMOBIL MOTOR”. Laporan Tugas Akhir ini diajukan untuk memenuhi salah satu syarat penyelesaian Program Studi D-IV Sistem Informasi pada Politeknik STMI Jakarts.

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

1. Orang tua dan keluarga tercinta yang telah memberikan doa, dukungan, pengorbanan, semangat dan kasih sayang hingga saat ini.
2. Dr. Mustofa, ST, MT selaku Direktur Politeknik STMI Jakarta.
3. Bapak Drs. Jacob Saragih, MM, selaku Ketua Program Studi Sistem Informasi dan dosen Pembimbing.
4. Ahmad Juniar, Skom, MT, selaku asisten dosen Pembimbing yang telah membantu penyelesaian Laporan Tugas Akhir.
5. Bapak Agus, selaku Divisi HRD yang telah mengorganisir jalannya Kerja Lapangan di PT Suzuki Indomobil Motor.
6. Bapak Tarmidi, selaku pembimbing di PT Suzuki Indomobil Motor yang telah membimbing dalam melaksanakan Kerja Lapangan.
7. Seluruh dosen POLITEKNIK STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
8. Kepada John Yansin dan Shendy Fajar yang sudah menemani, memotivasi, membantu dalam penyelesaian Tugas Akhir ini.

9. Rekan-rekan mahasiswa Sekolah Tinggi Manajemen Industri Program Studi Sistem Informasi Angkatan 2011 atas kebersamaan selama ini.

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

Jakarta, November 2016

Penulis

DAFTAR ISI

	Halaman
Halaman Judul	i
Lembar Persetujuan Dosen Pembimbing	ii
Lembar Pengesahan.....	iv
Lembar Bimbingan Tugas Akhir	v
Lembar Pernyataan Keaslian	vii
Abstrak	viii
Kata Pengantar	ix
Daftar Isi	x
Daftar Gambar	xiii
Daftar Tabel	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah.....	3
1.5 Manfaat Penelitian	3
1.6 Sistematika Penulisan	3
BAB II LANDASAN TEORI	6
2.1 Pengertian Rancang Bangun	6
2.2 Konsep Dasar Sistem	6
2.2.1 Karakteristik Sistem.....	8
2.3 Konsep Dasar Informasi	10
2.3.1 Siklus Informai	10
2.3.2 Kualitas Informasi.....	11

2.4	Konsep Dasar Sistem Informasi.....	12
2.4.1	Komponen Sistem Informasi	12
2.5	Siklus Hidup Pengembangan Sistem	14
2.6	Metodologi Pengembangan Sistem.....	15
2.7	Perencanaan untuk penerimaan.....	17
2.8	Pengertian Komponen.....	18
2.9	Bagan Alir (<i>Flowchart</i>).....	19
2.10	<i>Unified Modeling Language</i> (UML).....	21
2.10.1	<i>Use Case Diagram</i>	22
2.10.2	<i>Activity Diagram</i>	23
2.10.3	<i>Sequence Diagram</i>	24
2.10.4	<i>Class Diagram</i>	25
2.10.5	<i>Component Diagram</i>	26
2.10.6	<i>Deployment Diagram</i>	27
2.11	Kamus Data dan <i>Database</i>	28
2.12	HIPO (<i>Hierarchy plus Input Process Output</i>)	28
2.13	PHP	30
2.14	MySQL.....	31
BAB III	METODOLOGI PENELITIAN	33
3.1	Metodologi Penelitian	33
3.2	Jenis dan Sumber Data	33
3.3	Pengumpulan Data	34
3.4	Metode Pengembangan Sistem	34
3.5	Kerangka Penelitian	35
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA	38
4.1	Sejarah Perusahaan.....	38
4.1.1	Program Perusahaan.....	38
4.1.2	Visi dan Misi Perusahaan.....	39

4.1.3	Struktur Organisasi Perusahaan.....	39
4.1.4	Jaringan Distribusi.....	42
4.2	Proses <i>Assembly Engine</i>	43
4.3	Sistem Informasi yang Berjalan	45
4.3.1	Arus Dokumen.....	45
4.3.2	Alur Proses.....	47
4.3.3	Diagram Use Case Sistem Informasi yang Berjalan... ..	48
BAB V	ANALISIS DAN PEMBAHASAN	53
5.1	Analisis Sistem Informasi Berjalan.....	53
5.2	Analisis Kebutuhan Sistem.....	53
5.2.1	<i>Use Case Diagram</i> Yang Diusulkan	56
5.2.2	<i>Activity Diagram</i> Yang Diusulkan	60
5.2.3	<i>Sequence Diagram</i> Yang Diusulkan	67
5.2.4	<i>Class Diagram</i> Yang Diusulkan	72
5.2.5	Kamus Data Yang Diusulkan.....	73
5.2.6	<i>Deployment Diagram</i> Yang Diusulkan	76
5.2.7	<i>Hierarchy Plus Input Process Output (HIPO)</i>	77
5.2.8	<i>Flowchart</i> Sistem Informasi Pengendalian Komponen	77
5.2.9	Perancangan <i>Interface</i>	79
5.3	Spesifikasi Perangkat Lunak (<i>Software</i>).....	84
5.4	Spesifikasi Perangkat Keras (<i>Hardware</i>).....	84
BAB VI	KESIMPULAN DAN SARAN	85
6.1	Kesimpulan	85
6.2	Saran.....	85
	DAFTAR PUSTAKA	86
	LAMPIRAN	L-1

DAFTAR GAMBAR

Gambar II.1	Siklus Informasi.....	11
Gambar II.2	Blok Sistem Informasi Yang Berinteraksi.....	13
Gambar II.3	Langkah Pengembangan <i>Prototype Evolutionary</i>	16
Gambar II.4	Filsafat tata ruang yang luas	18
Gambar II.5	Klasifikasi <i>Diagram UML</i>	21
Gambar II.6	<i>Visual Table of Contents</i>	29
Gambar II.7	<i>Overview Diagram</i>	30
Gambar III.1	Kerangka Penelitian	37
Gambar IV.1	Struktur Organisasi PT SIM.....	40
Gambar IV.2	Struktur Organisasi PT SIM <i>Plant Cakung</i>	41
Gambar IV.3	Struktur Organisasi divisi Seksi <i>Assembling 4W</i>	41
Gambar IV.4	Jaringan Distribusi Suzuki Group	42
Gambar IV.5	Hasil <i>Assy Counter Line</i> PT Suzuki Indomobil Motor	43
Gambar IV.6	Hasil <i>Assy Main Line 1</i> PT Suzuki Indomobil Motor.....	44
Gambar IV.7	Hasil <i>Assy Main Line 2</i> PT Suzuki Indomobil Motor	44
Gambar IV.8	Laporan kerusakan dan Bon Permintaan Barang	45
Gambar IV.9	Laporan permintaan komponen.....	46
Gambar IV.10	Laporan penerimaan Komponen	46
Gambar IV.11	<i>Flowmap</i> Sistem yang berjalan	48
Gambar IV.12	<i>Use Case Diagram</i> Sistem Informasi Penerimaan Komponen <i>Assembling 4W</i>	49
Gambar V.1	<i>Flowmap</i> Sistem Informasi Penerimaan Komponen <i>Assembly</i> <i>Engine 4W</i>	55
Gambar V.2	<i>Use Case Diagram</i> Usulan	56
Gambar V.3	<i>Activity Diagram Login</i>	61
Gambar V.4	<i>Activity Diagram</i> Mengelola Data Master.....	62
Gambar V.5	<i>Activity Diagram</i> input form permintaan komponen	63
Gambar V.6	<i>Activity Diagram</i> form Penerimaan komponen.....	64
Gambar V.7	<i>Activity Diagram</i> Melihat data komponen	65

Gambar V.8	<i>Activity Diagram</i> Input form permintaan barang	66
Gambar V.9	<i>Activity Diagram</i> Mengecek stok barang.....	67
Gambar V.10	<i>Sequence Diagram</i> Proses Login.....	68
Gambar V.11	<i>Sequence Diagram</i> Mengelola Data Master.....	68
Gambar V.12	<i>Sequence Diagram</i> Permintaan Komponen	69
Gambar V.13	<i>Sequence Diagram</i> Membuat Penerimaan Komponen	70
Gambar V.14	<i>Sequence Diagram</i> Melihat Penerimaan Komponen	70
Gambar V.15	<i>Sequence Diagram</i> Permintaan Barang.....	71
Gambar V.16	<i>Sequence Diagram</i> Mengecek Stok	72
Gambar V.17	<i>Class Diagram</i> Sistem Informasi Penerimaan Komponen	72
Gambar V.18	<i>Deployment Diagram</i> Yang Diusulkan	76
Gambar V.19	HIPO Yang Diusulkan.....	77
Gambar V.20	<i>Flowchart</i> Yang Diusulkan	78
Gambar V.21	Perancangan <i>Form Login</i>	79
Gambar V.22	Perancangan Halaman Utama	79
Gambar V.23	Perancangan <i>Form Data Master</i> Komponen	80
Gambar V.24	Perancangan <i>Form Data Master Vendor</i>	80
Gambar V.25	Perancangan <i>Form Data Master Line</i> produksi.....	81
Gambar V.26	Perancangan <i>Form Data Master User</i>	81
Gambar V.27	Perancangan <i>Form Permintaan</i> Komponen.....	82
Gambar V.28	Perancangan <i>Form Penerimaan</i> Komponen	82
Gambar V.29	Perancangan <i>Form Permintaan</i> Barang.....	83
Gambar V.30	Perancangan <i>Form Daftar Penerimaan</i> Komponen.....	83

DAFTAR TABEL

Tabel II.1	Simbol-Simbol <i>Flowchart</i>	20
Tabel II.2	Simbol-Simbol <i>Use Case Diagram</i>	22
Tabel II.3	Simbol-simbol <i>Activity Diagram</i>	23
Tabel II.4	Simbol-simbol <i>Sequence Diagram</i>	24
Tabel II.5	Simbol-simbol <i>Class Diagram</i>	25
Tabel II.6	Simbol-simbol <i>Component Diagram</i>	26
Tabel II.7	Simbol-simbol <i>Deployment Diagram</i>	27
Tabel II.8	Beberapa Tipe Data Pada MySQL	32
Tabel IV.1	Definisi Aktor <i>Use Case Diagram</i>	49
Tabel IV.2	Definisi <i>Use Case</i> Membuat Rencana Produksi	50
Tabel IV.3	Definisi <i>Use Case</i> Membuat Rencana Permintaan Komponen	50
Tabel IV.4	Definisi <i>Use Case</i> Melakukan Proses Produksi.....	50
Tabel IV.5	Definisi <i>Use Case</i> Melakukan Pengecekan produksi	51
Tabel IV.6	Definisi <i>Use Case</i> Membuat <i>Form</i> Permintaan Komponen	51
Tabel IV.7	Definisi <i>Use Case</i> Membuat laporan penerimaan komponen	51
Tabel IV.8	Definisi <i>Use Case</i> Memngirim komponen	52
Tabel IV.9	Definisi <i>Use Case</i> Membuat form bon permintaan barang	52
Tabel V.1	Kebutuhan Sistem Informasi Pelatihan Karyawan	53
Tabel V.2	Definisi Aktor <i>Use Case Diagram</i> Sistem Informasi Usulan	57
Tabel V.3	<i>Use Case Diagram</i> Login Aktor.....	57
Tabel V.4	<i>Use Case Diagram</i> Mengelola Data Master	57
Tabel V.5	<i>Use Case</i> input form permintaan komponen	58

Tabel V.6 <i>Use Case Diagram input form penerimaan komponen</i>	58
Tabel V.7 <i>Use Case Diagram Melihat data komponen</i>	59
Tabel V.8 <i>Use Case Diagram Input form permintaan barang</i>	59
Tabel V.9 <i>Use Case Diagram Mengecek Stok komponen</i>	60
Tabel V.10 Spesifikasi Tabel <i>Users</i>	73
Tabel V.11 Spesifikasi Tabel <i>mst_komponen</i>	73
Tabel V.12 Spesifikasi Tabel <i>mst_lini produksi</i>	74
Tabel V.13 Spesifikasi Tabel <i>mst_vendor</i>	74
Tabel V.14 Spesifikasi Tabel <i>permintaan</i>	75
Tabel V.15 Spesifikasi Tabel <i>penerimaan</i>	75
Tabel V.16 Spesifikasi Tabel <i>permintaan_barang</i>	76
Tabel V.17 Spesifikasi Perangkat Keras	84

BAB I

PENDAHULUAN

1.1 Latar Belakang

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

Sistem informasi dibangun untuk mempermudah penyajian informasi kepada pemakainya. Sistem informasi banyak dimanfaatkan oleh perusahaan-perusahaan industri, baik manufaktur maupun jasa. Sistem informasi digunakan sebagai alat untuk pendukung pengambilan keputusan dalam membantu meningkatkan keakuratan penyediaan komponen.

Komponen merupakan bahan setengah jadi yang selanjutnya diproses menjadi bahan jadi yaitu *engine*. *Engine* sendiri terdiri dari banyak komponen yang dirakit sesuai dengan spesifikasi yang telah ditentukan. Hal ini mengharuskan perusahaan harus mengetahui jumlah komponen yang dibutuhkan dalam pembuatan *engine* sehingga proses produksi bisa berjalan sesuai dengan rencana produksi.

PT Suzuki Indomobil Motor yang terletak di daerah Cakung merupakan perusahaan industri yang bergerak di bidang otomotif. Produk yang dihasilkan adalah *engine* yang digunakan pada kendaraan roda dua (sepeda motor) dan roda empat (mobil). PT Suzuki Indomobil Motor terdiri dari beberapa departemen yang saling berkaitan dalam melakukan kegiatan produksi. Salah satu departemen yang menangani proses produksi adalah Departemen Produksi, Sedangkan Seksi *Assembling Engine 4W* yang berfungsi merakit setiap komponen *engine* kendaraan roda empat (mobil).

PT Suzuki Indomobil Motor sudah menerapkan sistem informasi dalam menjalankan kegiatannya. Tetapi, masih terdapat beberapa departemen yang belum

menerapkan sistem informasi dalam pengolahan datanya. pada Seksi *Assembling Engine 4W* masih menggunakan program aplikasi *Microsoft Excel*. Setiap komponen yang penerimaannya kurang jumlahnya dan komponen yang perlu penggantian harus ditulis terlebih dahulu di formulir permintaan barang. Seksi *Assembly Engine 4W* dan seksi *PMC (Production Material Control)* sering terjadinya ketidak sesuaian antara rencana jumlah produksi dengan jumlah komponen yang tersedia dikarenakan terlambatnya komponen yang datang dan kelebihan stok pada beberapa komponen Yang menyebabkan terjadinya penambahan jam operasional produksi dan kelebihan kapasitas pada rak komponen.

Dengan memperhatikan hal-hal di atas dengan merencanakan pembuatan suatu sistem informasi yang dapat menangani pengolahan data penerimaan. Oleh karena itu, dibuatlah Tugas Akhir ini dengan judul “Rancang Bangun Sistem Informasi Penerimaan Komponen *Assembling Engine 4W* Berbasis Web Menggunakan PHP 5.5 dan MySQL 5.6 PT Suzuki Indomobil Motor”.

1.2 Permasalahan

Permasalahan yang terjadi pada Departemen Produksi Seksi *Assembling Engine 4W* adalah sebagai berikut:

1. Penerimaan komponen yang jumlahnya kurang dan penerimaan penggantian komponen untuk *assembling* ke *pmc* memakan waktu yang lama karena harus disertakan formulir permintaan barang dan di sertakan komponen yang bermasalah, bila komponen perlu diganti.
2. Penerimaan komponen seksi *Assembly Engine 4W* yang tidak sesuai antara kapasitas rak komponen dengan penerimaan jumlah komponen oleh seksi *PMC (Production Material Control)* karena sistem yang ada menangani penerimaan komponen masih menggunakan *Microsoft Excel*

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Penerimaan komponen yang jumlahnya kurang dan penerimaan penggantian komponen untuk *assembling* ke *pmc* dapat melakukannya dengan mudah dan

cepat karena sudah terkomputerisasi dan saling terhubung antara seksi pmc dan seksi *assembling*.

2. Membuat sistem informasi penerimaan komponen *assembly engine 4W* berbasis web menggunakan *database* sehingga memudahkan penerimaan jumlah komponen agar sesuai dengan kapasitas rak komponen karena *assembling* dapat melihat komponen apa saja dan jumlahnya berapa yang sudah diterima pmc.

1.4 Batasan Masalah

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

1. Penelitian dilakukan di Departemen Produksi Seksi *Assembling Engine 4W*.
2. Analisis dan penelitian hanya mengenai sistem informasi penerimaan komponen *assembling engine 4W* dalam bentuk laporan di PT Suzuki Indomobil Motor.
3. Penelitian tidak membahas mengenai proses produksi *Assembling Engine 4W* yang dilakukan di PT Suzuki Indomobil Motor

1.5 Manfaat Tugas Akhir

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

1. Bagi mahasiswa
Memberikan wawasan dan pengalaman kepada mahasiswa dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi untuk memberikan informasi yang akurat.
2. Bagi perusahaan
Hasil penelitian Rancang Bangun Sistem Penerimaan komponen *Assembling Engine 4W* diharapkan dapat mempermudah dalam pengolahan data, mempermudah jalannya produksi dan pengambilan keputusan.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam Tugas Akhir ini terbagi dalam beberapa pokok bahasan yaitu:

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Bab ini membahas teori tentang pengertian sistem, informasi, sistem informasi, pengembangan sistem, Diagram Alur (*Flowchart*), *Unified Modeling Language* (UML) sebagai alat bantu untuk membuat rancangan konseptual, pemrograman PHP, *database* MySQL, XAMPP dan teori-teori lain yang berhubungan dengan perancangan sistem.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang jenis dan sumber data, metode pengumpulan data, serta langkah-langkah yang akan dilakukan dalam pemecahan masalah termasuk metodologi pengembangan sistem yang digunakan yaitu metode *evolutionary prototype*.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini akan menguraikan tentang hasil pengamatan yang telah dilakukan, seperti tentang sekilas perusahaan dan Departemen Produksi Seksi *Assembly Engine 4W* sebagai objek penelitian yang mencakup pengolahan data komponen *assembling engine 4W*, dokumen yang terlibat, laporan yang dihasilkan oleh sistem dan analisis sistem yang berjalan

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi tentang analisis dan perancangan sistem yang diusulkan. Analisis dan perancangan sistem yang diusulkan menggunakan OOAD (*Object Oriented Analysis and Design*), dengan *tools* pemodelan UML (*Unified Modeling Language*), Diagram alir. Selanjutnya dibahas tentang perancangan sistem meliputi, perancangan program dengan *flowchart*, perancangan basis data, perancangan tampilan dan implementasi sistem.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran dalam penerapan sistem informasi penerimaan komponen *assembling engine 4W* untuk perusahaan dan pengembangan sistem yang ingin dilakukan selanjutnya

BAB II

LANDASAN TEORI

2.1 Pengertian Rancang Bangun

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

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

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

2.2 Konsep Dasar Sistem

Menurut McLeod (2008), 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. Sistem itu sendiri menjadi sebuah dasar sistem informasi yang akan dihasilkan, sistem akan membuat sebuah jaringan kerja sebagai relasi dalam menjalankan suatu prosedur dengan tujuan tertentu. Sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*systema*) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi.

Istilah ini sering dipergunakan untuk menggambarkan suatu set entitas yang berinteraksi dengan suatu model matematika. Sistem juga merupakan kesatuan bagian-bagian yang saling berhubungan yang berada dalam suatu wilayah serta memiliki item-item penggerak, contoh umum seperti negara.

Sistem dipandang sebagai cara atau metode untuk mencapai suatu tujuan. Dapat disimpulkan bahwa sistem adalah kumpulan obyek-obyek yang saling berinteraksi dan bekerja bersama-sama untuk mencapai tujuan tertentu dalam lingkungan yang kompleks. Obyek yang dimaksud di sini adalah bagian-bagian dari sistem, seperti *input*, proses, *output*, penerimaan umpan balik, dan batasan-batasan, di mana setiap bagian ini mempunyai beberapa nilai atau harga yang bersama-sama menggambarkan keadaan sistem pada suatu saat tertentu.

Kata sistem banyak sekali digunakan dalam percakapan sehari-hari, dalam forum diskusi maupun dokumen ilmiah. Kata ini digunakan untuk banyak hal, dan pada banyak bidang pula, sehingga mempunyai makna beragam. Dalam pengertian umum, sebuah sistem adalah sekumpulan benda yang memiliki hubungan di antara mereka. Supaya dapat berjalan dengan baik diperlukan kerjasama diantara unsur-unsur yang terkait dalam suatu sistem. Sistem dapat terdiri dari sistem-sistem bagian (subsistem). Sebagai contoh, sistem komputer dapat terdiri dari subsistem perangkat keras dan perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil lagi atau terdiri dari komponen-komponen.

Subsistem perangkat keras dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar. Hal di atas serupa dengan kutipan buku Analisis dan Desain Sistem Informasi, yaitu subsistem-subsistem saling berinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran sistem tersebut dapat tercapai. Interaksi dari subsistem-subsistem sedemikian rupa sehingga dicapai suatu kesatuan yang terpadu atau terintegrasi. (Jogiyanto, 2005)

2.2.1 Karakteristik Sistem

Struktur dari sebuah sistem adalah kaitan antara obyek dan kemampuan. Hubungan antara berbagai obyek dan kemampuan menentukan sistem. Contoh hubungan fungsional antara obyek dan kemampuan yang terbentuk secara alami adalah manusia dan komputer. Sedangkan hubungan disfungsional terjadi mungkin karena desain struktur yang kurang baik sehingga obyek sistem tidak mampu bekerja dengan baik.

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen-komponen (*components*), batas sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran (*objectives*) atau tujuan (*goal*). Berikut ini merupakan karakteristik sebuah sistem:

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem

Batas sistem (*boundary*) merupakan daerah yang membatasi antara satu sistem dengan sistem lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem

Lingkungan luar (*environment*) dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedangkan lingkungan

luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran (*output*) dari satu subsistem akan menjadi masukan (*input*) untuk subsistem yang lainnya dengan melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan Sistem

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk mendapatkan keluaran.

6. Keluaran Sistem

Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain.

7. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan lain menjadi keluaran berupa barang jadi.

8. Sasaran Sistem

Suatu sistem pasti memiliki tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. (Jogiyanto, 2005)

2.3 Konsep Dasar Informasi

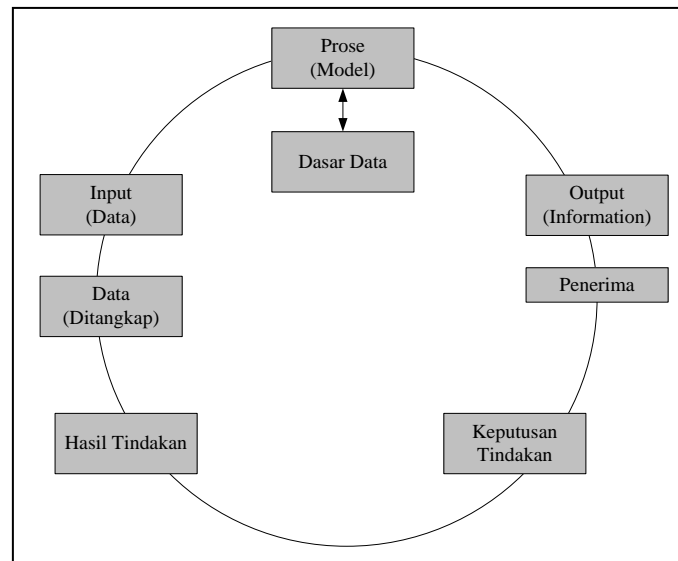
Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi, sehingga informasi ini sangat penting di dalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil dan akhirnya berakhir. Definisi informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian (*event*) adalah sesuatu yang terjadi pada saat tertentu. Kesatuan nyata (*fact* dan *entity*) adalah suatu obyek nyata seperti tempat, benda dan orang yang betul-betul ada dan terjadi. (Jogiyanto, 2005)

2.3.1 Siklus Informasi

Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk menghasilkan informasi. Untuk memperoleh informasi yang bermanfaat bagi penerimanya, perlu untuk dijelaskan bagaimana siklus yang terjadi atau dibutuhkan dalam menghasilkan informasi. (Jogiyanto, 2005)

Pertama data dimasukkan ke dalam model yang umumnya memiliki urutan proses tertentu dan pasti, setelah diproses akan dihasilkan informasi tertentu yang bermanfaat bagi penerima (*level management*) sebagai dasar dalam membuat suatu keputusan atau melakukan tindakan tertentu, Dari keputusan atau tindakan tersebut akan menghasilkan kejadian-kejadian tertentu yang akan digunakan kembali sebagai data yang nantinya akan dimasukkan ke dalam model (proses), begitu seterusnya. Dengan demikian akan membentuk suatu siklus informasi (*information cycle*) atau siklus pengolahan data (*data processing cycle*) yang dapat dilihat pada gambar II.1 berikut:



Gambar II.1 Siklus Informasi
(Sumber: Jogiyanto, 2005)

2.3.2 Kualitas Informasi

Menurut Jogiyanto (2005) kualitas informasi (*quality of information*) sangat dipengaruhi atau ditentukan oleh tiga hal, yaitu:

1. Relevan (*relevancy*)

Berarti informasi harus memberikan manfaat bagi pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila ditujukan kepada ahli teknik perusahaan.

2. Akurat (*accuracy*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak bias menyesatkan, dan harus jelas mencerminkan maksudnya. Ketidakakuratan dapat terjadi karena sumber informasi (data) mengalami gangguan atau kesengajaan sehingga merusak atau merubah data-data asli tersebut.

3. Tepat waktu (*timeliness*)

Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat (usang). Informasi yang usang tidak mempunyai nilai yang baik, sehingga kalau digunakan sebagai dasar dalam pengambilan keputusan akan berakibat fatal

atau kesalahan dalam keputusan dan tindakan. Kondisi demikian menyebabkan mahalnya nilai suatu informasi, sehingga kecepatan untuk mendapatkan, mengolah dan mengirimkannya memerlukan teknologi-teknologi terbaru.

2.4 Konsep Dasar Sistem Informasi

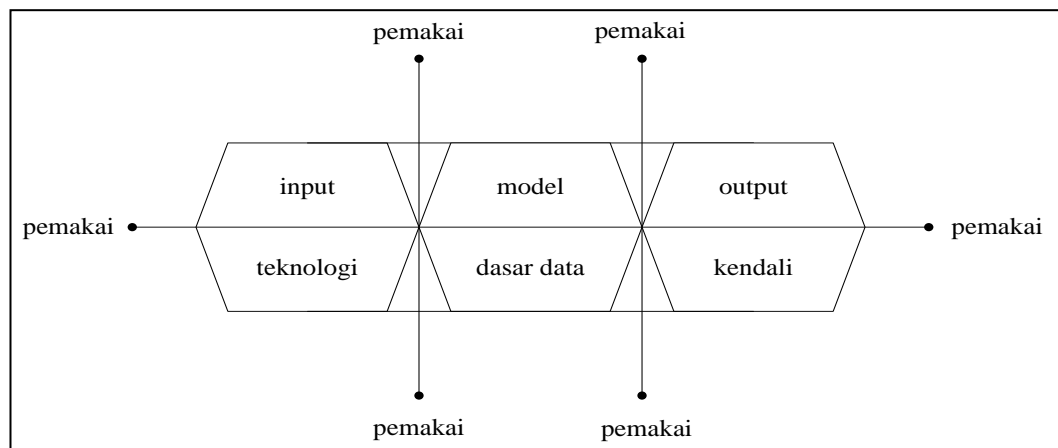
Konsep dasar sistem informasi ini terlebih dahulu akan diketahui tentang apa yang dimaksud dengan konsep dasar sistem seperti karakteristik sistem. Konsep dasar informasi seperti kualitas informasi dan nilai informasi.

Informasi dapat diperoleh dari sistem informasi (information system) atau disebut juga dengan processing system atau information processing system. Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. (Jogiyanto, 2005)

Sistem informasi dapat pula sebagai pendukung pengambilan keputusan, koordinasi tiap bagian, dan kendali/kontrol sistem. Sistem informasi dapat juga membantu para manajer dan karyawan untuk meneliti permasalahan, memvisualisasikan pokok-pokok yang kompleks, dan menciptakan produk-produk baru.

2.4.1 Komponen Sistem Informasi

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



Gambar II.2 Blok Sistem Informasi Yang Berinteraksi
(Sumber: Jogiyanto, 2005)

1. Blok Masukan

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

2. Blok Model

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

3. Blok Keluaran

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

4. Blok Teknologi

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

5. Blok Basis Data

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

6. Blok Kendali

Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperatur air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidakefisienan, sabotase dan lain sebagainya. Beberapa penerimaan perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi. (Jogiyanto, 2005)

2.5 Siklus Hidup Pengembangan Sistem

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

1. Perancangan Sistem

merancang output, input, struktur file, program, prosedur, perangkat keras dan perangkat lunak yang diperlukan untuk mendukung sistem informasi.

2. Analisis Sistem

menganalisis dan mendefinisikan masalah dan kemungkinan solusinya untuk sistem informasi dan proses organisasi.

3. **Pembangunan desain dan Testing Sistem**
membangun perangkat lunak yang diperlukan untuk mendukung sistem dan melakukan testing secara akurat. Melakukan instalasi dan testing terhadap perangkat keras dan mengoperasikan perangkat lunak
4. **Implementasi Sistem**
beralih dari sistem lama ke sistem baru, melakukan pelatihan dan panduan seperlunya.
5. **Evaluasi Sistem**
mengevaluasi sejauh mana sistem telah dibangun dan seberapa bagus sistem telah dioperasikan.
6. **Operasi dan Perawatan**
mendukung operasi sistem informasi dan melakukan perubahan atau tambahan fasilitas.

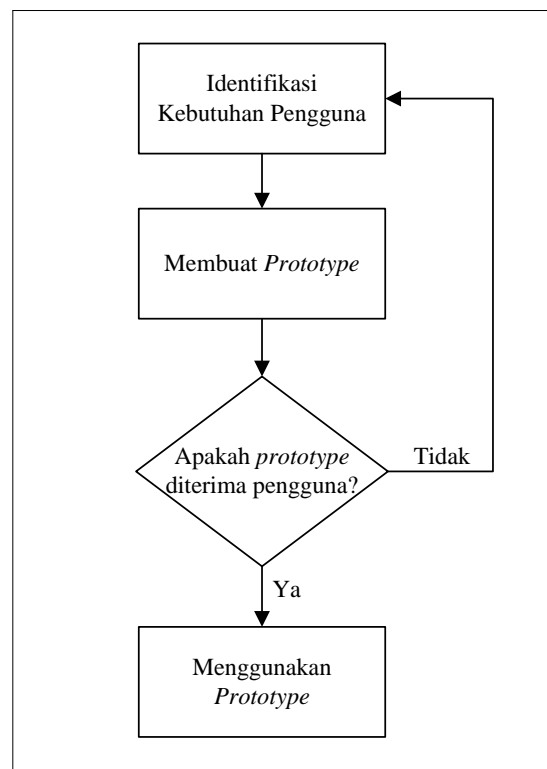
2.6 Metodologi Pengembangan Sistem

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

Sebuah metodologi berbasis *prototyping* melakukan tahap identifikasi, desain, dan pelaksanaan bersamaan, dan ketiga tahap tersebut dilakukan berulang kali dalam suatu siklus sampai sistem selesai. Prototipe pertama biasanya bagian pertama sistem yang akan digunakan pengguna. Hal ini ditunjukkan kepada pengguna dan sponsor proyek yang akan memberikan komentar, yang digunakan untuk menganalisis, mendesain, dan mengimplementasikan kembali prototipe kedua yang menyediakan fitur yang lebih sedikit. Proses ini berlanjut dalam suatu siklus sampai para analis, pengguna, dan sponsor setuju bahwa prototipe menyediakan fungsionalitas yang cukup untuk diinstal dan digunakan dalam organisasi. Setelah prototipe (sekarang disebut sistem) terinstal, perbaikan terus dilakukan sampai diterima sebagai sistem baru. (Mc.Leod, 2008)

Ada tiga jenis metode *prototype* yang dikembangkan, yaitu:

1. *Prototype evolutionary*, yaitu *prototype* yang terus menerus disempurnakan sampai memiliki seluruh fungsionalitas yang dibutuhkan pengguna dari sistem yang baru. (Mc.Leod, 2008) Beberapa langkah dalam metode pengembangan *prototype evolutionary* digambarkan seperti berikut ini:



Gambar II.3 Langkah Pengembangan *Prototype Evolutionary*
(Sumber: Mc.Leod, 2008)

2. *Prototype persyaratan*, yaitu *prototype* yang dikembangkan sebagai salah satu cara untuk mendefinisikan persyaratan-persyaratan fungsional dari sistem baru ketika pengguna tidak mampu mengungkapkan dengan jelas apa yang mereka inginkan. (McLeod, 2008)
3. *Throwaway Prototyping*
Metodologi *throwaway prototyping* dilakukan pada titik yang berbeda dalam SDLC. Metodologi *throwaway prototyping* ini memiliki tahap

analisis yang relatif menyeluruh, yang digunakan untuk mengumpulkan informasi dan mengembangkan ide-ide untuk konsep sistem. Namun, banyak fitur yang disarankan oleh pengguna mungkin tidak dipahami dengan baik, dan hal ini menjadi tantangan teknis untuk dipecahkan. Setiap masalah ini diperiksa oleh analis, perancang, dan pembuat desain *prototype*. (Mc.Leod, 2008)

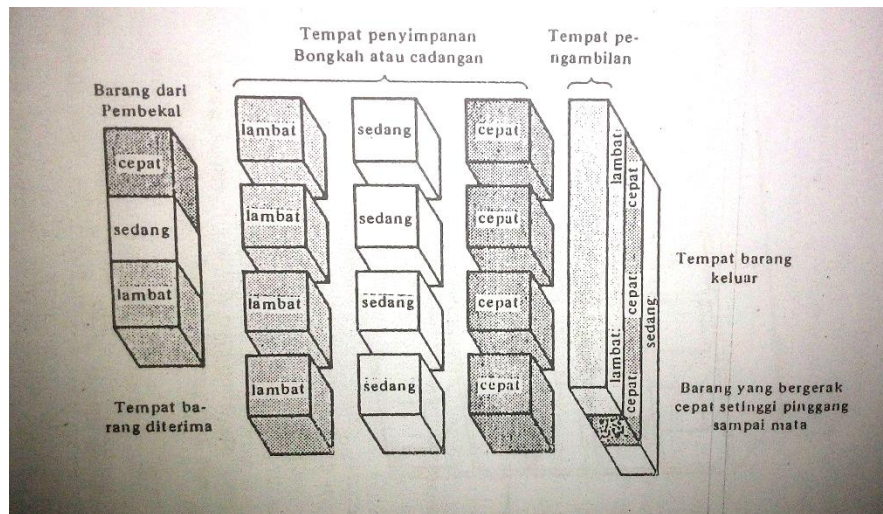
Dengan metode *prototyping* ini pengembang dan pengguna dapat saling berinteraksi selama proses pembuatan sistem. Sering terjadi, seorang pengguna hanya mendefinisikan secara umum apa yang dikehendakinya tanpa menyebutkan secara detail *output* apa saja yang dibutuhkan, pemrosesan dan data-data apa saja yang dibutuhkan. (Mc. Leod, 2008)

2.7 Perencanaan Untuk Penerimaan

Perencanaan gerak barang dalam gudang didasarkan pada tiga tahap pekerjaan pokok dalam gudang, yaitu (Warman, 1971):

1. Penerimaan dan penyimpanan pertama atas barang bongkah.
2. Pemecahan, seleksi, dan penyusunan kembali dalam bongkah lagi.
3. Gerak barang keluar dari gudang.

Perencanaan harus dibuat sedemikian rupa, sehingga semua gerak ini berjalan selancar mungkin dan sinambung. Tujuannya jelas bahwa barang gerak cepat lebih baik diletakkan dekat tempat pengambilan barang, sehingga mengurangi seringnya terjadi gerakan bolak-balik.



Gambar II.4 Filsafat Tata Ruang yang Luas
Sumber: Warman (1971)

Gambar tersebut menunjukkan prinsip-prinsip tata ruang di mana barang itu ditata berdasarkan kecepatan gerakannya. Dalam hal ini barang yang bergerak lambat tidak boleh menghambat laju barang yang bergerak cepat yang sedang dikirim ke gudang (Warman, 1971).

2.8 Pengertian Komponen

Bahan baku (komponen) merupakan bahan langsung, yaitu bahan yang membentuk suatu kesatuan yang tidak terpisahkan dari produk jadi. Bahan baku adalah bahan utama atau bahan pokok dan merupakan komponen utama dari suatu produk. Bahan baku biasanya mudah ditelusuri dalam suatu produk dan harganya relative tinggi dibandingkan dengan bahan pembantu. Misalkan produk kursi rotan bahan bakunya rotan. Adapun bahan pembantu dari produk kursi rotan, seperti paku, lem kayu, dempul, dan lain-lain. Besarnya bahan baku yang dimiliki perusahaan ditentukan oleh beberapa faktor, antara lain:

1. Anggaran produksi

Semakin besar produksi yang dianggarkan semakin besar bahan baku yang disediakan. Sebaliknya semakin kecil produksi yang dianggarkan semakin kecil juga bahan baku yang disediakan.

2. Harga beli bahan baku

Semakin tinggi harga beli bahan baku, semakin tinggi persediaan bahan baku yang direncanakan. Sebaliknya semakin rendah harga bahan baku yang dibeli, semakin rendah persediaan bahan baku yang direncanakan.

3. Biaya penyimpanan bahan baku di gudang (*carrying cost*) dalam hubungannya dengan biaya extra yang dikeluarkan sebagai akibat kehabisan persediaan (*stockout cost*).
4. Ketepatan pembuatan standar pemakaian bahan baku. Semakin tepat standar bahan baku dipakai yang dibuat, semakin kecil persediaan bahan baku yang direncanakan. Sebaliknya bila standar bahan baku dipakai yang dibuat sulit untuk mendekati ketepatan, maka persediaan bahan baku yang direncanakan akan besar.
5. Ketepatan pemasok (penjual bahan baku) dalam menyerahkan bahan baku yang dipesan.
6. Jumlah bahan baku setiap kali pesan.

2.9 Bagan Alir (*Flowchart*)

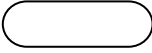


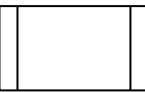
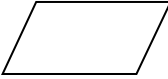

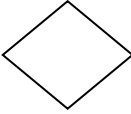
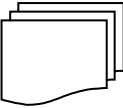
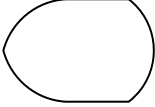
Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Pada waktu akan menggambar suatu bagan alir, analisis sistem atau pemrogram dapat mengikuti pedoman-pedoman sebagai berikut ini: (Jogiyanto, 2005)

1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhirnya.
4. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan, misalnya:
 - a. Persiapkan dokumen
 - b. Hitung gaji



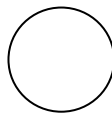
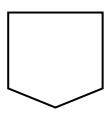
5. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakanlah simbol-simbol bagan alir yang standar.

Simbol-simbol standar yang digunakan oleh analis sistem untuk membuat bagan alir dokumen yang menggambarkan sistem tertentu dapat dilihat pada tabel berikut ini:

Tabel II.1 Simbol-Simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1.		Terminal	Awal dan akhir dari suatu proses.
2.		Garis akhir	Arus dari suatu proses.
3.		Proses	Proses pengolahan data.
4.		Proses terdefinisi	Permulaan sub program/proses menjalankan program.
5.		<i>Input/output</i> data	Mewakili data masukan atau keluaran.
6.		<i>Manual operation</i>	Menggambarkan operasi yang dilakukan secara manual.
7.		<i>Decision</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya.
8.		Dokumen rangkap	Menggambarkan dokumen asli dan tembusannya.
9.		<i>Display</i>	Menampilkan <i>output</i> .

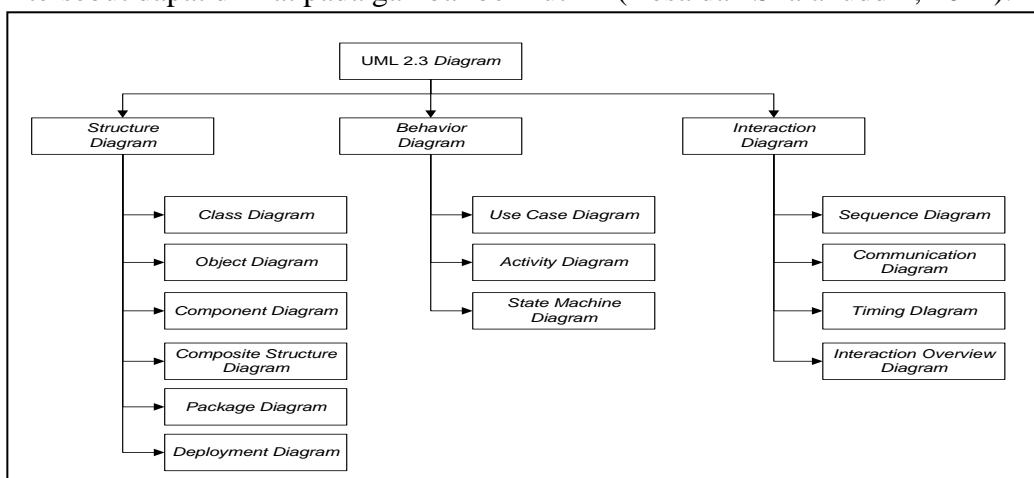
Tabel II.1 Simbol-Simbol *Flowchart* (Lanjutan)

10.		Arsip sementara	Menunjukkan tempat penyimpanan dokumen.
11.		Arsip permanen	Menunjukkan tempat penyimpanan dokumen secara permanen yang tidak akan diproses lagi.
12.		<i>On page connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman.
13.		<i>Off page connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda.

(Sumber: Jogiyanto, 2005)

2.10 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek. Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori tersebut dapat dilihat pada gambar berikut ini (Rosa dan Shalahuddin, 2011).



Gambar II.5 Klasifikasi Diagram UML

(Sumber: Rosa dan Shalahuddin, 2011)

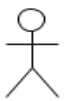
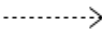

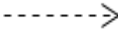
Berikut ini penjelasan singkat dari pembagian kategori tersebut (Rosa dan Shalahuddin, 2011):

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

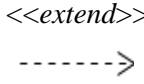



2.10.1 Use Case Diagram

Use case 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.

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

Simbol	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>dependent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
<i><<include>></i> 	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .

Tabel II.2 Simbol-Simbol *Use Case Diagram* (Lanjutan)

	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>Sistem</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem.



(Sumber: Rosa dan Shalahuddin, 2011)

Setelah membuat *use case* menurut kebutuhan, maka perlu juga dibuat pendeskripsian atau scenario *use case* dan aktor apa yang terdapat di dalamnya.







2.10.2 *Activity Diagram*

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem yang akan dibangun bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem yang akan dibangun.

Tabel II.3 Simbol-simbol *Activity Diagram*

Simbol	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi

Tabel II.3 Simbol-simbol *Activity Diagram* (Lanjutan)


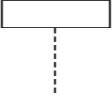
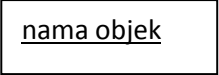
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Decision</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya.
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran
	<i>Join Node</i>	Beberapa aliran yang pada tahap tertentu berubah menjadi satu aliran
	<i>Directional Association</i>	Menghubungkan antar proses yang saling berhubungan dan membentur suatu alur proses

(Sumber: Rosa dan Shalahuddin, 2011)


2.10.3 *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Banyaknya *sequence diagram* yang harus digambar adalah minimal sebanyak pendefinisian *use case*.

Tabel II.4 Simbol-simbol *Sequence Diagram*

Simbol	Nama	Keterangan
	Aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi dibuat.
	<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	Objek	Menyatakan objek yang berinteraksi pesan.

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




	waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
1 : nama_metode() →	Pesan tipe <i>call</i>	Menyatakan suatu objek memanggil operasi atau metode yang ada pada objek lain atau dirinya sendiri.
1 : Masukan →	Pesan tipe <i>send</i>	Menyatakan suatu objek mengirimkan data ke objek lain.
1 : Keluaran ⋯→	Pesan tipe <i>return</i>	Menyatakan suatu objek telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu.

(Sumber: Rosa dan Shalahuddin 2011)



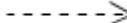

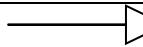
2.10.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa dan Shalahuddin, 2011).

Tabel II.5 Simbol-simbol *Class Diagram*

Simbol	Nama	Keterangan
	<i>Bidirectional Association</i>	Hubungan dimana salah satu kelas mengirimkan pesan kepada kelas lain, kemudian kelas yang lain mengirimkan pesan kepada kelas yang mengirimnya pesan.
	<i>N-ary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.

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

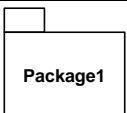
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Relasi ini menunjukkan bahwa sebuah kelas mengacu kepada kelas lainnya.
	<i>Directional Association</i>	Menggambarkan bahwa pesan atau urutan kejadian terjadi dari hanya salah satu kelas, sedangkan kelas yang lain pasif.
	<i>Generalization</i>	Adalah relasi pewarisan antara dua kelas.

(Sumber: Rosa dan Shalahuddin, 2011)

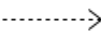

2.10.5 Component Diagram

Component diagram mengandung *component*, *interface* dan *relationship*. Hal yang penting pada *component* adalah *component* mewakili potongan-potongan yang independen yang bisa dipesan dan diperbaharui sewaktu-waktu. *Component* dihubungkan melalui *interface* yang diimplementasikan (Rosa dan Shalahuddin, 2011).

Tabel II.6 Simbol-simbol *Component Diagram*

Simbol	Nama	Keterangan
	<i>Package</i>	Merupakan sebuah kumpulan dari satu atau lebih komponen
	<i>Link</i>	Relasi antar objek
	<i>Component</i>	Komponen sistem

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



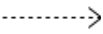
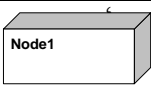
	<i>Dependency</i>	Hubungan suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Interface</i>	Sebagai antarmuka komponen

(Sumber: Rosa dan Shalahuddin, 2011)

2.10.6 *Deployment Diagram*

Deployment Diagram menyediakan gambaran bagaimana sistem secara fisik akan terlihat. Sistem terdiri dari *node-node* di mana setiap *node* diwakili oleh sebuah kubus dan garis yang menghubungkan antara kubus tersebut menunjukkan hubungan antara kedua *node* tersebut (Rosa dan Shalahuddin, 2011).

Tabel II.7 Simbol-simbol *Deployment Diagram*

Simbol	Nama	Keterangan
	<i>Package</i>	Merupakan sebuah bungkus dari satu atau lebih <i>node</i>
	<i>Link</i>	Relasi antar objek
	<i>Dependency</i>	Hubungan pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Node</i>	Perangkat keras dan perangkat lunak

(Sumber: Rosa dan Shalahuddin, 2011)

2.11 Kamus Data Dan Database

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

dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem.

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

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

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Himpunan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file* atau tabel atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.12 HIPO (*Hierarchy plus Input Process Output*)

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

HIPO dapat digunakan sebagai alat pengembangan sistem dan teknik dokumentasi program dan penggunaan HIPO ini mempunyai sasaran utama sebagai berikut:

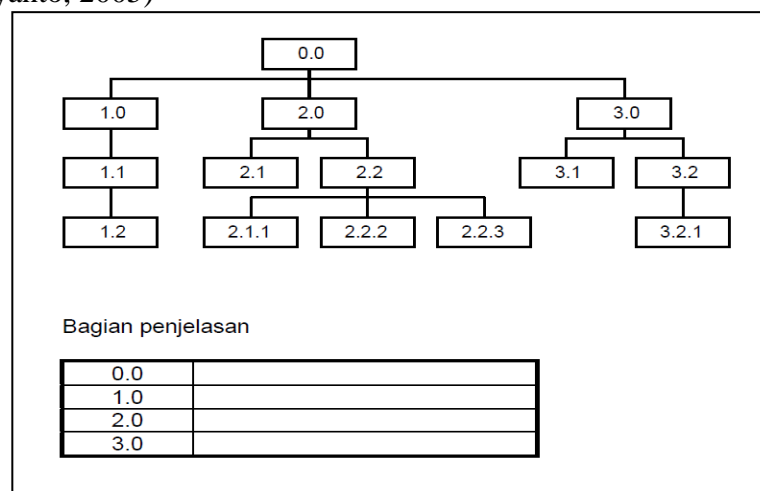
1. Untuk menyediakan suatu struktur guna memahami fungsi-fungsi dari sistem.

2. Untuk lebih menekankan fungsi-fungsi yang harus diselesaikan oleh program, bukan menunjukkan pernyataan-pernyataan program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan dari *input* yang harus digunakan dan *output* yang harus dihasilkan oleh masing-masing fungsi pada tiap-tiap tingkatan.
4. Untuk menyediakan *output* yang tepat dan sesuai dengan kebutuhan-kebutuhan pemakai.

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

1. Daftar Isi Visual/ *Visual Tabel of Contents* (VTOC)

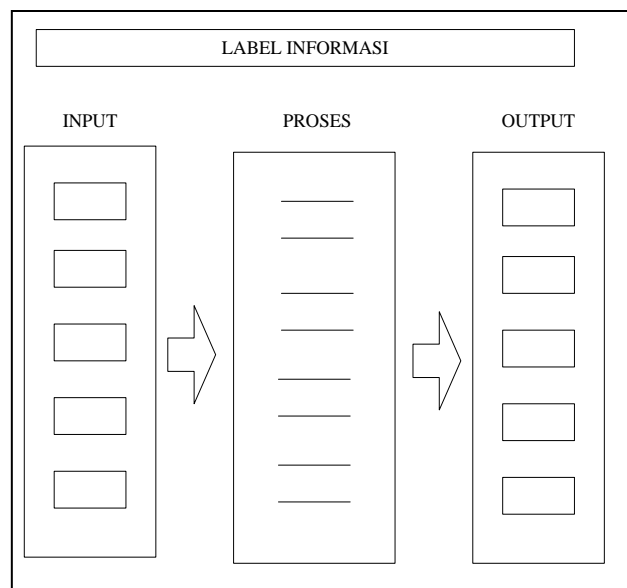
Visual tabel of contents menggambarkan seluruh program HIPO baik rinci maupun ringkasan yang terstruktur. Pada diagram ini nama dan nomor dari program HIPO diidentifikasi. Struktur paket diagram dan hubungan fungsi juga diidentifikasi dalam bentuk hirarki. Keterangan masing-masing fungsi diberikan pada bagian penjelasan yang diikutsertakan dalam diagram ini. *Visual tabel of contents* ini dapat digambarkan sebagai berikut. (Jogiyanto, 2005)



Gambar II.6 *Visual Tabel of Contents*
(Sumber: Jogiyanto, 2005)

2. *Overview diagram*

Overview diagram menunjukkan secara garis besar hubungan dari *Input*, proses dan *Output*. Bagian *Input* menunjukkan item-item data yang akan digunakan oleh bagian proses. Bagian proses berisi sejumlah langkah-langkah yang menggambarkan kerja dari fungsi. Bagian *Output* berisi dengan item-item data yang dihasilkan atau dimodifikasi oleh langkah-langkah proses. *Overview diagram* ini dapat digambarkan sebagai berikut:



Gambar II.7 *Overview Diagram*
(Sumber: Jogiyanto, 2005)

3. *Detail Diagram*

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

2.13 PHP

PHP (PHP: *Hypertext Preprocessor*) merupakan bahasa pemrograman berbasis *web* yang memiliki kemampuan untuk memproses data secara dinamis. PHP dikatakan sebagai sebuah *server-side embedded script language*, artinya sintaks-sintaks dan perintah yang kita berikan akan sepenuhnya dijalankan oleh

server tetapi disertakan pada halaman HTML biasa. Pada umumnya, semua aplikasi yang dibangun menggunakan PHP akan memberikan hasil pada *Web Browser*, tetapi prosesnya secara keseluruhan dijalankan di *server*. *Server* akan melakukan beberapa hal sebagai berikut:

1. Membaca permintaan dengan skrip PHP berasal dari *browser*.
2. Mencari halaman/*page* di *server* (*server pages*).
3. Melakukan *processing* melalui instruksi yang diberikan oleh PHP untuk melakukan modifikasi pada halaman/*page*.
4. Mengirim kembali halaman tersebut kepada *client* melalui *internet* atau *intranet* yang merupakan proses *echo/print*.

PHP memiliki kelebihan yang tidak dimiliki oleh bahasa *script* yang sejenis. PHP secara mendasar dapat mengerjakan semua yang dapat dikerjakan oleh program CGI, seperti mendapatkan data dari form, menghasilkan isi halaman *web* yang dinamik, dan menerima *cookies*. (Sidik, 2004)

2.14 MySQL

MySQL adalah sebuah program pembuat dan pengelola *Database* atau yang sering disebut dengan DBMS (*Database Management Sistem*), sifat dari DBMS ini adalah *open source*. MySQL sebenarnya produk yang berjalan pada *platform* Linux, dengan adanya perkembangan dan banyaknya pengguna serta lisensi dari *Database* ini adalah *open source*, maka para pengembang kemudian merilis versi Windows. (Nugroho, 2007)

SQL adalah suatu bahasa permintaan yang terstruktur dan telah distandarkan untuk semua program pengakses *Database* seperti Oracle, PostgreSQL, SQL Server, dan lain-lain. (Kadir, 2008)

Pada MySQL masing-masing tipe data memiliki kegunaan dan keterangan untuk digunakan dalam merancang tabel. Berikut ini beberapa tipe data yang terdapat pada MySQL:

Tabel II.8 Beberapa Tipe Data Pada MySQL

Tipe Data	Ukuran	Keterangan
CHAR	M	Menampung maksimal M karakter (kombinasi huruf, angka, dan simbol-simbol). Jumlah memori yang dibutuhkan selalu M byte. M terbesar adalah 255.
VARCHAR	M	Karakter yang disimpan maksimal M karakter. Jumlah memori yang dibutuhkan tergantung jumlah karakter. M bisa mencapai 65535.
DATE	8 byte	Menyatakan tanggal.
TIME	8 byte	Menyatakan waktu (jam:menit:detik).
TINYINT	1 byte	Bilangan antara -128 sampai dengan +127.
SMALLINT	2 byte	Bilangan antara -32768 sampai dengan +32768.
INT	4 byte	Bilangan antara -2147683648 sampai dengan +2147683647.
FLOAT	4 byte	Bilangan pecahan.
DOUBLE	8 byte	Bilangan pecahan dengan presisi tinggi.
BOOL	1 byte	Untuk menampung nilai <i>true</i> (benar) dan <i>false</i> (salah). Identik dengan TINYINT.
ENUM	-	Menyatakan suatu tipe yang nilainya tertentu (disebutkan dalam pendefinisian).
TEXT	-	Menyimpan teks yang ukurannya sangat panjang.
BLOB	-	Untuk menyimpan data biner (misalnya gambar atau suara).

(Sumber: Kadir, 2008)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

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

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

3.2 Jenis dan Sumber Data

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

1. Data primer

Data yang diperoleh melalui pengamatan langsung terhadap sistem yang sedang berjalan dan wawancara dengan pegawai Departemen Produksi Seksi *Assembling Engine 4W* sebagai sumber informasinya. Dalam penelitian ini data tersebut berupa data alur proses *update* dan data yang mendukung informasi yang dihasilkan terhadap komponen *assembling engine 4W* pada PT Suzuki Indomobil Motor

2. Data sekunder

Data penelitian yang diperoleh secara tidak langsung melalui media perantara atau data yang diperoleh dan dicatat oleh pihak lain yang telah disusun dan dipublikasikan. Data sekunder yang digunakan dalam penelitian ini laporan-laporan, buku-buku dan kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian.

3.3 Pengumpulan Data

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

1. Observasi

Melakukan pengamatan langsung pada departemen Produksi Seksi *Assembling Engine 4W* Indonesia. Hasil pengamatan yang dilakukan menjadi landasan penulis dalam melakukan perancangan aplikasi yang akan dibuat.

Adapun teknik studi lapangan yang digunakan yaitu:

2. Wawancara

Pihak yang diwawancarai adalah para pengguna sistem tersebut untuk memperoleh keterangan mengenai alur proses *input* data komponen *engine* dalam menghasilkan laporan harian dan juga bulanan.

3. Studi Pustaka

Studi pustaka yaitu metode pengumpulan data dengan melakukan pencarian referensi yang berkaitan dengan pemrograman PHP dan *database* MySQL dari berbagai referensi, baik itu referensi elektronik yang didapat dari internet maupun referensi dari buku teks. Referensi yang diperoleh, kemudian dikaji sebagai dasar penulis dalam menyelesaikan penelitian.

3.4 Metode Pengembangan Sistem

Metode ini menggunakan pemodelan dengan menggunakan diagram-diagram yang terdapat dalam *Unified Modeling Language* (UML). Sedangkan metode pengembangan perangkat lunak yang digunakan yaitu dengan model *prototype* jenis *evolutionary prototype*. Tahapan-tahapan dalam *evolutionary prototype* adalah sebagai berikut (Mcleod, 2011):

1. Mengidentifikasi kebutuhan pengguna

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

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

3.5 Kerangka Penelitian

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

1. Studi pendahuluan yang dilakukan yaitu dengan membaca buku literatur, *browsing* internet serta sumber-sumber lain dalam lingkup perkuliahan maupun di luar lingkup perkuliahan yang berhubungan dengan judul dan permasalahan tugas akhir.
2. Mengidentifikasi dan merumuskan permasalahan yang ditemukan saat dilakukan penelitian pada sistem penerimaan komponen yang sedang berjalan di Departemen Produksi Seksi *Assembling Engine 4W* PT Suzuki Indomobil Motor.
3. Melakukan studi literatur untuk mendapatkan landasan teori yang dapat mendukung pemecahan masalah yang sedang diteliti. Teori yang didapatkan merupakan langkah awal agar permasalahan lebih dipahami dengan benar sesuai dengan kerangka berpikir ilmiah.
4. Menentukan tujuan penelitian. Menentukan tujuan dari dilakukannya penelitian yaitu merancang bangun sistem informasi penerimaan komponen

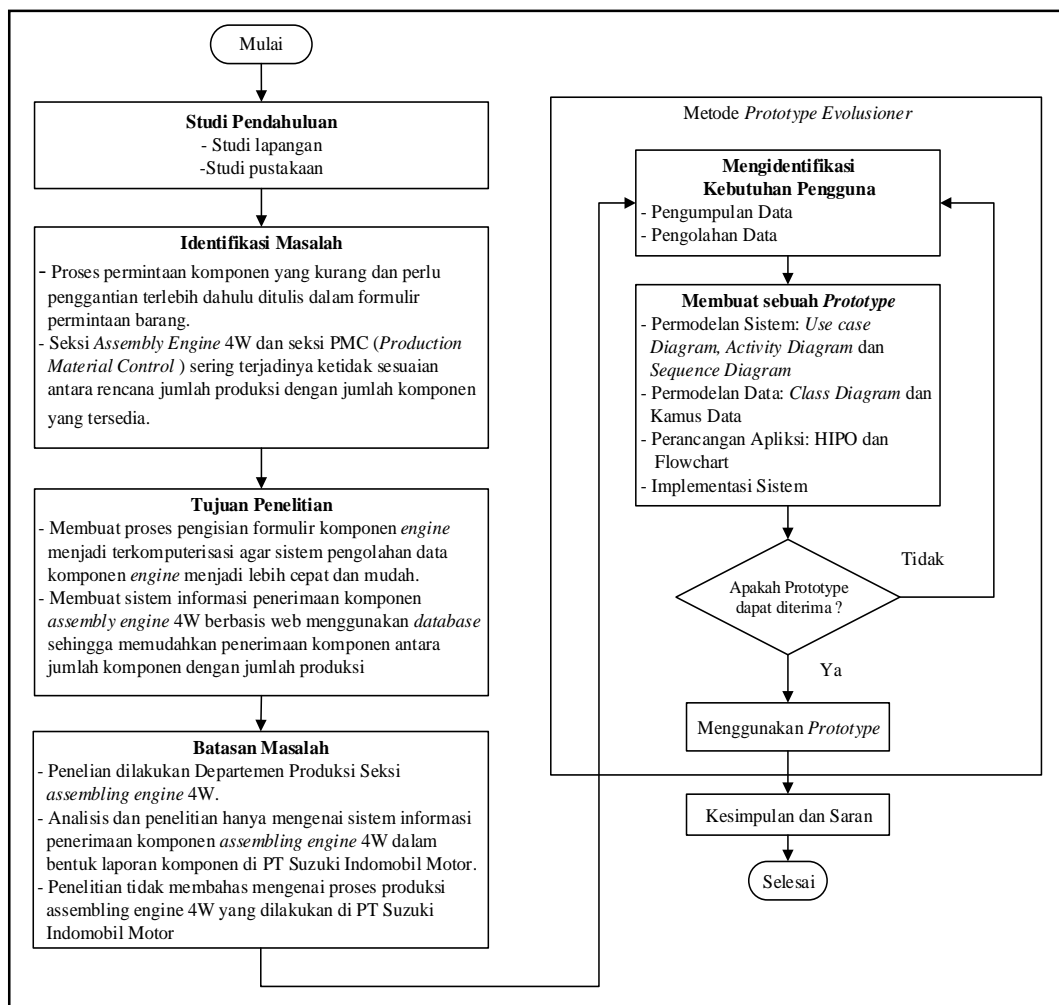
yang dapat menyajikan informasi komponen *assembling engine* 4W dan laporan komponen *engine*.

5. Menentukan pendekatan pemrograman sistem, *tools* yang digunakan untuk pemodelan dan metodologi yang akan digunakan dalam pengembangan perangkat lunak. Pendekatan pemrograman sistem yang dipilih untuk rancang bangun sistem informasi penerimaan komponen *assembling engine* 4W ini adalah pendekatan berorientasi objek dengan *tools* untuk pemodelannya adalah UML dan metodologi pengembangan perangkat lunak yang digunakan adalah *evolutionary prototype*.
6. Mengumpulkan data yang diperlukan dengan cara observasi langsung terhadap sistem yang berjalan, serta wawancara dengan pihak yang mempunyai hubungan dan pengetahuan terhadap sistem yang berjalan.
7. Melakukan analisis terhadap data yang telah diperoleh sebelumnya, sehingga dapat diketahui kebutuhan *user* terhadap aplikasi yang akan dibuat.
8. Membuat pemodelan sistem, pemodelan data dan desain sistem. Pemodelan sistem berbasis objek dengan menggunakan UML (*Unified Modeling Language*) berupa *use case diagram*, *activity diagram*, *sequence diagram*. Sedangkan pemodelan data dengan menggunakan *class diagram* dan kamus data. Untuk desain sistem menggunakan *flowchart* dan HIPO (*Hierarchy plus Input-Proses-Output*).
9. Membuat perancangan *database*, perancangan antarmuka dan perancangan arsitektur sistem.
10. Pengodean (*coding*) sistem yang telah dirancang dengan menggunakan pemrograman PHP dengan *tools* untuk penulisannya notepad++ dan perangkat lunak manajemen basis data MySQL.
11. Pada tahap ini, *prototype* yang diusulkan akan dilihat apakah sudah sesuai dengan keinginan *user* atau belum. Jika belum maka dilakukan analisis data kembali, tetapi apabila sesuai maka tahap selanjutnya adalah implementasi penggunaan *prototype* tersebut.
12. Sebelum melakukan implementasi sistem usulan, ada 2 (dua) hal yang harus dipenuhi yaitu perangkat keras yang akan digunakan yaitu *notebook* atau

Personal Computer (PC) dan perangkat lunak yaitu *Apache Service* dan *MySQL Service*.

13. Setelah *prototype* sistem usulan diterima maka tahap selanjutnya adalah menarik kesimpulan dari penelitian yang telah dilakukan di PT Suzuki Indomobil Motor dan memberikan saran yang membangun bagi perusahaan dan bagi peneliti selanjutnya dalam penggunaan dan pengembangan sistem penerimaan komponen *assembly engine* 4W.

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



Gambar III.1 Kerangka Penelitian

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Perusahaan

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

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

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

PT Suzuki Indomobil Motor mempunyai beberapa *Overseas Sales Subsidiaries*. Lokasi yang terbesar berada di Perancis, Italia, Inggris, Amerika, Selandia Baru, Jerman, dan Kanada.

Lokasi kantor pusat PT Suzuki Indomobil Motor berada di Wisma Indomobil di Jalan MT. Haryono, Kav. 8 Jakarta Timur. Kantor Pusat ini didukung oleh 314 karyawan. Lokasi pabriknya tersebar di beberapa tempat, antara lain di Pulogadung, Cakung, dan Tambun.

4.1.1 Program Perusahaan

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

1. GDS (Gerakan Disiplin Suzuki)
Program untuk meningkatkan kedisiplinan dengan menerapkan aturan-aturan yang wajib dipatuhi oleh seluruh pegawai PT Suzuki Indomobil Motor.
2. GKM (Gugus Kendali Mutu)
Tolak ukur untuk seluruh bagian produksi dalam mengendalikan dan meningkatkan mutu di PT Suzuki Indomobil Motor.
3. K3 (Keselamatan dan Kesehatan Kerja)
Panduan dalam menjaga keselamatan dan kesehatan seluruh pegawai PT Suzuki Indomobil Motor saat mereka bekerja.
4. Usulan
Program untuk mewadahi setiap pegawai yang memiliki saran dan usulan.
5. *Kaizen*
Teori yang diterapkan oleh PT Suzuki Indomobil Motor untuk meningkatkan kinerja pegawai dalam kesuksesan perusahaan.
6. CS (*Customer Satisfaction*)
Merupakan program pada PT Suzuki Indomobil Motor dalam meningkatkan kepuasan pelanggan dan menjalin hubungan baik dengan para pelanggan.

4.1.2 Visi dan Misi Perusahaan

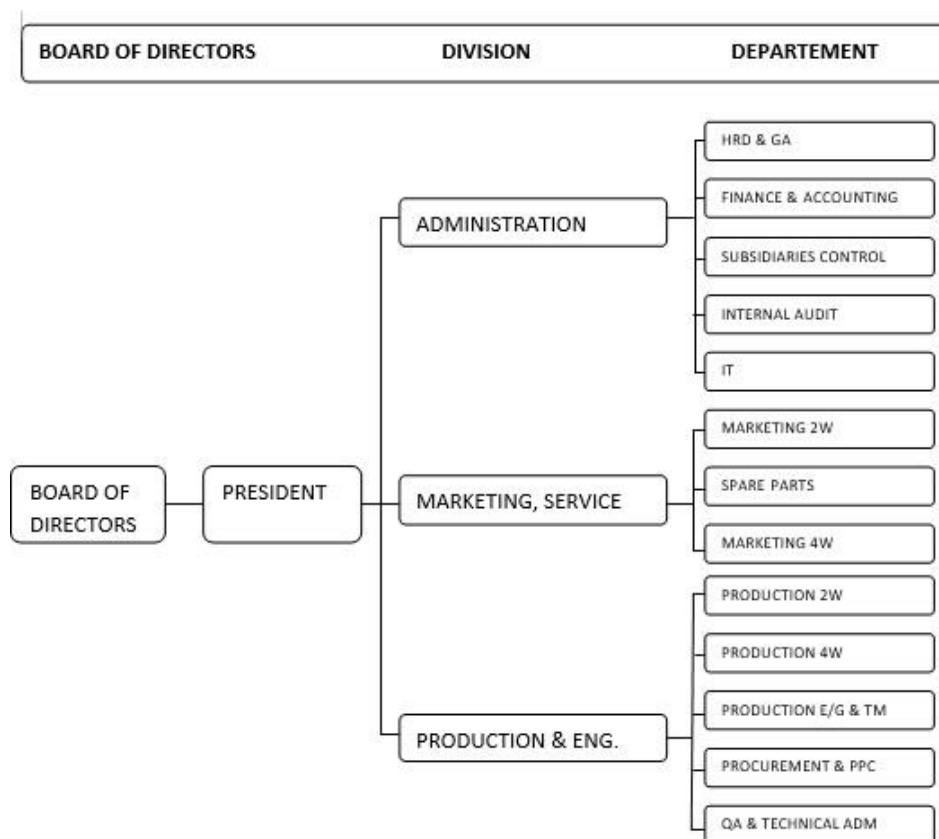
PT Suzuki Indomobil Motor adalah perusahaan yang bergerak di bidang industri otomotif, mempunyai visi yaitu menjadi perusahaan terkemuka di dalam Suzuki Global Operation yang dihargai dan dikagumi di Indonesia. Sedangkan misi dari PT Suzuki Indomobil Motor yaitu menginginkan pertumbuhan dan perkembangan Perusahaan yang berimbang berdasarkan azas kerja keras, integrasi dan kebersamaan untuk selalu mencapai hasil lebih baik dalam mendahului harapan pelanggan.

4.1.3 Struktur Organisasi Perusahaan

Organisasi adalah suatu bentuk atau wadah dari kelompok manusia dalam usaha untuk mencapai tujuan. Agar organisasi dapat bekerja dengan baik maka

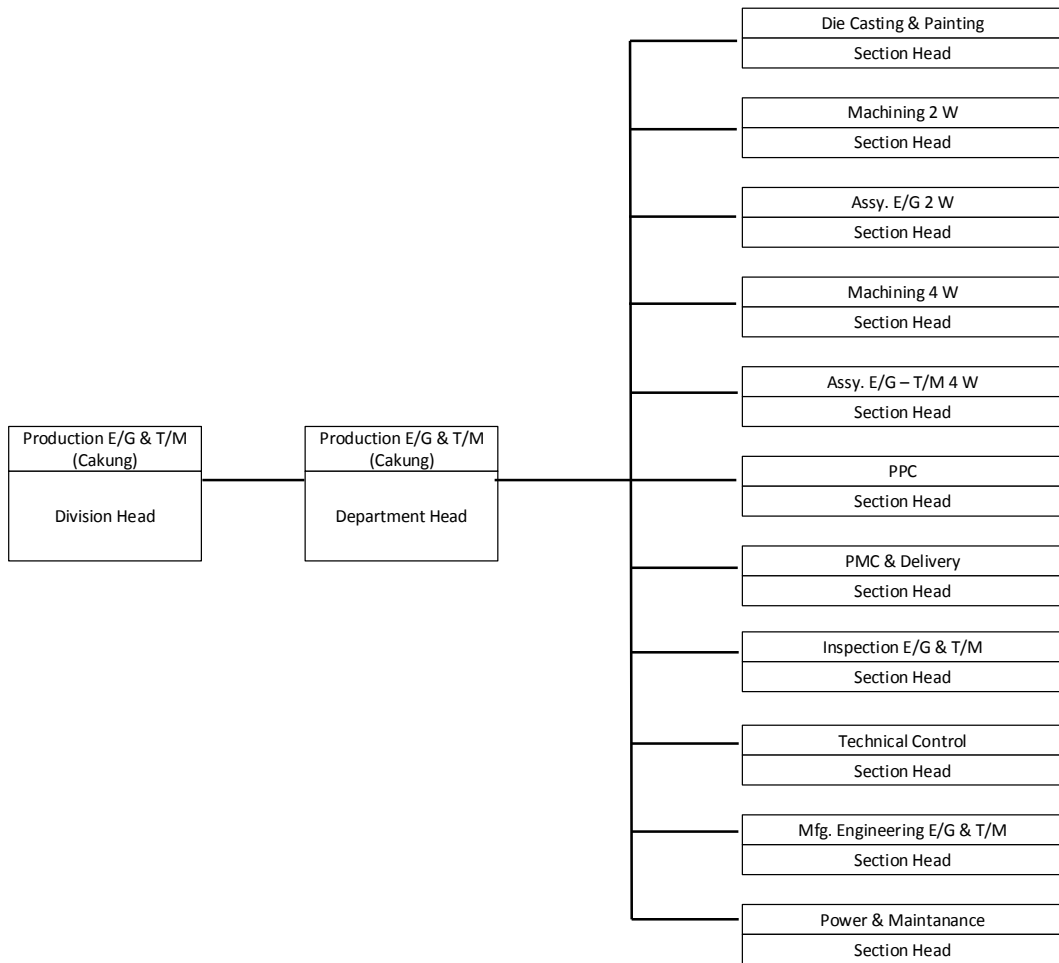
diperlukan struktur organisasi. Struktur organisasi merupakan suatu tingkatan dari suatu fungsi dan tanggung jawab dalam sebuah instansi, dimana memperlihatkan hubungan garis wewenang dan pertanggungjawaban antara jabatan dan peranan dalam suatu organisasi.

PT Suzuki Indomobil Motor memiliki struktur organisasi perusahaan mulai dari struktur organisasi PT Suzuki Indomobil Motor secara umum sampai dengan struktur organisasi pada level seksi khususnya seksi *Planning Material Control*, seperti yang tertera pada gambar-gambar dibawah ini:



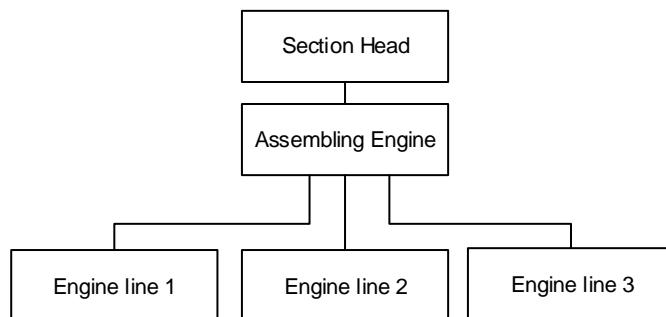
Gambar IV.1 Struktur Organisasi PT SIM

Sumber: PT Suzuki Indomobil Motor (2015)



Gambar IV.2 Struktur Organisasi PT SIM *Plant Cakung*

Sumber: PT Suzuki Indomobil Motor (2015)



Gambar IV.3 Struktur Organisasi divisi Seksi *Assembling 4W*

Sumber: PT Suzuki Indomobil Motor (2014)

Berikut adalah tugas dan tanggung jawab dari seksi *Assembling Engine 4W* antara lain:

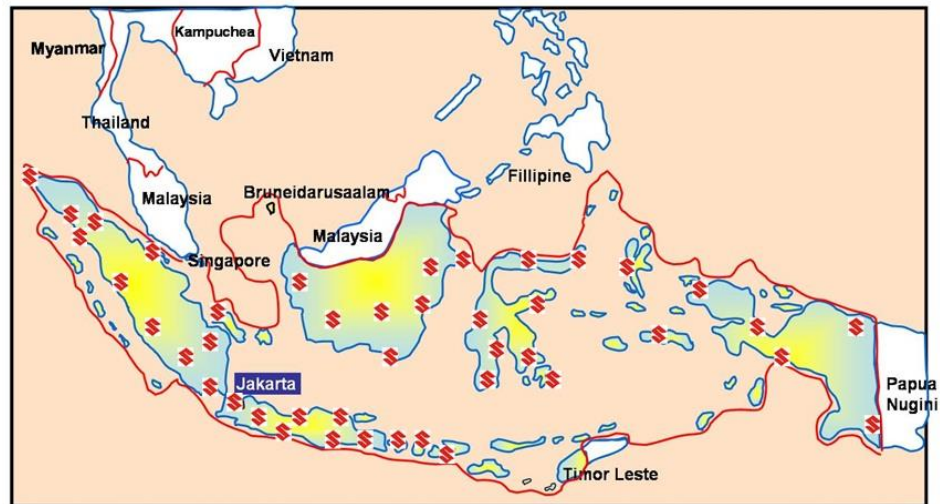
1. Merakit setiap komponen yang ada menjadi sebuah *engine 4W*.
2. Memastikan bahwa setiap *engine* yang dibuat tidak ada yang cacat.
3. Setiap komponen akan di cek jumlahnya oleh operator.
4. Pelaksanaan kegiatan produksi harus selalu memperhatikan aspek kesehatan dan keselamatan kerja.

Berdasarkan tugas dan tanggung jawab Seksi *Assembling Engine 4W* yang telah dijelaskan diatas, berikut ini merupakan tugas dan tanggung jawab dari setiap aktor yang ikut berperan dalam proses produksi, antara lain:

1. *PPC (Production Planning and Control)*
PPC bertugas untuk mengatur jadwal produksi dalam menentukan hasil jumlah unit produksi dengan memperhatikan *inventory* yang ada sehingga terdapat kesesuaian jumlah unit yang diproduksi.
2. *PMC (Production Material Control)*
PMC bertugas untuk mengatur mengatur dan menyiapkan semua kelengkapan dan penggunaan yang dibutuhkan di *section Assembling*. Bagian PMC ini juga yang mengatur masuknya komponen dari vendor serta pemberian komponen ke area produksi.
3. *Assembling*
Mengurusi semua kegiatan perakitan dan mempunyai tiga *line* yang dibagi menurut tipe mobil.
4. *Suplier*
Perusahaan yang bekerja sama untuk mengirimkan komponen yang dibutuhkan untuk proses produksi.

4.1.4 Jaringan Distribusi

PT Suzuki Indomobil Motor memiliki lebih dari 1300 dealer di Indonesia, diseluruh dunia terdapat kurang lebih 36 pabrik manufaktur, dengan jaringan distribusi kurang lebih 23 negara.



Gambar IV.4 Jaringan Distribusi Suzuki Group

Sumber: PT Suzuki Indomobil Motor (2015)

4.2 Proses *Assembly Engine*

Proses *assembly engine* 4W di PT Suzuki Indomobil Motor terdiri dari tiga bagian yaitu:

1. Proses *Assy Counter Line*

Proses *assy counter line* merupakan proses *assembly engine* 4W yang melibatkan bagian luar *engine* seperti *cylinder head*, *cover cylinder head*, *crankcase*, *cover timing*, *pully cranksaft*, *injector*, *intake MNT*, *knalpot*, *monti*, *alternator* dan *pan oil*. Hasil *assy counter line* diperlihatkan pada Gambar IV.5.



Gambar IV.5 Hasil *Assy Counter Line* PT Suzuki Indomobil Motor

(Sumber: PT Suzuki Indomobil Motor, 2015)

2. Proses Assy Main Line

Proses *assy main line 1* merupakan proses *assembly engine 4W* yang melibatkan bagian dalam *engine*, seperti *ring pistone*, *piston*, *counter bolt*, *sealant pant oil*, *crankshaft*, *disc cluth*, *cover cluth*, *valve* dan *plat housing*. Hasil *assy main line 1* diperlihatkan pada Gambar IV.6.

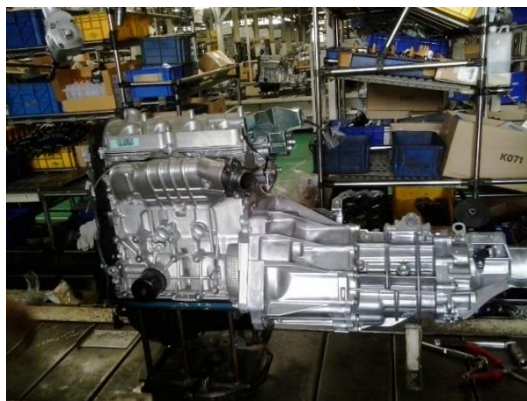


Gambar IV.6 Hasil Assy Main Line 1 PT Suzuki Indomobil Motor

(Sumber: PT Suzuki Indomobil Motor, 2015)

3. Proses Assy Main Line 2

Proses *assy main line 2* merupakan proses *assembly engine 4W* yang melibatkan bagian *finishing engine*, seperti *transmisi*. Hasil *assy main line 2* diperlihatkan pada Gambar IV.7.



Gambar IV.7 Hasil Assy Main Line 2 PT Suzuki Indomobil Motor

(Sumber: PT Suzuki Indomobil Motor, 2015)

4.3 Sistem Informasi yang Berjalan

Pada seksi *assembling* 4W, dalam melaksanakan penerimaan komponen *assembling* masih dilakukan dengan penulisan manual dan di *input* menggunakan *Microsoft Excel*, sehingga untuk melakukan *update* tiap harinya memerlukan waktu yang cukup lama. Penyebabnya dikarenakan data komponen mesin harus diurutkan berdasarkan nomor part dan namanya

Selanjutnya, jika melakukan permintaan komponen *engine* yang kurang harus atau perlu pergantian maka pihak *assembling* harus menggunakan laporan *form* permintaan barang kepada PMC (*Production Material Control*). Terkadang terjadi kesalahan penulisan permintaan komponen karena dikarenakan kesalahan operator dan sulitnya melakukan pergantian dikarenakan harus membawa komponen yang bermasalah.

4.3.1 Arus Dokumen

Dokumen yang terlibat dalam sistem informasi penerimaan kualitas *assembling* adalah sebagai berikut:

1. Form Permintaan Barang

Form laporan kerusakan dan bon permintaan barang adalah lembar kertas yang berisikan bon jumlah komponen dan jenis kerusakan pada setiap *engine*.

LAPORAN KERUSAKAN
 BON PERMINTAAN BARANG

No. 9-5-2016 Bagian: Assy 4W Cost Center: 923

NO	TYPE	NO. BARANG	NAMA BARANG	LOKAS SUBJECT										QUANTITY		AREA TEMUAN	JENIS KERUSAKAN		
				OH	DC	MC	PT	AB	DL	SU	GA	TS	RUSAK	PERISAL/PERISAL					
1		0150-0600	Selt Oil Seal Yonster													18	18	Assy/Ga	Seraf
2		0150-0600	Dish Generator													12	12	Assy/Ga	Penyok

DITRIMBAH OLEH: _____ DITERIMA OLEH: [Signature] BAGIAN PEMOHOK: _____ MENGETAHUI: _____

SAL. GUCANG: _____ PEMOHOK: _____ KAPOR / KASIB: _____ KAGE: _____ SKM / DM / DK: _____ BAGIAN GA: _____

Keterangan:

1. PUS - Gudang

2. Mada - Perawatan

3. Haki - Accounting

4. Kemas - GA

 Keterangan Internal:

OH = Old House proses Blue PT SM

AB = Assembling

DL = Delivery

SU = Supply

 Kriteria Laporan:

1. Proses - Proses Subalternasi

2. Proses - Proses Subalternasi

3. Proses - Proses Subalternasi

4. Proses - Proses Subalternasi

5. Proses - Proses Subalternasi

6. Proses - Proses Subalternasi

7. Proses - Proses Subalternasi

8. Proses - Proses Subalternasi

9. Proses - Proses Subalternasi

10. Proses - Proses Subalternasi

 Catatan:

Area Temuan adalah bagian dari proses di dalam lokal, meliputi Project (DP / Sub Process)

Gambar IV.8 Laporan kerusakan dan Bon Permintaan Barang

(Sumber: PT Suzuki Indomobil Motor, 2015)

2. Laporan Permintaan Komponen

Laporan permintaan komponen merupakan data *form* laporan permintaan komponen yang belum diolah.

NO.	VENDOR	NO. PART	NAMA PART	Q'TY
1	ADYAWINSA DINAMIKA KARAWANG, PT	33228-61J20	COVER COMP,CAM SENSOR	3192
2	ADYAWINSA DINAMIKA KARAWANG, PT	33228-61J30	LID,CAM SENSOR COVER	3192
3	ARAI RUBBER SEAL INDONESIA, PT	12934-67LA0	SEAL,VALVE STEM	110256
4	ART PISTON INDONESIA, PT	12111-61J10-0A0	PISTON [MERAH]	3584
5	ART PISTON INDONESIA, PT	12111-61J10-0B0	PISTON [BIRU]	9376
6	ART PISTON INDONESIA, PT	12111-61J60-0A0	PISTON [MERAH]	4800
7	ART PISTON INDONESIA, PT	12111-61J60-0B0	PISTON [BIRU]	11392
8	AUTOMOTIVE FASTENERS AOYAMA, PT	01550-0830A	BOLT TIMING CHAIN TENSIONER	12500
9	AUTOMOTIVE FASTENERS AOYAMA, PT	01550-0835A	BOLT MUFFLER MOUNTING	22728
10	AUTOMOTIVE FASTENERS AOYAMA, PT	01550-0855A	BOLT CRANKCASE	25200
11	AUTOMOTIVE FASTENERS AOYAMA, PT	01550-0865A	BOLT GENERATOR BRACKET	6300
12	AUTOMOTIVE FASTENERS AOYAMA, PT	01550-0875A	BOLT,CYLINDER HEAD NO.2	7000
13	AUTOMOTIVE FASTENERS AOYAMA, PT	01550-1050A	BOLT,STARTING MOTOR	9750
14	ASTRA OTOPARTS TBK, PT	11170-61J02	COVER COMP,CYLINDER HEAD	3183
15	ASTRA OTOPARTS TBK, PT	11170-775A0	COVER COMP,CYLINDER HEAD	3984
16	BANDO INDONESIA, PT	17521-78A02	BELT, WATER PUMP	144
17	BANDO INDONESIA, PT	17521-86550	BELT,WATER PUMP	7187
18	CATURINDO AGUNG JAYA RUBBER, PT	09250-30003	PLUGINSPECTION TIMING	6684
19	CATURINDO AGUNG JAYA RUBBER, PT	18571-85R00	CAP, PURGE UNION	288
20	CHEMCO HARAPAN NUSANTARA, PT	17561-61J00	CAP,WATER THERMOSTAT	3215
21	CIKARANG PERKASA MFG, PT	17131-80000	HUB,ENGINE COOLING FAN	288
22	CIKARANG PERKASA MFG, PT	17570-80C00	CAP, WATER OUTLET	7086

Gambar IV.9 Laporan permintaan komponen

(Sumber: PT Suzuki Indomobil Motor, 2015)

3. Laporan Penerimaan komponen

Laporan penerimaam komponen merupakan data *form* laporan komponen yang belum diolah menggunakan *Microsoft Excel*.

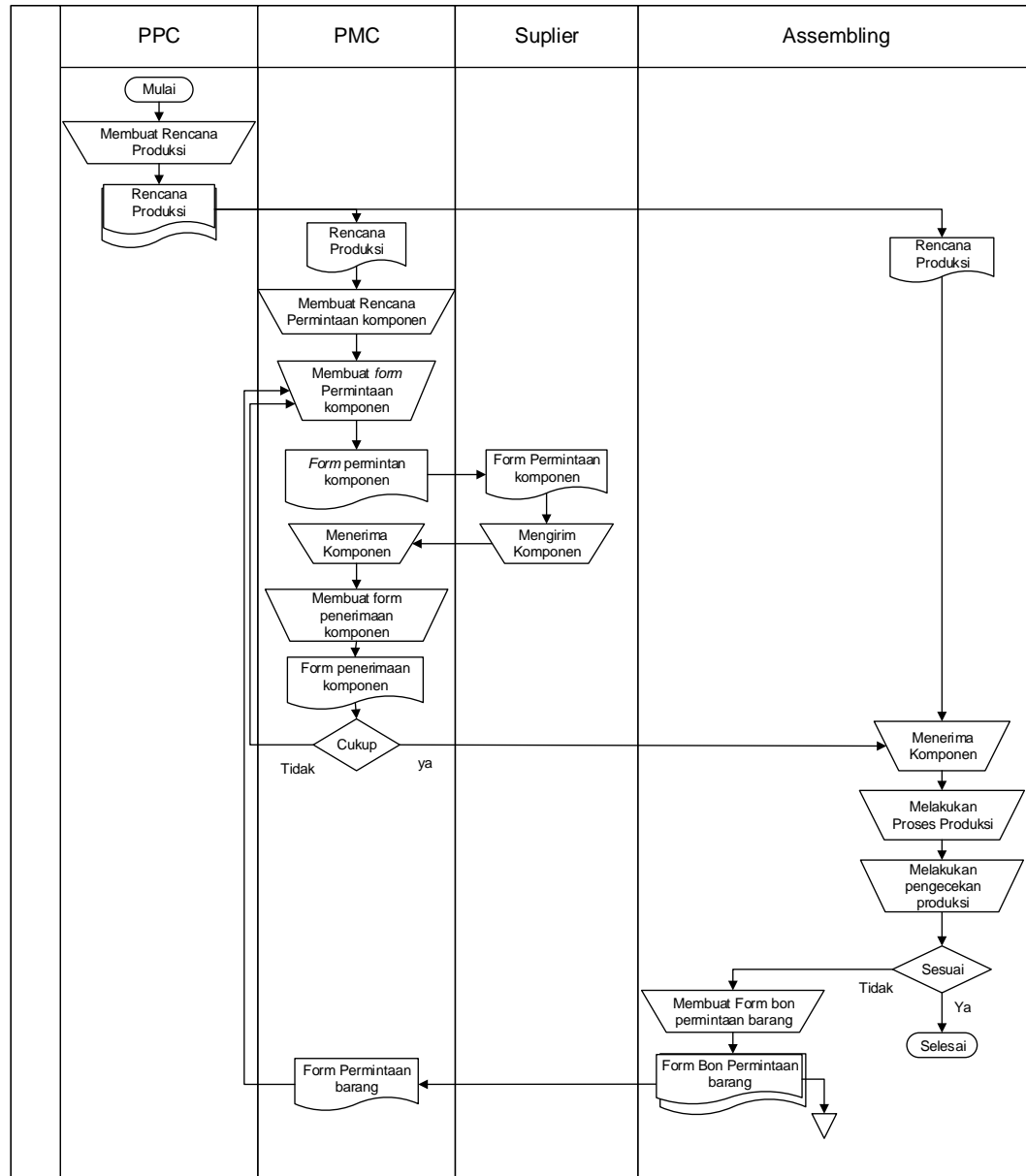
Gambar IV.10 Laporan penerimaan Komponen

(Sumber: PT Suzuki Indomobil Motor, 2015)

4.3.2 Alur Proses

Adapun prosedur sistem informasi penerimaan komponen pada seksi *assembling*:

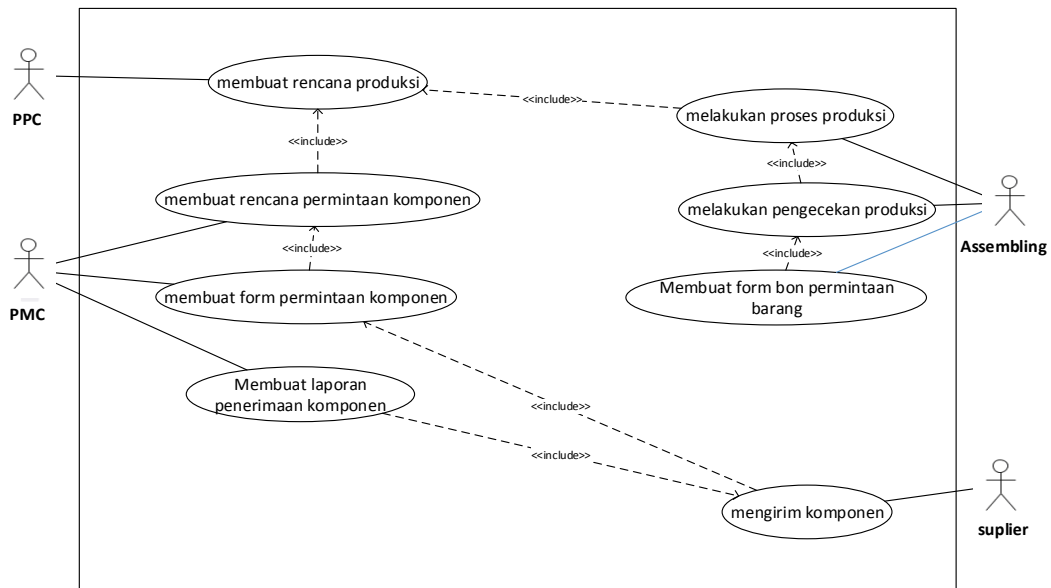
1. *Production Planning and Control* (PPC) memberikan *form* rencana produksi kepada seksi *Production Material Control* (PMC) dan seksi *Assembling*.
2. PMC membuat rencana permintaan komponen sesuai dengan tipe *engine* yang akan diproduksi.
3. *Supplier* mempersiapkan komponen yang telah dipesan.
4. *Supplier* mengirim komponen
5. PMC menerima komponen dan melakukan pemeriksaan terhadap kualitas dan kesesuaian jumlah komponen *engine* yang akan diproduksi.
6. PMC membuat form penerimaan komponen.
7. *Assembling* menerima komponen.
8. *Asembling* melakukan proses produksi.
9. *Assembling* melakukan pengecekan hasil produksi. jika lolos maka akan dikirimkan ke pihak tambun dan jika tidak lolos maka akan di cari tahu penyebabnya terlebih dahulu.
10. *Assembling* akan membuat *form* permintaan barang terhadap penerimaan komponen yang jumlahnya kurang dan perlu penggantian karena rusak saat proses diberikan kepada bagian PMC.



Gambar IV.11 *Flowmap* Sistem yang berjalan
 Sumber: Hasil Pengolahan Data, 2015

4.3.3 Diagram *Use Case* Sistem Informasi yang Berjalan

Sistem informasi penerimaan komponen *assembling* 4W memiliki keterkaitan antar aktor di dalam seksi tersebut, diantaranya *Production Planning and Control* (PPC), *Production Material Control* (PMC), *Assembling* dan *Supplier*. Setiap aktor memiliki kepentingan-kepentingan yang berkaitan dengan sistem tersebut. Gambar IV.12 menjelaskan tentang penggambaran sistem informasi penerimaan komponen *Assembling* 4W berlangsung.



Gambar IV.12 Use Case Diagram Sistem Informasi Penerimaan Komponen
Assembling 4W

Sumber: Hasil Pengolahan Data, 2015

Berikut ini adalah Deskripsi *use case diagram* yang berjalan pada sistem informasi penerimaan komponen *Assembling 4W* di PT Suzuki Indomobil Motor.

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi penerimaan komponen *assembly engine 4W* dapat dilihat pada tabel IV.1 berikut.

Tabel IV.1 Definisi Aktor Use Case Diagram

No	Aktor	Deskripsi
1	PPC	Orang yang bertugas untuk melakukan perencanaan terhadap jumlah hasil produksi.
2	PMC	Orang yang bertugas untuk melakukan perencanaan permintaan komponen untuk pihak <i>assembling</i> .
3	<i>Assembling</i>	Orang yang bertugas merakit komponen menjadi sebuah <i>engine</i> .

Tabel IV.1 Definisi Aktor *Use Case Diagram*

4	Suplier	Orang yang mengirimkan komponen untuk proses produksi.
---	---------	--

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.2 Definisi *Use Case* Membuat Rencana Produksi

Nama Use Case	Membuat Rencana Produksi
<i>Primary Actor</i>	PPC
<i>Brief Description</i>	Membuat <i>Form</i> hasil produksi <i>assembling</i> berdasarkan perhitungan produksi normal
<i>Relationship</i>	<i>Include:-</i>
<i>Normal Flow of Events</i>	PPC membuat <i>Form</i> hasil produksi <i>assembling</i>

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.3 Definisi *Use Case* Membuat Rencana Permintaan Komponen

Nama Use Case	Membuat Rencana Permintaan Komponen
<i>Primary Actor</i>	PMC
<i>Brief Description</i>	Membuat rencana permintaan komponen sesuai dengan hasil jumlah rencana produksi
<i>Relationship</i>	<i>Include:</i> membuat rencana produksi
<i>Normal Flow of Events</i>	PMC membuat daftar <i>suplier</i> komponen berikut jumlahnya

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.4 Definisi *Use Case* Melakukan Proses Produksi

Nama Use Case	Melakukan Proses Produksi
<i>Primary Actor</i>	<i>Assembling</i>
<i>Brief Description</i>	Melakukan proses produksi perakitan <i>engine</i>
<i>Relationship</i>	<i>Include:</i> membuat rencana produksi
<i>Normal Flow of Events</i>	<i>Assembling</i> melakukan proses produksi

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.5 Definisi *Use Case* Melakukan Pengecekan produksi

Nama Use Case	Melakukan Pengecekan produksi
<i>Primary Actor</i>	<i>Assembling</i>
<i>Brief Description</i>	Melakukan pengecekan terhadap hasil produksi produksi perakitan <i>engine</i>
<i>Relationship</i>	<i>Include</i> : Melakukan proses produksi
<i>Normal Flow of Events</i>	<i>Assembling</i> melakukan proses produksi

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.6 Definisi *Use Case* Membuat *Form* Permintaan Komponen

Nama Use Case	Membuat <i>Form</i> Permintaan Komponen
<i>Primary Actor</i>	PMC
<i>Brief Description</i>	Membuat <i>form</i> permintaan komponen
<i>Relationship</i>	<i>Include</i> : membuat rencana permintaan komponen
<i>Normal Flow of Events</i>	PMC membuat <i>form</i> permintaan komponen untuk diberikan kepada <i>supplier</i>

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.7 Definisi *Use Case* Membuat laporan penerimaan komponen

Nama Use Case	Membuat laporan komponen
<i>Primary Actor</i>	PMC
<i>Brief Description</i>	Membuat laporan berdasarkan komponen yang lulus pengecekan.
<i>Relationship</i>	<i>Include</i> : mengirim komponen
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. PMC melakukan bongkar muat dan mengecek dokumen 2. PMC memeriksa komponen untuk dipisahkan komponen yang rusak dan yang baik 3. PMC membuat laporan komponen

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.8 Definisi *Use Case* Memngirim komponen

Nama Use Case	Mengirim Komponen
<i>Primary Actor</i>	<i>Suplier</i>
<i>Brief Description</i>	Melakukan pengiriman komonen
<i>Relationship</i>	<i>Include: -</i>
<i>Normal Flow of Events</i>	<i>suplier</i> mengirim komponen

(Sumber: Hasil Pengolahan Data, 2015)

Tabel IV.9 Definisi *Use Case* Membuat form bon permintaan barang

Nama Use Case	Membuat form bon permintaan barang
<i>Primary Actor</i>	<i>Assembling</i>
<i>Brief Description</i>	Membuat <i>form</i> laporana bon permintaan barang
<i>Relationship</i>	<i>Include: melakukan pengecekan produksi</i>
<i>Normal Flow of Events</i>	Jumlah komponen yang kurang dan hasil produksi yang gagal diperbaiki dibuatkan <i>form</i> Laporan permintaan barang.

(Sumber: Hasil Pengolahan Data, 2015)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Sistem Informasi Berjalan

Berdasarkan hasil pengumpulan dan pengolahan data yang dilakukan pada sistem informasi yang sedang berjalan di Departemen Produksi Seksi *Assembling Engine 4W* PT Suzuki Indomobil Motor, dapat disimpulkan bahwa Departemen Produksi Seksi *Assembling Engine 4W* tidak memiliki sistem informasi terkomputerisasi yang khusus untuk menangani kebutuhan informasi penerimaan komponen *assembling engine 4W*.

5.2 Analisis Kebutuhan Sistem

Berdasarkan hasil analisis sistem informasi yang berjalan, maka diajukan sistem informasi usulan yang khusus untuk menangani kebutuhan penerimaan komponen di Departemen Produksi Seksi *Assembling Engine 4W*. Berikut ini adalah Table 5.1 analisis kebutuhan sistem informasi penerimaan komponen di Departemen Produksi Seksi *Assembling Engine 4W* PT Suzuki Indomobil Motor.

Tabel V.1 Kebutuhan Sistem Informasi Usulan

Kebutuhan Sistem	
Nama Proyek	Sistem informasi penerimaan komponen
Proses Sistem Berjalan	<ol style="list-style-type: none">1. Membuat laporan permintaan kompenen2. Membuat laporan penerimaan komponen3. Membuat laporan bon permintaan barang
Data Masukan (<i>input</i>)	<ol style="list-style-type: none">1. Data <i>user</i>2. Dats <i>Master</i>3. Data Permintaan komponen4. Data Penerimaan komponen5. Data Bon permintaan barang

Tabel V.1 Kebutuhan Sistem Informasi Usulan (Lanjutan)

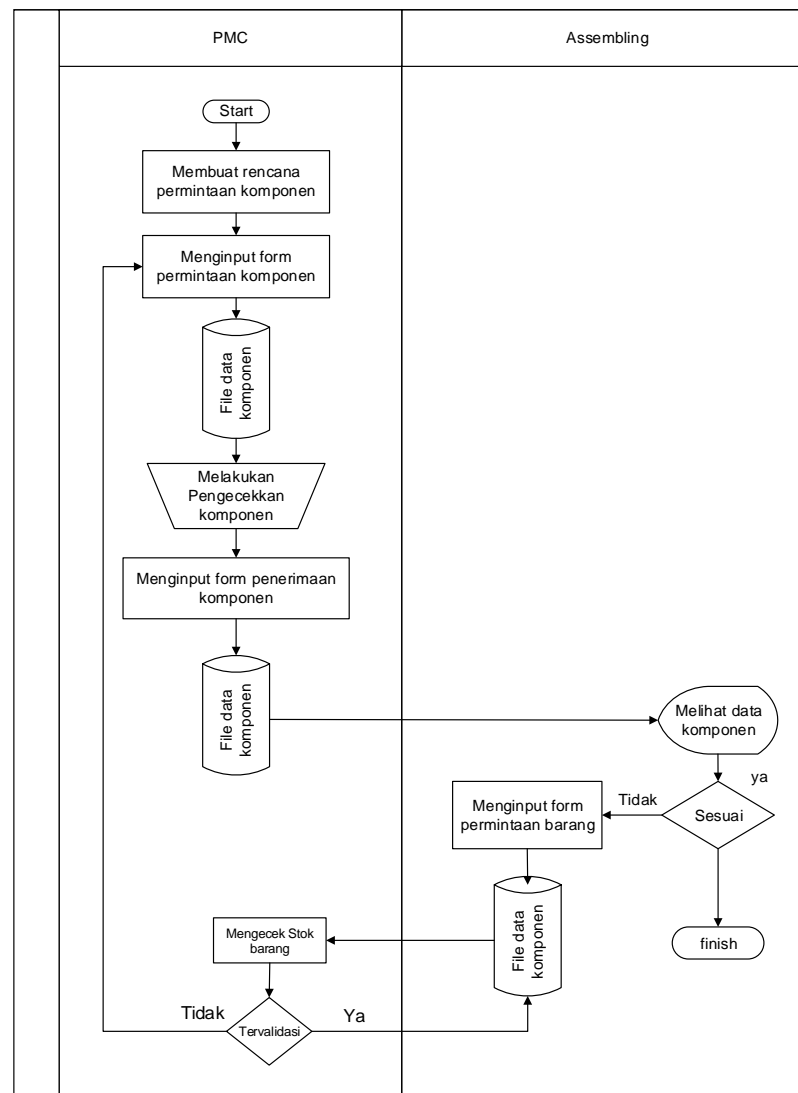
Data Keluaran (<i>output</i>)	<ol style="list-style-type: none"> 1. Informasi <i>user</i> 2. Informasi <i>master</i> 3. Informasi permintaan komponen 4. Informasi penerimaan komponen 5. Informasi permintaan barang
Model Sistem Informasi	<ol style="list-style-type: none"> 1. Input : Data <i>user</i> Proses : Mengelola data <i>user</i> Output : Informasi <i>user</i> 2. Input : Data <i>master</i> Proses : Mengelola data <i>master</i> Output : Informasi <i>master</i> 3. Input : Data permintaan komponen Proses : Mengelola data komponen yang akan dipesan Output : Informasi permintaan komponen 4. Input : Data penerimaan komponen Proses : Mengelola data komponen yang diterima Output : Informasi penerimaan komponen 5. Input : Data permintaan barang Proses : Mengelola data komponen yang bermasalah Output : Informasi permintaan barang

(Sumber: Hasil Analisis Data, 2016)

Flowmap sistem informasi penerimaan komponen *assembling engine 4W* yang diusulkan sebagai berikut:

1. PPC membuat rencana hasil produksi
2. PMC membuat rencana kedatangan komponen dan melakukan *input* form permintaan komponen
3. Data yang telah diinput masuk ke dalam *database* komponen.
4. PMc melakukan pengecekan komponen dan melakukan *input* form penerimaan komponen
5. Data yang telah diinput masuk ke dalam *database* komponen.

6. *Assembling* melihat data komponen untuk melakukan evaluasi apakah komponen yang tersedia mencukupi rencana produksi.
 7. *Assembling* melakukan *input* form permintaan barang
 8. Data yang telah diinput masuk ke dalam *database* komponen.
 9. PMC Mengecek stok barang untuk melakukan penggantian komponen bila tidak cukup maka PMC akan melakukan *input* form permintaan komponen
- Flowmap* sistem informasi penerimaan komponen *assembling engine 4W* usulan digambarkan dalam Gambar V.1.

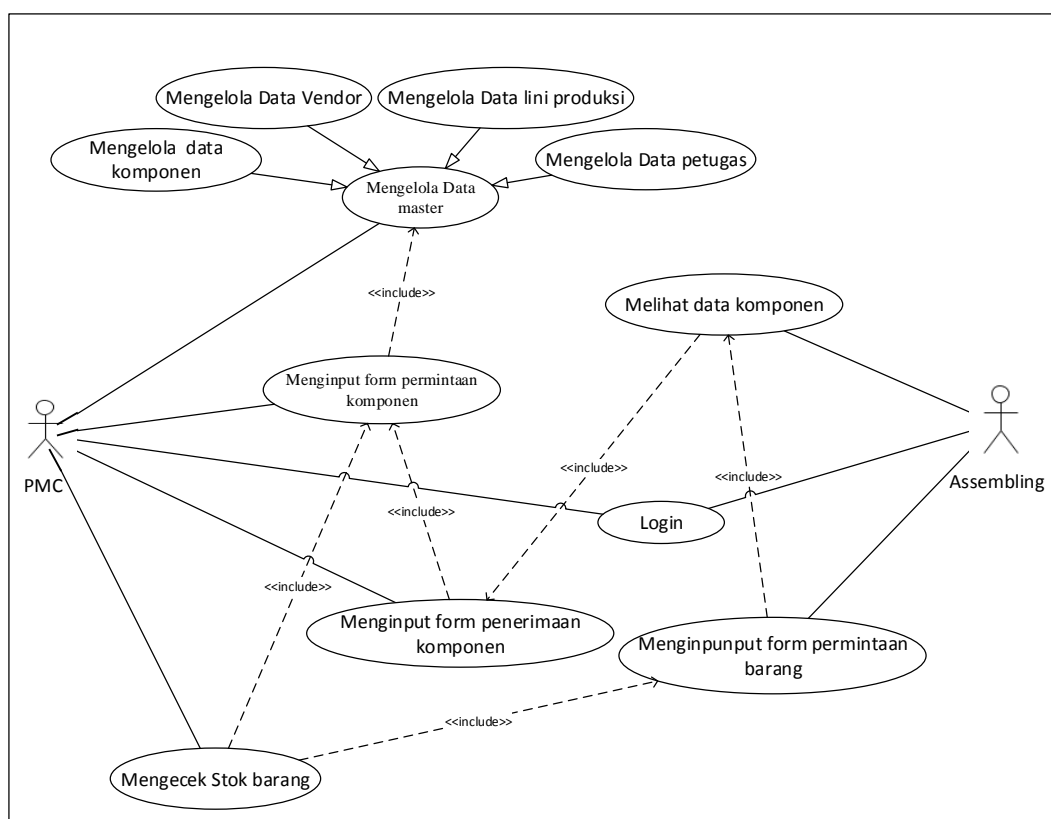


Gambar V.1 *Flowmap* Sistem Informasi Penerimaan Komponen *Assembly Engine 4W* (Sumber: Hasil Analisis, 2016)

5.2.1 Use Case Diagram Yang Diusulkan

Use Case Diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan aplikasi yang akan dibuat. *Use case diagram* ini memiliki 2 (dua) aktor yaitu PMC dan *Assembling*.

Untuk menjelaskan lebih detail tentang apa yang dapat dilakukan dalam sistem informasi penerimaan komponen, berikut merupakan *use case diagram* sistem yang diajukan.



Gambar V.2 Use Case Diagram Usulan
(Sumber: Hasil Analisis Data, 2016)

Berikut adalah skenario *use case diagram* sistem penerimaan komponen yang diusulkan:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi penerimaan komponen *assembly engine 4W* usulan dijelaskan pada Tabel V.2.

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

No	Aktor	Deskripsi
1	PMC	Orang yang bertugas untuk melakukan perencanaan permintaan komponen untuk pihak <i>assembling</i> .
2	<i>Assembling</i>	Orang yang bertugas merakit komponen menjadi sebuah <i>engine</i> .

(Sumber: Hasil Analisis, 2016)

2. Skenario *Use Case Diagram* Login Aktor

Tabel V.3 *Use Case Diagram* Login Aktor

<i>Use Case</i>	<i>Login Aktor</i>
<i>Actor</i>	PMC dan <i>Assembling</i>
<i>Description</i>	Sistem ini memungkinkan aktor mengakses sistem penerimaan komponen
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu <i>login</i> 2. Sistem merespon dengan menampilkan halaman untuk aktor melakukan <i>login</i>

(Sumber: Hasil Analisis Data, 2016)

3. Skenario *Use Case Diagram* Mengelola Data Master

Tabel V.4 *Use Case Diagram* Mengelola Data Master

<i>Use Case</i>	<i>Mengelola Data Master</i>
<i>Actor</i>	PMC
<i>Brief Description</i>	<i>Use case</i> ini memungkinkan PMC mengelola data master komponen, mengelola data master lini produksi, mengelola data vendor dan mengelola data petugas
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu data master 2. Aktor memilih data <i>user</i>, data komponen, data lini produksi dan data vendor 3. Kemudian aktor menginput data kedalam form yang disediakan

Tabel V.4 *Use Case Diagram* Mengelola Data Master (Lanjutan)

	4. Aktor dapat melakukan <i>update</i> dan menghapus data tersebut.
--	---

(Sumber: Hasil Analisis Data, 2016)

4. Skenario *Use Case Diagram* Menginput form permintaan komponenTabel V.5 *Use Case* Menginput form permintaan komponen

<i>Use Case</i>	Menginput form permintaan komponen
<i>Actor</i>	PMC
<i>Brief Description</i>	<i>Use case</i> ini memungkinkan PMC membuat jadwal kebutuhan komponen
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih form permintaan 2. Aktor memilih data baru untuk melakukan input jumlah komponen berdasarkan vendor yang telah tersedia sesuai yang di butuhkan 3. Aktor menyimpan data dan dapat menghapus data tersebut.

(Sumber: Hasil Analisis Data, 2016)

5. Skenario *Use Case Diagram* Menginput form penerimaan komponenTabel V.6 *Use Case Diagram* Menginput form penerimaan komponen

<i>Use Case</i>	Menginput form penerimaan komponen
<i>Actor</i>	PMC
<i>Description</i>	<i>Use case</i> ini memungkinkan PMC menginput form penerimaan komponen
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih form penerimaan 2. Aktor memilih data baru untuk melakukan input jumlah komponen berdasarkan tipe yang telah tersedia sesuai yang di butuhkan 3. Aktor menyimpan data dan dapat menghapus data tersebut.

(Sumber: Hasil Analisis Data, 2016)

6. Skenario *Use Case Diagram* Melihat data komponenTabel V.7 *Use Case Diagram* Melihat data komponen

<i>Use Case</i>	Melihat data komponen
<i>Actor</i>	<i>Assembling</i>
<i>Description</i>	<i>Use case ini memungkinkan Assembling melihat jumlah komponen dan data transaksi penerimaan komponen.</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu <i>Assembling</i> 2. Aktor hanya bisa melihat dan tidak dapat merubah data tersebut namun aktor bisa mencetak untuk bahan evaluasi.

(Sumber: Hasil Analisis Data, 2016)

7. Skenario *Use Case Diagram* Menginput form permintaan barangTabel V.8 *Use Case Diagram* Menginput form permintaan barang

<i>Use Case</i>	Menginput form permintaan Barang
<i>Actor</i>	<i>Assembling</i>
<i>Description</i>	<i>Use case ini memungkinkan Assembling untuk melakukan permintaan dan penggantian komponen</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor membuka menu form permintaan barang 2. Kemudian memasukan data komponen yang diinginkan 3. Aktor menyimpan data, dapat menghapus data dan dapat mencetak sebagai evaluasi.

(Sumber: Hasil Analisis Data, 2016)

8. Skenario *Use Case Diagram* Mengecek Stok komponenTabel V.9 *Use Case Diagram* Mengecek Stok komponen

<i>Use Case</i>	Mengecek Stok Komponen
<i>Actor</i>	PMC

Tabel V.9 *Use Case Diagram* Mengecek Stok komponen

<i>Description</i>	<i>Use case</i> ini memungkinkan bagi PMC yang ingin melihat komponen apa saja dibutuhkan <i>Assembling</i>
<i>Normal Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu form permintaan barang 2. Kemudian aktor dapat mengecek daftar komponen yang dibutuhkan <i>assembling</i>. 3. Aktor melakukan input permintaan komponen bila komponen tidak ada atau mulai sedikit

(Sumber: Hasil Analisis Data, 2016)

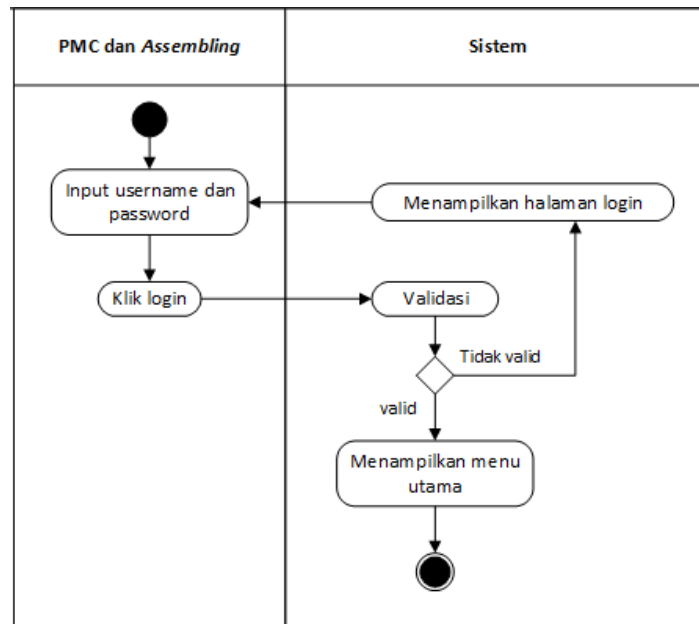
5.2.2 *Activity Diagram* Yang Diusulkan

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Berikut ini adalah *activity diagram* yang diperlukan untuk dmerancang dan membangun sistem informasi penerimaan komponen:

Activity diagram dibuat agar lebih memahami sistem yang akan dibuat. Berikut merupakan gambar *activity diagram* yang diusulkan untuk sistem informasi penerimaan komponen:

1. *Activity diagram Login*

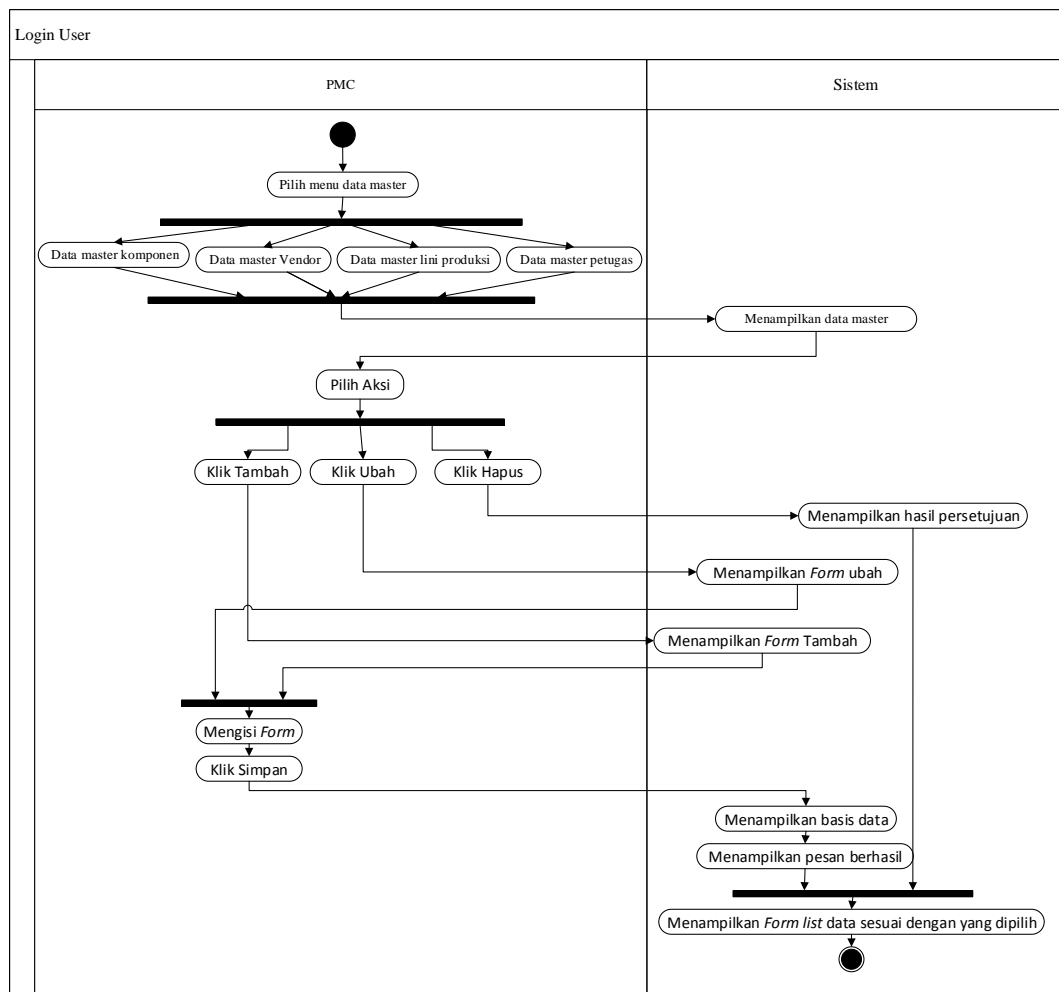
Activity diagram Login merupakan aktivitas awal untuk masuk kedalam sistem informasi penerimaan komponen. Tidak sembarang orang bisa masuk kedalam sistem aplikasi, urutan proses sistem dan kegiatan dalam melakukan *login* adalah dimulai dari penginputan *Username* dan *password* sampai dengan masuk ke halaman menu utama sistem informasi pelatihan karyawan. Berikut adalah Gambar V.3 *Activity Diagram Login User*.



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

2. Activity diagram Mengelola Data Master

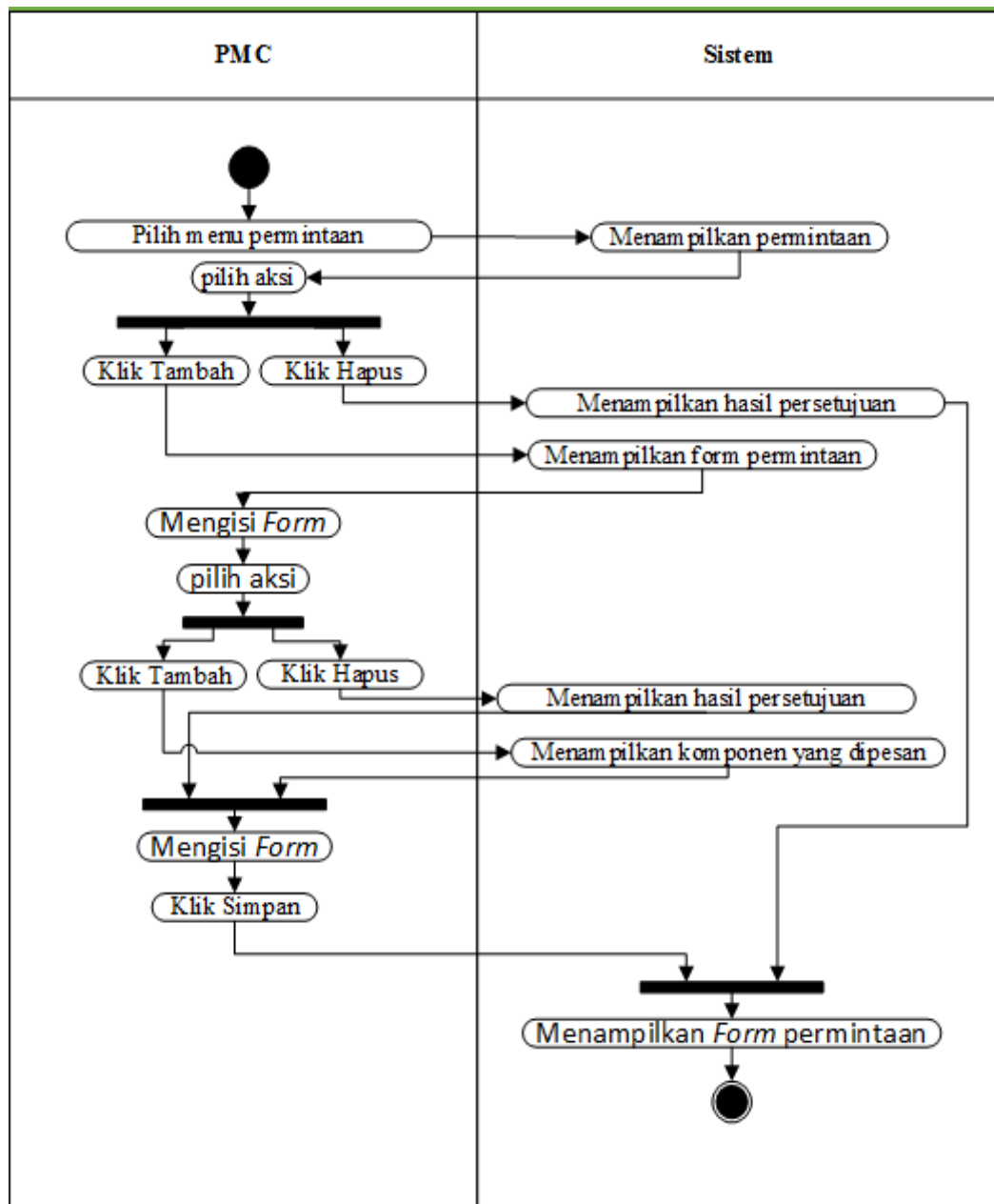
Activity berikut ini menjelaskan aktivitas yang dilakukan ketika *user* memilih menu data master. Dimana *user* dapat melakukan aktivitas *input*, *update*, dan *delete* data master. Berikut adalah Gambar V.4 Activity Diagram Mengelola Data Master.



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

3. Activity diagram menginput form permintaan komponen

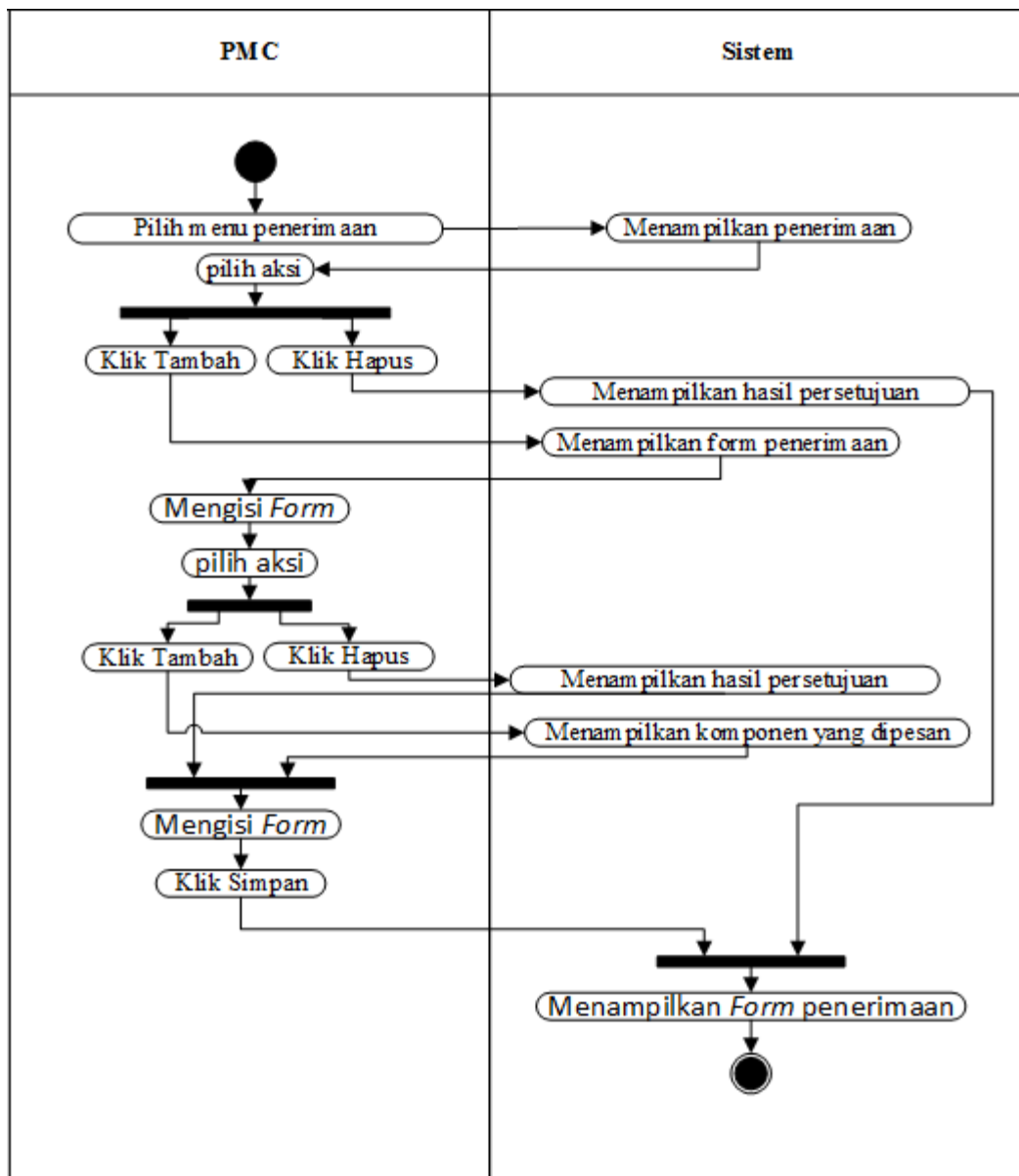
Activity diagram berikut ini menjelaskan aktivitas yang dilakukan ketika memilih menu permintaan komponen. Dimana *user* memasukan data-data yang dibutuhkan untuk membuat permintaan komponen. Hasil yang dikeluarkan berupa tabel sesuai dengan data yang diinputkan. Berikut adalah Gambar V.5 Activity Diagram permintaan komponen.



Gambar V.5 Activity Diagram Menginput form permintaan komponen
(Sumber: Hasil Analisis Data, 2016)

4. Activity diagram membuat form penerimaan komponen

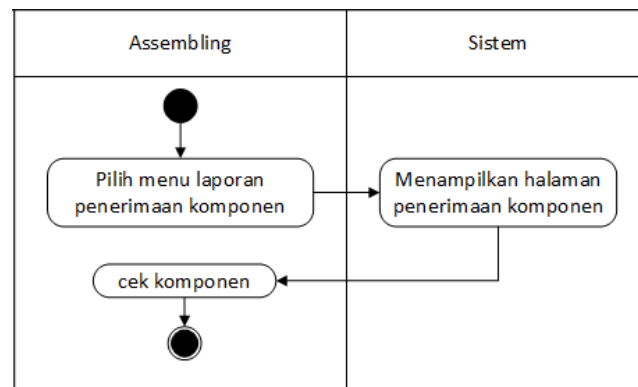
Activity diagram berikut ini menjelaskan aktivitas yang dilakukan ketika memilih menu penerimaan komponen. Dimana user memasukan data-data yang dibutuhkan berdasarkan komponen yang telah di cek. Hasil yang dikeluarkan berupa tabel sesuai dengan data yang diinputkan. Berikut ini adalah Gambar V.6 Activity Diagram Membuat form penerimaan komponen.



Gambar V.6 Activity Diagram form Penerimaan komponen
(Sumber: Hasil Analisis Data, 2016)

5. Activity diagram Melihat data komponen

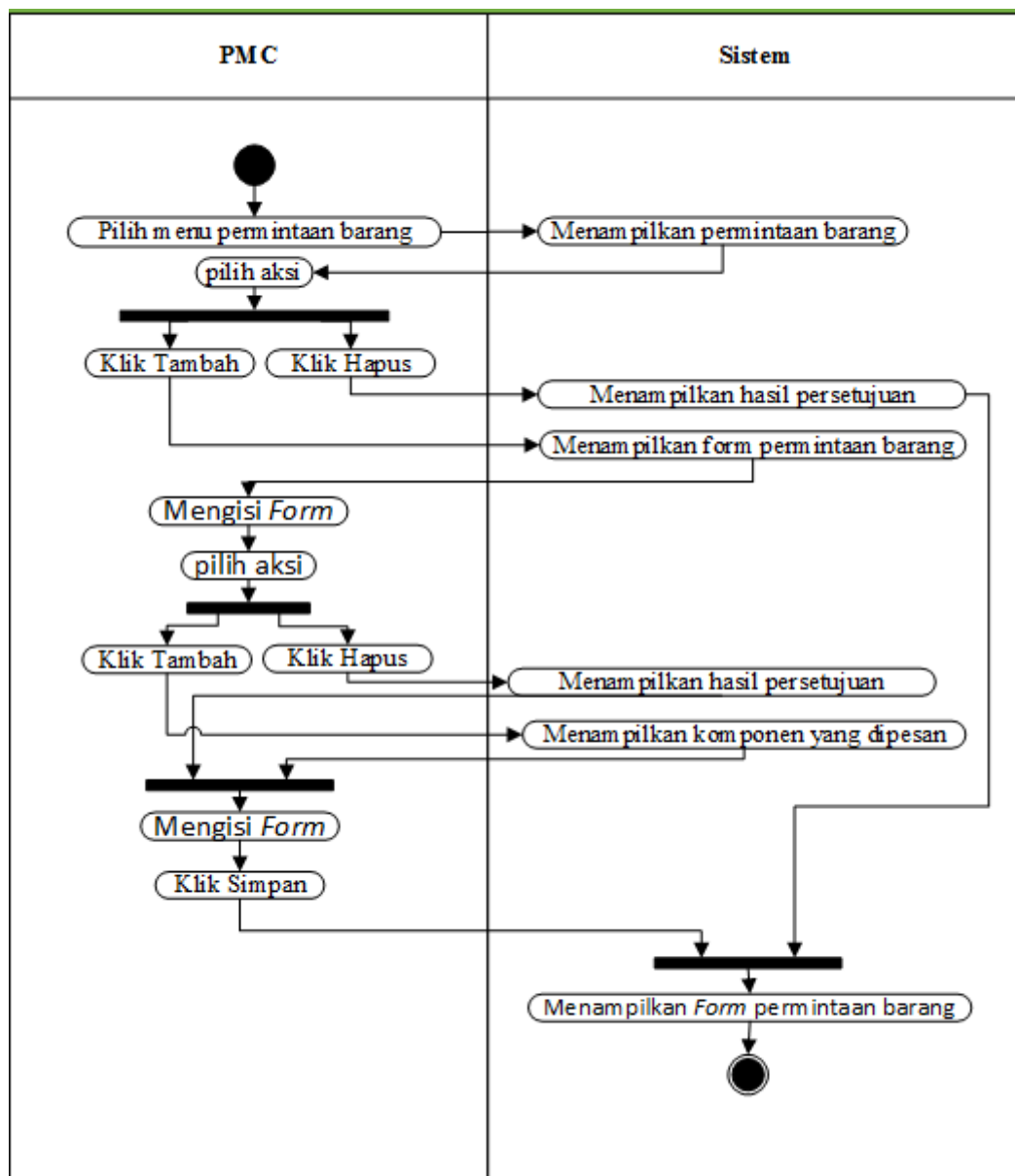
Activity diagram berikut ini menjelaskan aktivitas yang dilakukan ketika memilih menu laporan penerimaan komponen. Hasil yang dikeluarkan berupa tabel sesuai dengan data yang diinputkan. Berikut ini adalah Gambar V.7 Activity Diagram melihat data komponen



Gambar V.7 *Activity Diagram* Melihat data komponen
(Sumber: Hasil Analisis Data, 2016)

6. *Activity diagram* menginput form permintaan barang

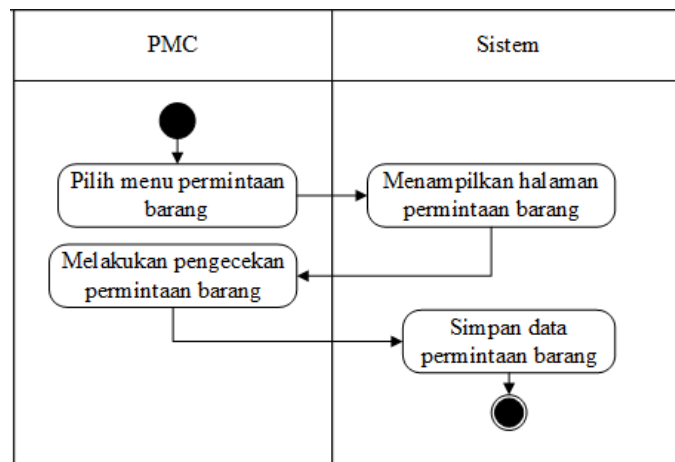
Activity berikut ini menjelaskan aktivitas yang dilakukan ketika *user* memilih menu form permintaan barang Berikut adalah Gambar V.8 *Activity diagram* Input form permintaan barang



Gambar V.8 *Activity Diagram* Input form permintaan barang
(Sumber: Hasil Analisis Data, 2016)

7. *Activity diagram* Mengecek stok barang

Activity berikut ini menjelaskan aktivitas yang dilakukan oleh PMC yang dilakukan saat mengecek jumlah komponen dan melakukan permintaan komponen bila terjadi kekurangan. Berikut adalah Gambar V.9 *Activity diagram* Mengecek stok barang.



Gambar V.9 Activity Diagram Mengecek stok barang
(Sumber: Hasil Analisis Data, 2016)

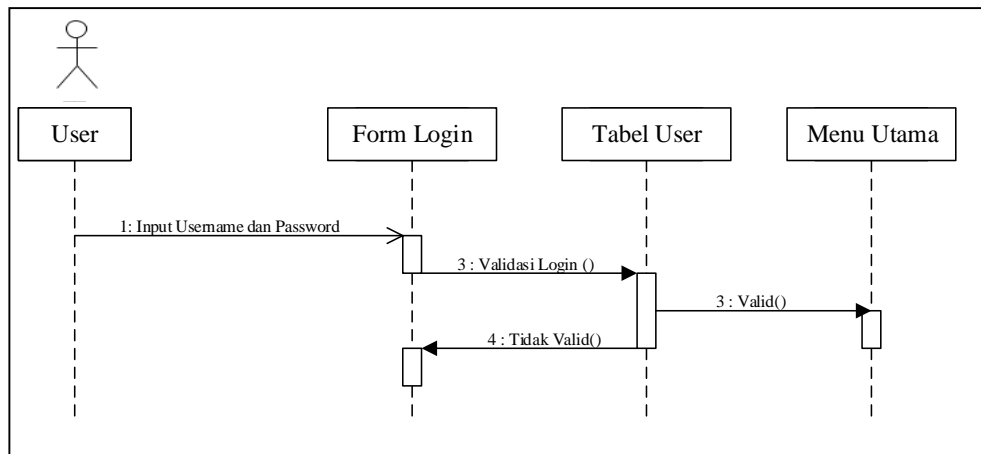
5.2.3 Sequence Diagram Yang Diusulkan

Sequence diagram menjelaskan secara detail urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari *use case*. *Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (pengguna, *display* dan sebagainya) berupa *message* yang digambarkan terhadap waktu.

Sequence diagram digunakan untuk memodelkan pengiriman *message* antar *objects*. Berikut merupakan *sequence diagram* yang diusulkan untuk sistem informasi pelatihan karyawan:

1. Sequence Diagram Proses Login

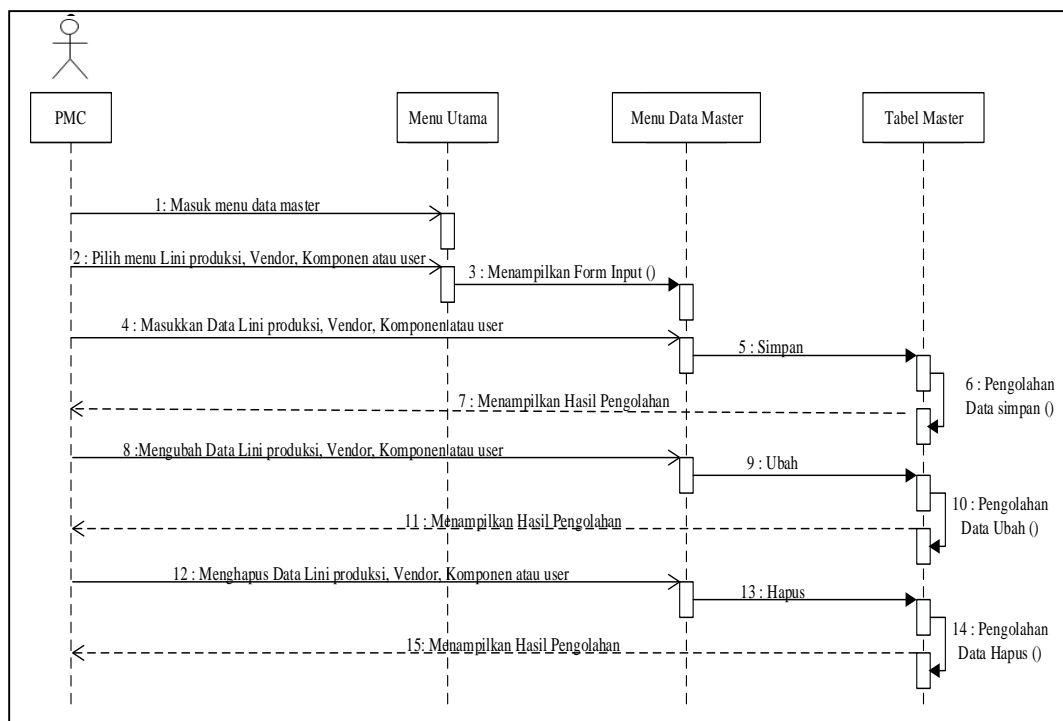
Sequence Diagram login menggambarkan proses yang terjadi dalam *login*. Prosesnya dimulai dengan masuk kedalam *form login* terlebih dahulu, kemudian selanjutnya *actor* dapat melakukan *login* dengan memasukkan *Username* dan *password* untuk dapat masuk kedalam sistem. Proses *login* itu sendiri diikuti juga dengan validasi yang dilakukan oleh sistem, dimana terdapat proses verifikasi *Username* dan *password*. Apabila validasi berhasil maka *actor* tersebut dapat masuk dan menggunakan sistem.



Gambar V.10 *Sequence Diagram* Proses Login
(Sumber: Hasil Analisis Data, 2016)

2. *Sequence Diagram* Mengelola Data Master

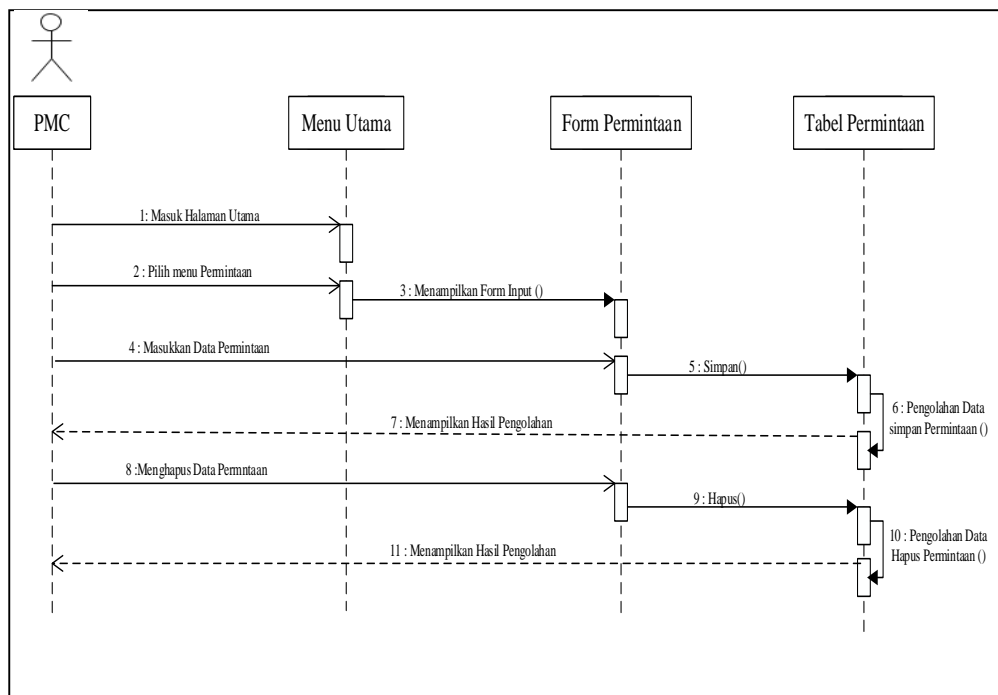
Sequence diagram mengelola data master dapat dilakukan apabila telah melalui proses *login*. Setelah itu masuk kedalam halaman utama dan memilih menu data master lalu mengisi data-data yang diperlukan, apabila data tidak lengkap maka sistem akan menolak menyimpan data tersebut.



Gambar V.11 *Sequence Diagram* Mengelola Data Master
(Sumber: Hasil Analisis Data, 2016)

3. *Sequence Diagram* Membuat Permintaan Komponen

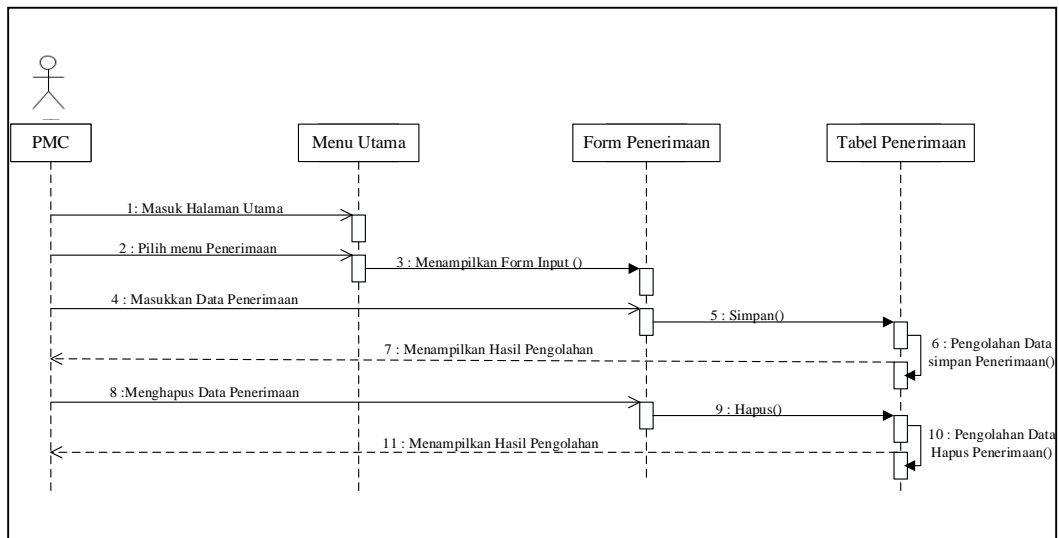
Sequence diagram input data permintaan komponen dapat dilakukan apabila telah melalui proses *login*. Setelah itu masuk kedalam halaman utama dan memilih menu permintaan kemudian mengisi data-data yang diperlukan, apabila data tidak lengkap maka sistem akan menolak menyimpan data tersebut.



Gambar V.12 *Sequence Diagram* Permintaan Komponen
(Sumber: Hasil Analisis Data, 2016)

4. *Sequence Diagram* Membuat Penerimaan Komponen

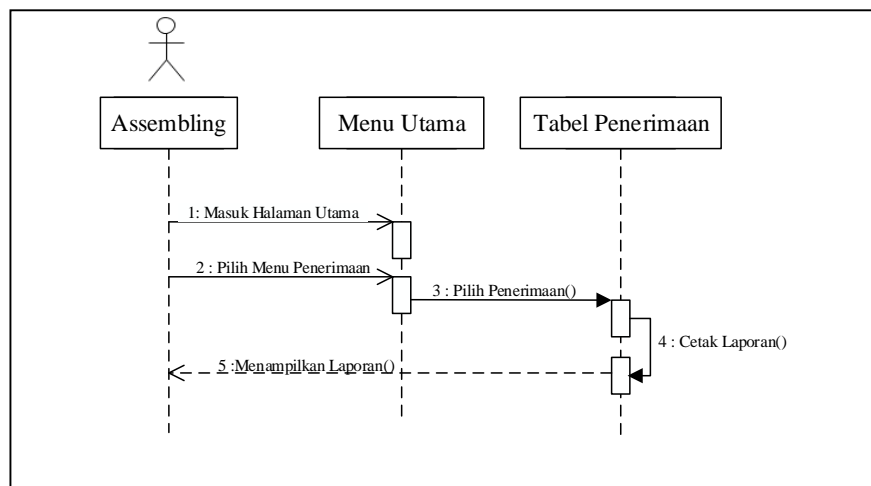
Sequence Diagram input data penerimaan komponen dapat dilakukan apabila telah melalui proses *login*. Setelah itu masuk ke dalam halaman utama dan memilih menu penerimaan, kemudian masukan data-data yang diperlukan, apabila data tidak lengkap maka sistem akan menolak menyimpan data tersebut.



Gambar V.13 *Sequence Diagram* Membuat Penerimaan Komponen
(Sumber: Hasil Analisis Data, 2016)

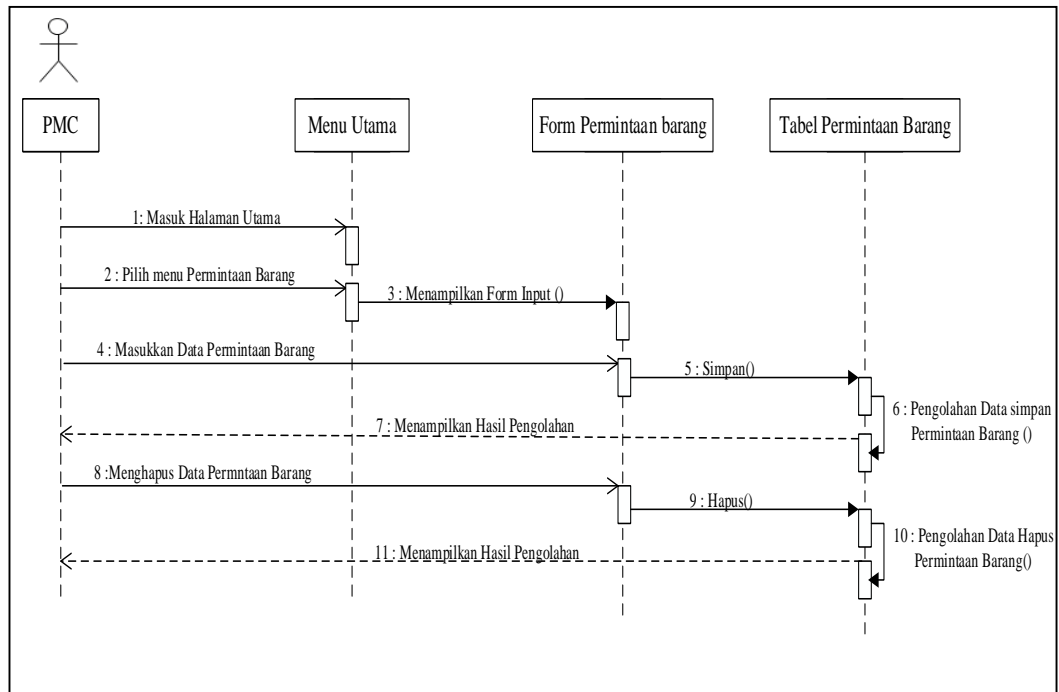
5. *Sequence Diagram* Melihat Penerimaan Komponen

Sequence Diagram input data daftar riwayat pelatihan dapat dilakukan apabila telah melalui proses *login*. Setelah itu masuk ke halaman utama dan memilih menu daftar riwayat pelatihan, kemudian melakukan penginputan data-data yang diperlukan. Apabila data tidak lengkap maka sistem akan menolak menyimpan data tersebut.



Gambar V.14 *Sequence Diagram* Melihat Penerimaan Komponen
(Sumber: Hasil Analisis Data, 2016)

6. *Sequence Diagram* Membuat Permintaan Barang

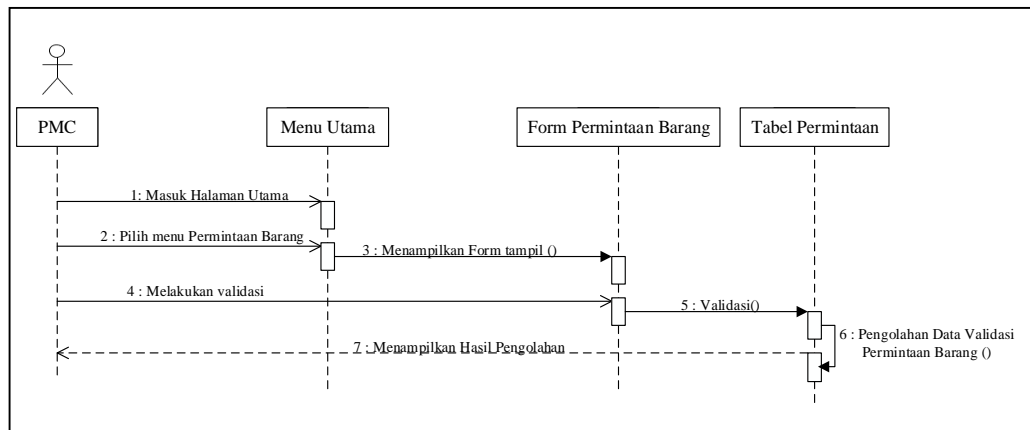


Gambar V.15 *Sequence Diagram* Permintaan Barang
(Sumber: Hasil Analisis Data, 2016)

Sequence Diagram input data penerimaan barang dapat dilakukan apabila telah melalui proses *login*. Setelah itu masuk kedalam halaman utama dan memilih menu penerimaan barang, kemudian masukan data-data yang diperlukan, apabila data tidak lengkap maka sistem akan menolak menyimpan data tersebut.

7. *Sequence Diagram* Mengecek Stok

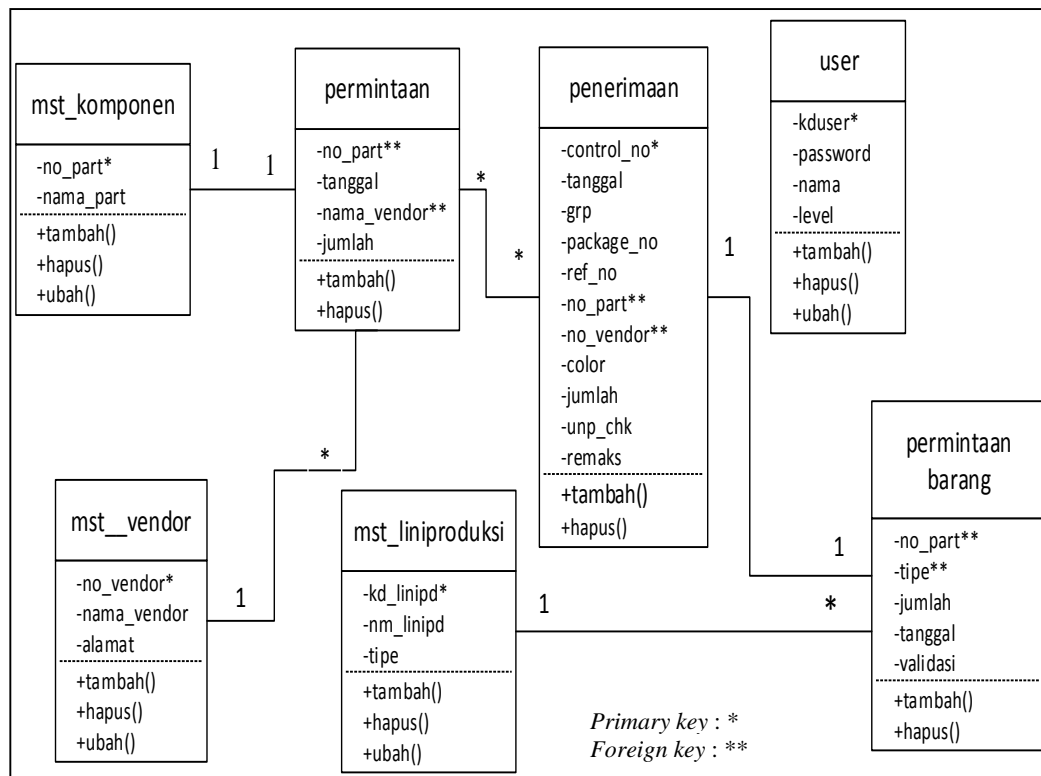
Sequence Diagram Mengecek Stok merupakan proses yang dilakukan oleh PMC untuk memberikan informasi terhadap jumlah komponen apakah di ganti atau melakukan penambahan jumlah komponen.



Gambar V.16 *Sequence Diagram* Mengecek Stok
(Sumber: Hasil Analisis Data, 2016)

5.2.4 *Class Diagram* Yang Diusulkan

Class diagram memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model desain dari suatu sistem. *Class Diagram* membantu dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Berikut merupakan *class diagram* dalam sistem informasi penerimaan Komponen.



Gambar V.17 *Class Diagram* Sistem Informasi Penerimaan Komponen
(Sumber: Hasil Analisis Data, 2016)

5.2.5 Kamus Data

Kamus data merupakan suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem. Pada tahap perancangan sistem kamus data digunakan untuk merancang *input* dan *database*. Dengan adanya kamus data pada perancangan dapat memudahkan membuat rancangan *database* untuk sistem aplikasi yang dibangun. Berikut ini merupakan kamus data yang terdapat pada sistem informasi penerimaan komponem:

1. Spesifikasi Tabel user

Nama Tabel : *user*

Fungsi : Untuk menyimpan data *user*

Tipe : *Data Master*

Tabel V.10 Spesifikasi Tabel *User*

No	Nama Elemen	Tipe	Panjang	Keterangan	Deskripsi
1	kduser	Varchar	10	<i>Primary Key</i>	Kode akses user
2	nama	Varchar	15		Nama user
3	level	Varchar	10		Nama seksi
4	password	Varchar	15		Kata kunci

(Sumber: Hasil Analisis Data, 2016)

2. Spesifikasi Tabel mst_komponen

Nama Tabel : *mst_komponen*

Fungsi : Untuk menyimpan data komponen

Tipe : *Data Master*

Tabel V.11 Spesifikasi Tabel *mst_komponen*

No	Nama Elemen	Tipe	Panjang	Keterangan	Deskripsi
1	no_part	Varchar	15	<i>Primary Key</i>	Nomor part
2	nama_part	varchar	30		Nama part

(Sumber: Hasil Analisis Data, 2016)

3. Spesifikasi Tabel mst_liniproduksi

Nama Tabel : mst_liniproduksi

Fungsi : Untuk menyimpan data lini produksi

Tipe : Data *Master*

Tabel V.12 Spesifikasi Tabel mst_liniproduksi

No	Nama Elemen	Tipe	Panjang	Keterangan	Deskripsi
1	kd_linipd	Char	7	<i>Primary Key</i>	Kode lini produksi
2	nm_linipd	varchar	8		Nama lini Produksi
3	tipe	Char	3		Tipe priduksi

(Sumber: Hasil Analisis Data, 2016)

4. Spesifikasi Tabel mst_vendor

Nama Tabel : mst_vendor

Fungsi : Untuk menyimpan data vendor

Tipe : Data *Master*

Tabel V.13 Spesifikasi Tabel mst_vendor

No	Nama Elemen	Tipe	Panjang	Keterangan	Deskripsi
1	no_vendor	Char	4	<i>Primary Key</i>	Nomor vendor
2	nama_vendor	Varchar	40		Nama vendor
3	alamat	Varchar	100		Alamat vendor

(Sumber: Hasil Analisis Data, 2016)

5. Spesifikasi Tabel permintaan

Nama Tabel : permintaan

Fungsi : Untuk menyimpan data permintaan komponen

Tipe : Data Transaksi

Tabel V.14 Spesifikasi Tabel permintaan

No	Nama Elemen	Tipe	Panjang	Keterangan	Deskripsi
1	no_part	Varchar	15	<i>Primary Key</i>	Nomor part
2	tanggal	Date			Tanggal
3	nama_vendor	varchar	40	<i>Foreign Key</i>	Nama vendor
4	jumlah	Int	7		Jumlah

(Sumber: Hasil Analisis Data, 2016)

6. Spesifikasi Tabel penerimaan

Nama Tabel : penerimaan

Fungsi : Untuk menyimpan data penerimaan komponen

Tipe : Data Transaksi

Tabel V.15 Spesifikasi Tabel Penerimaan

No	Nama Elemen	Tipe	Panjang	Keterangan
1	control_no	Char	3	<i>Primary Key</i>
2	Tanggal	Date		
3	Grp	Char	2	
4	package_no	char	6	
5	ref_no	Char	10	
6	no_part	varhar	15	<i>Foreign Key</i>
7	no_vendor	Char	4	<i>Foreign Key</i>
8	Color	varchar	5	
9	Jumlah	Int	4	
10	Package	Char	2	
11	unp_chk	Char	2	
12	Remaks	varchar	20	

(Sumber: Hasil Analisis Data, 2016)

7. Spesifikasi Tabel permintaan_barang

Nama Tabel : permintaan_barang

Fungsi : Untuk menyimpan data permintaan barang

Tipe : Data Transaksi

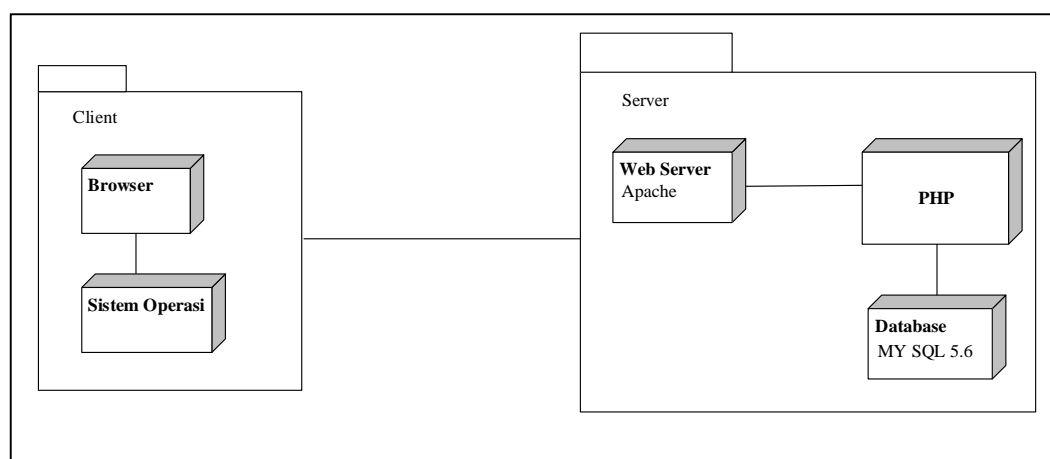
Tabel V.16 Spesifikasi Tabel permintaan_barang

No	Nama Elemen	Tipe	Panjang	Keterangan
1	no_part	Char	15	<i>Primary Key</i>
2	tipe	Char	3	<i>Foreign Key</i>
3	tanggal	Date		
4	jumlah	Int	7	
5	validasi	Varchar	11	

(Sumber: Hasil Analisis Data, 2016)

5.2.6 Deployment Diagram Yang Diusulkan

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Menggambarkan pembuatan sistem informasi penerimaan komponen berbasis *web* yang memanfaatkan PHP 5.5 dan MY SQL 5.6 sebagai desain sistemnya. Seperti yang dapat dilihat pada Gambar V.20 sebagai berikut:

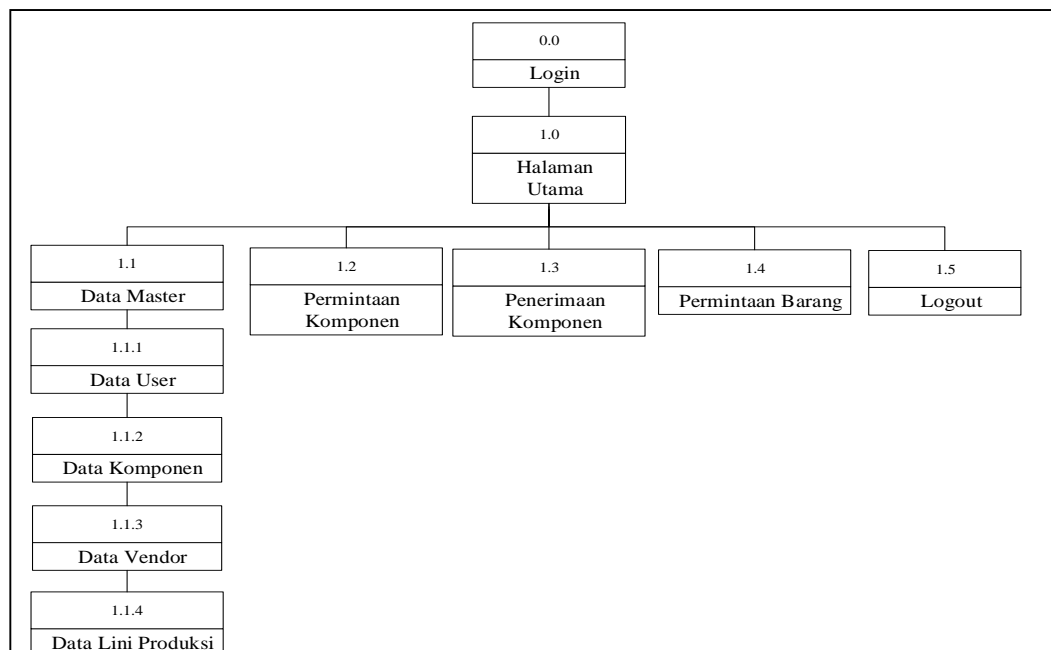


Gambar V.18 *Deployment Diagram* Yang Diusulkan

(Sumber: Hasil Analisis Data, 2016)

5.2.7 *Hierarchy plus Input Process Output (HIPO)*

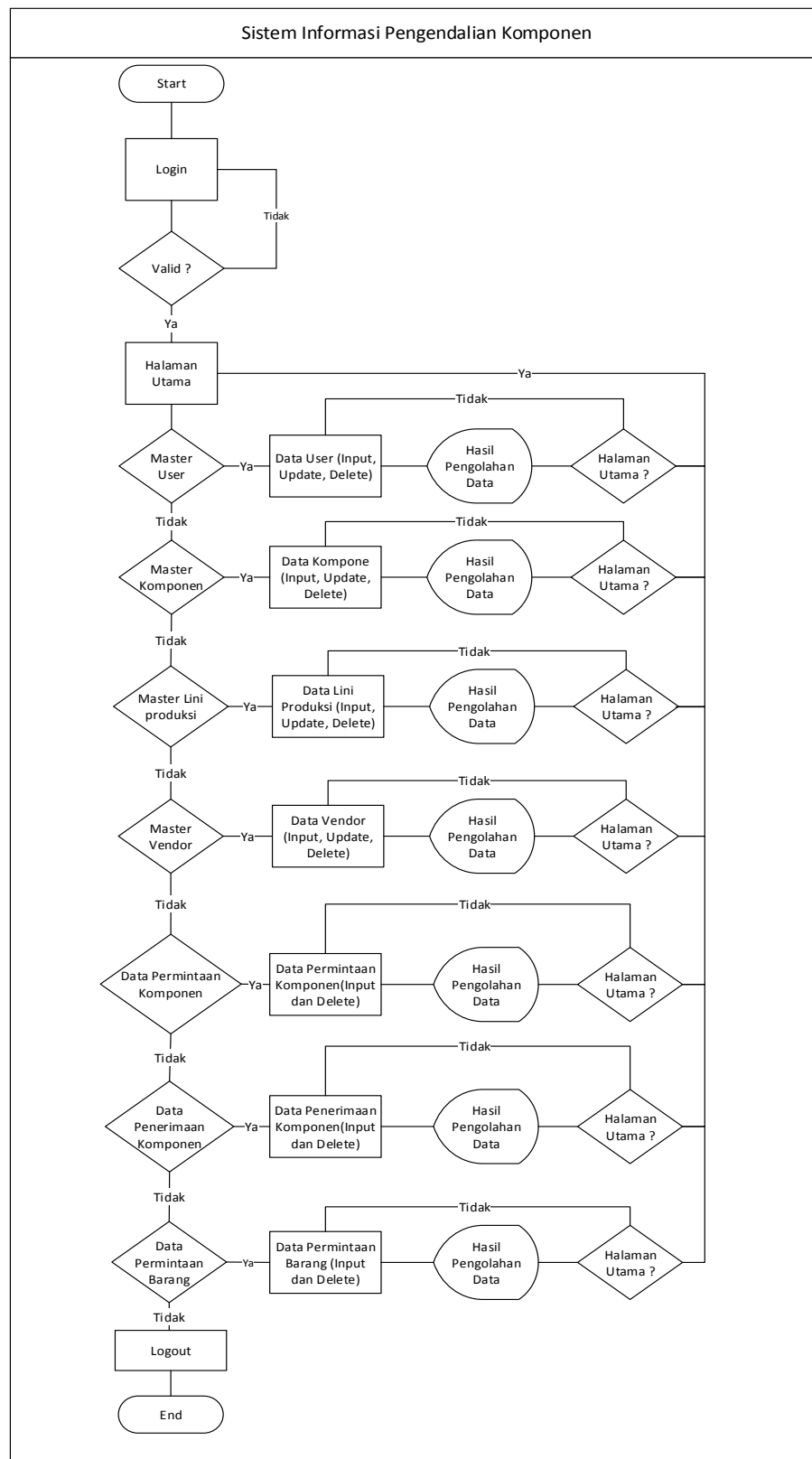
HIPO merupakan alat desain dan teknik dokumentasi dalam siklus pengembangan sistem. Dalam hal ini pembuatan sistem informasi pelatihan karyawan dibuat menggunakan perangkat lunak PHP untuk bahas pemrograman, serta MySQL yang berfungsi sebagai perangkat lunak basis data. Berikut adalah Gambar V.19 *Hierarchy plus Input Process Output (HIPO)* yang diusulkan.



Gambar V.19 HIPO Yang Diusulkan
(Sumber: Hasil Analisis Data, 2016)

5.2.8 *Flowchart Sistem Informasi Penerimaan Komponen*

Flowchart sistem informasi penerimaan komponen dapat menggambarkan alur logika yang sebenarnya. Bagian ini juga memperjelas urutan prosedur sistem dan spesifikasi proses. Berikut adalah *flowchart* sistem informasi pelatihan karyawan.



Gambar V.20 Flowchart Yang Diusulkan
(Sumber: Hasil Analisis Data, 2016)

5.2.9 Perancangan *Interface*

Perancangan *interface* merupakan gambaran dari program atau aplikasi yang diusulkan. Berikut ini merupakan perancangan *interface* sistem informasi penerimaan komponen:

1. Form *Login*

Form Login digunakan untuk membedakan hak akses pengguna. Melalui *Form login* ini pengguna yang boleh masuk sistem adalah pengguna yang mengetahui *Username* dan *password* atau pengguna yang memiliki wewenang untuk menggunakan sistem. Berikut adalah Gambar V.21 perancangan *form login*.

Login User
<div style="margin-bottom: 10px;"><input style="width: 80%; height: 20px;" type="text"/></div> <div style="margin-bottom: 10px;"><input style="width: 80%; height: 20px;" type="password"/></div> <div style="margin-bottom: 10px;"><input style="width: 40%; height: 20px; border-radius: 10px;" type="button" value="Login"/></div>

Gambar V.21 Perancangan Form *Login*
(Sumber: Hasil Analisis Data, 2016)

2. Halaman Utama

Halaman Utama berisi mengenai informasi sistem informasi penerimaan komponen. Berikut adalah Gambar V.22 perancangan halaman utama.

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR	
	<p>SELAMAT DATANG</p> <p>ID User :</p> <p>Posisi :</p> <p>Tanggal dan waktu</p>

Gambar V.22 Perancangan Halaman Utama
(Sumber: Hasil Analisis Data, 2016)

3. Halaman Data Master Komponen

Data Master Komponen berisi informasi mengenai komponen yang ada didalam *engine*. Berikut adalah Gambar V.23 perancangan *Form* data master komponen

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR												
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Data Master</div> <div style="border: 1px solid black; padding: 5px;">Komponen</div>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Komponen</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Tambah</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 20%;">No Part</th> <th style="width: 55%;">Nama Part</th> <th style="width: 20%;"></th> </tr> </thead> <tbody> <tr> <td style="height: 20px;"></td> <td></td> <td></td> <td style="text-align: right; vertical-align: bottom;">Ubah Hapus</td> </tr> </tbody> </table>				No	No Part	Nama Part					Ubah Hapus
No	No Part	Nama Part										
			Ubah Hapus									

Gambar V.23 Perancangan *Form* Data Master Komponen
(Sumber: Hasil Analisis Data, 2016)

4. Halaman Data Master Vendor

Data Master Vendor berisi informasi mengenai vendor yang mensupply komponen. Berikut adalah Gambar V.24 perancangan *form* data master Vendor.

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR														
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Data Master</div> <div style="border: 1px solid black; padding: 5px;">Vendor</div>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Vendor</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Tambah</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 20%;">No Vendor</th> <th style="width: 30%;">Nama Vendor</th> <th style="width: 35%;">Alamat</th> <th style="width: 10%;"></th> </tr> </thead> <tbody> <tr> <td style="height: 20px;"></td> <td></td> <td></td> <td></td> <td style="text-align: right; vertical-align: bottom;">Ubah Hapus</td> </tr> </tbody> </table>				No	No Vendor	Nama Vendor	Alamat						Ubah Hapus
No	No Vendor	Nama Vendor	Alamat											
				Ubah Hapus										

Gambar V.24 Perancangan *Form* Data Master Vendor
(Sumber: Hasil Analisis Data, 2016)

5. Halaman Data Master Lini Produksi

Data Master Lini Produksi berisi informasi mengenai lini produksi tiap produksi. Berikut adalah Gambar V.25 perancangan *form* data master Lini produksi.

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR															
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Data Master</div> <div style="border: 1px solid black; padding: 5px;">Lini Produksi</div>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Lini Produksi</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Tambah</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 20%;">Kode lini produksi</th> <th style="width: 30%;">Nama Line</th> <th style="width: 30%;">Tipe line</th> <th style="width: 15%;"></th> </tr> </thead> <tbody> <tr> <td style="height: 20px;"></td> <td></td> <td></td> <td></td> <td style="text-align: center; vertical-align: bottom;">Ubah Hapus</td> </tr> </tbody> </table>					No	Kode lini produksi	Nama Line	Tipe line						Ubah Hapus
No	Kode lini produksi	Nama Line	Tipe line												
				Ubah Hapus											

Gambar V.25 Perancangan *Form* Data Master Lini produksi
(Sumber: Hasil Analisis Data, 2016)

6. Halaman Data Master User

Data Master *Trainer* berisi informasi mengenai User tiap seksi. Berikut adalah Gambar V.26 perancangan *form* data master *user*

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR																	
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Data Master</div> <div style="border: 1px solid black; padding: 5px;">User</div>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">User</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">Tambah</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 15%;">Kode</th> <th style="width: 25%;">Nama</th> <th style="width: 15%;">Level</th> <th style="width: 30%;">password</th> <th style="width: 10%;"></th> </tr> </thead> <tbody> <tr> <td style="height: 20px;"></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center; vertical-align: bottom;">Ubah Hapus</td> </tr> </tbody> </table>					No	Kode	Nama	Level	password							Ubah Hapus
No	Kode	Nama	Level	password													
					Ubah Hapus												

Gambar V.26 Perancangan *Form* Data Master *User*
(Sumber: Hasil Analisis Data, 2016)

7. Halaman Permintaan Komponen

Permintaan Komponen akan dibuat oleh PMC dengan cara menambahkan rencana permintaan komponen untuk satu periode. Berikut adalah Gambar V.27 perancangan *form* permintaan Komponen.

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR																	
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Permintaan</div> <div style="border: 1px solid black; padding: 5px;">Permintaan</div>	Permintaan	Tambah	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 20%;">No part</th> <th style="width: 30%;">Nama part</th> <th style="width: 20%;">Vendor</th> <th style="width: 15%;">Jumlah</th> <th style="width: 10%;"></th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td style="text-align: center;">Hapus</td> </tr> </tbody> </table>			No	No part	Nama part	Vendor	Jumlah							Hapus
No	No part	Nama part	Vendor	Jumlah													
					Hapus												

Gambar V.27 Perancangan *Form* Permintaan Komponen
(Sumber: Hasil Analisis Data, 2016)

8. Halaman Penerimaan Komponen

Halaman ini dapat diakses oleh PMC untuk membuat data berdasarkan jumlah komponen yang diterima oleh pihak PMC. Berikut adalah Gambar V.28 perancangan *form* Penerimaan Komponen

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR																																											
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Penerimaan</div> <div style="border: 1px solid black; padding: 5px;">Penerimaan</div>	Penerimaan	Tambah	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 5%;">Tanggal</th> <th style="width: 5%;">Control No</th> <th style="width: 5%;">GRP</th> <th style="width: 5%;">Package No</th> <th style="width: 5%;">Ref No</th> <th style="width: 5%;">No Part</th> <th style="width: 5%;">No Vendor</th> <th style="width: 5%;">Color</th> <th style="width: 5%;">Package</th> <th style="width: 5%;">Unp chk</th> <th style="width: 5%;">Remaks</th> <th style="width: 5%;">Tipe</th> <th style="width: 5%;">Jumlah</th> <th style="width: 10%;"></th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td style="text-align: center;">Hapus</td> </tr> </tbody> </table>											No	Tanggal	Control No	GRP	Package No	Ref No	No Part	No Vendor	Color	Package	Unp chk	Remaks	Tipe	Jumlah																Hapus
No	Tanggal	Control No	GRP	Package No	Ref No	No Part	No Vendor	Color	Package	Unp chk	Remaks	Tipe	Jumlah																														
														Hapus																													

Gambar V.28 Perancangan *Form* Penerimaan Komponen
(Sumber: Hasil Analisis Data, 2016)

9. Halaman Permintaan Barang

Halaman ini berfungsi untuk melakukan permintaan komponen kepada PMC untuk mengganti atau menambah jumlah komponen. Berikut adalah Gambar V.29 perancangan *form* permintaan Barang.

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR																				
<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Permintaan Barang</div> <div style="border: 1px solid black; padding: 2px;">Permintaan Barang</div>	<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Permintaan Barang</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px; text-align: center;">Tambah</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 15%;">Kode</th> <th style="width: 25%;">Nama</th> <th style="width: 5%;">Tipe</th> <th style="width: 15%;">Tanggal</th> <th style="width: 15%;">Jumlah</th> <th style="width: 20%;"></th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td style="text-align: right;">Hapus</td> </tr> </tbody> </table>						No	Kode	Nama	Tipe	Tanggal	Jumlah								Hapus
No	Kode	Nama	Tipe	Tanggal	Jumlah															
						Hapus														

Gambar V.29 Perancangan *Form* Permintaan Barang
(Sumber: Hasil Analisis Data, 2016)

10. Halaman Daftar Penerimaan

Halaman ini ini dapat diakses oleh Assembling yaitu merupakan hasil akhir dari kegiatan penerimaan komponen. Berikut adalah Gambar V.30 perancangan *form* Penerimaan.

SISTEM INFORMASI PENERIMAAN KOMPONEN PT SUZUKI INDOMOBIL MOTOR																																												
<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Penerimaan</div> <div style="border: 1px solid black; padding: 2px;">Penerimaan</div>	<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">Penerimaan</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">No</th> <th style="width: 5%;">Tanggal</th> <th style="width: 5%;">Control No</th> <th style="width: 5%;">GRP</th> <th style="width: 5%;">Package No</th> <th style="width: 5%;">Ref No</th> <th style="width: 5%;">No Part</th> <th style="width: 5%;">No Vendor</th> <th style="width: 5%;">Color</th> <th style="width: 5%;">Package</th> <th style="width: 5%;">Unp chk</th> <th style="width: 5%;">Remaks</th> <th style="width: 5%;">Tipe</th> <th style="width: 5%;">Jumlah</th> <th style="width: 20%;"></th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td style="text-align: right;">Hapus</td> </tr> </tbody> </table>														No	Tanggal	Control No	GRP	Package No	Ref No	No Part	No Vendor	Color	Package	Unp chk	Remaks	Tipe	Jumlah																Hapus
No	Tanggal	Control No	GRP	Package No	Ref No	No Part	No Vendor	Color	Package	Unp chk	Remaks	Tipe	Jumlah																															
														Hapus																														

Gambar V.30 Perancangan *Form* Daftar Penerimaan Komponen
(Sumber: Hasil Analisis Data, 2016)

5.3 Spesifikasi Perangkat Lunak (*Software*)

Untuk mendukung kelancaran aplikasi ini selain diperlukan perangkat keras (*hardware*) juga diperlukan perangkat lunak (*software*) yang terdiri dari sistem operasi dan aplikasi *database*. Adapun perangkat lunak yang diperlukan adalah sebagai berikut:

1. Sistem Operasi : *Microsoft Windows 7*

2. *Web Server* : *Apache*
3. *Database Server* : *MySQL*
4. *Design Interface* : *PHP*
5. *Web Browser* : *Internet Explorer, Google Chrome*

5.4 Spesifikasi Perangkat Keras (*Hardware*)

Perangkat keras (*hardware*) adalah suatu perangkat fisik komputer yang digunakan untuk memasukkan, memproses, menyimpan dan mengeluarkan hasil mengelola data dalam bentuk informasi. Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain.

Tabel V.17 Spesifikasi Perangkat Keras

Perangkat Keras	<i>Specification</i>
<i>Processor</i>	Core i3
<i>Main Memory</i>	2 GB
<i>Hard Disk</i>	500 GB
<i>Graphic Adapter (VGA)</i>	2 GB

(Sumber: Hasil Analisis Data, 2016)

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil penelitian maka dapat diambil kesimpulan bahwa:

1. Sistem informasi penerimaan komponen *assembly engine* 4W berbasis web ini dapat mempercepat dan mempermudah proses permintaan komponen yang jumlahnya kurang dan komponen yang perlu penggantian karena pengisian formulir permintaan barang sudah dilakukan secara terkomputerisasi.
2. Sistem informasi penerimaan komponen *assembling engine* 4W ini dapat mempermudah Seksi *Assembly Engine* 4W dan seksi PMC (*Production Material Control*) untuk mengetahui jumlah komponen yang tersedia karena sudah saling terhubung dengan menggunakan *database* MySQL.

6.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem informasi pengendalian komponen *assembling engine* 4W ini selanjutnya adalah sebagai berikut:

1. Dalam penerapan sistem yang lama ke sistem yang baru sebaiknya dilakukan secara bertahap, hal ini untuk menghindari terjadinya kegagalan terhadap sistem yang baru.
2. Perlunya pemeliharaan terhadap sistem yang telah dibuat agar tetap berjalan dengan baik.

DAFTAR PUSTAKA

- Amrullah, 1986, Unified Modeling Language (UML), Jakarta
- Indrajit. 2001. *Pengertian Sistem Menurut Para Ahli Definisi dan Karakteristik Elemen*. www.sarjanaku.com/2012/11/. (Tanggal akses 30 April 2014).
- Jogiyanto, H. M. 2005. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Andi Offset
- Kadir, Abdul. 2008. *Tuntunan Praktis Belajar Database Menggunakan MySQL*, C.V Andi Offset. Yogyakarta
- Kristanto, Harianto. 2004. *Konsep & Perancangan Database*, Andi Offset, Yogyakarta.
- McLeod, Jr., Raymond; Schell, George P. 2011. *Sistem Informasi Manajemen*. Edisi 10. Jakarta: Salemba Empat.
- Nazir, Moh. 2011. *Metode Penelitian*. Bogor: Ghalia Indonesia.
- Nasution, Andi Hakim (2003) *Sistem Manufaktur*. Jakarta : Bhratara Karya.
- Nugroho. 2007. "Membuat Aplikasi Database SQL Server dengan Visual Basic 6.0", Gava Media, Yogyakarta.
- Rosa dan Shalahuddin, M. 2011. *Rekayasa Perangkat Lunak*. Bandung: Modula.
- Sutanta Edhy. 2003. *Sistem Informasi Manajemen*. Graha Ilmu. Yogyakarta.

Sidharta, Lani. 1995. *Pengantar Sistem Informasi Bisnis*. Jakarta:PT Elex Media Komputindo.

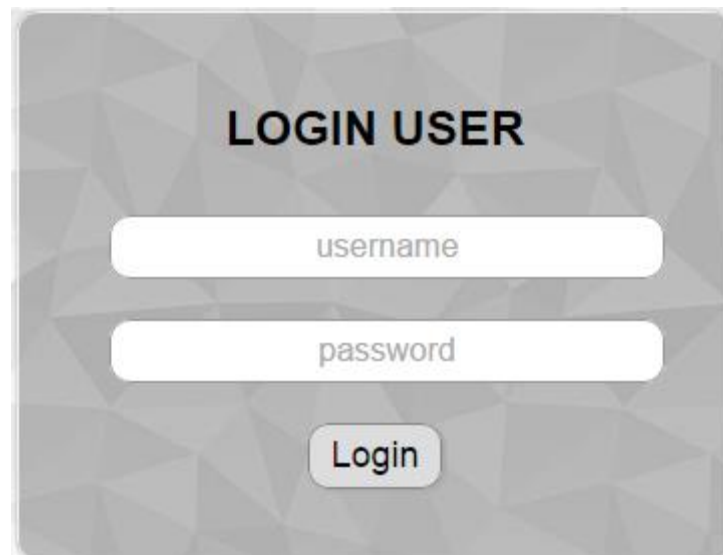
Sidik. 2004. *Pemrograman Web dengan PHP*, Informatika, Bandung.

Warman, John. 1971. *Manajemen Pergudangan*. Cetakan Pertama. Jakarta: Sinar Harapan.

Wikipedia. *Definisi sistem*, <http://id.wikipedia.org/wiki/Sistem>, (Tanggal Akses; 2 September 2014)

TAMPILAN PROGRAM

1. Login



A screenshot of a web application's login page. The page has a grey background with a geometric pattern. At the top center, the text "LOGIN USER" is displayed in bold black font. Below this, there are two white input fields with rounded corners. The first field is labeled "username" and the second is labeled "password". Below the input fields is a white button with rounded corners labeled "Login".

2. Menu Utama



3. Menu Permintaan

RANCANG BANGUN SISTEM INFORMASI PENERIMAAN KONPONEN
PT SUZUKI INDOMOBIL MOTOR

Beranda Permintaan x

Data Baru Cetak

Data Permintaan

No	Tanggal	No Part	Vendor	Jumlah
1	2016-11-09	12111-61J10-0A0	AUTOMOTIVE FASTENERS AOYAMA, P	32
2	2016-11-06	11170-61J02	ASTRA OTOPARTS TBK. PT	34
3	2016-11-09	01550-1050A	AUTOMOTIVE FASTENERS AOYAMA, P	3
4	2016-11-06	01550-0865A	ASTRA OTOPARTS TBK. PT	2

Halaman : 1 |

4. Menu Penerimaan

RANCANG BANGUN SISTEM INFORMASI PENERIMAAN KONPONEN
PT SUZUKI INDOMOBIL MOTOR

Beranda Permintaan x Penerimaan x

Data Baru Cetak

Data Penerimaan

No	Tanggal	Control No	GRP	Package No	Ref no	No Part	No Vendor	Color	Package	Up chk	remaks	Tipe	Jumlah
1	2016-11-06	1	1	1	1-1-358	12181-75F70-0A0	004	1	1	1	1	YL0 23	Hapus

Halaman : 1 |

5. Menu Permintaan Barang

RANCANG BANGUN SISTEM INFORMASI PENERIMAAN KONPONEN
PT SUZUKI INDOMOBIL MOTOR.

Permintaan Barang

No	Tanggal	Kode Line	No Part	Tipe	Jumlah	Validasi
1	2016-11-06	Line 3	01550-0865A	Y9J	6	Ya Hapus

Halaman : 1 |

6. Menu Validasi

RANCANG BANGUN SISTEM INFORMASI PENERIMAAN KONPONEN
PT SUZUKI INDOMOBIL MOTOR.

Validasi Permintaan Barang

No Part: 01550-0865A

Apakah Permintaan disetujui?

LISTING PROGRAM

1. Login

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>Login Suzuki</title>
  <link rel="stylesheet" type="text/css" href="mycss/login.css" />
  <script type="text/javascript" src="jquery_easyui/jquery.min.js"></script>
  <script type="text/javascript" src="libs2_js/login.js"></script>
</head>

<body>
</br>
<div class="lg-container">
  <div></div>
  <h2><center>
    <p>LOGIN USER</p> </center></h2>
    <form action="akses/p_login.php" id="lg-form" name="lg-form"
method="post" >
      <div>
        <label for="username">Username:</label>
        <input type="text" name="username" id="username"
placeholder="username"/>
      </div>
      <div>
        <label for="password">Password:</label>
        <input type="password" name="password" id="password"
placeholder="password" />
      </div>
      <div>
        <button type="submit" id="login">Login</button>
```

```

        </div>
    </form>
    <div id="message"></div>
</div>
<div id="footer">
    © Afryan Azhari-1311022</div>
</body>
</html>

```

2. Menu Utama

```

<?php
session_start();
if(empty($_SESSION['kd_user'])&&empty($_SESSION['level'])){
header("location:login.php");
}
else{
$level=$_SESSION['level'];}
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-
1" />
<title>Suzuki</title>
<link rel="stylesheet" type="text/css" href="mycss/index.css">
<link
rel="stylesheet"
type="text/css"
href="jquery_easyui/themes/default/easyui.css">
<link
rel="stylesheet"
type="text/css"
href="jquery_easyui/themes/icon.css">
<link
rel="stylesheet"
type="text/css"
href="jquery_easyui/themes/panel.css" />
<script type="text/javascript" src="jquery_easyui/jquery.min.js"></script>
<script
type="text/javascript"
src="jquery_easyui/jquery.easyui.min.js"></script>

<script>
function addTab(title, url){
    if ($('#tt').tabs('exists', title)){
        $('#tt').tabs('select', title);
    } else {

```

```

var content = '<iframe scrolling="auto"
frameborder="0"
style="width:100%;height:100%;"></iframe>';
$('#tt').tabs('add',{
    title:title,
    content:content,
    closable:true
});
}
</script>

</head>

<body>
<a href="#" target="_self">
<div id="header"> </div></a>

<div id="navigasi">
<div style="width:200px;height:100%;padding:5px;float:left;
margin-right:10px;background:#7190E0;">
<a href="#" class="easyui-linkbutton" iconCLS="icon-user"
plain="true" onClick="addTab('Beranda','index.php')">Beranda</a><br />

<?php if ($level=="pmc"){?>
<div class="easyui-panel" title="Data master" collapsed="true"
collapsible="true" style="width:200px;height:auto;padding:10px;">
<a href="#" class="easyui-linkbutton" iconCLS="icon-box-fill"
plain="true"
onClick="addTab('UserTampil','UserTampil.php')">User</a><br />
<a href="#" class="easyui-linkbutton" iconCLS="icon-box-fill"
plain="true"
onClick="addTab('Komponen','KomponenTampil.php')">Komponen</a>
<br />
<a href="#" class="easyui-linkbutton" iconCLS="icon-truck"
plain="true"
onClick="addTab('Vendor','VendorTampil.php')">Vendor</a><br />
</div>
<div class="easyui-panel" title="Permintaan" collapsed="true"
collapsible="true" style="width:200px;height:auto;padding:10px;">
<a href="#" class="easyui-linkbutton" iconCLS="icon-trolley"
plain="true"
onClick="addTab('Permintaan','PermintaanTampil.php')">Permintaan</a>
<br />
</div>

```

```

    <div class="easyui-panel" title="Penerimaan" collapsed="true"
collapsible="true" style="width:200px;height:auto;padding:10px;">
    <a href="#" class="easyui-linkbutton" iconCls="icon-truck"
plain="true"
onClick="addTab('Penerimaan','PenerimaanTampil.php')">Penerimaan</a
><br />
    </div>
    <div class="easyui-panel" title="Permintaan Barang" collapsed="true"
collapsible="true" style="width:200px;height:auto;padding:10px;">
    <a href="#" class="easyui-linkbutton" iconCls="icon-box-fill"
plain="true"
onClick="addTab('Permintaan
Barang','LPermintaanBarangTampil.php')">Permintaan Barang</a><br />
    </div>
    <?php }?>

    <?php if ($level=="assembling"){?>
    <div class="easyui-panel" title="Data Master" collapsed="true"
collapsible="true" style="width:200px;height:auto;padding:10px;">
    <a href="#" class="easyui-linkbutton" iconCls="icon-box-fill"
plain="true"
onClick="addTab('Line','LineTampil.php')">Line
Produksi</a><br />
    </div>
    <div class="easyui-panel" title="Penerimaan" collapsed="true"
collapsible="true" style="width:200px;height:auto;padding:10px;">
    <a href="#" class="easyui-linkbutton" iconCls="icon-truck"
plain="true"
onClick="addTab('Penerimaan','LapPenerimaan.php')">Penerimaan
komponen</a><br
    />
    </div>
    <div class="easyui-panel" title="Permintaan Barang" collapsed="true"
collapsible="true" style="width:200px;height:auto;padding:10px;">
    <a href="#" class="easyui-linkbutton" iconCls="icon-truck"
plain="true"
onClick="addTab('PermintaanBarang','PermintaanBarangTampil.php')">P
ermintaan barang</a><br />
    </div>
    <?php }?>

    <div style="width:180px;height:auto;padding:5px;float:left;
margin-right:10px;background:#7190E0;">
    <a href="akses/logout.php" class="easyui-linkbutton"
iconCls="icon-logout" plain="true">Logout</a>

    </div>
    </div>

```

```

</div>

<div id="isi">
<div id="tt" class="easyui-tabs" style="height:500px;">
<div title="Beranda" style="padding-top:20px; text-align:center;
">
<h1><strong> Selamat Datang </strong> </h1></br>
</br>
<h2>ID User : <?php echo $_SESSION['kduser']; ?> </h2>
<h2>Posisi : <?php echo $_SESSION['level']; ?> </h2></br>
<?php
$GMT = (5 *3600);
$hari=array("Minggu","Senin","Selasa","Rabu","Kamis","Jumat","
Sabtu");
$bulan=array("Januari","Februari","Maret","April","Mei","Juni","J
uli","Agustus","September","Oktober","November","Desember");
$tanggal[0]=$hari[date("w",time())];
$tanggal[1]=date("d",time());
$tanggal[2]=date("n",time());
$tanggal[3]=date("Y",time());
$tanggal[4]=date("H",time() + $GMT);
$tanggal[5]=date("i",time() + $GMT);

echo "<p class=\"tanggal\">";
echo "<h2>$tanggal[0], ";
//echo "$tanggal[1] " . $bulan[$tanggal[2]]." $tanggal[3]<br />";
echo "$tanggal[1]"/"."$tanggal[2]"/"."$tanggal[3]<br />";
echo "$tanggal[4] : $tanggal[5] WIB <br></h2>";
echo "</b></p>";
?>
</div>

</div>
</div>

<div id="footer">
©Afryan Azhari - 1311022</div>
</body>
</html>

```