

**RANCANG BANGUN APLIKASI PERENCANAAN *INDIRECT*
MATERIAL PADA DEPARTEMEN PPIC BERBASIS
DESKTOP MENGGUNAKAN VISUAL BASIC .NET 2010 DAN
SQL SERVER 2008 DI PT SHOWA INDONESIA
MANUFACTURING**

TUGAS AKHIR

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

**OLEH
DESI HUTAMI
1312044**



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

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir : RANCANG BANGUN APLIKASI PERENCANAAN
INDIRECT MATERIAL PADA DEPARTEMEN PPIC
BERBASIS DESKTOP MENGGUNAKAN VISUAL
BASIC.NET 2010 DAN SQL SERVER 2008 DI PT
SHOWA INDONESIA MANUFACTURING

Disusun Oleh :
Nama : Desi Hutami
Nim : 1312044
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 15 November 2016.


Jakarta, 18 November 2016

Dosen Pembimbing



Dedy Trisanto, S. Kom, MMSI
NIP. 197805052005021002

Dosen Penguji



Dr. Rizky Kramandita, S. Kom, MT
NIP. 197403022002121001

Ketua Penguji



Drs. Jacob Saragih, MM
NIP. 195404281986031002

Dosen Penguji

 21/11/2016

Ulil Hamida ST, MT
NIP. 198103272005022001

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN APLIKASI PERENCANAAN
INDIRECT MATERIAL PADA DEPARTEMEN PPIC
BERBASIS DESKTOP MENGGUNAKAN VISUAL
BASIC.NET 2010 DAN SQL SERVER 2008 DI PT
SHOWA INDONESIA MANUFACTURING.

Disusun Oleh :
Nama : Desi Hutami
Nim : 1312044
Program Studi : Sistem Informasi
Tanggal Seminar : Jumat, 04 November 2016
Tanggal Sidang : Selasa, 15 November 2016
Tanggal Lulus : Selasa, 15 November 2016

Jakarta, 21 November 2016

Menyetujui
Dosen Pembimbing



Dedy Trisanto, S. Kom, MMSI
NIP. 197805052005021002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN APLIKASI PERENCANAAN
INDIRECT MATERIAL PADA DEPARTEMEN PPIC
BERBASIS DESKTOP MENGGUNAKAN VISUAL
BASIC.NET 2010 DAN SQL SERVER 2008 DI PT
SHOWA INDONESIA MANUFACTURING.

Disusun Oleh :

Nama : Desi Hutami

Nim : 1312044

Program Studi : Sistem Informasi

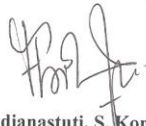
Tanggal Seminar : Jumat, 04 November 2016

Tanggal Sidang : Selasa, 15 November 2016

Tanggal Lulus : Selasa, 15 November 2016

Jakarta, 21 November 2016

Menyetujui
Asisten Pembimbing



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



POLITEKNIK STMI JAKARTA

d.h. SEKOLAH TINGGI MANAJEMEN INDUSTRI

Jl. Letjen Suprpto No. 26 Cempaka Putih, Jakarta 10510

Telp : (021) 42886064 Fax : (021) 42888206

www.stmi.ac.id



LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Desi Hutami
NIM : 1312044
Judul Tugas Akhir : Rancang Bangun Aplikasi Perencanaan Indirect Material pada Departemen PPIC Berbasis Desktop Menggunakan Visual Basic .Net 2010 Dan Sql Server 2008 Di Pt Showa Indonesia Manufacturing
Pembimbing : Dedy Trisanto, S.Kom, MMSI
Asisten Pembimbing : Fifi L. Hadianastuti, S.Kom, M.Kes

Tanggal	BAB	Keterangan	Paraf
25-07-2016	I	Revisi pendahuluan	
01-08-2016	I	Revisi Bab I	
22-08-2016	II	Revisi Bab II	
26-08-2016	III	Revisi bab III	
02-09-2016	IV	Revisi bab IV	
22-09-2016	IV	Revisi Bab IV	
29-09-2016	V	Revisi Usecase	
04-10-2016	V	Revisi Bab V	
11-10-2016	V	Revisi Class diagram	
17-10-2016	VI	Revisi Bab VI	
20-10-2016	Seluruh	Pengecekan seluruh bab	

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif

Drs. Jacob Saragih, MM
NIP. 195404281986031002

Pembimbing

Dedy Trisanto, S.Kom, MMSI
NIP. 197805052005021002













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



LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Desi Hutami
 NIM : 1312044
 Judul Tugas Akhir : Rancang Bangun Aplikasi Perencanaan Indirect Material pada Departemen PPIC Berbasis Desktop Menggunakan Visual Basic .Net 2010 Dan Sql Server 2008 Di Pt Showa Indonesia Manufacturing
 Pembimbing : Dedy Trisanto, S.Kom, MMSI
 Asisten Pembimbing : Fifi L. Hadianastuti, S.Kom, M.Kes

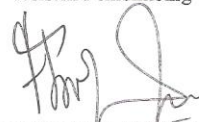
Tanggal	BAB	Keterangan	Paraf
1 Agustus 2016	I	Bimbingan Cover dan Bab I	
10 Agustus 2016	I, II	Bimbingan Bab I dan Bab II	
18 Agustus 2016	I, II	Bimbingan Bab I dan Bab II	
26 Agustus 2016	I, III	Bimbingan Bab I dan Bab III	
5 September 2016	III	Bimbingan Bab III	
19 September 2016	III IV	Bimbingan Bab III dan Bab IV	
28 September 2016	IV	Bimbingan Bab IV	
3 Oktober 2016	IV, V	Bimbingan Bab IV dan Bab V	
18 Oktober 2016	V, VI	Bimbingan Bab V dan Bab VI	
20 Oktober 2016	V, VI	Bimbingan Bab V dan Bab VI	

Mengetahui,
 Ketua Program Studi
 Sistem Informasi Industri Otomotif



Drs. Jacob Saragih, MM
 NIP. 195404281986031002

Asisten Pembimbing



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



LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Desi Hutami

Nim : 1312044

Program Studi : Sistem Informasi

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

“Rancang Bangun Aplikasi Perencanaan Indirect Material Pada Departemen PPIC Berbasis Desktop Menggunakan Visual Basic.Net 2010 Dan Sql Server 2008 Di PT Showa Indonesia Manufacturing”

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

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

Jakarta, November 2016


Desi Hutami



ABSTRAK

PT Showa Indonesia Mfg merupakan perusahaan yang bergerak di bidang industri manufaktur yang berfokus dalam produksi *Shock absorber*. PT Showa Indonesia Mfg membutuhkan pengembangan sebuah aplikasi perencanaan *indirect material* mengenai perhitungan perencanaan *indirect material* dan penyetujuan dokumen. Sistem informasi perencanaan *indirect material* yang ada pada PT Showa Indonesia Mfg dalam perencanaannya masih kurang maksimal karena tidak sesuai dengan pemakaian sebulan dan lamanya proses pengajuan *purchase order* sehingga terjadi kerusakan dan kehilangan dokumen. Rancang bangun aplikasi perencanaan *indirect material* sangat diperlukan dalam perusahaan untuk melakukan proses perencanaan *indirect material*. Aplikasi yang diusulkan akan mempermudah dan mengintegrasikan proses pengolahan data sehingga membantu bagian dalam sistem untuk saling bertukar informasi dan mengambil keputusan dengan cepat. Pengembangan sistem aplikasi perencanaan *indirect material* menggunakan metode prototipe evolusioner. Pemodelan sistem dan data menggunakan *unified modeling language (UML)* dan *flowchart*. Perancangan aplikasi perencanaan *indirect material* ini menggunakan *hierarchy plus input-process-output (HIPO)*, *flowchart* program dan perancangan tampilan antarmuka. Pembuatan aplikasi perencanaan *indirect material* menggunakan Visual Basic .NET 2010 dan SQL Server 2008. Aplikasi perencanaan *indirect material* diharapkan dapat memudahkan perusahaan dalam proses perencanaan *indirect material* terutama dalam kalkulasi perencanaan bulanan dan memudahkan dalam validasi dokumen *purchase order*. Untuk penerapan sistem baru, disarankan untuk melakukan sosialisasi kepada bagian terkait dan pemeliharaan aplikasi agar dapat berjalan dengan baik.

Kata Kunci :Perencanaan *indirect material*, validasi dokumen, prototipe evolusioner, Visual Basic .net 2010, SQL Server 2008, UML.

KATA PENGANTAR

Puji syukur penulis panjatkan bagi Allah SWT , yang atas berkat dan kasih-Nya sehingga penulis mampu menyelesaikan Tugas Akhir dengan judul **“Rancang Bangun Aplikasi Perencanaan Indirect Material pada Departemen PPIC Berbasis Desktop Menggunakan Visual Basic .Net 2010 Dan SQL Server 2008 Di Pt Showa Indonesia Manufacturing”**.

Penulisan Tugas Akhir ini bertujuan untuk memenuhi sebagian syarat penyelesaian jenjang D-4 pada program studi Sistem Informasi pada Politeknik STMI Jakarta.

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

1. Bapak Dr. Mustofa, ST, MT selaku Direktur Politeknik STMI Jakarta.
2. Bapak Drs. Jacob Saragih, MM. selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
3. Bapak Dedy Trisanto, S.Kom, MMSI. selaku dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
4. Ibu Fifi Lailasari Hadianastusi, S.Kom, M.Kes. selaku asisten dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
5. Bapak Sigit Widodo dan Ibu Dewi Santi selaku orang tua penulis yang selalu mendoakan, memberikan motivasi dan pengorbanannya baik segi moril, materi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
6. Bapak Tofik Wijanarko, Bapak Dedi Ubaidilah, Bapak Suparno, Bapak Deny Rosadi, serta seluruh karyawan di Departemen Planning Production

Inventory Control PT Showa Indonesia Manufacturing yang telah membantu dalam penyusunan Tugas Akhir.

7. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
8. Seluruh teman-teman jurusan Sistem Informasi angkatan 2012 yang telah memberikan semangat dan motivasi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
9. Serta semua pihak yang telah membantu penulis dan yang tidak dapat disebutkan satu per satu.

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

Jakarta, 20 Oktober 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	xi
Daftar Gambar	xv
Daftar Tabel	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah.....	3
1.5 Manfaat Penelitian	4
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Rancang Bangun	6
2.2 Konsep Dasar Sistem	6
2.2.1 Sistem.....	6
2.2.2 Karakteristik Sistem.....	7

2.3 Konsep Dasar Informasi	10
2.3.1 Siklus Informasi	10
2.4 Pengertian Aplikasi	11
2.5 Pengertian Perencanaan	11
2.6 Pengertian <i>Indirect Material</i>	12
2.7 <i>Agregat Planning</i>	13
2.7.1 Metode Perhitungan <i>Agregat Planning</i>	15
2.8 <i>System Development Life Cycle</i>	18
2.8.1 Desain Terstruktur.....	18
2.8.2 <i>Rapid Application Development (RAD)</i>	20
2.8.3 <i>Prototyping</i>	22
2.9 <i>Flowchart</i>	23
2.10 <i>Unified Modeling Language (UML)</i>	25
2.11 Kamus Data	32
2.12 HIPO (<i>Hierarchy plus Input-Process-Output</i>).....	33
2.13 Visual Basic .NET 2010.....	34
2.14 SQL Server.....	38
BAB III METODOLOGI PENELITIAN	42
3.1 <u>Metodologi Penelitian</u>	42
3.2 Identifikasi Masalah	43
3.3 Pengumpulan Data	45
3.4 Metode Pengembangan Sistem	44
3.5 Kerangka Penelitian	45
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA	49
4.1 Sekilas Tentang PT Showa Indonesia Mfg.	50
4.2 Profil Perusahaan	51
4.3 Visi, Misi, dan Tujuan Perusahaan	52

4.4	<u>Struktur Organisasi</u>	52
4.5	Data <i>Indirect Material</i>	57
4.6	Proses Perencanaan <i>Indirect Material</i>	59
4.6.1.	Analisis Dokumen.....	59
4.7	<u>Analisis Sistem yang Sedang Berjalan</u>	64
4.7.1.	Flowchart yang Sedang Berjalan di Perencanaan <i>Indirect Material</i>	64
4.8	<u>Sistem Berjalan dengan Use Case Diagram</u>	66
BAB V	ANALISIS DAN PEMBAHASAN	69
5.1	Analisis Kebutuhan Sistem	69
5.2	Prosedur Sistem Informasi Perencanaan <i>Indirect Material</i> Usulan	70
5.2.1	Prosedur Sistem Informasi Perencanaan <i>Indirect Material</i> Usulan.....	70
5.2.2	<i>Use Case Diagram</i> Usulan.....	72
5.2.3	<i>Activity Diagram</i> Usulan.....	80
5.2.4	<i>Sequence Diagram</i> Usulan.....	88
5.2.5	<i>Class Diagram</i> Usulan	95
5.2.6	Kamus Data.....	98
5.2.7	<i>Deployment Diagram</i> Usulan.....	99
5.3	Analisis Desain Program.....	100
5.3.1	HIPO	100
5.3.2	<i>Flowchart</i> Program	101
5.3.3	Perancangan <i>Interface Program</i>	106
BAB VI	KESIMPULAN DAN SARAN	115
6.1.	Kesimpulan	112
6.2.	Saran.....	112

DAFTAR PUSTAKA	xxi
LAMPIRAN	L-1

DAFTAR GAMBAR

Gambar II.1 Karakteristik Sistem	9
Gambar II.2 Siklus Informasi.....	11
Gambar II.3 <i>Waterfall Development</i>	19
Gambar II.4 <i>Paralell Development</i>	20
Gambar II.5 <i>Rapid Application Development</i>	21
Gambar II.6 <i>Evolutionary Prototype Model</i>	22
Gambar II.7 UML Diagram	26
Gambar II.8 <i>Visual Table Of Contents</i>	34
Gambar II.9 Antarmuka Visual Basic .NET 2010	35
Gambar II.10 Antarmuka Pembuatan Project Baru	35
Gambar II.11 <i>Title Bar</i> Visual Basic .NET 2010.....	36
Gambar II.12 Menu Bar dan Toolbar Visual Basic .NET 2010.....	36
Gambar II.13 <i>Form</i> Visual Basic .NET 2010	36
Gambar II.14 <i>Toolbox</i> Visual Basic .NET 2010.....	37
Gambar II.15 <i>Solution Explorer</i> Visual Basic .NET 2010.....	37
Gambar II.16 <i>Form</i> Visual Basic .NET 2010	38
Gambar III.1 Kerangka Penelitian	47
Gambar III.2 Kerangka Penelitian (Lanjutan)	48
Gambar IV.1 Struktur Organisasi PT Showa Indonesia Mfg	52
Gambar IV.2 Struktur Organisasi Divisi PPIC	54
Gambar IV.3 <i>Indirect Material Consumption Tools</i>	60
Gambar IV.4 Dokumen Kalkulasi Pemesanan <i>Indirect Material</i>	60
Gambar IV.5 Dokumen <i>Forecasting Indirect Material</i>	62
Gambar IV.6 Contoh Dokumen <i>Purchase Requisition</i>	62
Gambar IV.7 <i>Purchase Order Plan</i>	63
Gambar IV.8 <i>Purchase Order Release</i>	63

Gambar IV.9 Flowmap Pemesanan yang Sedang Berjalan di PPIC PT Showa Indonesia Mfg	65
Gambar IV.10 <i>Use Case</i> yang Sedang Berjalan di PPIC PT Showa Indonesia Mfg.	66
Gambar V.1 <i>Flowmap</i> Sistem Informasi Perencanaan <i>Indirect Material</i> Usulan	71
Gambar V.2 <i>Use Case</i> Diagram Sistem Informasi Perencanaan <i>Indirect Material</i> Usulan	72
Gambar V.3 <i>Activity Diagram</i> Login	80
Gambar V.4 <i>Activity Diagram</i> Mengelola Data Master	81
Gambar V.5 <i>Activity Diagram</i> Kalkulasi Order.....	82
Gambar V.6 <i>Activity Diagram</i> Membuat <i>Forecasting</i> Bulanan	83
Gambar V.7 <i>Activity Diagram</i> Membuat <i>Purchase Requisition</i>	84
Gambar V.8 <i>Activity Diagram</i> Membuat <i>Purchase Order</i>	85
Gambar V.9 <i>Activity Diagram</i> View <i>Purchase Order</i>	85
Gambar V.10 <i>Activity Diagram</i> Menyetujui <i>Purchase Order</i>	86
Gambar V.11 <i>Activity Diagram</i> Melakukan Pemesanan	87
Gambar V.12 <i>Activity Diagram</i> Membuat Laporan Bulanan	87
Gambar V.13 <i>Sequence Diagram</i> Login.....	88
Gambar V.14 <i>Sequence Diagram</i> Mengelola Data Master.....	89
Gambar V.15 <i>Sequence Diagram</i> Kalkulasi Order Bulanan	90
Gambar V.16 <i>Sequence Diagram</i> Membuat <i>Forecasting</i> Bulanan	90
Gambar V.17 <i>Sequence Diagram</i> Membuat <i>Purchase Requisition</i>	91
Gambar V.18 <i>Sequence Diagram</i> Membuat <i>Purchase Order</i>	92
Gambar V.19 <i>Sequence Diagram</i> View <i>Purchase Order</i>	92
Gambar V.20 <i>Sequence Diagram</i> Menyetujui <i>Purchase Order</i>	93
Gambar V.21 <i>Sequence Diagram</i> Melakukan Pemesanan	93
Gambar V.22 <i>Sequence Diagram</i> Membuat Laporan Bulanan	94
Gambar V.23 <i>Class Diagram</i> Aplikasi Perencanaan <i>Indirect Material</i>	94
Gambar V.24 <i>Deployment Diagram</i> Aplikasi Perencanaan <i>Indirect Material</i>	99
Gambar V.25 HIPO Aplikasi Perencanaan <i>Indirect Material</i>	101
Gambar V.26 <i>Flowchart</i> Aplikasi Perencanaan <i>Indirect Material</i>	102
Gambar V.27 Form Login.....	103

Gambar V.28 Menu Utama Aplikasi Perencanaan <i>Indirect Material</i>	104
Gambar V.29 Form <i>User</i>	104
Gambar V.30 Form Part.....	105
Gambar V.31 Form Hak Akses.....	106
Gambar V.32 Form Kalkulasi Order.....	107
Gambar V.33 <i>Form Forecasting</i>	107
Gambar V.34 <i>Form Purchase Requisition</i>	108
Gambar V.35 Form <i>Purchase Order</i>	109
Gambar V.36 Form <i>View PO</i>	109
Gambar V.37 Form Validasi	110
Gambar V.38 Form Laporan Pemesanan Bulanan.....	110
Gambar V.39 Rancangan Tampilan Laporan Pemesanan Bulanan	111

DAFTAR TABEL

Tabel II.1 Simbol-Simbol <i>Flowchart</i>	24
Tabel II.2 Simbol-simbol Use Case Diagram	27
Tabel II.3 Simbol-Simbol <i>Activity Diagram</i>	28
Tabel II.4 Simbol-simbol <i>Sequence Diagram</i>	29
Tabel II.5 Simbol-Simbol <i>Class Diagram</i>	31
Tabel II.6 Simbol-simbol <i>Deployment Diagram</i>	32
Tabel II.7 Tipe Data Numerik <i>Integer</i>	39
Tabel II.8 Tipe Data Numerik <i>Floating Point</i>	39
Tabel II.9 Tipe Data <i>String Text/Blob</i>	40
Tabel II.10 Tipe Data Numerik Floating Point	40
Tabel II.11 Tipe Data Tanggal dan Waktu.....	41
Tabel IV.1 Definisi Aktor Use Case Diagram	67
Tabel IV.2 Definisi Use Case Diagram.....	67
Tabel V.1 Kebutuhan Sistem Aplikasi Perencanaan <i>Indirect Material</i>	69
Tabel V.2 Definisi Aktor <i>Use Case Diagram</i>	73
Tabel V.3 Definisi <i>Use Case Diagram</i>	73
Tabel V.4 Use Case Description Login.....	74
Tabel V.5 <i>Use Case Description</i> Mengelola Data Master	75
Tabel V.6 <i>Use Case Description</i> Membuat Kalkulasi Order Bulanan	75
Tabel V.7 Use Case Description Membuat Dokumen <i>Forecasting Indirect Material</i>	76
Tabel V.8 <i>Use Case Description</i> Membuat <i>Purchase Requisition</i>	77
Tabel V.9 <i>Use Case Description</i> Membuat <i>Purchase Order</i>	77
Tabel V.10 <i>Use Case Description</i> View <i>Purchase Order</i>	78
Tabel V.11 Use Case Description Melakukan Persetujuan <i>Purchase Order</i>	78
Tabel V.12 <i>Use Case Description</i> Melakukan Pemesanan.....	79

Tabel V.13 <i>Use Case Description</i> Membuat Laporan Pemesanan Bulanan.....	80
Tabel V.14 Tabel <i>User</i>	95
Tabel V.15 Tabel Part	95
Tabel V.16 Tabel Kalkulasi	96
Tabel V.17 Tabel Detail Kalkulasi.....	96
Tabel V.18. Tabel <i>Forecasting</i>	97
Tabel V.19. Tabel Detail <i>Forecasting</i>	97
Tabel V.20. Tabel Purchase Requisition.....	97
Tabel V.21. Tabel Detail Purchase Requisition	98
Tabel V.22. Tabel Purchase Order	98
Tabel V.23. Tabel Detail Purchase Order	99

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan di bidang teknologi sekarang ini membuat berkembang pesatnya sistem informasi yang ada pada dewasa ini dibandingkan sebelumnya. Terutama pada perusahaan manufaktur yang berlomba-lomba untuk meningkatkan produksi mereka melalui bantuan dari teknologi. Tidak dapat dipungkiri jika kemajuan teknologi bisa menjadi sebuah terobosan baru bagi proses bisnis yang sedang berjalan untuk meningkatkan kinerja mereka dalam menjalankan proses bisnis tersebut.

Permintaan dari konsumen akan sebuah produk bisa sangat bervariasi tergantung dengan kebutuhannya. Salah satunya adalah kebutuhan atas otomotif seperti kendaraan bermotor. Permintaan yang tinggi ini mendorong pasar industri berlomba untuk memproduksi produk yang dibutuhkan dengan standar kualitas yang terbaik sesuai dengan kebutuhan para konsumen namun tetap memiliki keuntungan yang diinginkan oleh perusahaan. Dalam memenuhi permintaan ini, perusahaan melakukan perencanaan dan pengendalian produksinya dengan baik sehingga produksi yang akan dilaksanakan secara efektif dan efisien. Salah satu proses dalam perencanaan dan pengendalian produksi ini adalah perencanaan bahan baku.

Bahan baku adalah bahan yang digunakan dalam membuat produk dimana bahan tersebut secara menyeluruh tampak pada produk jadinya. Bahan baku terbagi menjadi dua bagian yaitu bahan baku langsung (*direct material*) dan bahan baku tidak langsung (*indirect material*). Bahan baku langsung merupakan bahan baku yang merupakan bagian suatu produk jadi. Sedangkan bahan baku tidak langsung adalah bahan baku yang ikut berperan dalam proses produksi tetapi tidak secara langsung tampak pada produk jadi yang dihasilkan.

PT Showa Indonesia Mfg adalah sebuah perusahaan yang bergerak di bidang manufaktur komponen otomotif. Perusahaan ini adalah produsen *Shock Absorber* atau yang biasa dikenal dengan peredam kejut dan memiliki kantor yang berada di Jakarta

Utara sedangkan pabrik produksinya berada di daerah Cikarang. PT Showa Indonesia Mfg memproduksi *Shock Absorber* yang berbeda sesuai dengan merek dan jenisnya. Dalam menjalankan kegiatan bisnisnya, PT Showa Indonesia Mfg telah menerapkan sistem perencanaan dan pengendalian produksi yang terbukti mampu membantu perusahaan dalam membantu proses produksi untuk memenuhi permintaan pasar lokal maupun internasional.

Dalam PT Showa Indonesia Mfg terdapat sistem perencanaan dan pengendalian produksi dan inventori. Perencanaan dan pengendalian khususnya bahan baku tidak langsung dilakukan melalui perhitungan secara terkomputerisasi dengan bantuan *spreadsheet* yang kemudian dimasukkan ke sistem ERP IFS (*Industrial and Financial Systems*) yang ada di perusahaan. Pengolahan data perencanaan ini diperlukan setiap bulannya untuk melakukan peramalan pembelian bahan baku tidak langsung yang akan dihitung untuk pembelian bulan selanjutnya. Untuk mendukung ketepatan dalam perencanaan peramalan yang akurat maka perlu dirancang suatu sistem informasi yang dapat mengurangi kelemahan pada sistem informasi yang sebelumnya yaitu perencanaan *indirect material* yang dilakukan tidak sesuai dengan pemakaian selama sebulan dan penumpukan dokumen untuk divalidasi. Setelah kelemahan tersebut diperbaiki maka diharapkan akan meningkatkan produktivitas produksi.

Berdasarkan latar belakang di atas, maka Tugas Akhir ini berjudul “RANCANG BANGUN APLIKASI PERENCANAAN *INDIRECT MATERIAL* PADA DEPARTEMEN PPIC BERBASIS DESKTOP MENGGUNAKAN VISUAL BASIC.NET 2010 DAN SQL SERVER 2008 DI PT SHOWA INDONESIA MANUFACTURING”.

1.2 Pokok Permasalahan

Adapun pokok permasalahan setelah mengamati sistem pemesanan *indirect material* yang berjalan pada PT Showa Indonesia Manufacturing adalah sebagai berikut:

1. Proses perencanaan *indirect material* kurang tepat karena perhitungan yang dilakukan tidak sesuai dengan pemakaian selama sebulan. Hal ini mengakibatkan keadaan stok material tidak aman.
2. Proses pengajuan *purchase order* masih dikirim secara manual dan mengalami penumpukan dokumen karena *dept head* tidak langsung memvalidasi sehingga terjadi kerusakan dan kehilangan dokumen.

1.3 Tujuan Penelitian

Adapun tujuan dari Tugas Akhir ini adalah membuat aplikasi perencanaan *indirect material* yang mempunyai fungsi sebagai berikut:

1. Merancang bangun aplikasi perencanaan *indirect material* berdasarkan rumus *agregat planning* dan parameter yang sudah ditentukan agar perencanaan *indirect material* sesuai dengan kebutuhan dan mempunyai keadaan stok yang aman.
2. Membuat sistem peringatan untuk *dept head* ketika dokumen *purchase order* sudah dikirim melalui email sehingga proses validasi *purchase order* bisa langsung dilakukan oleh *dept head*.

1.4 Batasan Masalah

Dalam penyusunan Tugas Akhir agar lebih fokus dan terarah, maka perlu diadakan batasan-batasan sebagai berikut:

1. Pengamatan mencakup pada data pemesanan *indirect material Consumption Tools* untuk mengelola permasalahan perencanaan pemesanan pada Departemen PPIC PT Showa Indonesia Manufacturing.
2. Membuat aplikasi pengolahan data dan menghasilkan laporan perencanaan *indirect material*.

3. Tidak membahas *direct material* (bahan baku langsung).

1.5 Manfaat Tugas Akhir

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

1. Bagi Penulis

Sebagai alat untuk mempraktekkan teori-teori yang telah diperoleh selama perkuliahan, sehingga penulis dapat menambah pengetahuan secara praktis tentang masalah-masalah yang dihadapi oleh perusahaan.

2. Bagi Perusahaan

Hasil penelitian ini diharapkan dapat mempermudah kinerja perusahaan dalam proses perencanaan *indirect material*. Aplikasi perencanaan *indirect material* dapat mempercepat waktu perencanaan *indirect material* dan juga dapat menghasilkan laporan secara efektif dan efisien.

3. Bagi Pihak Lain

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

1.6 Sistematika Penulisan

Adapun sistematika penulisan Tugas Akhir ini adalah sebagai berikut:

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Bab ini membahas tentang pengertian sistem informasi, pengertian aplikasi, jenis aplikasi, pengertian perencanaan, pengertian *indirect material*, pengertian *agregat planning*, metodologi pengembangan sistem, pengertian HIPO, *Flowchart*, *Unified Modeling Language (UML)*, *visual basic.net* dan *MySQL*.

BAB III METODOLOGI PENELITIAN

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

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini menguraikan tentang hasil pengamatan selama penelitian pada PT Showa Indonesia Mfg dilakukan, seperti data material, data pemakaian material selama bulan dan laporan pembelian material yang dihasilkan oleh sistem.

BAB V ANALISA DAN PEMBAHASAN

Bab ini berisi analisis rinci dari pengolahan data, yaitu mulai dari analisis sistem yang meliputi diagram alir sistem usulan, model sistem UML, model data, hierarki program (HIPO), *flowchart* program, perancangan program, dan spesifikasi sistem informasi yang dibutuhkan untuk menjalankan program.

BAB VI KESIMPULAN DAN SARAN

Pada bab ini dikemukakan kesimpulan secara keseluruhan yang diperoleh dari hasil penelitian dan saran-saran dalam penerapan Aplikasi Perencanaan *Indirect Material* untuk perusahaan dan pengembangan selanjutnya.

BAB II

LANDASAN TEORI

2.1 Rancang Bangun

Rancang bangun adalah tahap dari setelah analisis dari siklus pengembangan sistem yang merupakan pendefinisian dari kebutuhan-kebutuhan fungsional, serta menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi, termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat keras dan perangkat lunak dari suatu sistem (Jogiyanto, 2005).

2.2 Konsep Dasar Sistem

2.2.1 Sistem

Sistem berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) adalah suatu kesatuan yang terdiri dari 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. Berikut pengertian sistem menurut beberapa ahli antara lain sebagai berikut:

1. 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 (McLeod, 2001).
2. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (Jogiyanto, 2005).

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.

Pengertian sistem tergantung pada latar belakang cara pandang orang yang mencoba mendefinisikan. Menurut hukum, sistem dipandang dari kumpulan aturan-aturan yang membatasi, baik oleh kapasitas sistem itu sendiri maupun lingkungan dimana sistem itu berada, untuk menjamin keserasian dan keadilan. Menurut rekayasa, sistem dipandang sebagai proses masukan (*input*) yang ditransformasikan menjadi keluaran (*output*) tertentu.

2.2.2 Karakteristik Sistem

Model umum sebuah sistem adalah *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem.

Menurut pendapat Jogiyanto (2005), dalam bukunya yang berjudul Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur, Teori dan Praktik Aplikasi Bisnis mempunyai beberapa karakteristik atau sifat-sifat tertentu, antara lain:

1. Komponen (*component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batasan sistem (*boundary*)

Daerah yang membatasi antara sistem yang satu dengan yang lainnya atau dengan lingkungan luar. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem.

3. Lingkungan luar (*environment*)

Segala sesuatu di luar dari batas sistem yang mempengaruhi operasi dari suatu 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 (*interface*)

Suatu 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 suatu 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 (*input*)

Energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh di dalam sistem komputer, program A adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya, dan data adalah *signal input* untuk diolah menjadi informasi. Pada sistem informasi, masukan dapat berupa data transaksi, data non transaksi (misal: surat pemberitahuan) dan instruksi.

6. Keluaran (*output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran merupakan masukan untuk subsistem yang lain atau kepada suprasistem. Hasil dari pemrosesan, dapat berupa keluaran yang berguna (informasi, produk) atau keluaran yang tidak berguna (limbah). Pada sistem informasi, keluaran dapat berupa: informasi, saran, dan cetakan laporan.

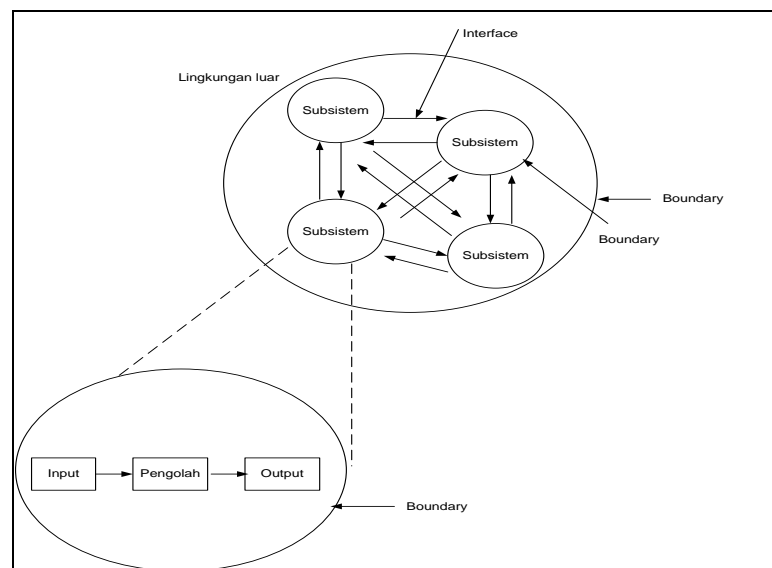
7. Sasaran sistem (*objective*)

Suatu tujuan yang ingin dicapai oleh suatu sistem. 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.

8. Pengolahan sistem

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

Berikut adalah gambaran karakteristik sistem:



Gambar II.1 Karakteristik Sistem
(Sumber: Jogiyanto, 2005)

Konsep dari sebuah sistem memungkinkan orang untuk berkonsentrasi pada suatu sistem tertentu dari seluruh hirarki sistem. Batas-batas suatu sistem biasa sebuah sistem sehubungan dengan batas-batas ini adalah sebagai berikut:

1. Buatlah daftar dari semua elemen yang membentuk sistem tersebut dan buatlah batasan sistem tersebut. Segala sesuatu yang berada dalam ruang tersebut disebut sistem, dan yang berada di luar disebut lingkungan.
2. Buatlah semua arus yang melewati batas. Arus dari lingkungan ke dalam sistem adalah *input*, dan arus dari dalam batas keluar adalah *output*.

3. Identifikasi semua elemen yang memberikan kontribusi dalam mencapai sasaran yang ditentukan bagi sistem ini, dan masukkan ke dalam batas lingkaran, jika belum termasuk.

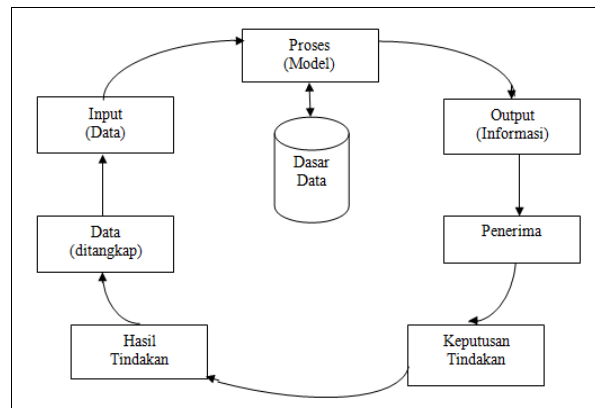
2.3 Konsep Dasar Informasi

Sumber informasi adalah data. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Data merupakan jamak dari bentuk tunggal data *item*. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. (Jogiyanto, 2005)

Data yang diolah untuk menghasilkan informasi menggunakan suatu model proses tertentu. Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan yang berarti menghasilkan tindakan yang lain sehingga membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus.

2.3.1 Siklus Informasi

Menurut Jogiyanto (2005), data agar menjadi lebih berarti dan berguna dalam bentuk informasi maka perlu diolah menjadi suatu model tertentu. Data yang telah diolah tersebut kemudian diterima oleh penerima, lalu penerima membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, dan diproses kembali lewat suatu model dan seterusnya sehingga membentuk suatu siklus. Siklus ini disebut dengan siklus informasi (*information cycle*) atau disebut pula siklus pengolahan data (*processing cycles*). Berikut gambaran siklus informasi menurut Jogiyanto (2005):



Gambar II.2 Siklus Informasi
(Sumber: Jogiyanto, 2005)

2.4 Pengertian Aplikasi

Perangkat lunak aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna (wikipedia).

Sedangkan menurut Jogiyanto (2005) aplikasi merupakan penerapan, menyimpan suatu hal, data, permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk menerapkan atau mengimplementasikan hal atau permasalahan yang ada sehingga berubah menjadi suatu bentuk yang baru tanpa menghilangkan nilai-nilai dasar dari hal data, permasalahan dan pekerjaan itu sendiri.

2.5 Pengertian Perencanaan

Dalam ilmu manajemen menjelaskan bahwa salah satu fungsi pokok manajemen adalah perencanaan. Ilmu manajemen menjelaskan fungsi pokok manajemen terdiri dari perencanaan, koordinasi, pelaksanaan, pengawasan dan evaluasi. Perencanaan merupakan salah satu fungsi pokok manajemen yang pertama harus dijalankan, sebab tahap awal dalam melakukan aktivitas perusahaan sehubungan dengan pencapaian tujuan organisasi perusahaan adalah dengan membuat perencanaan (Sinulingga, 2013).

Definisi perencanaan tersebut menjelaskan bahwa perencanaan merupakan suatu proses untuk mencapai tujuan perusahaan secara menyeluruh. Definisi

perencanaan dapat disimpulkan bahwa perencanaan menggunakan beberapa aspek yaitu (Sinulingga, 2013):

1. Penentuan tujuan yang akan dicapai.
2. Memilih dan menentukan cara yang akan ditempuh untuk mencapai tujuan atas dasar alternatif yang dipilih.
3. Usaha-usaha atau langkah-langkah yang ditempuh untuk mencapai tujuan atas dasar alternatif yang dipilih.

Selain aspek tersebut, perencanaan juga mempunyai manfaat bagi perusahaan sebagai berikut:

1. Dengan adanya perencanaan, maka pelaksanaan kegiatan dapat diusahakan dengan efektif dan efisien.
2. Tujuan yang telah ditetapkan tersebut, dapat dicapai dan dilakukan koreksi atas penyimpangan-penyimpangan yang timbul seawal mungkin.
3. Dapat mengidentifikasi hambatan-hambatan yang timbul dengan mengatasi hambatan dan ancaman.
4. Dapat menghindari adanya kegiatan pertumbuhan dan perubahan yang tidak terarah dan terkontrol.

Dalam suatu organisasi perencanaan memiliki posisi penting dari langkah-langkah berikutnya. Kematangan dan kesalahan dalam perencanaan mampu memberi pengaruh positif dan negatif pada masa yang akan datang, sehingga suatu perencanaan yang dibuat adalah selalu memikirkan dampak jangka panjang yang mungkin akan dialami.

2.6 Pengertian *Indirect Material*

Bahan baku pada proses produksi dibagi menjadi dua jenis menurut Gunawan Adisaputro, yaitu bahan baku langsung dan bahan baku tidak langsung. Bahan baku langsung atau *direct material* adalah semua bahan baku yang merupakan bagian daripada barang jadi yang di hasilkan. Biaya yang dikeluarkan untuk membeli bahan baku langsung ini mempunyai hubungan yang erat dan sebanding dengan jumlah barang jadi yang dihasilkan.

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

Sebagai contoh diambil adalah pada proses produksi kursi kayu, maka yang merupakan bagian *direct material* adalah kayu dan pada bagian *indirect material* adalah perekat kayu dan dasar cat untuk kursi yang dihasilkan. Pada dasarnya *indirect material* mendukung proses produksi sama halnya dengan *direct material* baik dalam hal pelengkap desain produk ataupun pendukung dari mesin yang digunakan pada proses produksinya. *Indirect material* ini masuk dalam perhitungan biaya produksi bulanan karena sangat penting untuk menyimpan persediaan bahan-bahan tersebut.

2.7 Agregat Planning

Aggregate Planning (AP) adalah suatu aktivitas operasional untuk menentukan jumlah dan waktu produksi pada waktu dimasa yang akan datang. AP juga didefinisikan sebagai usaha untuk menyamakan antara supply dan demand dari suatu produk atau jasa dengan jalan menentukan jumlah dan waktu input, transformasi, dan output yang tepat. Dimana keputusan AP dibuat untuk produksi, pengendalian staf, inventori, dan level pemesanan kembali.

Perencanaan Agregat (*agregat planning*) juga dikenal sebagai Penjadwalan Agregat adalah Suatu pendekatan yang biasanya dilakukan oleh para manajer operasi untuk menentukan kuantitas dan waktu produksi pada jangka menengah (biasanya antara 3 hingga 18 bulan ke depan). Perencanaan agregat dapat digunakan dalam menentukan jalan terbaik untuk memenuhi permintaan yang diprediksi dengan menyesuaikan nilai produksi, tingkat tenaga kerja, tingkat persediaan, pekerjaan lembur, tingkat subkontrak, dan variabel lain yang dapat dikendalikan. Keputusan Penjadwalan menyangkut perumusan rencana bulanan dan kuartalan yang mengutamakan masalah mencocokkan produktifitas dengan permintaan yang fluktuatif. Oleh karenanya perencanaan Agregat termasuk dalam rencana jangka menengah.

Proses perencanaan dapat digolongkan dalam tiga tingkatan yaitu:

1. *Long Range Plans*

Merupakan perencanaan lebih dari setahun yang menyangkut perencanaan produk baru, biaya perluasan dan sebagainya. *Long Range Plans* ditetapkan oleh manajer puncak.

2. *Intermediete Range Plans*

Merupakan rencana antara 3 sampai 18 bulan, menyangkut rencana penjualan, rencana produksi, rencana inventori, anggaran tenaga kerja dan sebagainya. *Intermediate range plans* ditetapkan oleh Manajer Operasi.

3. *Short Range Plans*

Merupakan rencana kurang dari tiga bulan yang menyangkut *job assignment, ordering, Job scheduling*. *Short Range Plans* ditetapkan oleh Manajer Operasi bersama dengan supervisor dan operator.

Dalam tiga tingkatan proses perencanaan tersebut, perencanaan agregat berada pada tingkatan kedua yaitu *Intermediate plans* yang menyangkut rencana produksi atau operasi perusahaan.

Perencanaan agregat membentuk keterkaitan antara perencanaan fasilitas di satu pihak dan penjadwalan di pihak lain. Perencanaan fasilitas membatasi keputusan perencanaan agregat. penjadwalan berkenaan dengan jangka waktu yang pendek (beberapa bulan atau kurang) dan dibatasi oleh keputusan perencanaan agregat. Perencanaan agregat berkaitan dengan perolehan sumber daya, sedangkan penjadwalan berkaitan dengan pengalokasian sumber daya yang tersedia terhadap pekerjaan dan pesanan tertentu. Jadi perbedaan dasar harus dilakukan antara perolehan sumber daya melalui penjadwalan.

Pada dasarnya perencanaan produksi agregat merupakan suatu proses penetapan tingkat output/kapasitas produksi secara keseluruhan guna memenuhi tingkat permintaan yang diperoleh dari peramalan dan pesanan dengan tujuan meminimalkan total biaya produksi.

Beberapa fungsi perencanaan agregat yaitu :

1. Menjamin rencana penjualan dan rencana produksi konsisten terhadap rencana strategi perusahaan

2. Alat ukur performansi proses perencanaan produksi
3. Menjamin kemampuan produksi konsisten terhadap rencana produksi
4. Memonitor hasil produksi aktual terhadap rencana produksi dan membuat penyesuaian
5. Mengatur persediaan produk jadi untuk mencapai target dan membuat penyesuaian
6. Mengarahkan penyusunan dan pelaksanaan jadwal induk produksi.

Perencanaan Agregat bertujuan untuk berusaha memperoleh suatu pemecahan yang optimal dalam biaya atau keuntungan pada periode perencanaan. Namun terdapat permasalahan strategis lain yang lebih penting daripada biaya rendah. Permasalahan strategis yang dimaksud itu antara lain mengurangi permasalahan tingkat ketenagakerjaan, menekan tingkat persediaan, atau memenuhi tingkat pelayanan yang lebih tinggi. Bagi perusahaan manufaktur, jadwal agregat bertujuan menghubungkan sasaran strategis perusahaan dengan rencana produksi, tetapi untuk perusahaan jasa, penjadwalan agregat bertujuan menghubungkan sasaran dengan jadwal pekerja. Ada empat hal yang diperlukan dalam perencanaan agregat antara lain:

1. Keseluruhan unit yang logis untuk mengukur penjualan dan output
2. Prediksi permintaan untuk suatu periode perencanaan jangka menengah yang layak pada waktu agregat.
3. Metode untuk menentukan biaya
4. Model yang menggabungkan prediksi dan biaya sehingga keputusan penjadwalan dapat dibuat untuk periode perencanaan

2.7.1 Metode Perhitungan Agregat Planning

Agregat Planning mempunyai metode-metode peramalan yang beragam sesuai dengan kebutuhan yang diperlukan oleh perusahaan. Metode ini antara lain adalah:

1. *Simple Average*

Metode *simple average* adalah metode yang mengambil rata-rata dari seluruh data observasi yang dikumpulkan untuk meramalkan data yang akan datang. Persamaannya adalah:

$$X = \frac{1}{T} (X_1 + X_2 + X_3 + \dots + X_t)$$

Dimana:

X : data peramalan yang akan datang

X_t : data observasi pada periode ke t

T : jumlah periode observasi

Metode *simple average* baik digunakan dimana ada standar jumlah pembelian di setiap barang dan perubahan harga yang tidak besar. metode ini juga mempunyai beberapa keuntungan dan kerugian yaitu:

Keuntungan:

- a. Mudah dikerjakan dan diaplikasikan
- b. Harga yang diberikan tidak terkena efek perubahan dari harga pembelian.
- c. Metode harga rata-rata cocok untuk kondisi dimana pembelian beda jenis tercampur sehingga indentifikasi jadi tidak dibutuhkan.
- d. Ketika jumlah pembelian stabil namun harganya berubah fluktuatif, maka metode ini cocok untuk digunakan.

Kekurangan:

- a. Keuntungan atau kerugian dalam peningkatan jumlah material dalam harga total sering terjadi tidak menjadi sama dalam biaya totalnya.
- b. Terlalu banyak keuntungan ataupun kerugian yang bisa dihasilkan dari metode ini. Sedangkan banyak pembelian berbeda berdasarkan jumlahnya.
- c. Berdasarkan fakta bahwa identitas material menghilang di gudang, pemeriksaan stok akhir akan menjadi sulit.
- d. Seringnya perhitungan tarif akan dibutuhkan jika sering melakukan pembelian, dengan demikian akan melibatkan banyak kegiatan

pencatatan atau pembukuan. Tarif rata-rata bisa diperbaiki karena habisnya stok yang ada meskipun tidak ada pembelian baru.

2. *Moving Average*

Moving average adalah metode yang menghitung rata-rata berdasarkan periode waktu yang sudah ditentukan dan akan bergerak mengikuti periode waktu tersebut. Teknik ini berguna untuk mendapatkan sebuah nilai yang berdasarkan periode waktu yang ditentukan dan dihitung secara berulang. Jika *moving average* yang akan dibuat akan bergerak selama 3 periode maka rata-rata akan dihitung berdasarkan 3 periode waktu dan akan bergerak maju mengikuti waktu selanjutnya. Rumus metode ini adalah:

$$\text{Rata - rata bergerak periode } t = \frac{Y_t + Y_{t-1} + Y_{t-2} + \dots + Y_{t-n+1}}{n}$$

Dalam metode ini, hanya dicantumkan hasil peramalan yang memberikan hasil terbaik dengan nilai eror terkecil karena perhitungan dilakukan beberapa kali melalui beberapa data. Metode ini dinilai sebagai metode yang baik digunakan dalam peramalan *agregat planning*.

3. *Simple Exponential Smoothing*

Simple exponential smoothing adalah metode eksponen paling mudah dan sering juga disebut *single exponential smoothing*. Metode ini cocok untuk peramalan data dengan tidak adanya tren atau pola musiman namun data juga bisa berubah seiring dengan perkembangan waktu. Metode ini juga cocok digunakan pada peramalan jangka pendek seperti untuk satu bulan kedepan. Rumus untuk metode *simple exponential smoothing* adalah

$$F_{t+1} = a \times X_t + (1 - a) \times F_t$$

Dimana:

- F_t : Peramalan untuk periode t
- $X_t + (1 - a)$: Nilai aktual periode waktu
- F_{t+1} : Peramalan pada waktu t + 1
- a : Konstanta perataan antara 0 dan 1

4. *Weighted Moving Average*

Weighted moving average adalah metode peramalan yang dikembangkan dari metode *Moving Average*. *Weighted moving average* meramalkan dengan beberapa data terakhir yang diberikan bobot yang berbeda berdasarkan jika pengaruh data yang baru adalah lebih besar dibandingkan dengan data yang lama terhadap keadaan di masa depan. Rumus *weighted moving average* adalah:

$$\text{Periode } P = (\text{Bobot}_{(p-n)} \times \text{data aktual}_{(p-n)} + \text{Bobot}_{(p-n+1)} \\ \times \text{data aktual}_{(p-n+1)} + \dots + \text{Bobot}_{(n-1)} \times \text{data aktual}_{(p-1)})$$

Dengan total bobot = 1.

2.8 System Development Life Cycle

System Development Life Cycle (SDLC) atau siklus hidup pengembangan sistem adalah proses pemahaman bagaimana sistem informasi dapat mendukung kebutuhan bisnis dengan merancang sebuah sistem, membangunnya, dan memberikan kepada *user* (Dennis et al, 2005). SDLC memiliki empat fase dasar: perencanaan, analisis, desain dan implementasi. Proyek-proyek yang berbeda dapat menekankan bagian-bagian yang berbeda dari SDLC atau pendekatan SDLC dalam cara yang berbeda, tapi semua proyek memiliki unsur-unsur empat fase tersebut. Setiap tahap itu sendiri terdiri dari serangkaian langkah, yang mengandalkan teknik yang menghasilkan penyerahan (khusus dokumen dan file yang memberikan pemahaman tentang proyek).

Metodologi pengembangan sistem merupakan kerangka formal dalam mengimplementasikan konsep SDLC untuk mengembangkan suatu sistem informasi. Ada beberapa jenis metodologi pengembangan sistem dan setiap jenisnya memiliki keunikan masing-masing berdasarkan fokus yang dijabarkan pada setiap tahapan SDLC (Dennis et al, 2005).

2.8.1 Desain Terstruktur

Kategori pertama dari metodologi pengembangan sistem adalah desain terstruktur. Metodologi desain terstruktur mengadopsi pendekatan langkah demi langkah formal SDLC yang bergerak secara logis dari tahap satu ke tahap

berikutnya. Secara umum sebuah tahap selesai sebelum tahap berikutnya dimulai (Dennis et al, 2005).

Model pengembangan sistem terdiri dari dua jenis, yaitu:

1. *Waterfall Development*

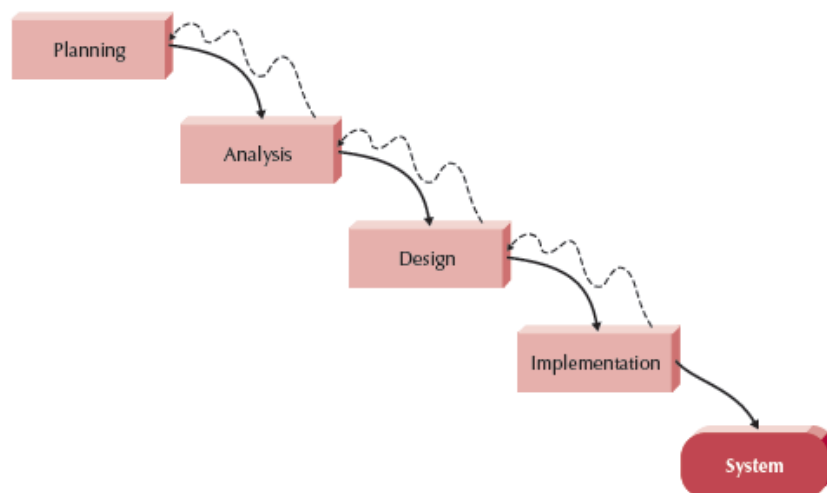
Waterfall Development merupakan suatu cara pengembangan *software* yang fase-fasenya berurutan. Sebuah fase tidak bisa dikerjakan sebelum fase sebelumnya telah selesai dikerjakan. Model ini adalah model yang muncul pertama kali yaitu sekitar tahun 1980 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai dalam *Software Engineering* (Dennis et al, 2005).

Kelebihan dari *Waterfall Development* adalah:

- a. Memahami kebutuhan sistem secara mendalam.
- b. Meminimalisir adanya perubahan kebutuhan selama proyek berjalan.

Kekurangan dari *Waterfall Development* adalah:

- a. Memerlukan waktu yang relatif lama untuk menyelesaikan proyek.
- b. Tidak adaptif terhadap perubahan, karena harus kembali ke proses awal.



Gambar II.3 *Waterfall Development*
Sumber: Dennis et al (2005)

2. *Parallel Development*

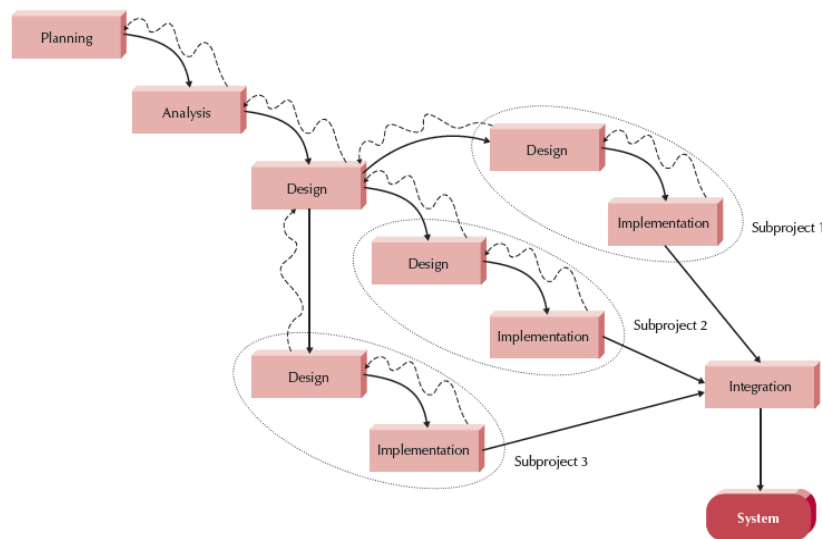
Parallel Development Methodology merupakan suatu cara pada SDLC yang melakukan fase *design* dan *implementation* secara paralel (Dennis et al, 2005).

Kelebihan dari *Parallel Development Methodology* adalah:

- a. Terdiri dari sub-sub proyek yang bisa dikerjakan paralel.
- b. Mempercepat waktu penyelesaian proyek.

Kekurangan dari *Parallel Development Methodology* adalah:

- a. Memerlukan lebih banyak *resource*.
- b. Masih menggunakan dokumen dalam bentuk kertas.
- c. Terkadang timbul masalah integrasi dari sub-sub proyek, sehingga butuh upaya penyesuaian.



Gambar II.4 Paralell Development
Sumber: Dennis et al (2005)

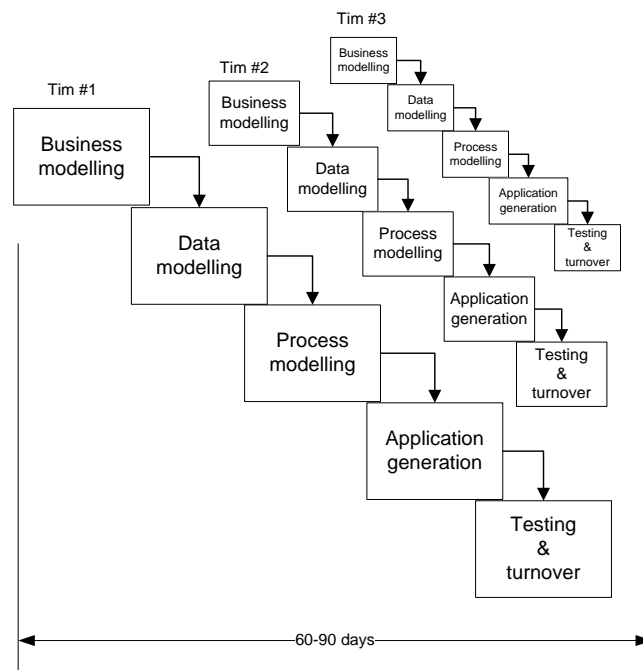
2.8.2 Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah salah satu alternatif dari SDLC yang digunakan untuk mengatasi keterlambatan proses *development*. Keunggulan RAD menggabungkan teknik SDLC, teknik *Joint Application Development* (JAD) dan *Computer Aided Software Engineering* (CASE Tools) yang bertujuan untuk membuat sistem dalam waktu singkat. Dengan cara ini, *user*

dapat lebih memahami sistem dan menyarankan revisi yang membawa sistem lebih dekat dengan apa yang dibutuhkan. RAD melibatkan *user* dalam proses *testing* sehingga dapat memangkas proses *development* yang panjang (Dennis et al, 2005).

Rapid Application Development (RAD) adalah model proses pengembangan perangkat lunak yang bersifat inkremental terutama untuk waktu pengerjaan yang pendek. Model RAD adalah adaptasi dari model *waterfall* versi kecepatan tinggi dengan menggunakan *waterfall* untuk pengembangan setiap komponen perangkat lunak (Pressman, 2005).

Model RAD membagi tim pengembang menjadi beberapa tim untuk mengerjakan beberapa komponen masing-masing tim pengerjaan dapat dilakukan secara paralel sebagaimana dapat dilihat pada gambar II.5.



Gambar II.5 *Rapid Application Development*
Sumber: Pressman (2001)

Berikut merupakan penjelasan dari *Rapid Application Development* (RAD)

1. Pemodelan bisnis

Pemodelan yang dilakukan untuk memodelkan fungsi bisnis untuk mengetahui informasi apa yang terkait proses bisnis, informasi apa saja yang

harus dibuat, siapa yang harus membuat informasi itu, bagaimana alur informasi itu, dan proses apa saja yang terkait informasi itu.

2. **Pemodelan data**

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atributnya beserta relasinya dengan data-data yang lain.

3. **Pemodelan proses**

Mengimplementasikan fungsi bisnis yang sudah didefinisikan terkait dengan pendefinisian data.

4. **Pembuatan aplikasi**

Mengimplementasikan pemodelan proses dan data menjadi program.

5. **Pengujian dan pergantian**

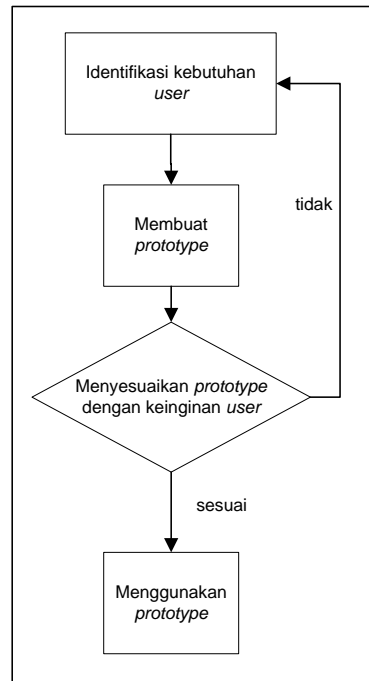
Pada tahap ini tim pengembang menguji komponen-komponen yang dibuat. Jika sudah teruji maka tim pengembang dapat beranjak untuk mengembangkan komponen lainnya.

2.8.3 Prototyping

Menurut McLeod (2004), *prototyping* adalah sebuah versi dari suatu sistem potensial yang menyediakan pengembang dan *user* dengan suatu gambaran tentang bagaimana sistem dalam bentuk sempurnanya akan berfungsi. McLeod (2004) mendefinisikan 2 (dua) tipe dari *prototype* yaitu:

1. ***Evolutionary Prototype***

Evolutionary prototype yaitu *prototype* yang secara terus menerus dikembangkan hingga *prototype* tersebut memenuhi fungsi dan prosedur yang dibutuhkan oleh sistem. Pada pendekatan evolusioner, suatu *prototype* dibangun berdasarkan pada kebutuhan dan pemahaman secara umum. *Prototype* kemudian diubah dan dievolusikan dari pada dibuang. *Prototype* yang dibuang biasanya digunakan dengan aspek sistem yang dimengerti secara luas dan dibangun atas kekuatan tahapan *evolutionary prototype*.



Gambar II.6 *Evolutionary Prototype Model*

Sumber: McLeod (2004)

- a. Identifikasi kebutuhan *user*, pengembang dan pemilik sistem melakukan diskusi dimana *user* atau pemilik sistem menjelaskan kepada pengembang tentang kebutuhan sistem yang mereka inginkan.
- b. Membuat *prototype*, pengembang membuat *prototype* dari sistem yang telah dijelaskan oleh pemilik sistem.
- c. Menyesuaikan *prototype* dengan keinginan pemilik sistem, pengembang menanyakan kepada *user* atau pemilik sistem tentang *prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.
- d. Menggunakan *prototype*, sistem mulai dikembangkan dengan *prototype* yang sudah dibuat.

2.9 *Flowchart*

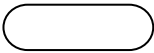



Flowchart adalah diagram yang menunjukkan alir di dalam program atau prosedur sistem secara logika. Diagram alir digunakan untuk alat bantu



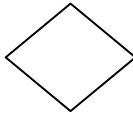
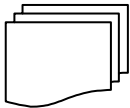
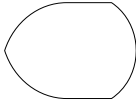


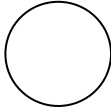
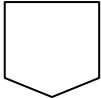
komunikasi dan dokumentasi. Pada waktu menggambar suatu diagram alir, analisis sistem dapat mengikuti ketentuan-ketentuan sebagai berikut (Jogiyanto, 2005):

1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.
3. Harus ditunjukkan 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.
5. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol-simbol bagan alir yang standar.

Simbol-simbol *flowchart* yang digunakan merupakan simbol-simbol *flowchart* standar yang dikeluarkan oleh ANSI dan ISO. Berikut simbol-simbol standar yang digunakan untuk menggambarkan diagram alir sesuai kegunaan simbol (Jogianto, 2005):

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.

No.	Simbol	Nama	Fungsi
5.		<i>Input/output data</i>	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> .
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 Model Language (UML)

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. UML merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri piranti lunak dan pengembangan sistem. (Rama dan Jones, 2008).

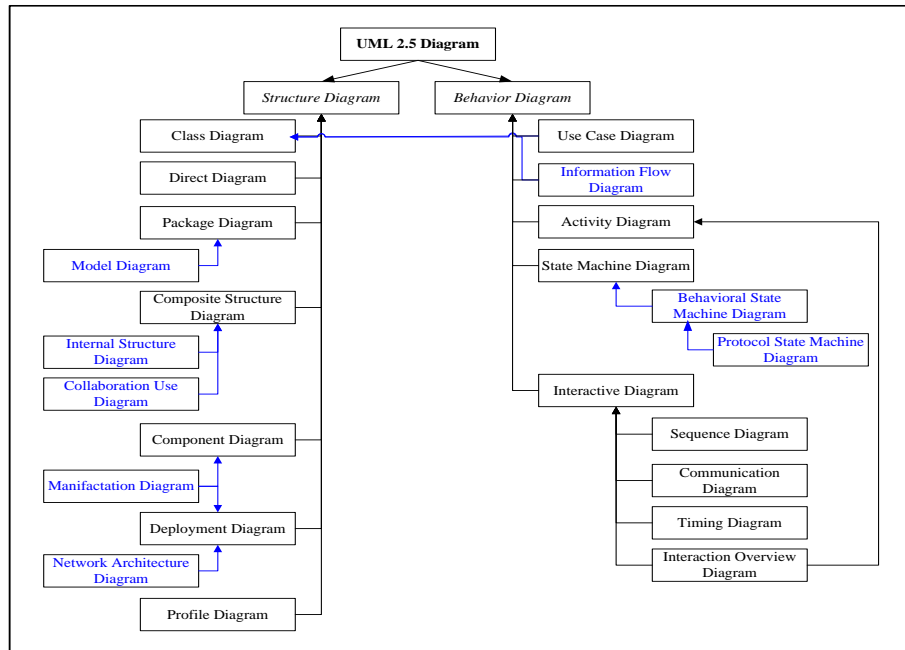
Di mulai pada bulan Oktober 1994, Booch, Rumbaugh dan Jacobson yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan memelopori usaha untuk penyatuan metodologi perancangan berorientasi objek. Pada tahun 1995 dirilis pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh *Object Management Group (OMG)*. (Dharwiyanti, 2015).

UML dikelompokkan menjadi 2 kategori, yaitu *Structure Diagrams* dan *Behavior Diagrams*.

Berikut ini penjelasan singkat dari pembagian kategori UML yaitu:

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.

Pengelompokan UML dapat dilihat pada gambar II.11.






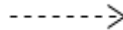


Gambar II.7 UML Diagram
(Sumber: uml-diagrams.org, 2015)

Adapun beberapa penjelasan dari macam-macam diagram tersebut adalah sebagai berikut (Munawar, 2005):

a. *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	Deskripsi
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</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>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
	<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>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem.






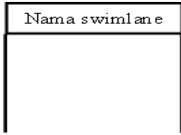
Sumber: Rosa dan Shalahuddin (2011)

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

b. *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 bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2011).

Tabel II.3 Simbol-Simbol *Activity Diagram*


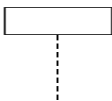
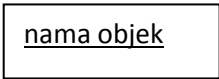

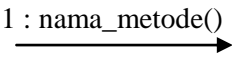
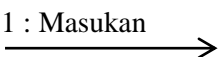
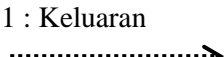
Simbol	Nama	Deskripsi
	Status awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan/ <i>decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari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

b. Sumber: Rosa dan Shalahuddin (2011)

3. *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeksripsikan 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	Deskripsi
	Aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi dibuat.
	<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	Objek	Menyatakan objek yang berinteraksi pesan.
	waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
	Pesan tipe <i>call</i>	Menyatakan suatu objek memanggil operasi atau metode yang ada pada objek lain atau dirinya sendiri.
	Pesan tipe <i>send</i>	Menyatakan suatu objek mengirimkan data ke objek lain.
	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)

4. *Class Diagram*

Class diagram merupakan diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas

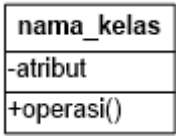




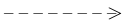

yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas sebagai berikut (Rosa dan Shalahuddin, 2011):

- a. Kelas main. Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
- b. Kelas yang menangani tampilan sistem. Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.
- c. Kelas yang diambil dari pendefinisian *use case*. Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*.
- d. Kelas yang diambil dari pendefinisian data. Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Di dalam kelas terdapat beberapa *keys* yang berfungsi untuk membedakan semua basis data dalam tabel secara unik. Pengertian *keys* menurut Connolly dan Begg (2010) dalam Bina Nusantara (2012) yaitu sebagai berikut:

- a. *Candidate key*
Candidate key adalah sejumlah kecil *attribute* yang secara unik mengidentifikasi setiap kejadian dari setiap tipe *entity*.
- b. *Primary key*
Primary key adalah *candidate key* yang terpilih untuk mendefinisikan secara unik pada setiap kejadian dari sebuah tipe *entity*.
- c. *Composite key*
Composite key adalah sebuah *candidate key* yang terdiri dari dua atau banyak *attribute*.
- d. *Foreign key*
Foreign key adalah himpunan *attribute* dalam data *relationship* yang cocok dengan *candidate key* dari beberapa *relationship* lainnya.

Tabel II.5 Simbol-Simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	<p>Kelas</p> 	Kelas pada struktur sistem.
2.	<p>Antarmuka/interface</p> 	Sama dengan konsep interface dalam pemrograman berorientasi objek.
3.	<p>Asosiasi/association</p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.
4.	<p>Asosiasi berarah</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
5.	<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.	<p>Kebergantungan</p> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
7.	<p>Agregasi/aggregation</p> 	Relasi antar kelas dengan makna semua-bagian (whole-part).

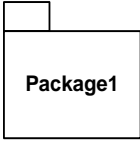
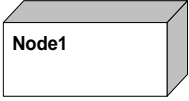
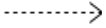

Sumber: Rosa dan Shalahuddin (2011)

5. *Deployment Diagram*

Deployment diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Deployment diagram* juga dapat digunakan untuk memodelkan hal-hal berikut (Rosa dan Shalahuddin, 2014):

- a. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node* dan *hardware*.
- b. Sistem *client/server*.
- c. Sistem terdistribusi murni.
- d. Rekayasa ulang aplikasi.

Tabel II.6 Simbol-simbol *Deployment Diagram*

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

Sumber: Rosa dan Shalahuddin (2014)

2.11 Kamus Data

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

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

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

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

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

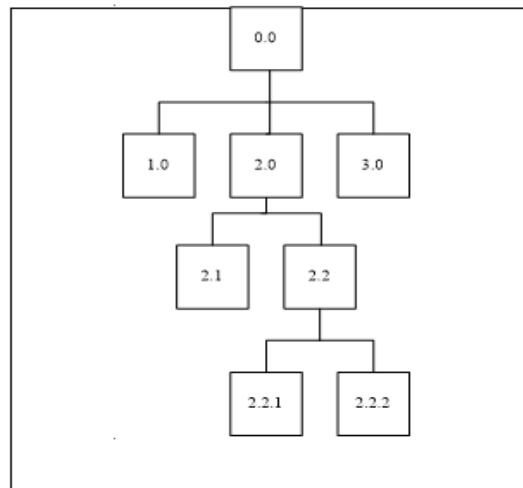
1. Untuk menyediakan suatu struktur guna memahami fungsi-fungsi dari sistem.
2. Untuk lebih menekankan fungsi-fungsi yang harus diselesaikan oleh program, bukannya menunjukkan statemen-statementen program yang digunakan untuk melaksanakan fungsi tersebut.
3. Untuk menyediakan penjelasan yang jelas dari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

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

1. *Visual Table Of Contents (VTOC)*

Visual table of contents menggambarkan hubungan fungsi-fungsi di sistem secara berjenjang, *visual table 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. Berikut adalah Gambar II.9 *Visual table of contents*.



Gambar II.8 *Visual Table Of Contents*
Sumber: Jogiyanto (2005)

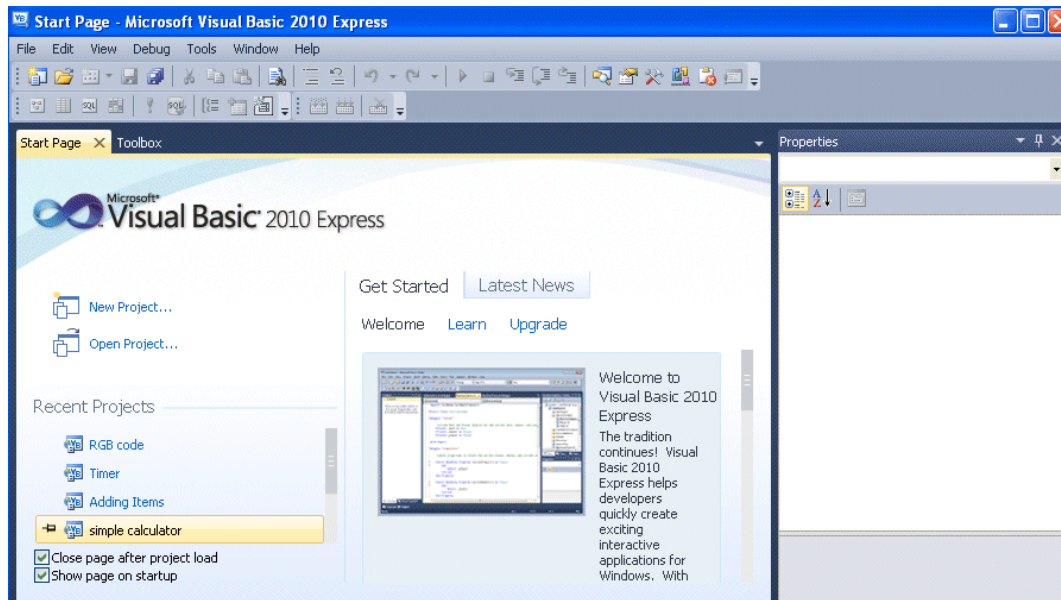
2.13 Visual Basic .NET 2010

Visual Basic .NET adalah salah satu bahasa pemrograman yang bisa digunakan untuk membangun aplikasi-aplikasi .NET di platform Microsoft .NET. tidak seperti generasi sebelumnya yaitu visual basic 6.0 yang lebih difokuskan untuk aplikasi desktop, Visual Basic .NET memungkinkan para pengembang membangun bermacam aplikasi, baik desktop maupun aplikasi web (Prasetyo, 2006).

Antarmuka atau lingkungan dari Visual Basic .NET tidak jauh berbeda dengan visual basic 6.0, kelebihanannya memiliki IDE (*Interface Development Environment*) yang lengkap dan terorganisasi, sehingga mudah bagi pengembang untuk mencari objek-objek atau komponen yang terdapat pada *toolbox* yang diinginkan (Darmayuda, 2010).

Untuk menjalankan Visual Basic .NET 2010 berarti menjalankan Visual Studio .NET 2010 dalam lingkungan pemrograman Visual Basic yaitu dengan cara klik Start > Programs > Microsoft Visual Studio 2010 > Microsoft Visual

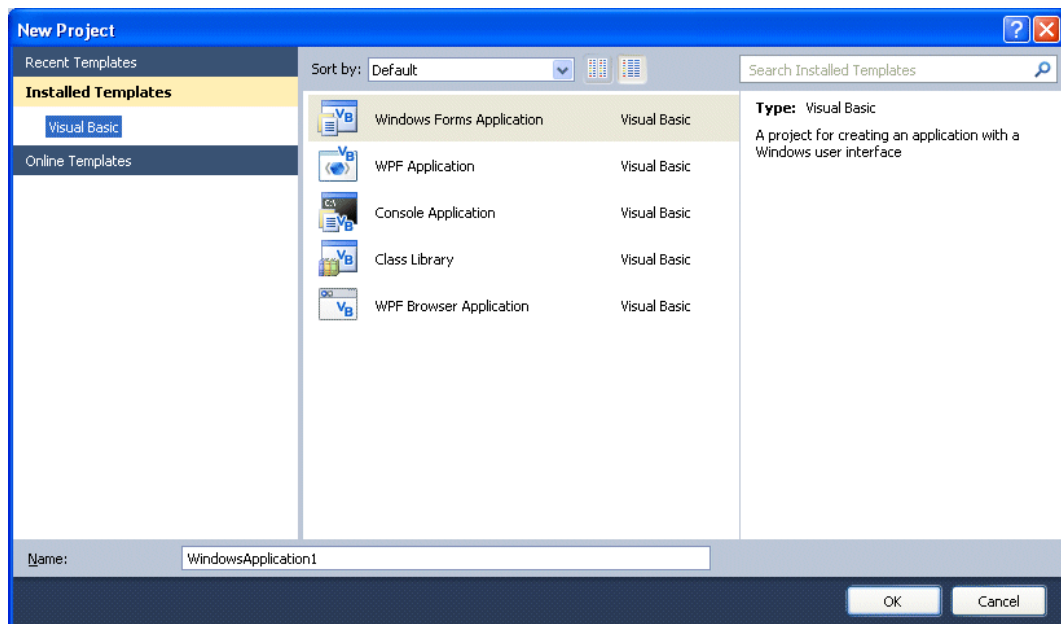
Studio 2010. Di dalam Visual Basic .NET 2010 terdapat beberapa komponen dasar seperti *title bar*, *menu bar*, *tool bar*, *solution explorer*, *toolbox*, *form*, dan *properties*. Antarmuka Visual Basic .NET 2010 dapat dilihat pada gambar II.9



Gambar II.9 Antarmuka Visual Basic .NET 2010

Sumber: <http://www.vbtutor.net/index.php/vb2010-lesson-1-introduction/>

Tahap selanjutnya adalah membuat *form* atau *project* baru yang tampilannya bisa dilihat di Gambar II.10 ini.



Gambar II.10 Antarmuka Pembuatan Project Baru

Sumber: <http://www.vbtutor.net/index.php/vb2010-lesson-1-introduction/>

Lingkungan kerja pada Visual Basic .NET 2010 umumnya memiliki lingkungan sebagai berikut:

1. *Title Bar*

Berfungsi menampilkan nama *project* yang aktif atau sedang dikerjakan.

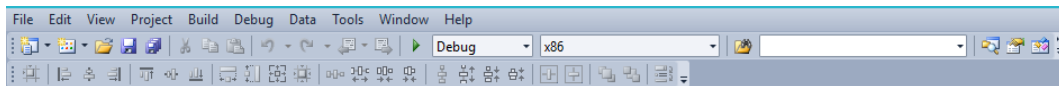


Gambar II.11 *Title Bar* Visual Basic .NET 2010

Sumber: Visual Basic.Net 2010

2. *Menu Bar*

Berfungsi untuk pengolahan fasilitas yang dimiliki Visual Basic .NET 2010. Dibawah *menu bar* ada *toolbar* yang berfungsi untuk melakukan perintah khusus secara cepat.

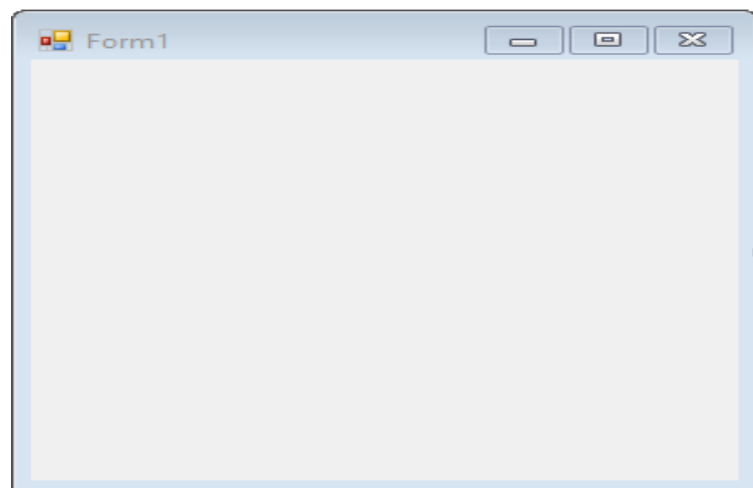


Gambar II.12 *Menu Bar* dan *Toolbar* Visual Basic .NET 2010

Sumber: Visual Basic.Net 2010

3. *Form*

Form adalah objek utama yang berfungsi untuk meletakkan objek-objek yang terdapat pada *toolbox* yang digunakan dalam melakukan perancangan sebuah tampilan program aplikasi.

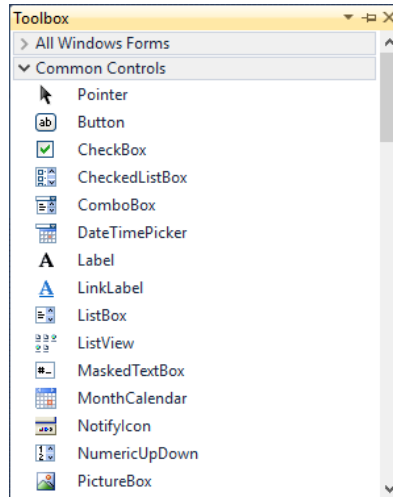


Gambar II.13 *Form* Visual Basic .NET 2010

Sumber: Visual Basic.Net 2010

4. Toolbox

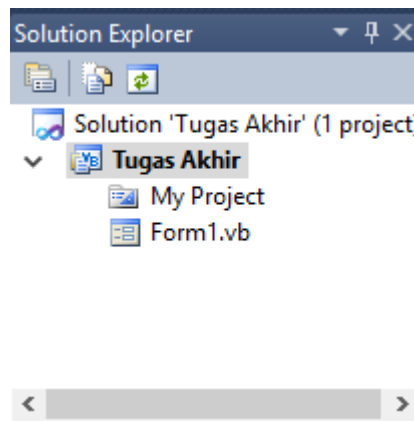
Berfungsi menyediakan objek-objek atau komponen yang digunakan dalam merancang sebuah *form* pada program aplikasi.



Gambar II.14 *Toolbox* Visual Basic .NET 2010
Sumber: Visual Basic.Net 2010

5. Solution Explorer

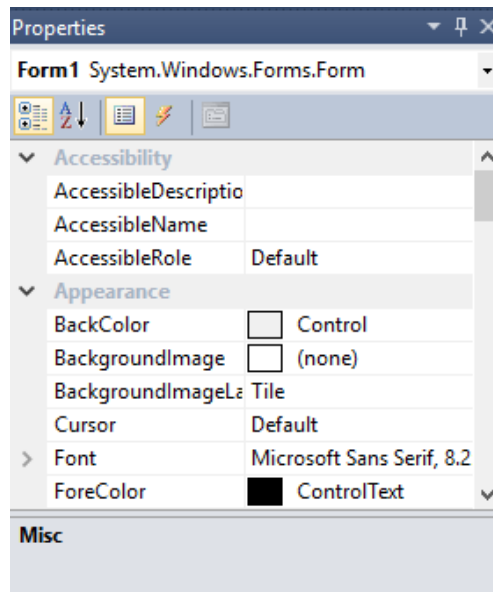
Berfungsi untuk menampilkan *project* berupa *file-file* pendukung yang terdapat pada program aplikasi.



Gambar II.15 *Solution Explorer* Visual Basic .NET 2010
Sumber: Visual Basic.Net 2010

6. Properties Windows

Berfungsi untuk mengatur *properties-properties* pada objek (*setting object*) yang diletakkan pada sebuah *form*.



Gambar II.16 *Form* Visual Basic .NET 2010
 Sumber: Visual Basic.Net 2010

2.14. SQL Server

Microsoft SQL Server adalah sebuah sistem manajemen basis data relasional (RDBMS) dari produk Microsoft. Bahasa kueri utamanya adalah *Transact SQL* yang merupakan implementasi dari SQL standar ANSI/ISO yang digunakan oleh Microsoft dan *Sybase*. Umumnya SQL Server digunakan di dunia bisnis yang memiliki baris data berskala kecil sampai dengan menengah, tetapi kemudian berkembang dengan digunakannya SQL Server pada basis data besar (Wikipedia, 2014)

SQL Server mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Terdapat tiga kategori tipe data yang didukung oleh SQL Server, yaitu tipe data numerik, string, serta penganggalan dan waktu. Sebuah data yang akan disimpan harus sesuai dengan tipe data yang bersangkutan (Wahana Komputer, 2010).

1. Data Numerik

SQL Server mendukung penyimpanan data standar numerik. Data numerik adalah salah satu bentuk data berupa angka, baik berupa bilangan bulat atau bilangan real. Bilangan bulat dapat berupa tipe data integer/int, tinyint, smallint,

dan lainnya. Sebaliknya bilangan real dapat menyimpan data berupa angka pecahan.

Tabel II.7 Tipe Data Numerik *Integer*

Tipe Data	Byte	Keterangan
<i>Tinyint</i>	1	-128 s/d 127 <i>signed</i> , 0-255 <i>unsigned</i>
<i>Smallint</i>	2	-32768 s/d 32767 <i>signed</i> , 0 s/d 65535 <i>unsigned</i>
<i>Mediumint</i>	3	-8388608 s/d 8388607 <i>signed</i> , 0 s/d 16777215 <i>unsigned</i>
<i>Integer/int</i>	4	-2147483648 s/d 2147483647 <i>signed</i> 0 s/d 4294967295 <i>unsigned</i>
<i>Bigint</i>	8	-9223372036854775808 s/d 9223372036854775807 <i>signed</i> 0 s/d 18446744073709551615 <i>unsigned</i>

Sumber: Wahana Komputer (2010)

Tabel II.8 Tipe Data Numerik *Floating Point*

Tipe Data	Byte	Keterangan
<i>Float (p)</i>	4 jika $0 \leq p \leq 24$ 8 jika $25 \leq p \leq 53$	P merepresentasikan presisi bit.
<i>Float</i>	4	Angka <i>floating point</i> kecil (<i>single-precision</i>)
<i>Double [Presisi], real</i>	8	Ukuran normal angka <i>floating point</i> (<i>double-precision</i>)
<i>Decimal/dec (M,D), Numeric (M<D)</i>	variasi	M adalah jumlah angka digit desimal dan D adalah angka di belakang tanda desimal.
<i>Bit (M)</i>	$(M+7)/8$	M adalah banyaknya bit setiap nilai. Bit

Tipe Data	Byte	Keterangan
		ditambahkan pada versi 5.0.5, sebelumnya merupakan sinonim <i>tinyint</i> (1).

Sumber: Wahana Komputer (2010)

2. Data String

Tipe data string, dapat menyimpan semua data baik karakter, angka, waktu, maupun tanggal. Data dapat pula merupakan kombinasi karakter dan angka. Tipe data string pada SQL Server dapat digolongkan menjadi dua, yaitu kelompok data yang berbentuk *Text/Blob* dan selain *Text/Blob*.

Tabel II.9 Tipe Data String *Text/Blob*

Tipe Data	Byte	Kapasitas Penyimpanan
<i>Tinytext</i> <i>Tinyblob</i>	2^8-1	L+1 $0 \leq L \leq 2^8-1$
<i>Text</i> <i>Blob</i>	$2^{16}-1$	L+2 $0 \leq L \leq 2^{16}-1$
<i>Mediumtext</i> <i>Mediumblob</i>	$2^{24}-1$	L+3 $0 \leq L \leq 2^{24}-1$
<i>Longtext</i> <i>Longblob</i>	$2^{32}-1$	L+4 $0 \leq L \leq 2^{32}-1$

Sumber: Wahana Komputer (2010)

Tabel II.10 Tipe Data String selain *TEXT/BLOB*

Tipe Data	Byte	Keterangan
<i>Varchar</i>	255	Tipe <i>varchar</i> menyimpan data sebanyak yang dimasukkan.
<i>Char</i>	255	Tipe <i>char</i> sama dengan tipe <i>varchar</i> , hanya tempat penyimpanan selalu tetap.
<i>Binary</i>	255	<i>Binary</i> mirip dengan <i>char</i> , hanya saja yang disimpan adalah nilai biner (byte) dari data yang disimpan.
<i>Varbinary</i>	255	Menyimpan nilai biner sebanyak data yang dimasukkan.

Tipe Data	Byte	Keterangan
<i>Enum</i>	N	Tipe data ini disebut juga tipe data validasi. Pada tipe data ini, data input telah dideklarasikan terlebih dahulu.
<i>Set</i>	N	Tipe <i>Set</i> memiliki fungsi yang sama dengan enum. N adalah banyaknya daftar input, tiap nilai bernilai 1 <i>byte</i> .

Sumber: Wahana Komputer (2010)

3. Tipe Data Penganggalan dan Waktu

Dalam menangani data tanggal dan waktu, SQL Server memiliki tipe data tersendiri.

Tabel II.11 Tipe Data Tanggal dan Waktu

Tipe Data	Byte	Keterangan
<i>Datetime</i>	8	Merupakan tipe data yang menyimpan tanggal dan jam. Formatnya YYYY-MM-DD HH:MM:SS.
<i>Date</i>	3	Tipe ini hanya menyimpan data tanggal. Formatnya YYYY-MM-DD.
<i>Timestamp</i>	4	Tipe data ini ditulis berjajar tanpa ada pembatas, menyimpan tanggal dan jam. Formatnya adalah YYYYMMDDHHMMSS.
<i>Time</i>	3	Tipe data ini hanya menyimpan data jam dengan format HH:MM:SS.
<i>Year</i>	1	Tipe data ini hanya menyimpan data tahun dengan format YYYY.

Sumber: Wahana Komputer (2010)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, dan memilih langkah-langkah sistematis (Hasan, 2002).

Dengan adanya metodologi penelitian, 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 berdasarkan fakta-fakta secara ilmiah. Kegiatan penelitian dikerjakan akan lebih terarah sesuai dengan prosedur yang telah ditetapkan namun tidak menutup kemungkinan akan terjadi variasi urutan penelitian yang tentunya disebabkan oleh permasalahan yang sedang diteliti. Metodologi penelitian yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Observasi

Tahap ini merupakan tahap melakukan pengamatan langsung terhadap pengguna sistem dan sistem yang sedang berjalan. Tahap ini melakukan observasi pada departemen PPIC seksi *indirect material* yang belum memiliki sistem perencanaan *indirect material* yang terencana dengan baik.

2. Wawancara

Pada tahap ini dilakukan wawancara dengan para pengguna sistem tersebut untuk memperoleh keterangan mengenai alur proses *input* data perencanaan *indirect material* mengenai kebutuhan sistem yang akan dikembangkan selanjutnya. Keterangan yang diperoleh bertujuan agar sistem selanjutnya dapat lebih mudah, cepat, dan akurat dalam melakukan penyampaian data dan penanganan dalam perencanaan *indirect material*.

3. Studi Literatur

Tahap ini merupakan tahap awal pengumpulan data dari sumber-sumber seperti buku yang berhubungan analisis dan desain sistem informasi, buku pemrograman menggunakan visual basic .net, dan juga mencari data tambahan yang diperlukan melalui internet.

3.2 Identifikasi Masalah

Identifikasi masalah adalah salah satu proses penelitian yang boleh dikatakan paling penting di antara proses lain. Masalah penelitian akan menentukan kualitas dari penelitian, bahkan juga menentukan apakah sebuah kegiatan bisa disebut penelitian atau tidak. Masalah penelitian secara umum bisa ditemukan lewat studi literatur atau lewat pengamatan lapangan (observasi).

Identifikasi masalah yang dilakukan di PT Showa Indonesia Mfg menemukan bahwa proses perencanaan *indirect material* setiap bulannya kurang tepat dengan penggunaan selama sebulan karena perhitungan yang dilakukan belum sesuai. Hal ini mengakibatkan keadaan stok material tidak aman. Selain itu masalah lain yang ada adalah proses pengajuan *purchase order* masih dikirim secara manual dan mengalami penumpukan dokumen karena *section head* tidak langsung memvalidasi sehingga terjadi kerusakan dan kehilangan dokumen.

3.3 Pengumpulan Data

Pengumpulan data merupakan sebuah sarana yang menampung dan mengumpulkan data-data dari berbagai sumber dan bentuk yang akan digunakan dalam keperluan menganalisis sistem untuk perancangan sistem usulan. Sumber data atau informasi penelitian ini berdasarkan kepada jenis data yang diperlukan.

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 PPIC *indirect material* sebagai sumber informasinya. Dalam penelitian ini data tersebut berupa aliran data perencanaan *indirect material*, prosedur perencanaan *indirect material*, prosedur

perhitungan kalkulasi perencanaan material pada seksi PPIC *indirect material* PT Showa Indonesia Mfg.

2. Data Sekunder

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

3.4 Metode Pengembangan Sistem

Metodologi pengembangan sistem merupakan kerangka formal dalam mengimplementasikan konsep SDLC untuk mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik) (Rosa dan Shalahuddin, 2014). Pada penelitian ini untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat pengembangan sistem. Dalam pengembangan sistem ini digunakan metode prototipe evolusioner.

Metode prototipe merupakan cara yang potensial disediakan oleh pengembang kepada calon pengguna dengan tujuan memperoleh umpan balik dari pengguna untuk ditingkatkan sampai sistem dianggap sempurna. prototipe evolusioner adalah prototipe yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi (McLeod, 2011).

Tahapan-tahapan dalam prototipe evolusioner adalah sebagai berikut:

1. Identifikasi kebutuhan pengguna

Pengembang melakukan diskusi dan observasi dengan Seksi *Indirect Material* PPIC di PT Showa Indonesia Mfg tentang kebutuhan sistem yang diinginkan.

2. Membuat prototipe

Pengembang membuat prototipe dari sistem yang telah dijelaskan oleh seksi *Indirect Material* PPIC di PT Showa Indonesia Mfg.

3. Menyesuaikan dan evaluasi prototipe dengan keinginan *user*

Pengembang menanyakan Seksi *Indirect Material* tentang prototipe yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.

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 sesuai dengan metodologi penelitian yang digunakan dalam penelitian tersebut. Penjelasan langkah-langkah atau tahapan dalam penelitian tersebut adalah sebagai berikut:

1. Studi Pendahuluan

Studi Pendahuluan merupakan langkah awal yang dilakukan untuk memulai penelitian. Pendahuluan dilakukan 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. Pendahuluan ini dimaksudkan untuk dapat mengetahui suatu gambaran yang jelas untuk mengetahui masalah yang sedang dihadapi. Langkah-langkah yang dilakukan dalam studi yang bertujuan untuk mendapatkan informasi adalah dengan cara observasi dan melakukan wawancara terhadap pihak-pihak terkait yang berada di PT Showa Indonesia Mfg.

2. Identifikasi Masalah

Pokok dari permasalahan yang menjadi objek penelitian ini adalah mengenai persoalan perencanaan *indirect material* pada seksi *indirect material* PPIC PT Showa Indonesia Mfg. Identifikasi dari permasalahan tersebut yaitu perencanaan *indirect material* belum sesuai dengan kebutuhan dan stok yang ada dan proses validasi penyetujuan perencanaan *indirect material* masih dilakukan secara manual sehingga belum efisien dari segi pemanfaatan waktu.

3. Tujuan Penelitian

Tujuan penelitian dari tugas akhir ini adalah merancang suatu aplikasi Perencanaan *Indirect Material* berbasis *desktop* untuk memudahkan kegiatan operasional seksi *Indirect Material* PPIC di PT Showa Indonesia Mfg dalam kegiatan perencanaan *indirect material*.

4. Batasan Masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya. Batasan masalah yang ada pada penelitian ini adalah pengamatan mencakup pada data pemesanan *indirect material* untuk mengelola permasalahan pemesanan pada Departemen PPIC PT Showa Indonesia Manufacturing, kemudian membuat aplikasi pengolahan data dan menghasilkan laporan perencanaan *indirect material* serta tidak membahas *direct material* atau bahan baku langsung.

5. Identifikasi Kebutuhan User

Identifikasi kebutuhan *user* merupakan langkah lanjutan dari pengolahan data, dimana semua spesifikasi sistem dan kebutuhannya dituangkan ke dalam sebuah dokumen persyaratan perangkat lunak. Identifikasi kebutuhan user didapatkan melalui dua proses yaitu proses pengumpulan data dan pengolahan data. Pengumpulan data didapatkan melalui gambaran umum tentang PPIC, laporan perencanaan *indirect material*, dan *flowchart* perencanaan *indirect material*. Selain itu pada pengolahan data dapat dilihat dari *usecase diagram* sistem perencanaan *indirect material* dan *activity diagram* sistem perencanaan *indirect material*.

6. Membuat prototipe

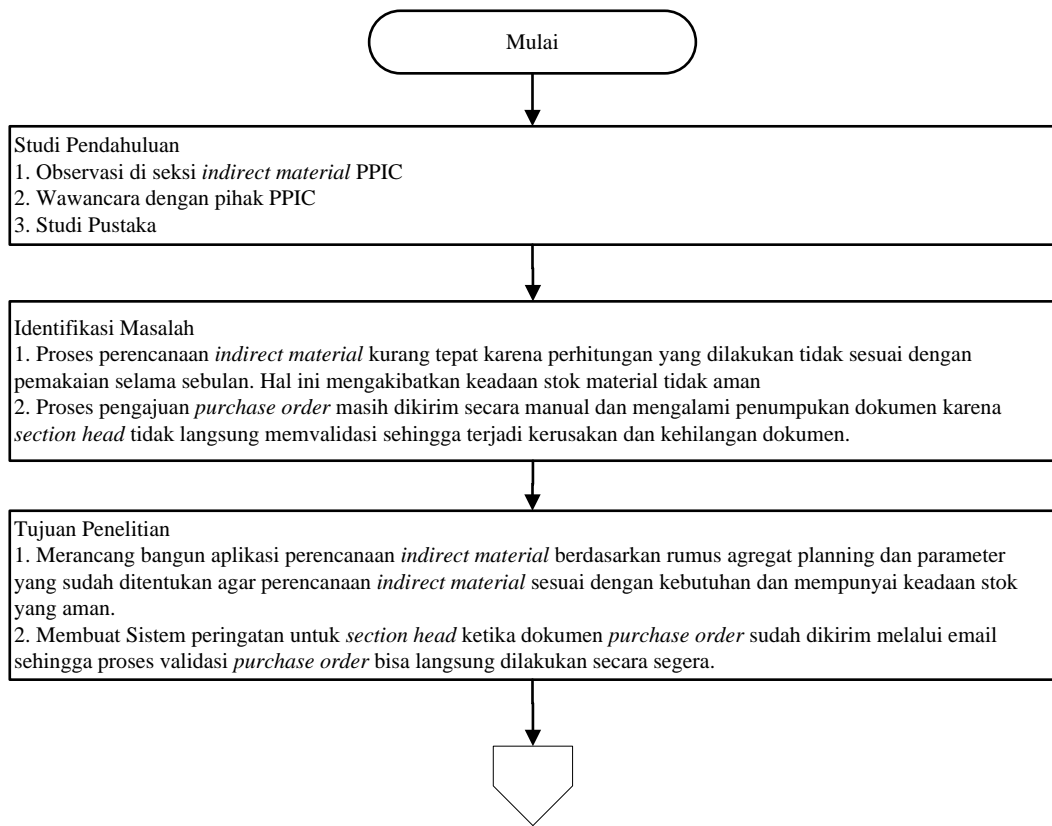
Membuat sebuah prototipe sesuai dengan analisis yang dilakukan. Pada tahap ini akan dirancang aplikasi tentang sistem perencanaan *indirect material*. Pembuatan prototipe dijabarkan menjadi 4 tahap. Tahap pertama adalah pemodelan sistem yang dibuat dengan *usecase diagram*, *activity diagram*, dan *sequence diagram*. Tahap selanjutnya adalah pemodelan data yang terdiri dari *class diagram* dan kamus data. Setelah dilakukan pemodelan data maka akan dilanjutkan dengan pemodelan implementasi dengan *deployment diagram*. Setelah ketiga tahap tadi dilakukan maka tahap terakhir adalah perancangan program dengan HIPO (*Hierarchy plus Input-Process-Output*), *flowchart* program dan rancangan program.

7. Menggunakan prototipe

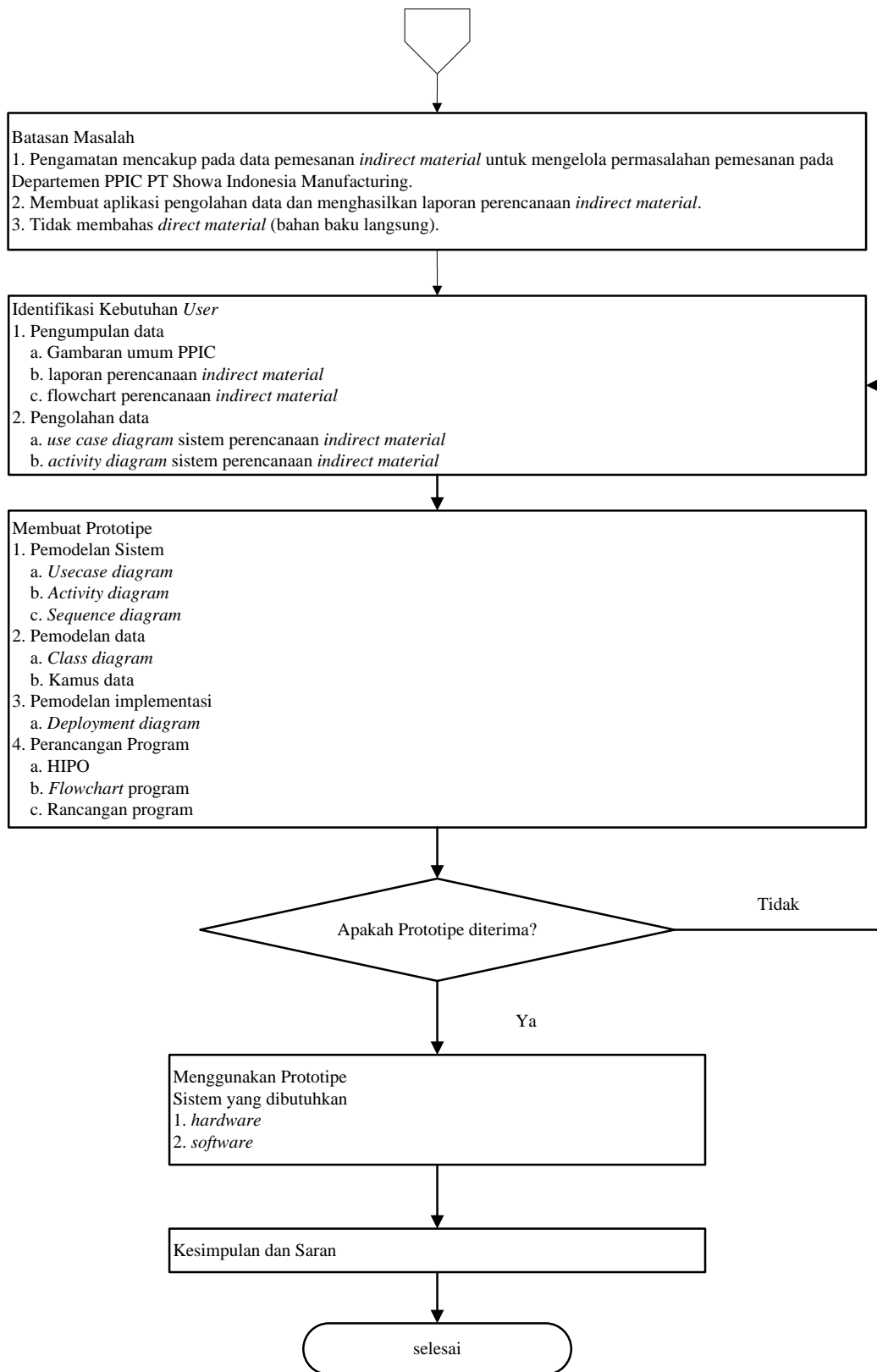
Pada tahap ini, *prototype* dari sistem yang diusulkan akan dilihat apakah sistem usulan tersebut sudah sesuai dengan keinginan pengguna atau belum. Jika belum, maka akan dilakukan pengumpulan dan menganalisis data kembali dan mencari tahu kebutuhan apa yang dibutuhkan oleh pengguna sistem, serta melakukan perancangan sistem ulang. Sedangkan jika sudah sesuai keinginan *user*, maka *prototype* tersebut dapat digunakan. Kemudian akan dibuat spesifikasi berupa *hardware* dan *software* yang menunjang prototipe ini.

8. Kesimpulan dan Saran

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



Gambar III.1 Kerangka Penelitian



Gambar III.2 Kerangka Penelitian (lanjutan)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sekilas Tentang PT Showa Indonesia Mfg.

PT Showa Indonesia Manufacturing (Mfg) adalah perusahaan hasil kerja sama antara Showa Corporation Jepang (55%) dengan PT Astra Honda Motor (45%). PT Showa Indonesia Mfg berdiri di Jakarta pada tanggal 8 Maret 1978 dan berlokasi di Jalan Raya Gaya Motor Sunter II Jakarta Utara dengan Akte Notaris Kartini Mulyadi SH Nomor 74. Pada saat itu PT Showa Indonesia Mfg hanya mempunyai lahan dengan luas tanah 18,142 m².

PT Showa Indonesia Mfg adalah salah satu perusahaan besar yang bergerak di bidang otomotif untuk kendaraan roda dua maupun roda empat, sebagian besar hasil produksinya adalah *Shock Absorber* atau sering disebut juga sebagai peredam kejut. Selain itu PT Showa Indonesia Mfg juga menghasilkan produk berupa *Front Fork, Rear Cushion, Steering Stem, Gear Housing* dan komponen otomotif lainnya.

PT Showa Indonesia Mfg memproduksi *Shock Absorber* untuk kendaraan roda dua dan roda empat. Awalnya PT Showa Indonesia Mfg hanya memproduksi produk *Shock Absorber* untuk kendaraan roda dua untuk motor jenis Honda tepatnya pada tahun 1979. Kemudian perusahaan tersebut mencoba untuk memproduksi *Shock Absorber* roda empat untuk mobil merk Honda dan Suzuki. Produksi tersebut dimulai pada tahun 1980.

PT Showa Indonesia Mfg didirikan pada tanggal 8 maret 1978 dengan luas tanah 18.142 m²., serta dengan modal 1.000.000,00 dolar yang terletak di Jalan Raya Gaya Motor Sunter II Jakarta Utara yang bergerak dibidang produksi komponen otomotif. Kemudian tanggal 23 mei 1996 PT Showa Indonesia Mfg melakukan perpindahan produksinya secara total ke Kawasan Industri Cikarang, Jawa Barat, Jl Jababeka VI kav.

28-32 Cikarang Bekasi 17530, dengan luas tanah 104.640 m² dan memiliki 6 gedung, yaitu gedung A,B,C,D,E dan F.

PT Showa Indonesia Manufacturing mulai melakukan ekspor ke Jepang pada tahun 1988 dan tahun 1989, mengekspor komponen *Shock Absorber* ke Malaysia dan Thailand, sementara produksi untuk Mitsubishi dimulai tahun 1990, dengan semakin bertambahnya produksi maka menuntut para manajemen untuk mengeskerikan pabrik ke tahap 11.

Seiring dengan berjalannya waktu PT Showa Indonesia Mfg berkembang dengan pesat. PT Showa Indonesia Mfg mencoba untuk melakukan ekspor keluar negeri, diantaranya adalah Jepang, Malaysia, dan Thailand. Produksi yang dihasilkan dan diekspor ke negara-negara pada tahun 1988-1990 berupa *spring* dan pada tahun 1990 memproduksi *Shock Absorber* untuk Mitsubishi. Dengan kualitas yang cukup menjamin, PT Showa Indonesia Mfg mempunyai konsumen tetap bahkan semakin bertambah baik dari dalam negeri maupun luar negeri.

4.2 Profil Perusahaan

Profil singkat PT Showa Indonesia Manufacturing sebagai berikut:

1. Alamat

Kantor dan pabrik : Gedung Graha Kirana (Representative office)

Jl. Yos Sudarso Kav. 88 Jakarta Utara

Cikarang Plant (Office dan Factory)

Jl. Jababeka VI Kav. 28 – 36 Jababeka Industrial Estate, Cikarang Bekasi

Telepon : (021) 8934855-56

Fax : (021) 8934875

2. Tanggal Pendirian : 8 Maret 1978

3. Aktivitas Industri : Produksi komponen otomotif, khususnya *Shock Absorber* untuk kendaraan roda dua dan roda empat dan produksi yang berhubungan lainnya.
4. Status Perusahaan : Perseroan Terbatas, dengan kepemilikan dibagi
 Showa Corporation Japan (55%)
 PT Astra Honda Motor (45%)
5. Produksi komersil : 1 Juni 1979
6. Jumlah Pegawai :
 Pria : 2.402 orang
 Wanita : 62 orang
 Ekspatriat : 5 orang
 Total : 2.469 orang
7. Luas Pabrik : 50,568.00 m2 (Land 1)
 31,040.00 m2 (Land 2)
 11,520.00 m2 (Land 3)
 10,000.00 m2 (Land 4)
 40,800 m2 (Factory)

4.3 Visi, Misi, dan Tujuan Perusahaan

PT Showa Indonesia Mfg merupakan perusahaan yang menjalankan fungsi seperti produksi dan penjualan. Visi dan misi dari perusahaan ini adalah:

Visi

Menyediakan komponen otomotif kelas dunia, terutama *Shock Absorber* dan *Steering Stem* untuk pasar global OEM (*Original Equipment Manufacturing*) dan REM (*Replacement Equipment Manufacturing*) dengan kualitas terbaik dan aman untuk pihak-pihak yang berkepentingan.

Misi

1. Menjadikan pemimpin Industri *Shock Absorber* OEM di Kawasan ASEAN dan sebagai pemasok global *Shock Absorber* dan *Steering Stem*.

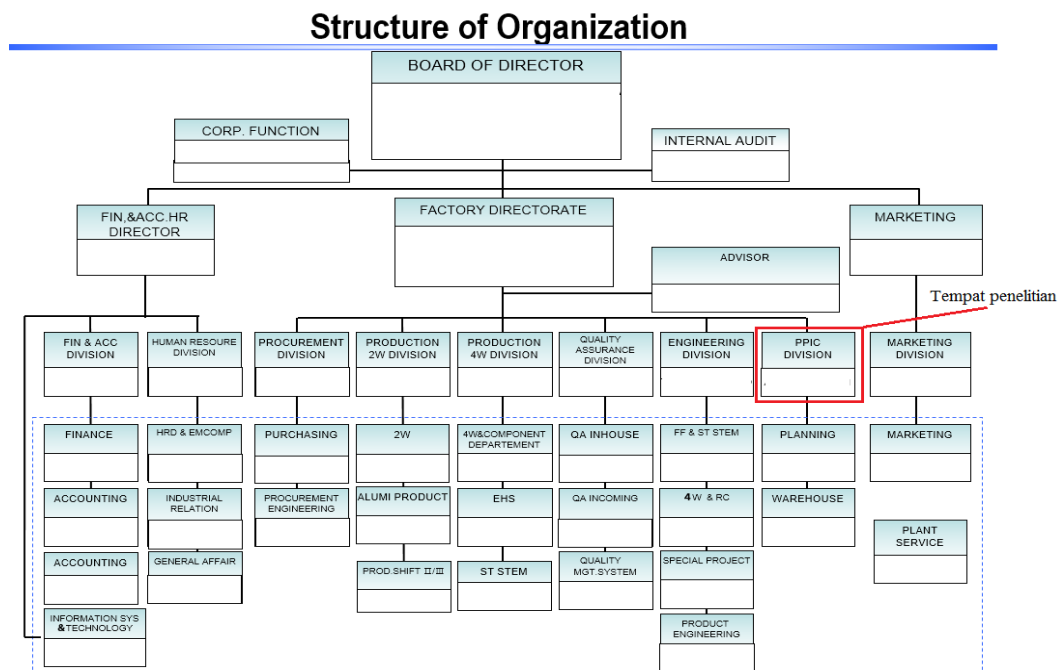
- Menciptakan kepuasan pelanggan dengan kualitas tertinggi, harga bersaing, sumber daya manusia yang kompeten, komitmen sebagai perusahaan yang ramah lingkungan, aman, nyaman efisien dan produktif.

Tujuan

- Mengurangi keluarnya devisa lebih banyak dari pembelian impor penuh (*buildup*) menjadi CKD (*Completely Knock Down*) yang dibuat dan dirakit di Indonesia
- Impor (*Assembling*)
- Menciptakan lapangan kerja baru.
- Memperkuat industri komponen kendaraan bermotor.
- Melakukan inovasi dan pengembangan produk.

4.4 Struktur Organisasi

Untuk menjalankan usahanya, setiap perusahaan membutuhkan suatu struktur organisasi yang baik. Struktur organisasi bisa diartikan sebagai suatu susunan dan hubungan antara setiap bagian serta posisi yang ada di dalam suatu perusahaan. Sebuah struktur organisasi menjelaskan tentang pembagian kerja, pelimpahan wewenang, kesatuan perintah dan tanggung jawab yang jelas. Sebuah struktur organisasi yang tersusun dengan baik akan memudahkan koordinasi, intergrasi, serta meningkatkan produktivitas sebuah perusahaan dalam mencapai tujuan yang diinginkan.



Gambar IV.1 Struktur Organisasi PT Showa Indonesia Mfg

Sumber: PT Showa Indonesia Manufacturing, 2015

Berikut adalah struktur organisasi perusahaan dan struktur organisasi PT Showa Indonesia Mfg:

1. *Board of Director*

Board of director bertugas untuk mengawasi kegiatan perusahaan yang berjalan. pengawasan yang dilakukan adalah dengan mengendalikan kegiatan administrasi, merencanakan dan mengembangkan sumber-sumber pendapatan serta pembelanjaan dan kekayaan perusahaan, dan mengkoordinasikan kegiatan pengadaan dan perlengkapan perusahaan.

2. *Financial and Accounting Human Resources Director*

Bertanggungjawab untuk mengatur keuangan dan sumber daya yang ada dalam perusahaan. kegiatan ini meliputi pengolahan pembelanjaan perusahaan, tenaga kerja, dan relasi antar perusahaan lainnya.

3. *Factory Directorate*

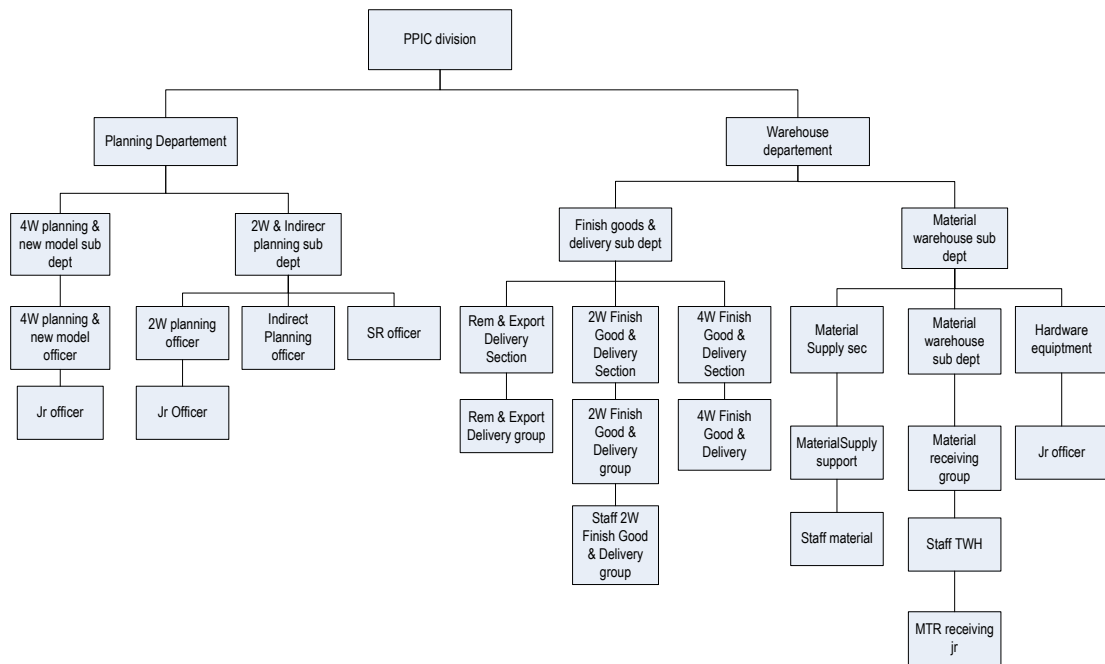
Bertanggungjawab terhadap proses produksi perusahaan. Termasuk didalamnya adalah proses produksi roda dua dan roda empat, perencanaan produksi, *engineering*, dan *procurement*.

4. *Marketing*

Membina hubungan yang harmonis dengan pelanggan dengan melakukan identifikasi harapan pelanggan sehingga pemesanan terus bertambah.

5. *PPIC (Production Planning dan Inventory Control)*

Bertanggung jawab terhadap ketersediaan material yang diproses produksi, perencanaan proses produksi, penyimpanan dan pengiriman produk kepada pelanggan yang tepat waktu dan jumlah.



Gambar IV.2 Struktur Organisasi Divisi PPIC

Sumber: PT Showa Indonesia Manufacturing, 2015

Tidak hanya di dalam perusahaan, suatu departemen yang ada di dalamnya juga memiliki struktur organisasinya sendiri. Dalam PT Showa Indonesia Mfg, departemen PPIC terbagi menjadi dua bagian yaitu divisi perencanaan dan divisi gudang. Pendeskripsian pekerjaan pada divisi PPIC departemen perencanaan adalah sebagai berikut:

1. *Staff Material Planner* (2W atau 4W)
 - a. Memasukan PO atau *Purchase Order* dari konsumen dan koordinasi dengan bagian *Marketing* terhadap pemenuhan pesanan dari kenaikan atau penurunan pesanan.
 - b. Melakukan perencanaan *material planning* berdasarkan rencana *inhouse* produksi, PO atau *forecast* konsumen dan mengeluarkan *purchase requisition* (PR) ke *Procurement*.
 - c. Melakukan pengamatan harian dan kontrol mengenai realisasi kedatangan *part* terhadap rencana kedatangan *part* yang telah dibuat, dan melakukan perubahan apabila diperlukan seperti permintaan harian.

- d. Amati dan kontrol kalkulasi pesanan terhadap kenaikan atau penurunan pesanan atau *forecast* konsumen, dan melakukan simulasi atau perubahan kalkulasi atau PR apabila diperlukan.
 - e. Melakukan koordinasi dengan departemen terkait (*Procurement, Marketing, Produksi, Quality*) mengenai realisasi kedatangan part terhadap rencana kebutuhan part tersebut.
 - f. Melaporkan keadaan *level stock*, DOH, dan lain-lain
 - g. Terlibat dalam aktivitas QCC dan SS
 - h. Mendukung setiap program kerja divisi (KPI Departemen dan Divisi).
2. *Staff Production Planner* (2W atau 4W)
- a. Memasukan PO atau *Purchase Order* dari konsumen dan koordinasi dengan bagian Pemasaran terhadap pemenuhan pesanan dari kenaikan atau penurunan pesanan.
 - b. Melakukan perencanaan produksi *work center* level FG (contoh perakitan) berdasarkan kebutuhan atau jadwal Konsumen, kapasitas dan kebijakan level stok FG
 - c. Melakukan pengamatan harian dan kontrol mengenai realisasi hasil produksi, kanban (permintaan harian konsumen) dan posisi stok FG.
 - d. Membuat perencanaan produksi harian dan koordinasi dengan seksi produksi untuk rencana H sampai dengan H+7.
 - e. Melakukan perencanaan kebutuhan sarana FG terkait kenaikan dan penurunan *Forecast* atau PO dan *new model*.
 - f. Melakukan koordinasi dengan departemen terkait (*Procurement, Marketing, Produksi, Quality*) mengenai realisasi produksi dan pengiriman konsumen.
 - g. Melaporkan hasil *hibikan* produksi, level stok, DOH dan lain-lain.
 - h. Terlibat dalam aktivitas QCC dan SS
 - i. Mendukung setiap program kerja divisi (KPI Departemen dan Divisi).

3. Staf
 - a. Pemeliharaan dan penyesuaian perubahan *basic data inventory* pada aplikasi IFS yang telah disetujui atasan dan departemen terkait.
 - b. Konfigurasi *basic data inventory* di PPIC.
 - c. Pengendalian *form* perubahan inventori karena MLO.
 - d. Pengendalian *form new* atau perubahan inventori part selain karena MLO.
 - e. Pengendalian pemastian setiap aktivitas untuk mendukung delivery new model ke konsumen dari tahap percobaan, pp1, 2, 3 sampai dengan produksi massal.
 - f. Koordinasi dan membantu masalah atau *Claim database inventory*.
 - g. Koordinasi dan membantu kustomisasi dan perkembangan terkait sistem atau aplikasi inventori basis data.
 - h. Melaporkan dan membantu data terkait dasar dan analisa bagian inventori.
 - i. Terlibat dalam aktivitas QCC dan SS.
 - j. Mendukung setiap program kerja divisi (KPI Departemen dan Divisi).
4. *Subdept Head Planning 4W dan System New Model*
 - a. Memimpin, merencanakan, mengatur, mengorganisir dan mengontrol perencanaan di bagian perencanaan 4W dan *System New Model*, antara lain:
 - 1) Merencanakan dan mengontrol perencanaan produksi *inhouse* berdasarkan kebutuhan atau jadwal konsumen, kapasitas dan kebijakan *level stock FG*.
 - 2) Merencanakan dan mengontrol perencanaan kedatangan *material part* dari pemasok, baik CKD atau Lokal dan Jasa berdasarkan PO, *Forecast* atau perencanaan produksi *inhouse* dan *level part material*.
 - 3) Menjamin keakurasian basis data inventori dan kelancaran sistem pada aplikasi IFS.
 - b. Mengkoordinasikan pengendalian aktivitas model baru pada sistem PPIC.
 - c. Membangun sistem dan organisasi PPIC stream 4W dan *System New Model* yang efektif.
 - d. Memonitor, mengontrol dan melaporkan pencapaian KPI Streamnya.

- e. Melaporkan realisasi produksi harian H-1, kondisi *level stock* FG dan rencana pengiriman ke konsumen di laporan pagi Direktorat.
 - f. Melakukan koordinasi dengan departemen terkait mengenai realisasi pencapaian produksi dan kelancaran pemenuhan pengiriman.
 - g. Mendorong berlangsungnya aktifitas QCC dan SS.
5. *Subdept Head Planning 2W*
- a. Memimpin, merencanakan, mengatur, mengorganisasi dan mengontrol perencanaan di bagian produksi, perencanaan, dan perencanaan material di Stream 2W, antara lain:
 - 1) Merencanakan dan mengontrol perencanaan produksi *inhouse* berdasarkan kebutuhan atau jadwal konsumen, kapasitas dan kebijakan level stock FG.
 - 2) Merencanakan dan mengontrol perencanaan kedatangan material part dari pemasok baik CKD atau Lokal dan jasa berdasarkan PO atau *Forecast* atau perencanaan produksi *inhouse* dan *level part material*.
 - b. Membangun sistem dan organisasi PPIC *stream 2W* yang efektif.
 - c. Memonitor, mengontrol, dan melaporkan pencapaian KPI stream 2W.
 - d. Melaporkan realisasi hasil produksi harian H-1, kondisi *level stock* FG dan rencana pengiriman ke konsumen di laporan pagi Direktorat.
 - e. Melakukan koordinasi dengan departemen terkait mengenai realisasi pencapaian produksi dan kelancaran pemenuhan pengiriman.
 - f. Mendorong berlangsungnya aktivitas QCC dan SS.

4.5 Data Indirect Material

Indirect material atau bahan baku tidak langsung adalah bahan baku yang tidak digunakan sebagai bahan utama dalam pembuatan suatu produk dan dianggap sebagai bahan pendukung dalam proses produksi ataupun bahan pelengkap. Sebagai contoh adalah ketika proses produksi sebuah kursi mabel, bahan baku utamanya adalah kayu sebagai rangka kayunya dan busa sebagai tambahan kursinya untuk menambah kenyamanan. Sedangkan bahan baku tidak langsung dari kursi mabel adalah pisau potong yang digunakan untuk memotong bentuk kursi dan jarum jahit untuk menjahit busa dan kain untuk memperindah tampilan kursi mabel tersebut.

Pada PT Showa Indonesia Manufacturing, bahan *indirect material* dibagi menjadi empat kategori yang masing-masing memiliki peranan dalam proses produksi, kategori-kategori tersebut adalah:

1. *Factory Supplier*

Factory Supplier adalah kategori *indirect material* yang berfungsi sebagai pendukung para pekerja dalam menjalankan proses produksi yang berjalan. Kategori ini merupakan kategori *indirect material* yang paling banyak digunakan setiap harinya dan berisi alat keselamatan kerja. Contoh *indirect material* pada kategori ini adalah sarung tangan, lap majun, masker, minyak, air dan lain-lain. Kategori ini merupakan kategori yang penggunaannya paling banyak dalam sebulan.

2. *Consumption Tools*

Consumption Tools adalah kategori pada *indirect material* yang langsung digunakan pada produktivitas mesin. Kategori ini berisi barang-barang yang berhubungan langsung dengan mesin seperti peralatan yang digunakan untuk membuat suatu bentuk produk. Contoh *indirect material* dalam kategori ini adalah armen plate, *slitting cutter*, *collet* dan lain-lain.

3. *Auxiliary Material*

Auxiliary Material adalah kategori yang memiliki *indirect material* yang berfungsi sebagai pemelihara mesin produksi. kategori ini menyimpan bahan-bahan yang berguna untuk memelihara dan merawat mesin untuk memperpanjang umur penggunaan mesin produksi. contoh bahan yang ada pada kategori ini adalah cat anti karat, minyak pelumas pisau mesin, bahan pembersih mesin dan lain-lain.

4. *Machinery Support*

Machinery Support disebut juga sebagai kategori suku cadang mesin. Kategori ini berisi berbagai bahan yang merupakan suku cadang pengganti mesin yang digunakan sewaktu-waktu ketika terjadi kerusakan. Kategori ini termasuk *slow moving* atau jarang digunakan sehingga penyediaan stok pada kategori ini dilakukan dalam jangka waktu yang berbeda dari yang lain seperti tiga bulan sekali atau setahun sekali. Contoh *indirect material* pada kategori ini adalah *timer* pada mesin, sensor, pengatur suhu dan lain-lain.

4.6 **Proses Perencanaan *Indirect Material***

Perencanaan *indirect material* berbeda dengan perencanaan *direct material* yang menggunakan MPP dan *Bill of Material* (BOM) sebagai pedoman dalam perencanaan bahan. Perencanaan *indirect material* tidak menggunakan MPP dan BOM sebagai pedoman perencanaannya melainkan menggunakan data penggunaan bahan *indirect material* yang berdasarkan rata-rata pemakaian selama tiga bulan sebelumnya. Data tersebut kemudian akan diperhitungkan untuk memenuhi kebutuhan bulan selanjutnya dan juga menyediakan stok untuk disimpan ketika pemakaian kebutuhan melebihi perkiraan. Selain itu juga dilihat apakah stok yang disimpan di gudang masih mencukupi atau tidak. Setelah melakukan perhitungan tadi maka bagian PPIC akan mencetak dokumen *purchase requisition* dan mengirimkan dokumen tadi ke *procurement* untuk kemudian membuat dokumen *purchase order plan* untuk disetujui oleh direksi dan merilis *purchase order release*. Setelah itu *procurement* baru bisa melakukan pembelian setelah *purchase order* dikirimkan kepada *supplier*.

4.6.1 **Analisis Dokumen**

Analisis dokumen berikut ini akan menjelaskan dokumen apa saja yang digunakan dalam proses perencanaan dan pengendalian stok *indirect material* di departemen PPIC PT Showa Indonesia Mfg. Dokumen-dokumen tersebut adalah sebagai berikut:

1. Dokumen *Indirect Material* jenis *Consumption Tools*

Indirect material yang digunakan untuk membangun sebuah produk *shock absorber* bermacam-macam jenisnya. Material ini dikategorikan kedalam *consumption tools* yang digunakan untuk mendukung berjalannya proses produksi *shock absorber*. Contoh material yang digunakan dalam proses pendukung produksi adalah yang ada di gambar IV.3 .

No	Part No	Part Name	Specification	UoM	Acc. Group
1	Tw0012418	BELT ABRASIVE RIBENT	Gra. 150 50X3250	pcs	CT
2	Tw0020067	ABRASIVE BELT	60 50 X 915	pcs	CT
3	Tw0009481	JIG SAW BLADE	NO. 1	pcs	CT
4	Tw0013038	TURNING INSERT - KORLOY	TCMT 110204 HMP NC 30	pcs	CT
5	Tw0009299	TURNING INSERT [AKAN NAMBAAH KARNA LINE PAKAI]	TCGT 090204 HFPNC6210	pcs	CT
6	Tw0018608	SCREW	FTKA 0410	pcs	CT
7	Tw0017921	MILLING HOLDER	RM8ACM 4050 HR-H	pcs	CT
8	Tw0020085	THREADING INSERT KORLOY	IR16-1.0 ISO PC 3030T	pcs	CT
9	Tw0019872	COLLET CHUCK PLATING DIA 20 X 80	PR4R-592-01-02-J	pcs	CT
10	Tw0019871	COLLET CHUCK PLATING DIA 20 X 59	PR4R-592-01-01-01-J	pcs	CT

Gambar IV.3 *Indirect Material Consumption Tools*

Sumber: PT Showa Indonesia Manufacturing, 2015

2. Dokumen Kalkulasi Pemesanan *Indirect Material*

Dokumen kalkulasi pemesanan *indirect material* adalah dokumen yang berisi tentang jumlah pemesanan yang akan dilakukan setelah melakukan peramalan pemesanan. Dokumen mencakup jumlah stok yang ada di gudang, jumlah pemakaian perhari, anggaran jumlah pembelian, status barang dan *safety stock*. Jika status barang dinyatakan “*over*” maka tidak perlu melakukan pembelian karena jumlah persediaan sudah berlebihan dan harus segera dihabiskan.

CALCULATION ORDER INDIRECT MATERIAL (CT)					22										
BULAN : NOVEMBER 2014					SEPT 2015										
No	Part No	Part Name	Specification	UoM	Add. Order		Reg Order		Budget	Jul-00	End Stock		Stock Level		
					Ideal	Order	Ideal	Order	Monthly	Use/Day	Qty	Remarks	Min	SS	Max
1	Tw0010194	ARMEN PLATE		pcs	-	-	-	-	22	1	160	OVER	37	35	100
2	Tw0010091	SPLITTING CUTTER	D. 75X4TXID25.4 234	pcs	-	-	-	-	90	4	192	OK	17	12	200
3	Tw0009638	SPLITTING CUTTER	D1.25X3X25.4 256	pcs	-	-	-	-	-	-	-	OK	-	-	-
4	Tw0019719	COLLET	KT12-1275 Ø8.4	pcs	-	-	-	1	-	-	5	OK	-	-	5
5	Tw0019751	ABRASIVE BELT RIBENT	#120. 50 x 3250	pcs	-	200	133	-	200	9	105	OK	38	27	300
6	Tw0012418	BELT ABRASIVE RIBENT	Gra. 150 50X3250	pcs	71	600	4,171	3,500	3,600	164	600	SIAGA3	671	491	5,000
7	Tw0020067	ABRASIVE BELT	60 50 X 915	pcs	-	200	489	400	500	23	205	OK	94	68	1,000
8	Tw0009481	JIG SAW BLADE	NO. 1	pcs	-	-	32	40	38	2	16	OK	8	5	100
9	Tw0013038	TURNING INSERT - KORLOY	TCMT 110204 HMP NC 30	pcs	-	-	-	-	-	-	40	OK	-	-	40
10	Tw0009299	TURNING INSERT [AKAN NAMBAAH KARNA LINE PAKAI]	TCGT 090204 HFPNC6210	pcs	87	400	1,167	700	1,000	45	100	SIAGA2	167	136	1,300

Gambar IV.4 Dokumen Kalkulasi Pemesanan *Indirect Material*

Sumber: PT Showa Indonesia Manufacturing, 2016

Kalkulasi material pada dokumen kalkulasi diatas mempunyai rumus-rumus yang dibuat oleh staf PPIC *Indirect material*. Ada tiga perhitungan yang dibutuhkan untuk membuat sebuah kalkulasi order, yaitu:

a. Order reguler

Order reguler mempunyai 2 kondisi yaitu jika keadaan stok terakhir melebihi stok minimal dan anggaran belanja bulanan maka tidak akan dilakukan order. Namun apabila kurang maka akan dilakukan order dengan rumus:

$$\begin{aligned} \text{Order reguler} &= \text{anggaran belanja} + \text{stok minimal} \\ &- \text{jumlah stok pada bulan sebelumnya} \end{aligned}$$

b. Order tambahan

Order tambahan hanya dilakukan jika stok terakhir kurang dari stok minimal. Order tambahan ini idealnya dihitung dengan rumus:

$$\text{Order tambahan} = \text{stok minimal} - \text{jumlah stok terakhir}$$

Namun pada aktualnya order tambahan dilakukan dengan jumlah minimum pembelian yang sudah ditentukan oleh perusahaan. Jika lebih maka akan dijadikan cadangan untuk bulan selanjutnya.

c. *Safety stock*

Safety stock dihitung dengan rumus sebagai berikut:

$$\text{Safety stock} = \text{Pemakaian selama sehari} \times \text{Lead time}$$

Lead time pada rumus diatas berarti waktu yang dibutuhkan untuk menerima barang yang dipesan yaitu selama tiga hari. Sedangkan pemakain selama sehari bisa dihitung dari anggaran belanja bulanan dibagi dengan jumlah hari kerja selama sebulan.

3. Dokumen *Forecasting Indirect Material*

Dokumen *forecasting indirect material* adalah dokumen yang berisi peramalan untuk pemakaian pada bulan selanjutnya. Peramalan ini dilakukan dengan menjumlahkan order reguler dengan order tambahan yang ada pada dokumen kalkulasi.

Pada dokumen *forecasting indirect material* dibawah, rumus yang digunakan adalah:

$$\text{Forecast order bulan } x = \text{order reguler} + \text{order tambahan}$$

Order tambahan dan order reguler diatas bisa dilihat pada dokumen kalkulasi yang sebelumnya sudah dihitung. Setelah semua material dihitung berapa yang akan dipesan maka akan dibuatkan *purchase requisition*.


FORECAST ORDER MATERIAL INDIRECT MATERIAL																
GROUP : CONSUMPTION TOOLS																
★ Consignment Part			★ Part Import				Forecast Order									
No	Part No	Part Name	Specification	UoM	Acc. Group	Lot	Jan-15	Feb-15	Mar-15	Apr-15	May-15	Jun-15	Jul-15	Aug-15	Sep-15	Oct-15
1	TW0010194	ARMEN PLATE		pcs	FS	10	-	-	-	-	-	-	-	-	-	-
949	TW0020497	BURNISHING DRILL	DIA 6.5 X DIA 8.5 X 100 L	pcs	CT			2	3	3						
950	TW0020498	STRAIGHT SHANK DRILL [SUIT	Die 8.8 mm [BLUE]	pcs	CT		200	180		450	200	350	1	300	380	
951	TW0020523	ANODA BAG [FINISHING WITH W	100 X 1000	pcs	CT				180	120	60					
952	TW0020565	ABRASIVE BELT	GRADE 320, 50 X 850 MM	pcs	CT				2,000							100
953	TW0020566	NEW ANODA PLATING 4W	PR4W-100-00-01-F	pcs	CT					216						
954	TW0020567	INSERT ROUGHING TURNING	WNMG 080408-MA UE6105	pcs	CT					110	50		50	70	80	
955	TW0020606	CARTRIDGE	STTCR - 08CA-09 MERK SAJ	pcs	CT				15	20	10					

Gambar IV.5 Dokumen Forecasting Indirect Material

Sumber: PT Showa Indonesia Manufacturing, 2016

4. Purchase Requisition

Dokumen ini berisi jenis part, deskripsi, spesifikasi, jumlah pemesanan dan tanggal tiba pemesanan barang. Dokumen ini dibuat setiap bulannya untuk mencatat barang apa saja yang akan dipesan untuk menjaga stok. Dokumen ini dicetak melalui form *purchase requisition* untuk diteruskan ke *procurement*.

PURCHASE REQUISITION							
Department : PPC Internal Dest : TWH Cost Center : Code Invest : Budget No :				Reg. Num : 177100 Date : 03 -pr 2016 Ordered For : SUP:P1IO T H			
No	Part No	Purchase Group	Description	Specification	Quantity	UoM	Schedule
1	TW0000418	ITR2	BARIUM CARBONAT (BACO3)	@ 25 KG	4	SACK	20 Apr 16
2	TW0000421	ITR5	ALKALINE BATTERY	SIZE AA 1.5V	20	pcs	15 Apr 16
3	TW0001230	ITR5	PAINT	1000-0102, MEDIU	20	CAN	18 Apr 16
4	TW0001264	ITR2	CLEANING CHEMICAL	ACTREL 3363L @ 2	4	DRUM	20 Apr 16
5	TW0002143	ITR2	FIXODINE FM	@ 25 KG	4	PAIL	20 Apr 16
6	TW0002474	ITR6	OXYGEN GASES	@ 8.5 KG	6	BTL	11 Apr 16
7	TW0002507	ITR4	COTTON GLOVE	NO.1 - 8 YARN	1,000	LSN	18 Apr 16
8	TW0002521	ITR4	WELDING GLOVE		100	PSG	20 Apr 16
9	TW0002593	ITR5	CHEMICAL AUTOMOTIVE GREASE	100A - YELLOW @	2	PAIL	18 Apr 16
10	TW0002659	ITR2	H N O 3	@ 35 KG	3	PAIL	20 Apr 16
11	TW0002778	ITR3	HEXAGON SOCKET HEAD CAP SC	M 5X15	200	pcs	18 Apr 16
12	TW0002796	ITR5	HEXAGON SOCKET HEAD CAP SC	M 6X15	200	pcs	18 Apr 16
13	TW0004876	ITR5	NUT	M 6	300	pcs	18 Apr 16
14	TW0005406	ITR2	P.A. PHOSPORIC ACID 85%	NO.ART.1.00573.250	1	BTL	18 Apr 16
15	TW0007026	ITR5	SCOTCH BRITE	7447	200	pcs	18 Apr 16
16	TW0007187	ITR3	SEAL	DHS 30	2	pcs	18 Apr 16
17	TW0008293	ITR5	PAINT MARKER	PURPLE	100	pcs	18 Apr 16
18	TW0008297	ITR5	PAINT MARKER	BLUE	100	pcs	18 Apr 16
19	TW0008298	ITR2	PAINT MARKER	SILVER	50	pcs	18 Apr 16
20	TW0008299	ITR5	PAINT MARKER	RED	150	pcs	18 Apr 16
21	TW0008301	ITR5	PAINT MARKER	WHITE	300	pcs	18 Apr 16
22	TW0008302	ITR5	PAINT MARKER	PINK	30	pcs	18 Apr 16
23	TW0008303	ITR5	PAINT MARKER	GREEN	200	pcs	18 Apr 16
24	TW0008304	ITR5	PAINT MARKER	YELLOW	50	pcs	18 Apr 16
25	TW0009175	ITR2	WASHING THINNER	CARGLOSS SI @ 20	10	CAN	20 Apr 16
26	TW0009255	ITR2	ZINC ANODE	SHG 999995%	200	kg	18 Apr 16
27	TW0009329	ITR2	COVERFLUX 100 BH	@ 25 KG	8	SACK	18 Apr 16
28	TW0009331	ITR2	DIE CASTING LUBRICANT	LUBEX 30SN @ 19	2	PAIL	20 Apr 16
29	TW0009340	ITR2	MACH-3 CONVENTION LIQUID	@ 20 LTR	5	CAN	18 Apr 16
30	TW0009342	ITR2	NICKEL SULPHATE (NISO4)	@ 20kg	50	SACK	18 Apr 16
					TOTAL QTY	3,321	
Note : ADDITIONAL ORDER CATEG SUPPLIER LOKAL APRIL 2016							
USER				FINANCE		DIRECTOR	
PREPARED	CHECK	APPROVAL	CHECK	APPROVAL	APPROVAL	RECEIVED	RELEASED
 Print Date : 04 Apr 2016 4:58:16 PM							

Gambar IV.6 Contoh Dokumen Purchase Requisition

Sumber: PT Showa Indonesia Manufacturing, 2016

5. Purchase Order

Purchase order adalah form yang diisi oleh staff *procurement* untuk menampilkan material apa saja yang akan dipesan. *Purchase order* ada dua macam yaitu *purchase order plan* dan *purchase order release*. *Purchase order plan* adalah *purchase order* yang sudah direncanakan berdasarkan *purchase requisition* PPIC sedangkan

purchase order release adalah *purchase order* yang sudah disetujui oleh direksi untuk dilakukan pemesanan dan siap untuk diteruskan ke *supplier* untuk memesan material yang dibutuhkan.

Overview - Purchase Order Lines Indirect Material							
Bulan : APRIL 2012 (All Category)							
No	Part No	Qty	Order No	Wanted Receipt Date	Status	Month	Year
1	TW0020795	1	N197115	29-09-15		Sep-15	2015
2	TW0011026	2	N197116	10-09-15		Sep-15	2015
3	TW0020699	4	N197118	10-09-15		Sep-15	2015
4	TW0000175	1	N197120	10-09-15		Sep-15	2015
5	TW0000202	1	N197120	10-09-15		Sep-15	2015
6	TW0008276	4	N197120	10-09-15		Sep-15	2015
7	TW0020771	1	N197120	02-10-15		Oct-15	2015
8	TW0020773	1	N197120	10-09-15		Sep-15	2015
9	TW0018639	12	N197122	10-09-15		Sep-15	2015
10	TW0018742	4	N197122	10-09-15		Sep-15	2015
11	TW0018946	4	N197122	10-09-15		Sep-15	2015
12	TW0020807	12	N197122	10-09-15		Sep-15	2015
13	TW0020808	24	N197122	10-09-15		Sep-15	2015
14	TW0020641	10	N197150	14-09-15		Sep-15	2015
15	TW0009255	600	N197225	04-09-15		Sep-15	2015

Gambar IV.7 *Purchase Order Plan*
 Sumber: PT Showa Indonesia Manufacturing , 2016

Register Purchase Order Arrival Indirect Material							
Bulan : APRIL 2012 (All Category)							
No	Part No	Qty	Order No	Wanted Receipt Date	Status	Month	Year
1	TW0019870	20	N196823	01-09-15 00:00		Sep-15	2015
2	TW0013655	1250	N196684	01-09-15 00:00		Sep-15	2015
3	TW0002517	400	N196687	01-09-15 00:00		Sep-15	2015
4	TW0020100	14	N196828	01-09-15 00:00		Sep-15	2015
5	TW0005014	1000	N196831	01-09-15 00:00		Sep-15	2015
6	TW0013492	2700	N196832	01-09-15 00:00		Sep-15	2015
7	TW0009357	200	N196835	01-09-15 00:00		Sep-15	2015
8	TW0009357	300	N196835	01-09-15 00:00		Sep-15	2015
9	TW0019859	2	N196836	01-09-15 00:00		Sep-15	2015
10	TW0019861	20	N196836	01-09-15 00:00		Sep-15	2015
11	TW0009530	500	N196838	01-09-15 00:00		Sep-15	2015
12	TW0009354	500	N196838	01-09-15 00:00		Sep-15	2015
13	TW0010005	150	N196842	01-09-15 00:00		Sep-15	2015
14	TW0009998	520	N196842	01-09-15 00:00		Sep-15	2015
15	TW0009527	10	N196846	01-09-15 00:00		Sep-15	2015

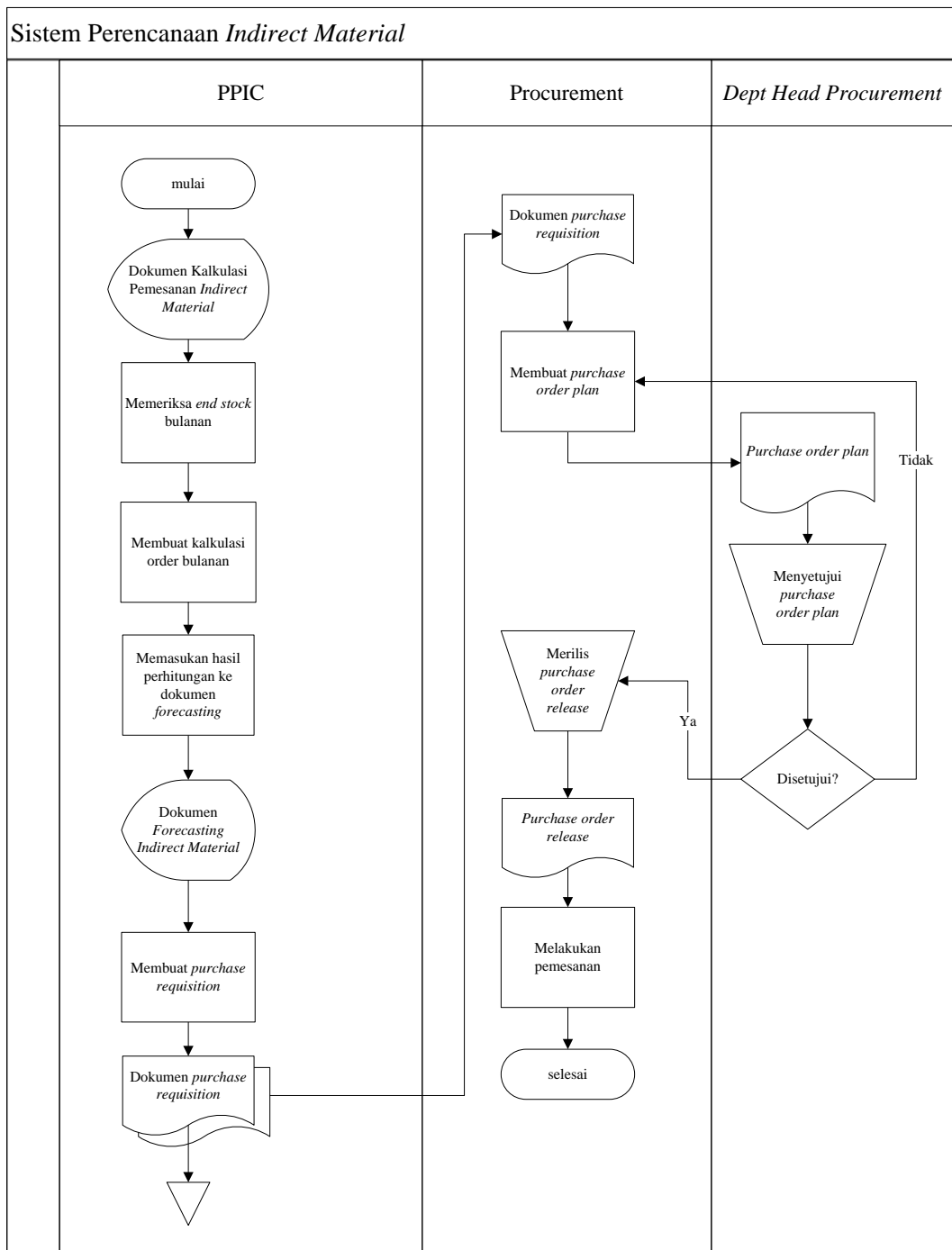
Gambar IV.8 *Purchase Order Release*
 Sumber: PT Showa Indonesia Manufacturing , 2016

4.7 Analisis Sistem yang Sedang Berjalan

Bagan alir (*Flowchart*) menunjukkan arus dari pekerjaan secara keseluruhan dari sistem termasuk dokumen-dokumennya. Bagan alir ini digunakan sebagai alat bantu komunikasi dan untuk dokumentasi perusahaan. *Flowchart* ini menjelaskan urutan dari prosedur yang ada di dalam sistem. Bagan alir juga menunjukkan apa saja yang dikerjakan di sistem yang berjalan.

4.7.1 *Flowchart* yang Sedang Berjalan di Perencanaan *Indirect Material*

1. Staf PPIC akan memeriksa keadaan end stock pada saat akhir bulan melalui basis data persediaan.
2. Kemudian staf PPIC akan membuat perhitungan kalkulasi pemesanan. Kalkulasi ini berdasarkan jumlah pemakaian selama tiga bulan sebelumnya dan anggaran belanja bulanan yang diberikan kepada seksi PPIC sebagai referensi. Setelah menghitung order reguler dan order tambahan serta *safety stock* maka kalkulasi selesai.
3. Staf yang sudah membuat perencanaan pemesanan kemudian akan membuat kalkulasi order untuk dipesan pada bulan depan.
4. Berdasarkan dokumen *forecasting* maka staf PPIC akan membuat *purchase requisition* kemudian akan dicetak menjadi dokumen dan diteruskan kepada *procurement*.
5. Staf *procurement* akan membuat *purchase order plan* berdasarkan *purchase requisition* yang diberikan oleh staf PPIC, *purchase order plan* akan dicetak kemudian akan diserahkan ke *Dept Head* untuk disetujui.
6. *Purchase order plan* yang sudah disetujui akan diubah statusnya menjadi *purchase order release* oleh staf *procurement*.
7. Staf *procurement* akan melakukan pemesanan material kepada supplier.



Gambar IV.9 *Flowmap* Pemesanan yang Sedang Berjalan di PPIC PT Showa Indonesia

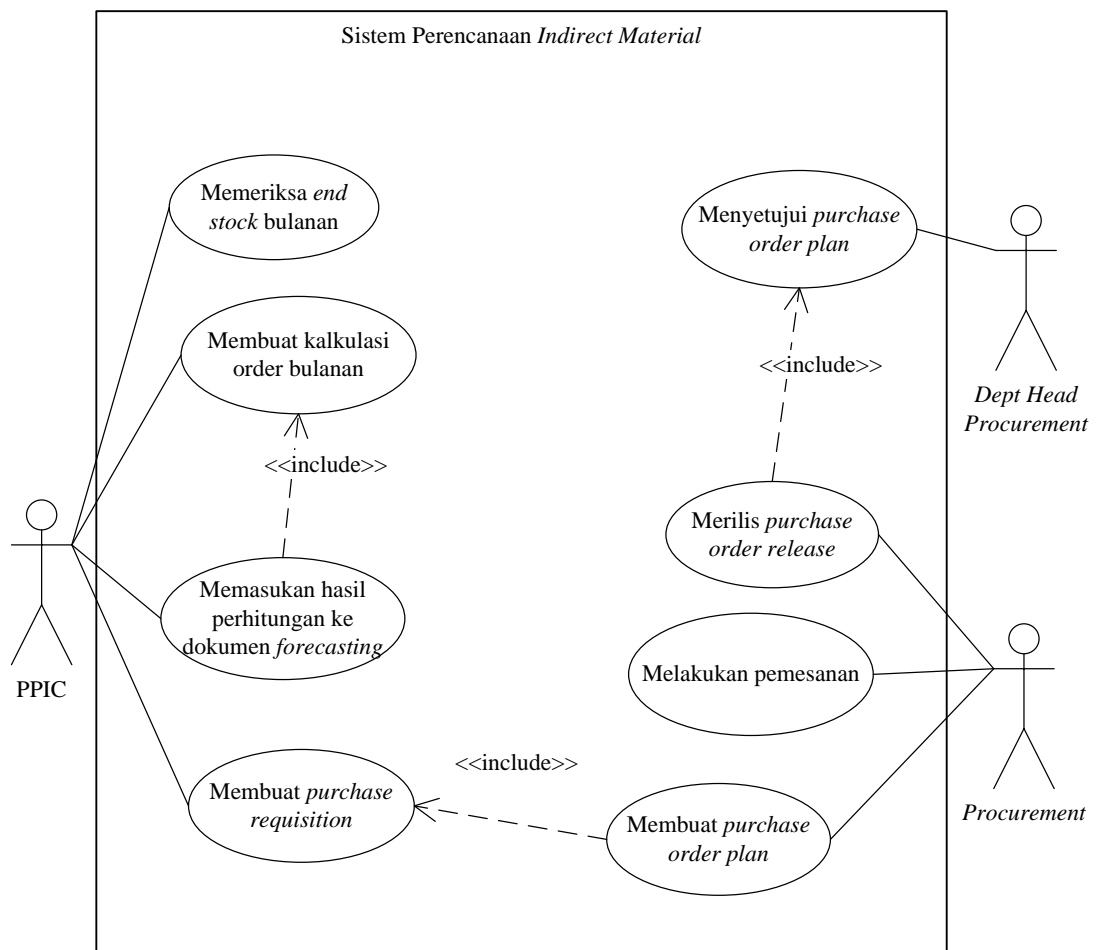
Mfg

Sumber: Pengolahan data, 2016

4.8 Sistem Berjalan dengan Use Case Diagram

Use case adalah konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata pengguna potensial. *Use case* terdiri dari sekumpulan skenario yang dilakukan oleh aktor. Sedangkan *use case diagram* memfasilitasi komunikasi antara analis dan pengguna serta diantara analis dan klien (Munawar, 2005).

Berikut ini adalah *use case diagram* pada proses perencanaan dan pengendalian stok *indirect material* pada departemen PPIC PT Showa Indonesia Mfg yang sedang berjalan:



Gambar IV.10 Use Case yang Sedang Berjalan di PPIC PT Showa Indonesia Mfg

Sumber: Pengolahan data, 2016

Penjelasan *use case diagram* proses perencanaan dan pengendalian stok *indirect material* pada departemen PPIC PT Showa Indonesia Mfg di atas dapat dijelaskan sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* proses perencanaan dan pengendalian stok *indirect material* pada departemen PPIC PT Showa Indonesia Mfg adalah:

Tabel IV.1 Definisi Aktor *Use Case Diagram*

No.	Aktor	Deskripsi
1.	PPIC	Melakukan pembuatan dokumen kalkulasi bulanan, memeriksa <i>end stock</i> , membuat dokumen <i>forecasting</i> , membuat <i>purchase requisition</i> .
2.	<i>Procurement</i>	Melakukan pembuatan dokumen <i>purchase order plan</i> , <i>purchase order release</i> , dan melakukan pemesanan.
3	<i>Dept Head Procurement</i>	Melakukan persetujuan untuk merilis <i>purchase order release</i> .

Sumber: Hasil pengolahan data, 2016

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* proses perencanaan dan pengendalian stok *indirect material* pada departemen PPIC PT Showa Indonesia Mfg adalah:

Tabel IV.2 Definisi *Use Case Diagram*

No.	<i>Use Case</i>	Deskripsi
1.	Memeriksa <i>end stock</i> bulanan	Merupakan proses memeriksa keadaan stok terakhir yang ada di gudang dari dokumen kalkulasi bulan sebelumnya.
2.	Membuat kalkulasi order bulanan	Membuat kalkulasi perhitungan pemesanan <i>indirect material</i> dengan pedoman anggaran belanja bulanan dan <i>end stock</i> terakhir.
3.	Memasukkan hasil perhitungan ke dokumen <i>forecasting</i>	Setelah membuat kalkulasi order dibuat maka hasil perhitungan akan dimasukkan ke dokumen <i>forecasting order</i> yang dibuat selama sebulan.
4.	Membuat <i>purchase requisition</i>	Mengisi form yang berisi part yang akan dipesan, spesifikasi dan jumlahnya serta tanggal penerimaannya berdasarkan dokumen <i>forecasting</i> .
5.	Membuat <i>purchase order plan</i>	Membuat kelanjutan dari <i>purchase requisition</i> yang dibuat oleh <i>procurement</i> . Isinya adalah jenis material, jumlah pesanan, tanggal tiba barang dan nama pemasoknya.
6.	Melakukan persetujuan <i>purchase order plan</i>	Proses penyetujuan <i>purchase order plan</i> yang sudah disetujui oleh <i>dept head procurement</i> mengenai jumlah pemesanan dan

		sesuai dengan anggaran yang diberikan.
7	Merilis <i>purchase order release</i>	Menerbitkan <i>purchase order</i> yang siap diberikan kepada
8	Melakukan pemesanan	Melakukan pemesanan kepada supplier berdasarkan <i>purchase order release</i> .

Sumber : Hasil pengolahan data, 2016

BAB V

ANALISIS DAN PEMBAHASAN

5.1. Analisis Kebutuhan Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, dibutuhkan perancangan aplikasi sistem informasi perencanaan *indirect material* sebagai sarana untuk memberikan informasi mengenai pelaporan pemesanan *indirect material* dari kegiatan perencanaan Departemen PPIC. Berikut adalah daftar kebutuhan sistem untuk aplikasi sistem informasi perencanaan *indirect material* pada Departemen PPIC.

Tabel V.1 Kebutuhan Sistem Aplikasi Perencanaan *Indirect Material*

Kebutuhan Sistem	
<i>Project Name</i>	Aplikasi Perencanaan <i>Indirect Material</i> .
<i>Project Sponsor</i>	Departemen PPIC
<i>Business Need</i>	<ol style="list-style-type: none"> 1. Terwujudnya sebuah aplikasi berbasis data yang dapat membantu dalam perhitungan perencanaan pemesanan <i>indirect material</i>. 2. Terwujudnya sistem informasi yang dapat membantu dalam memvalidasi persetujuan <i>Purchase Order</i>.
<i>Business Requirement</i>	Memberikan sistem informasi pemesanan <i>indirect material</i> yang dapat membantu Departemen PPIC dalam membuat laporan.
<i>Bussiness Value</i>	<ol style="list-style-type: none"> 1. Memudahkan dalam pembuatan laporan pemesanan

	<p><i>indirect material</i> sehingga laporan menjadi lebih informatif dan mudah dimengerti.</p> <p>2. Mampu memvalidasi <i>Purchase Order</i> secara terkomputerisasi.</p>
<i>Special Issues or Constrains</i>	-

Sumber: Hasil Analisa Data (2015)

5.2. Prosedur Sistem Informasi Perencanaan *Indirect Material* Usulan

Untuk mengatasi permasalahan dalam perencanaan *indirect material* yang ada pada Departemen PPIC diajukan usulan sistem baru dengan menerapkan penggunaan aplikasi komputer untuk pengolahan data yang akan mendukung beberapa proses bisnis yang sedang berjalan pada perusahaan. Dengan menggunakan aplikasi ini membuat sistem menjadi terintegrasi sehingga dapat menyajikan informasi secara cepat dan dapat meningkatkan fungsionalitas perencanaan *indirect material* yang ada pada Departemen PPIC di PT Showa Indonesia Mfg.

5.2.1. Prosedur Sistem Informasi Perencanaan *Indirect Material* Usulan

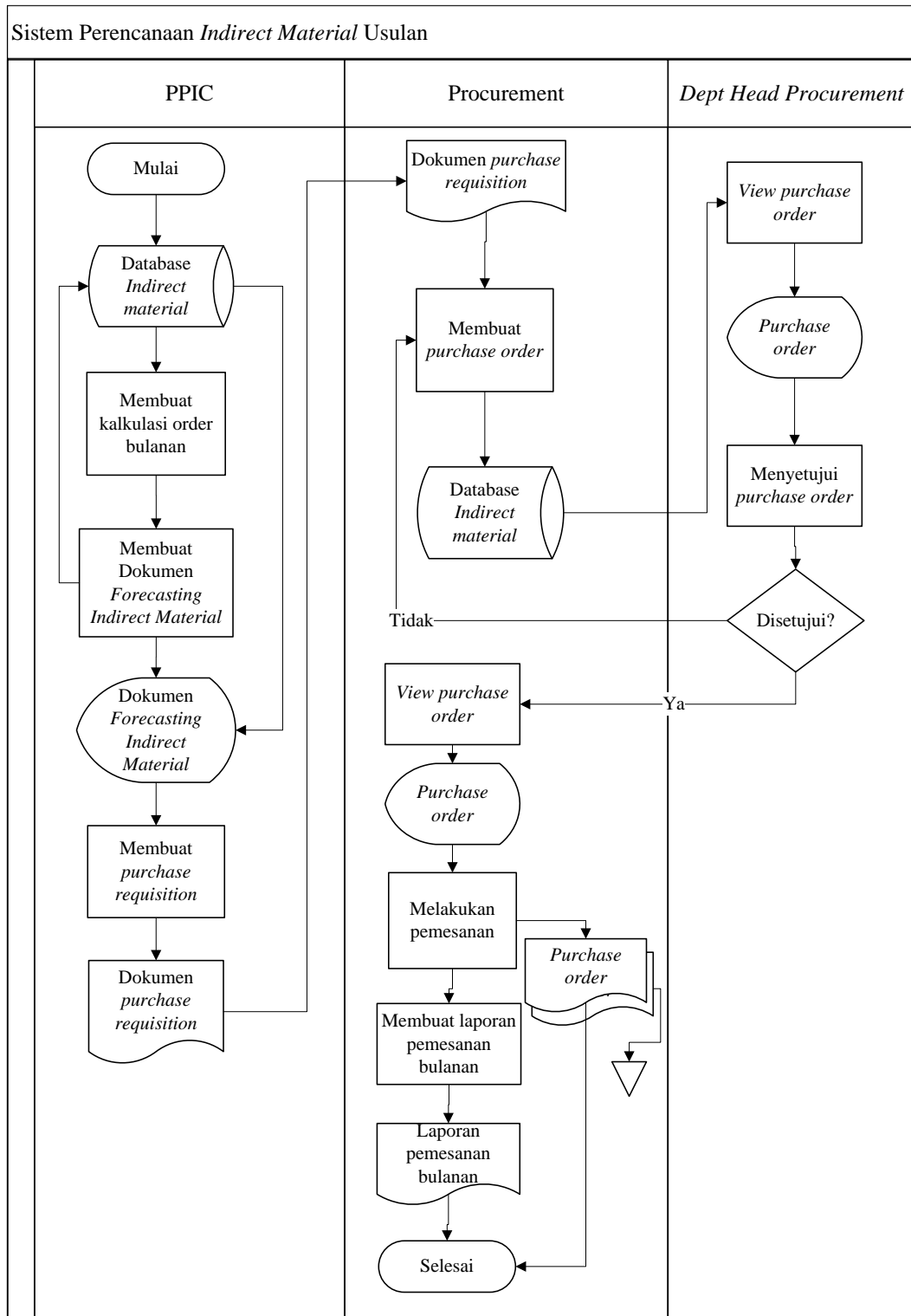
Prosedur sistem informasi perencanaan *indirect material* yang diusulkan melalui tahapan sebagai berikut:

1. Berdasarkan keadaan *end stock* bulan sebelumnya yang tersimpan di database *indirect material*, staf PPIC akan membuat perhitungan kalkulasi order bulanan.
2. Setelah melakukan kalkulasi order bulanan, maka staf PPIC akan memasukan hasil perhitungan ke *forecasting* bulanan yang berisi order reguler dan order tambahan selama sebulan.
3. *Forecasting* yang sudah dibuat maka akan dibuatkan *purchase requisition* yang berisi barang-barang apa saja yang akan dipesan selama sebulan, spesifikasinya, dan jumlah.
4. *Purchase requisition* yang sudah dibuat kemudian akan diserahkan kepada departemen *Procurement*. *Procurement* akan membuat *Purchase Order*

yang mengacu kepada *purchase requisition* dan kemudian akan dicetak dan diserahkan kepada *Dept Head Procurement*.

5. *Dept Head Procurement* akan menerima *Purchase Order* untuk kemudian disetujui atau tidak untuk melakukan pemesanan. Jika disetujui maka *Purchase Order* akan dikembalikan kepada *procurement* dan dilakukan pemesanan. Sebaliknya, jika tidak disetujui maka *Purchase Order* akan dibuat ulang. Setiap bulan akan dibuat laporan pemesanan bulanan berdasarkan *purchase order*.

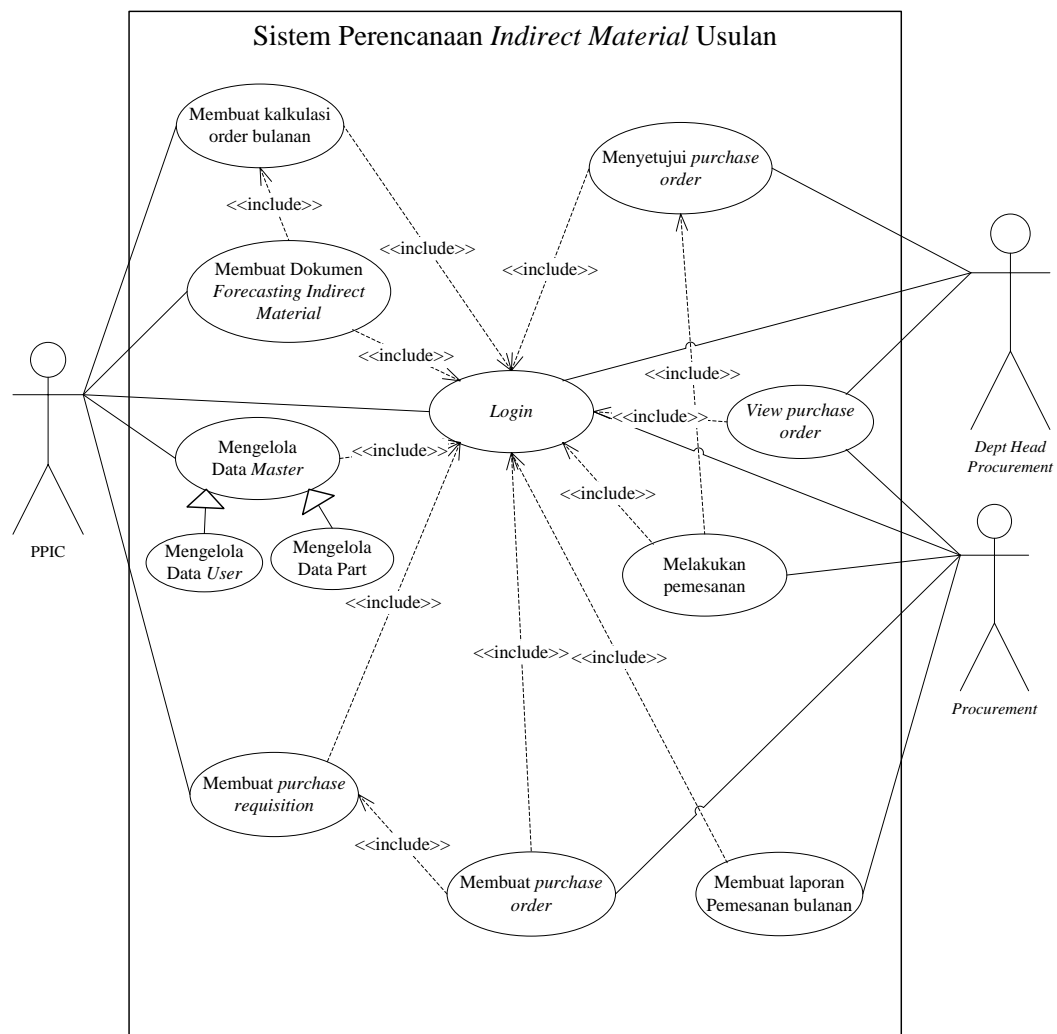
Seperti yang dapat dilihat pada Gambar V.1 adalah *Flowmap* Sistem Informasi Perencanaan *Indirect Material* Usulan sebagai berikut:



Gambar V.1 *Flowmap* Sistem Informasi Perencanaan *Indirect Material* Usulan
 Sumber: Hasil Analisa Data (2016)

5.2.2 Use Case Diagram Usulan

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



Gambar V.2 *Use Case Diagram* Sistem Informasi Perencanaan *Indirect Material* Usulan
 Sumber: Hasil Analisa Data (2016)

3. Definisi Aktor

Pendefinisian aktor pada *use case diagram* proses perencanaan dan pengendalian stok *indirect material* pada departemen PPIC PT Showa Indonesia Mfg adalah:

Tabel V.2 Definisi Aktor *Use Case Diagram*

No.	Aktor	Deskripsi
1.	PPIC	Melakukan pembuatan dokumen kalkulasi bulanan, membuat dokumen <i>forecasting</i> , membuat <i>purchase requisition</i> .
2.	<i>Procurement</i>	Melakukan pembuatan dokumen <i>Purchase Order</i> , dan melakukan pemesanan.
3	<i>Dept Head Procurement</i>	Melakukan persetujuan untuk menyetujui <i>Purchase Order</i> .

Sumber: Hasil Analisa Data, 2016

4. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* proses perencanaan dan pengendalian stok *indirect material* pada departemen PPIC PT Showa Indonesia Mfg adalah:

Tabel V.3 Definisi *Use Case Diagram*

No.	<i>Use Case</i>	Deskripsi
1	Membuat Kalkulasi Order Bulanan	Membuat kalkulasi perhitungan pemesanan <i>indirect material</i> dengan pedoman anggaran belanja bulanan dan <i>end stock</i> terakhir.
2.	Membuat Dokumen <i>Forecasting Indirect Material</i>	Setelah membuat kalkulasi order dibuat maka hasil perhitungan akan dimasukkan ke dokumen <i>forecasting indirect material</i> yang dibuat selama sebulan.
3.	Membuat <i>Purchase Requisition</i>	Mengisi form yang berisi part yang akan dipesan, spesifikasi dan jumlahnya serta tanggal penerimaannya berdasarkan dokumen <i>forecasting</i> .
4.	Membuat <i>Purchase Order</i>	Membuat kelanjutan dari <i>purchase requisition</i> yang dibuat oleh procurement. Isinya adalah jenis material, jumlah pesanan, tanggal tiba barang dan nama pemasoknya.
5.	<i>View Purchase</i>	Proses untuk menampilkan <i>Purchase Order</i> yang

	<i>Order</i>	sudah dibuat. Menampilkan daftar part yang akan dipesan, tanggal pemesanan dan jumlah yang akan dipesan.
5.	Melakukan Persetujuan <i>Purchase Order</i>	Proses penyetujuan <i>Purchase Order</i> yang sudah disetujui oleh <i>Dept Head Procurement</i> mengenai jumlah pemesanan dan sesuai dengan anggaran yang diberikan.
6.	Melakukan Pemesanan	Melakukan pemesanan kepada <i>supplier</i> berdasarkan <i>Purchase Order</i> yang sudah disetujui.
7.	Membuat Laporan Pemesanan Bulanan	Proses pembuatan laporan pemesanan bulanan yang akan dijadikan data pemesanan setiap bulannya.
8.	<i>Login</i>	Proses untuk melakukan <i>login</i> pada aplikasi
9.	Mengelola Data Master	Proses untuk melakukan pengolahan data master yang terdiri dari data master seperti data master user dan <i>indirect material</i> .

Sumber: Hasil Analisa Data, 2016

3. Skenario *Use Case Diagram*

Skenario jalannya masing-masing *use case* pada sistem informasi perencanaan *indirect material* usulan dapat dilihat pada poin berikut:

a. *Use Case Log in*

Berikut adalah skenario *Use Case login* yang terdapat pada Tabel V.4:

Tabel V.4. *Use Case Description Login*

Nama <i>Use Case</i>	<i>Login</i>
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan <i>User</i> masuk ke dalam sistem.
Aktor	<i>User</i> (PPIC, <i>Procurement</i> , <i>Dept Head Procurement</i>)
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User</i> membuka <i>Login Form</i>. 2. <i>User</i> memasukkan NPK dan <i>password</i> pada <i>Login Form</i>. 3. Sistem mengecek ke <i>database</i>, apakah NPK dan

	<p><i>password</i> benar.</p> <p>4. Jika NPK dan <i>password</i> benar, maka muncul tampilan menu utama.</p> <p>5. Jika NPK dan <i>password</i> salah (tidak valid), maka muncul <i>message</i> “<i>Wrong NPK or password</i>” pada <i>Login Form</i></p>
--	---

Sumber: Hasil Analisa Data, 2016

b. *Use Case* Mengelola Data Master

Berikut adalah skenario *Use Case* mengelola data master yang terdapat pada tabel V.5.

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

Nama <i>Use Case</i>	Mengelola Data Master
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses pengelolaan data master yang terdiri dari data <i>user</i> dan data <i>part</i> .
Aktor	<i>User</i> (PPIC)
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu file data master. 3. <i>User</i> memilih sub menu yang terdiri dari data <i>user</i> dan <i>part</i>. 4. <i>User</i> melakukan proses cari, tambah, ubah, dan hapus data di basis data masing masing pada data master.

Sumber: Hasil Analisa Data, 2016

c. *Use Case* Membuat Kalkulasi Order Bulanan

Berikut adalah skenario *Use Case* membuat kalkulasi order bulanan yang terdapat pada Tabel V.6:

Tabel V.6. *Use Case Description* Membuat Kalkulasi Order Bulanan

Nama <i>Use Case</i>	Membuat Kalkulasi Order Bulanan
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses perhitungan kalkulasi pemesanan <i>indirect material</i> .
Aktor	<i>User</i> (PPIC)
<i>Normal Flow</i>	1. <i>User Login</i> dan masuk ke tampilan menu utama.

<i>Events:</i>	<ol style="list-style-type: none"> 2. <i>User</i> memilih menu kalkulasi order. 3. Sistem menampilkan menu kalkulasi. 4. <i>User</i> memasukan kode kalkulasi bulanan. 5. <i>User</i> memasukan kode part, anggaran belanja bulanan, jumlah hari kerja, dan order tambahan. 6. <i>User</i> membuat perhitungan kalkulasi selama sebulan. 7. <i>User</i> memilih simpan untuk menyimpan data kalkulasi <i>indirect material</i>.
----------------	---

Sumber: Hasil Analisa Data, 2016

d. *Use Case* Membuat Dokumen *Forecasting Indirect Material*

Berikut adalah skenario *Use Case* memasukkan hasil perhitungan ke dokumen *forecasting* yang terdapat pada Tabel V.7:

Tabel V.7. *Use Case Description* Membuat Dokumen *Forecasting Indirect Material*

Nama <i>Use Case</i>	Membuat Dokumen <i>Forecasting Indirect Material</i>
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses pembuatan dokumen <i>forecasting</i> berdasarkan dokumen kalkulasi yang sudah dibuat.
Aktor	<i>User</i> (PPIC)
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu <i>Forecasting</i>. 3. Sistem menampilkan menu <i>Forecasting</i>. 4. <i>User</i> memasukan kode <i>Forecasting</i> dan bulan pemesanan. 5. <i>User</i> memasukan kode kalkulasi bulanan kemudian memilih kode part yang akan dipesan. 6. <i>User</i> memasukan jumlah order sesuai dengan kebutuhan. 7. <i>User</i> menyimpan dokumen <i>forecasting</i> untuk menyimpan data <i>forecasting</i> bulanan.

Sumber: Hasil Analisa Data, 2016

e. *Use Case Membuat Purchase Requisition*

Berikut adalah skenario *Use Case* membuat *purchase requisition* yang terdapat pada Tabel V.8:

Tabel V.8. *Use Case Description* Membuat *Purchase Requisition*

Nama <i>Use Case</i>	Membuat <i>Purchase Requisition</i>
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses pembuatan dokumen <i>Purchase Requisition</i> .
Aktor	User (PPIC)
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu <i>Purchase Requisition</i>. 3. Sistem menampilkan menu <i>Purchase Requisition</i>. 4. <i>User</i> memasukkan kode <i>Purchase Requisition</i>. 5. <i>User</i> memilih kode part yang akan dipesan. 6. <i>User</i> memasukkan jumlah order sesuai dengan kebutuhan. 7. <i>User</i> menyimpan dokumen <i>Purchase Requisition</i> untuk menyimpan data kemudian dokumen <i>Purchase Requisition</i> dikirimkan ke <i>Procurement</i>.

Sumber: Hasil Analisa Data, 2016

f. *Use Case Membuat Purchase Order*

Berikut adalah skenario *Use Case* membuat *Purchase Order* yang terdapat pada Tabel V.9:

Tabel V.9. *Use Case Description* Membuat *Purchase Order*

Nama <i>Use Case</i>	Membuat <i>Purchase Order</i>
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses pembuatan dokumen <i>Purchase Requisition</i> .
Aktor	User (<i>Procurement</i>)
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu . 3. Sistem menampilkan menu <i>Purchase Order</i>. 4. <i>User</i> memasukkan kode <i>Purchase Order</i>.

	<p>5. <i>User</i> memasukkan kode <i>Purchase Requisition</i> yang sesuai dengan <i>Purchase Requisition</i> yang sudah diterima.</p> <p>6. <i>User</i> memasukkan jumlah order sesuai dengan kebutuhan.</p> <p>7. <i>User</i> menyimpan dokumen <i>Purchase Order</i> kemudian dikirimkan dan diberikan kepada <i>Dept Head</i> untuk divalidasi.</p>
--	--

Sumber: Hasil Analisa Data, 2016

g. *Use Case View Purchase Order*

Berikut adalah skenario *Use case view Purchase Order* yang terdapat pada tabel V.10:

Tabel V.10. *Use Case Description View Purchase Order*

Nama <i>Use Case</i>	<i>View Purchase Order</i>
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses validasi <i>Purchase Order</i> apakah sesuai dengan anggaran belanja yang sudah diberikan.
Aktor	<i>User (Procurement, Dept Head Procurement)</i>
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu <i>Purchase Order</i>. 3. Sistem menampilkan menu <i>Purchase Order</i>. 4. <i>User</i> memasukkan kode <i>Purchase Order</i>. 5. Sistem menampilkan dokumen <i>Purchase Order</i>.

h. *Use Case Melakukan Persetujuan Purchase Order*

Berikut adalah skenario *Use Case* melakukan persetujuan *Purchase Order* yang terdapat pada Tabel V.11:

Tabel V.11. *Use Case Description Melakukan Persetujuan Purchase Order*

Nama <i>Use Case</i>	Melakukan Persetujuan <i>Purchase Order</i>
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses validasi <i>Purchase Order</i> apakah sesuai dengan anggaran belanja yang sudah diberikan.

Aktor	<i>Dept Head Procurement</i>
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu validasi. 3. Sistem menampilkan menu validasi. 4. <i>User</i> memasukan kode <i>Purchase Order</i>. 5. Sistem menampilkan dokumen <i>Purchase Order</i>. 6. <i>User</i> memeriksa jumlah pemesanan part berdasarkan anggaran belanja yang sudah diberikan. 7. Jika sesuai maka <i>user</i> akan memberikan tanda setuju untuk dilakukan pemesanan. Jika tidak maka dokumen akan dikembalikan ke <i>procurement</i> untuk dilakukan perubahan. 8. Dokumen yang sudah disetujui dikirimkan kembali kepada <i>Procurement</i>.

Sumber: Hasil Analisa Data, 2016

i. *Use Case* Melakukan Pemesanan

Berikut adalah skenario *Use Case* melakukan pemesanan yang terdapat pada Tabel V.12:

Tabel V.12. *Use Case Description* Melakukan Pemesanan

Nama <i>Use Case</i>	Melakukan Pemesanan
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses pemesanan yang dilakukan berdasarkan dokumen <i>Purchase Order</i> yang sudah divalidasi
Aktor	<i>Procurement</i>
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu <i>Purchase Order</i>. 3. Sistem menampilkan menu <i>Purchase Order</i>. 4. <i>User</i> memasukan kode <i>Purchase Order</i>. 5. Sistem akan menampilkan <i>Purchase Order</i> yang sudah divalidasi oleh <i>Dept Head</i>. 6. <i>User</i> mencetak dokumen <i>Purchase Order</i>.

	7. <i>User</i> melakukan pemesanan.
--	-------------------------------------

Sumber: Hasil Analisa Data, 2016

j. *Use Case* Membuat Laporan Pemesanan Bulanan.

Berikut adalah skenario *Use Case* membuat laporan pemesanan bulanan yang terdapat pada Tabel V.13:

Tabel V.13. *Use Case Description* Membuat Laporan Pemesanan Bulanan

Nama <i>Use Case</i>	Membuat Laporan Pemesanan Bulanan
Deskripsi <i>Use Case</i>	<i>Use Case</i> ini menggambarkan proses laporan pemesanan bulanan berdasarkan dokumen <i>Purchase Order</i> .
Aktor	<i>Procurement</i>
<i>Normal Flow</i> <i>Events:</i>	<ol style="list-style-type: none"> 1. <i>User Login</i> dan masuk ke tampilan menu utama. 2. <i>User</i> memilih menu laporan. 3. Sistem menampilkan menu laporan. 4. <i>User</i> memasukkan bulan pada laporan yang akan dicetak. 5. <i>User</i> membuat laporan pemesanan bulanan berdasarkan <i>Purchase Order</i> dan bulan yang dipilih kemudian akan disimpan dan dicetak.

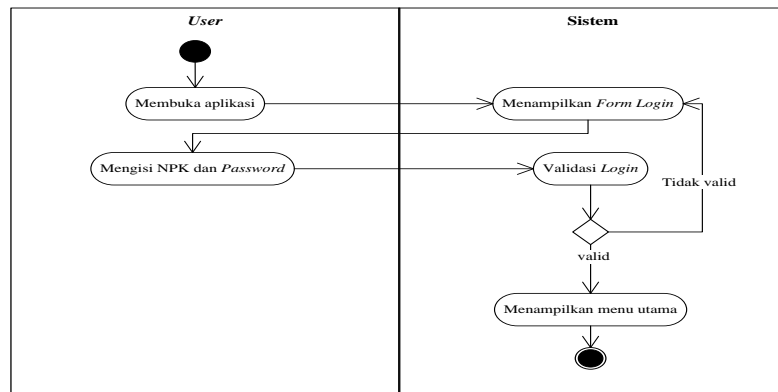
Sumber: Hasil Analisa Data, 2016

5.2.3. *Activity Diagram* Usulan

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar lebih memahami sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem informasi perencanaan *indirect material*.

1. *Activity Diagram Login*

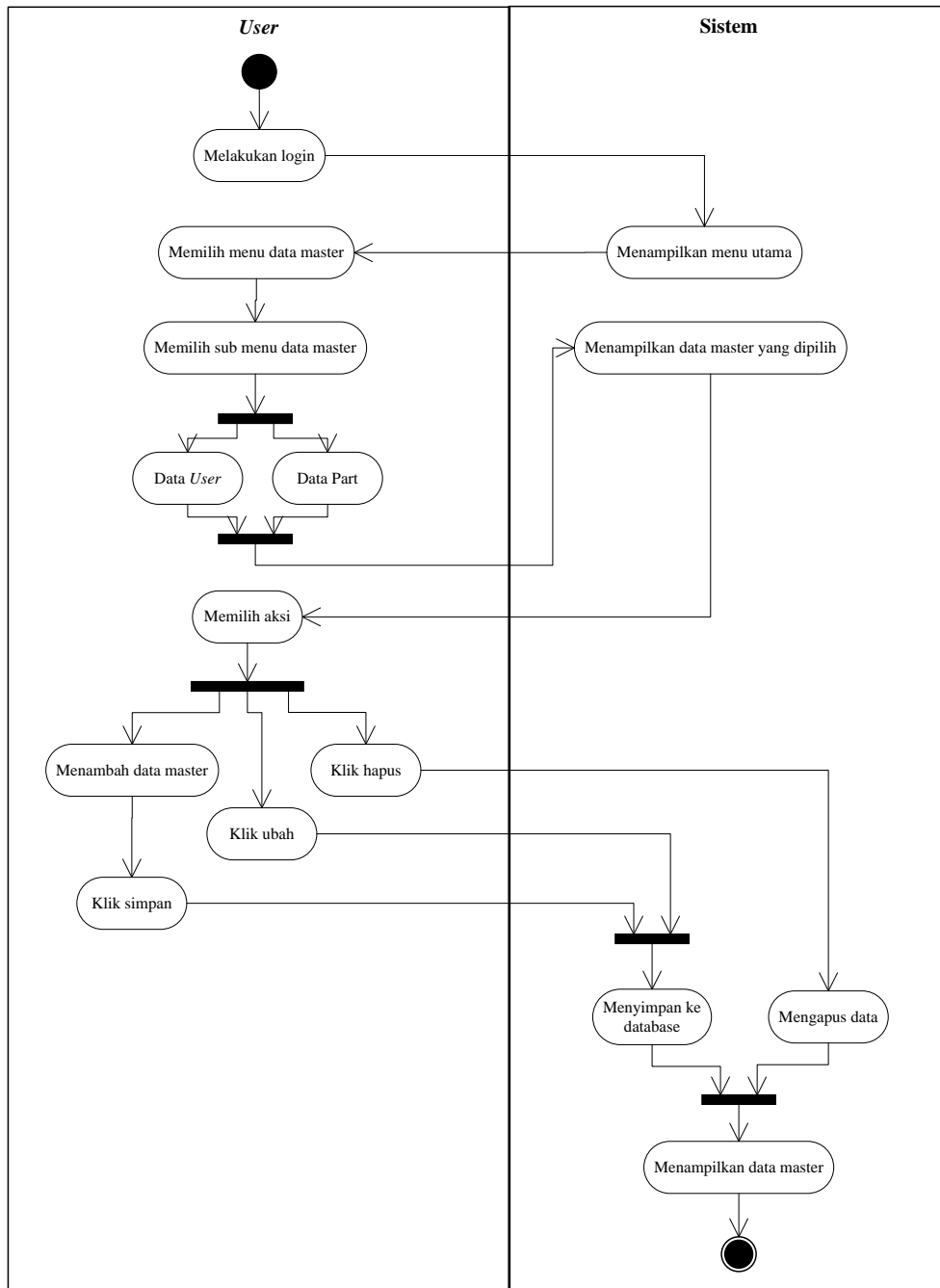
Activity diagram login menjelaskan alur aktivitas yang dilakukan oleh *user* yaitu PPIC, *Procurement* dan *Dept Head* untuk dapat masuk ke aplikasi perencanaan *indirect material*.



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

2. Activity Diagram Mengelola Data Master

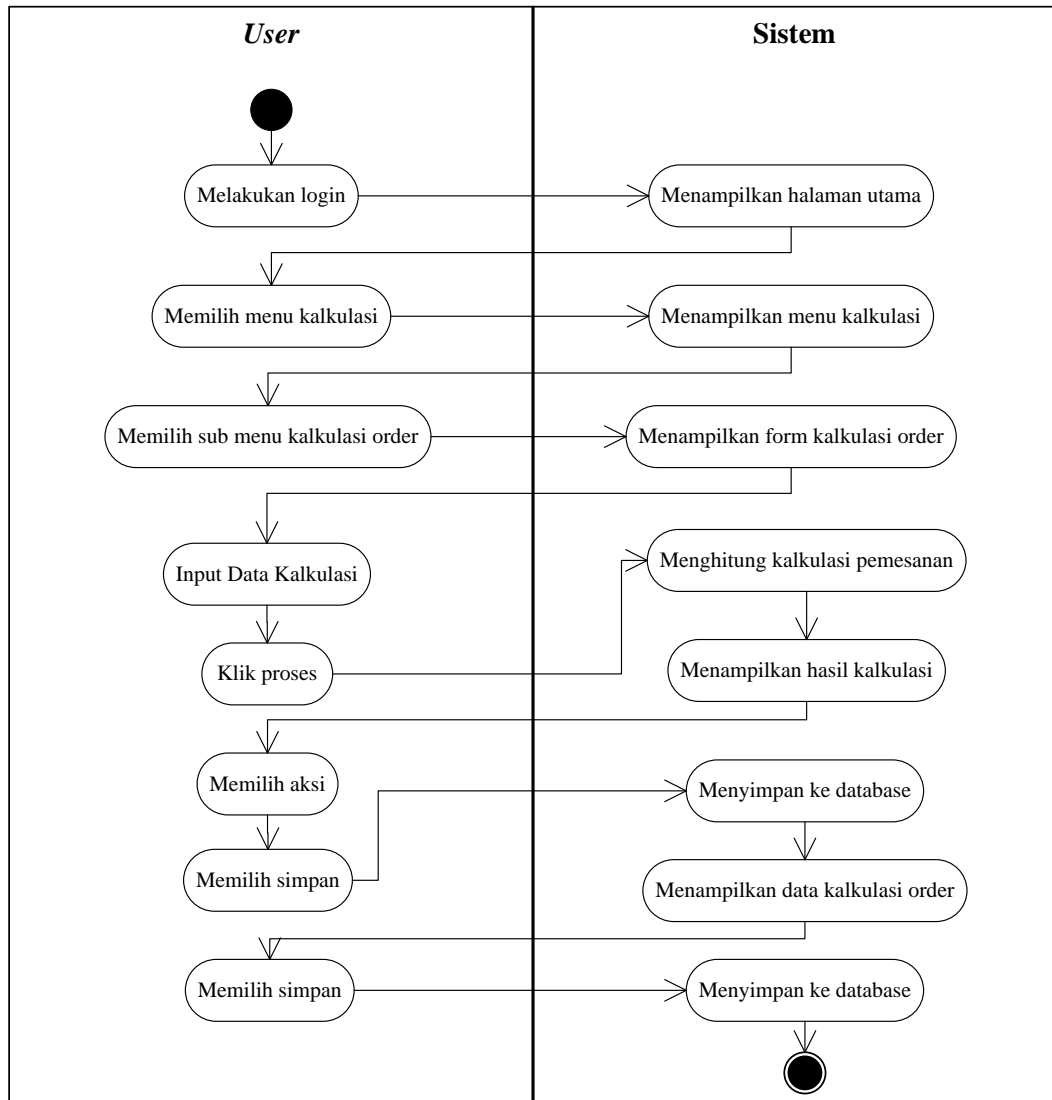
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam mengelola data *master*.



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

3. Activity Diagram Membuat Kalkulasi Order

Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam membuat kalkulasi order.

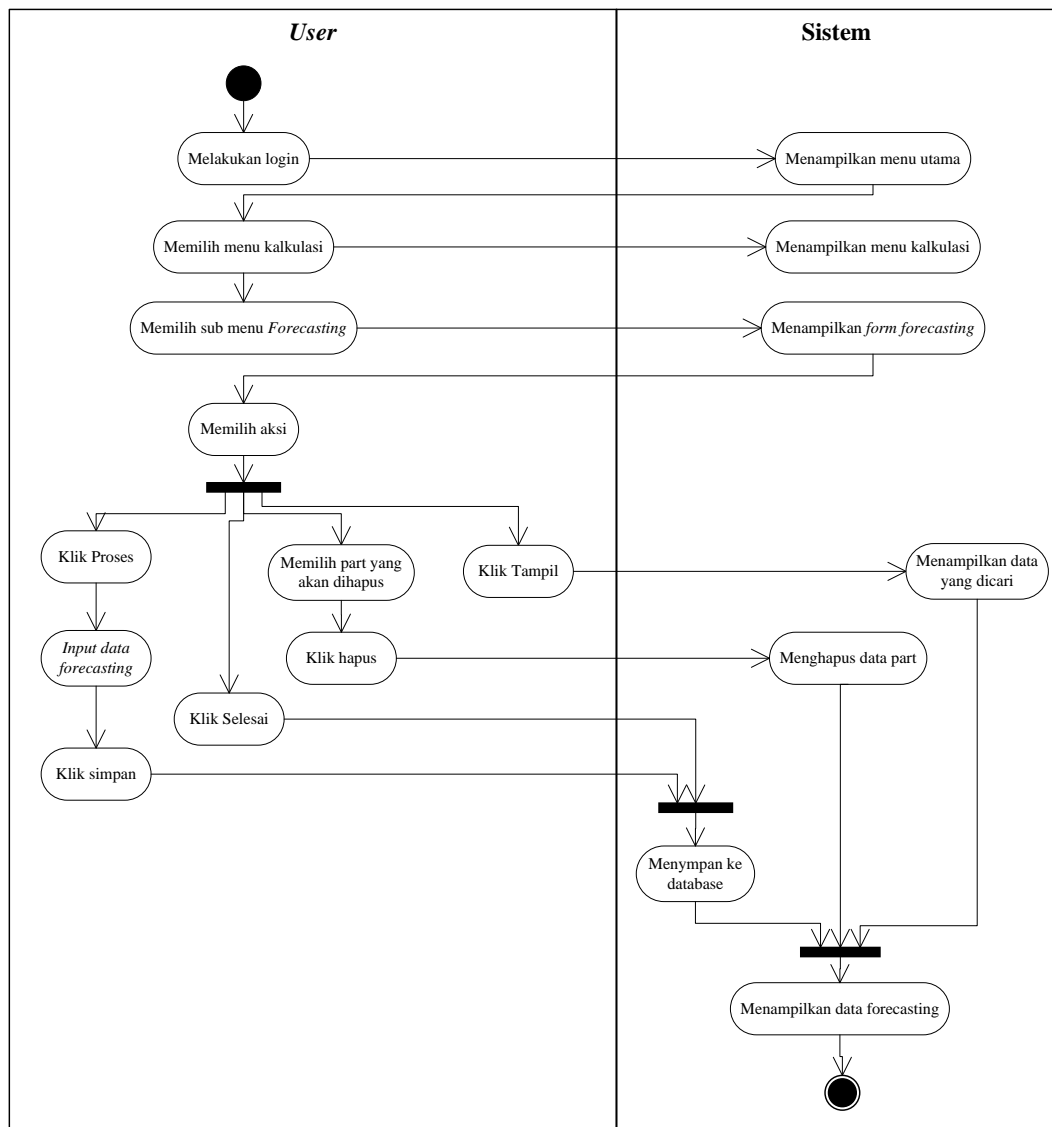


Gambar V.5 Activity Diagram Membuat Kalkulasi Order Bulanan

Sumber: Hasil Analisa Data (2016)

4. Activity Diagram Membuat Forecasting Bulanan

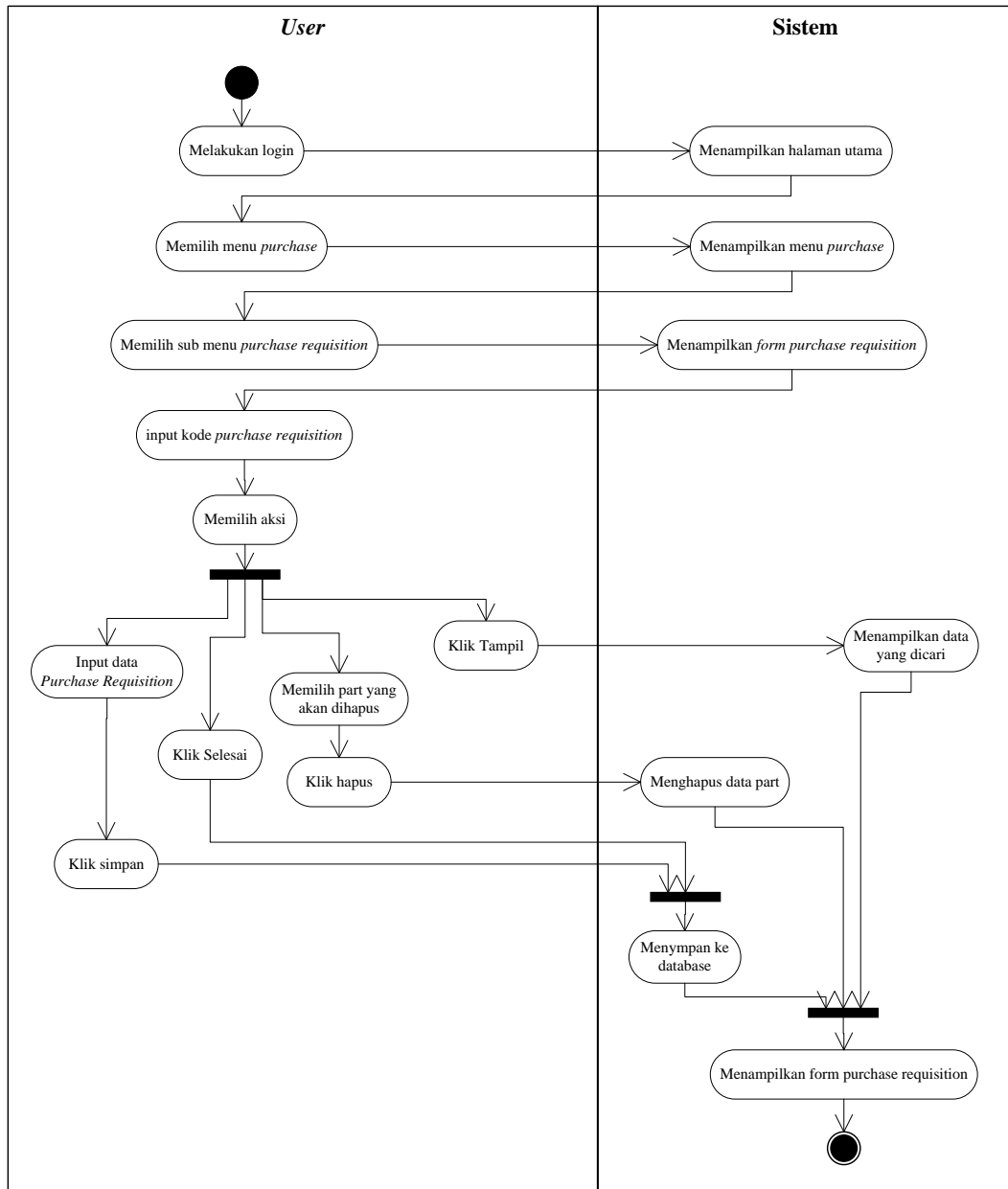
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam membuat forecasting bulanan.



Gambar V.6 Activity Diagram Membuat Forecasting Bulanan
 Sumber: Hasil Analisa Data (2016)

5. Activity Diagram Membuat Purchase Requisition

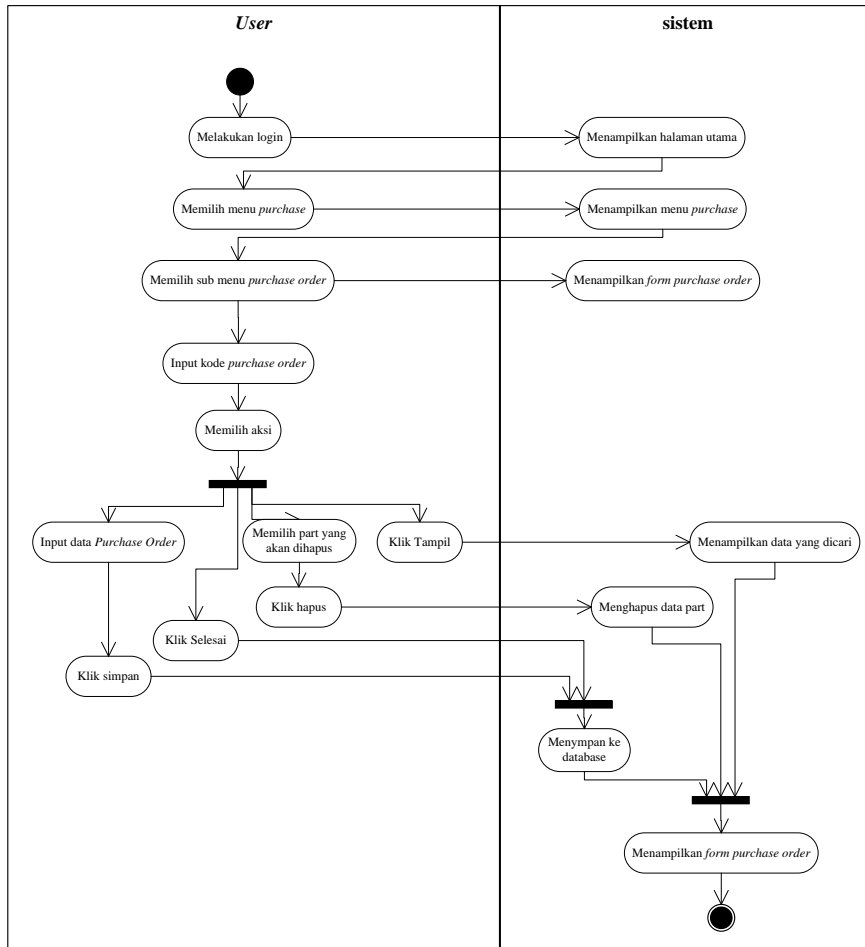
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam membuat purchase requisition.



Gambar V.7 Activity Diagram Membuat Purchase Requisition
 Sumber: Hasil Analisa Data (2016)

6. Activity Diagram Membuat Purchase Order

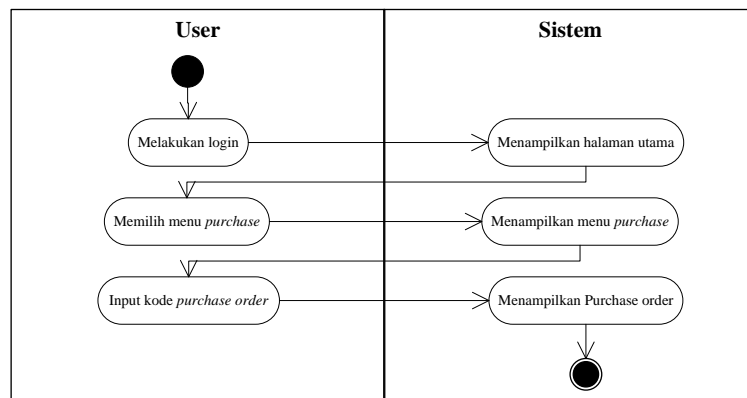
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam membuat Purchase Order.



Gambar V.8 Activity Diagram Membuat Purchase Order
 Sumber: Hasil Analisa Data (2016)

7. Activity Diagram View Purchase Order

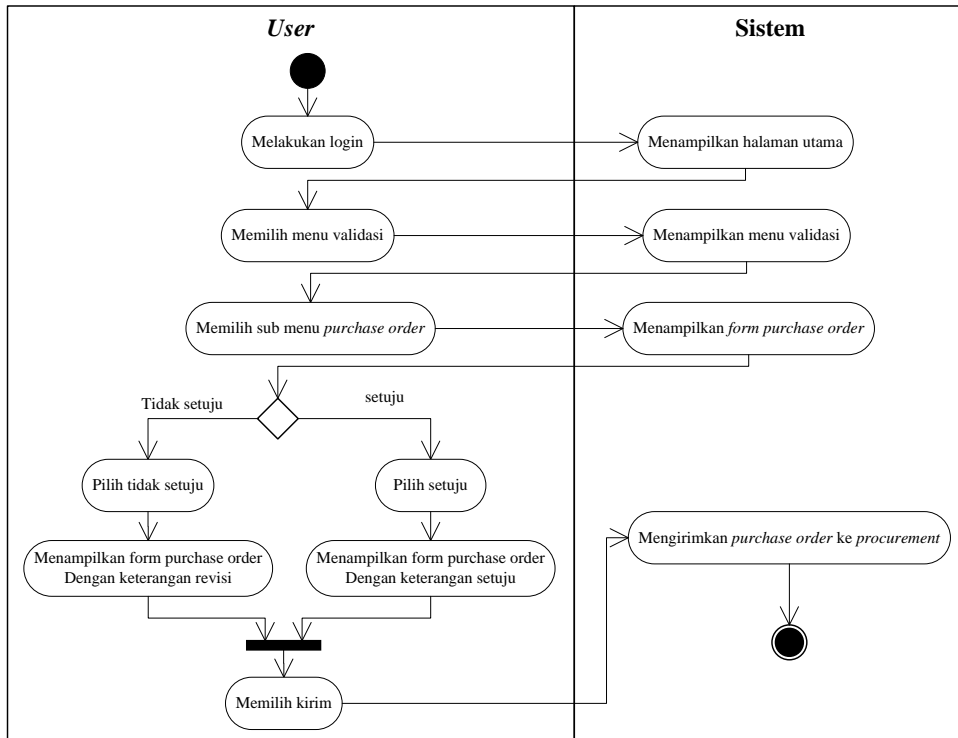
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam menampilkan Purchase Order



Gambar V.9 Activity Diagram View Purchase Order
 Sumber: Hasil Analisa Data (2016)

8. Activity Diagram Menyetujui Purchase Order

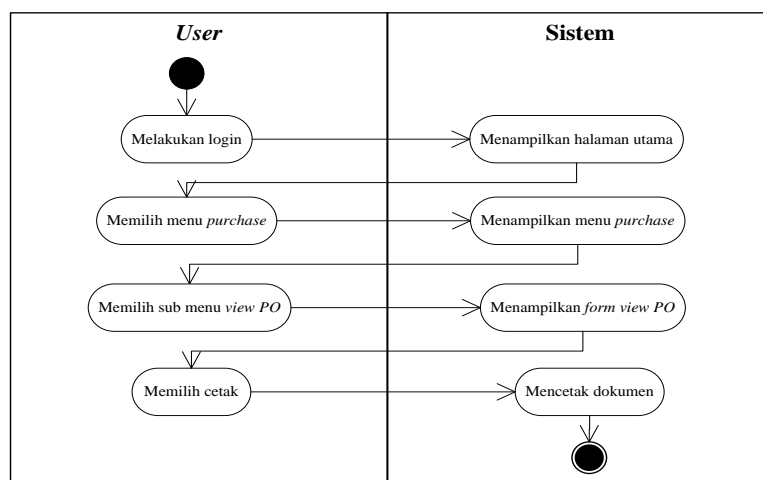
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam menyetujui Purchase Order.



Gambar V.10 Activity Diagram Menyetujui Purchase Order
Sumber: Hasil Analisa Data (2016)

9. Activity Diagram Melakukan Pemesanan

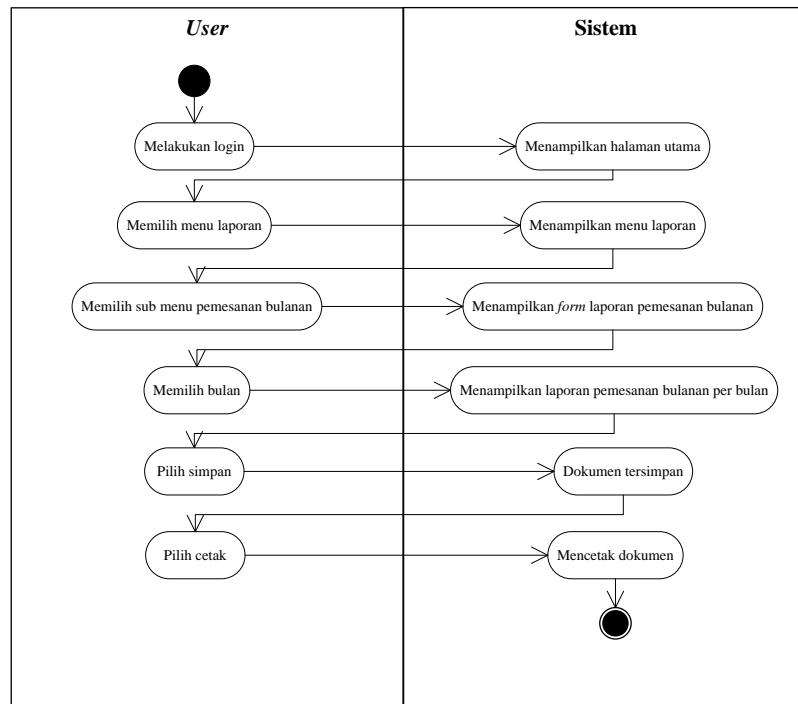
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam melakukan pemesanan.



Gambar V.11 Activity Diagram Melakukan Pemesanan
Sumber: Hasil Analisa Data (2016)

10. Activity Diagram Membuat Laporan Pemesanan Bulanan

Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam membuat laporan bulanan.



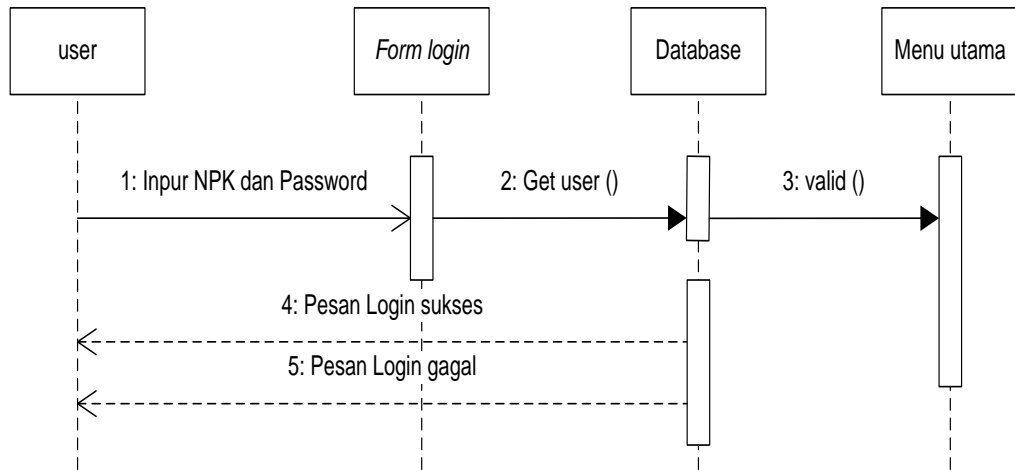
Gambar V.12 Activity Diagram Membuat Laporan Bulanan
Sumber: Hasil Analisa Data (2016)

5.2.4. Sequence Diagram Usulan

Sequence diagram menggambarkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu, di mana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada *diagram*. Hubungan yang ada pada gambar di bawah ini adalah proses yang dilakukan oleh sistem ketika melakukan proses yang sesuai dengan suatu objek *use case diagram*, berikut adalah *sequence diagram* pada aplikasi perencanaan *indirect material*:

1. Sequence Diagram Login

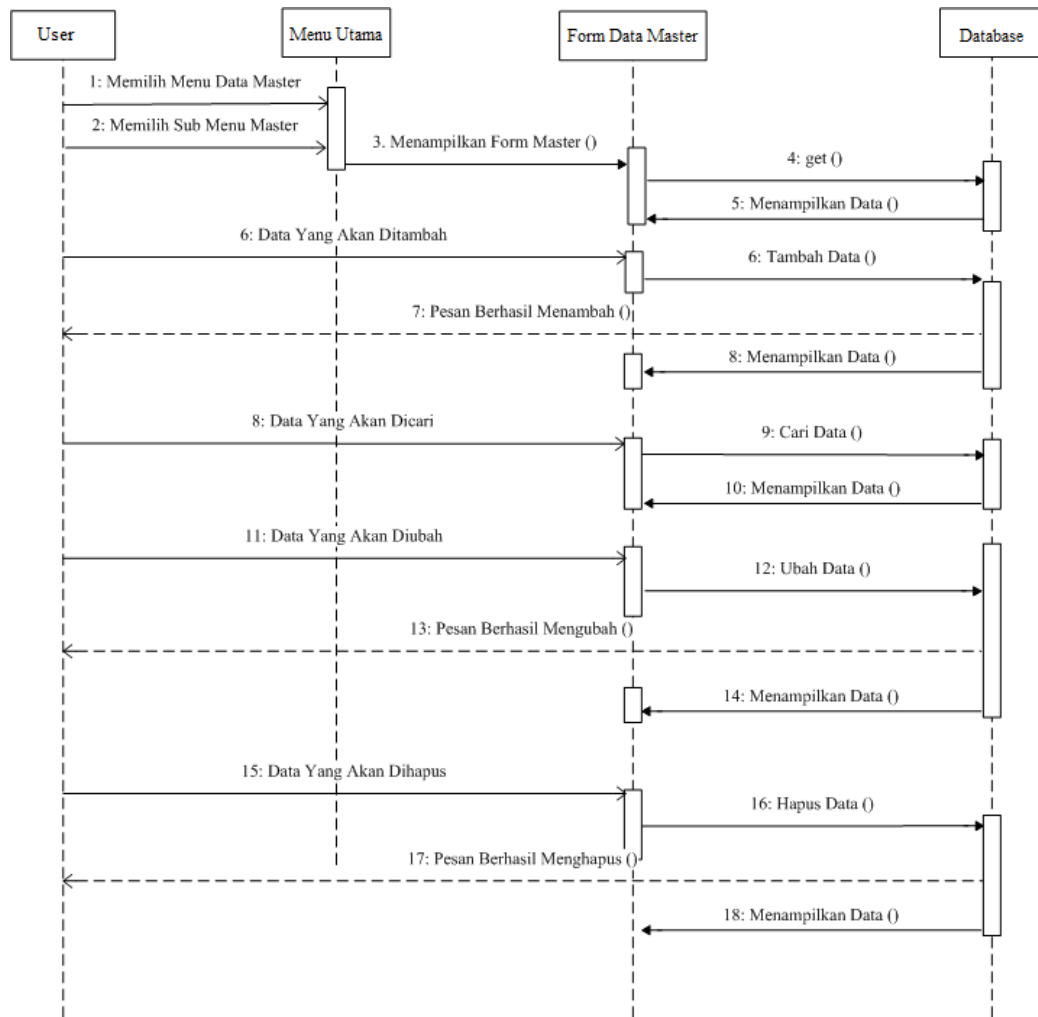
Sequence diagram login menjelaskan proses ini dilakukan oleh *user* sebelum masuk sistem. Proses ini dilakukan oleh *user* sebelum masuk ke sistem. Adapun *sequence diagram* dari *usecase login* adalah sebagai berikut.



Gambar V.13 *Sequence Diagram Login*
 Sumber: Hasil Analisa Data (2016)

2. *Sequence Diagram Mengelola Data Master*

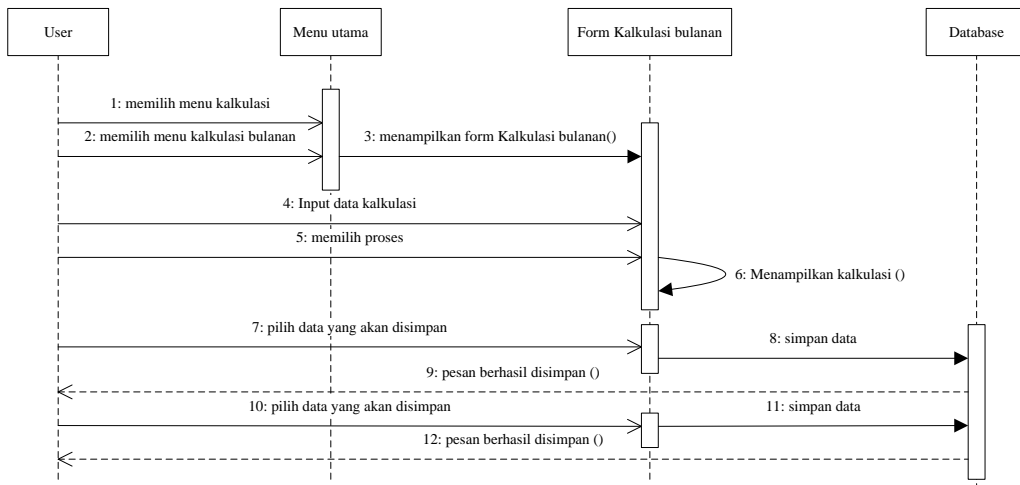
Sequence Diagram mengelola data *master* menjelaskan sebuah *sequence diagram* dalam proses menambah, menghapus dan mengubah data *master*. Data master pada aplikasi ini terdiri dari dua data yaitu data master user dan data master part. Adapun *sequence diagram* dari *use case* mengelola data *master* meliputi data *user* dan part yang dapat dilihat sebagai berikut:



Gambar V.14 *Sequence Diagram* Mengelola Data Master
 Sumber: Hasil Analisa Data (2016)

3. *Sequence Diagram* Membuat Kalkulasi Order Bulanan

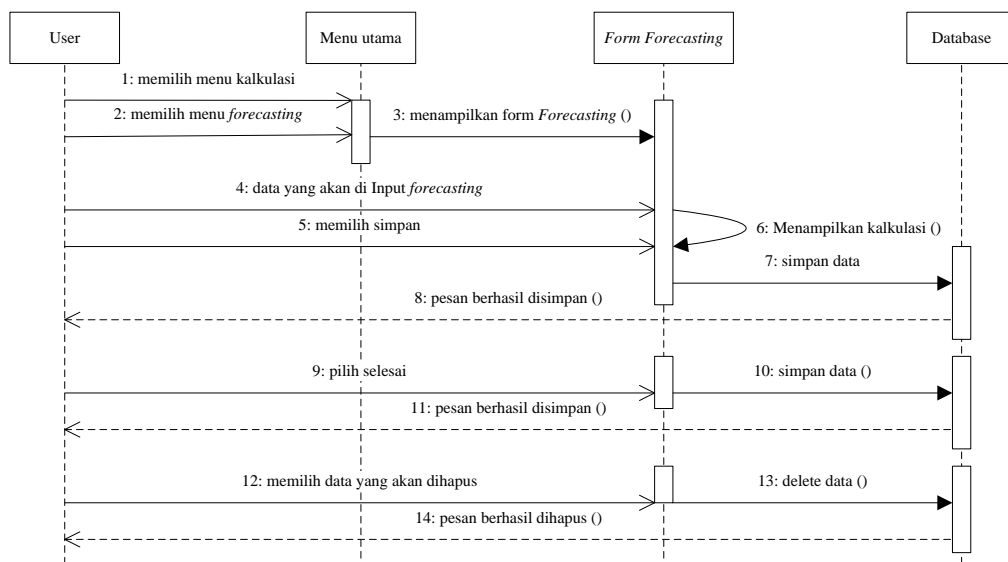
sequence diagram membuat kalkulasi order menjelaskan sebuah *sequence diagram* dalam sebuah proses menghitung kalkulasi order yang dilakukan oleh PPIC. Adapun *sequence diagram* membuat kalkulasi order bulanan dapat dilihat sebagai berikut.



Gambar V.15 *Sequence Diagram* Membuat Kalkulasi Order Bulanan
 Sumber: Hasil Analisa Data (2016)

4. *Sequence Diagram* Membuat *Forecasting* Bulanan

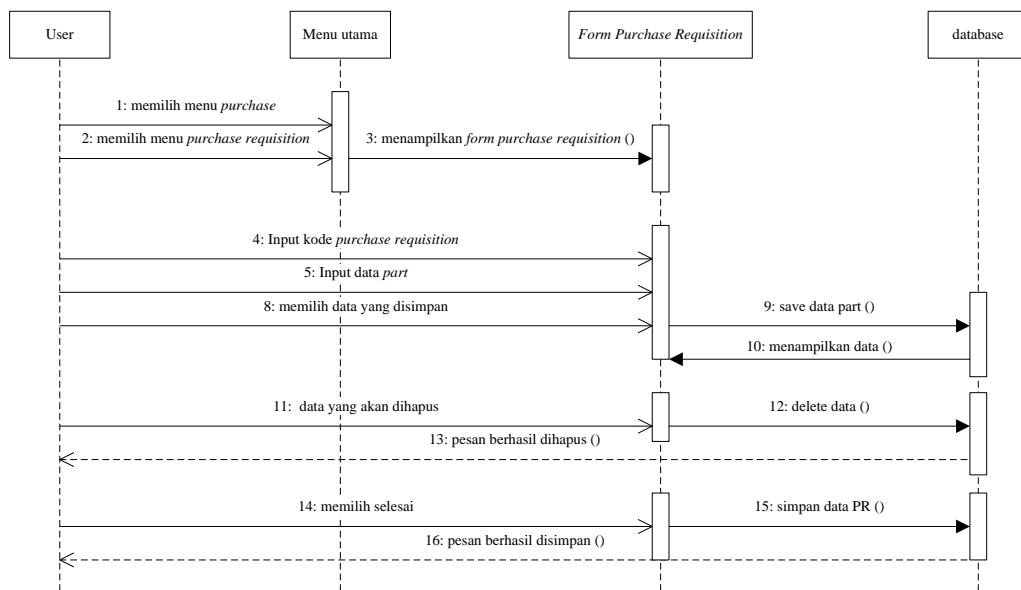
sequence diagram membuat *forecasting* bulanan menjelaskan sebuah *sequence diagram* dalam sebuah proses membuat *forecasting* selama satu bulan yang dilakukan oleh PPIC. Adapun *sequence diagram* membuat *forecasting* bulanan dapat dilihat sebagai berikut.



Gambar V.16 *Sequence Diagram* Membuat *Forecasting* Bulanan
 Sumber: Hasil Analisa Data (2016)

5. Sequence Diagram Membuat Purchase Requisition

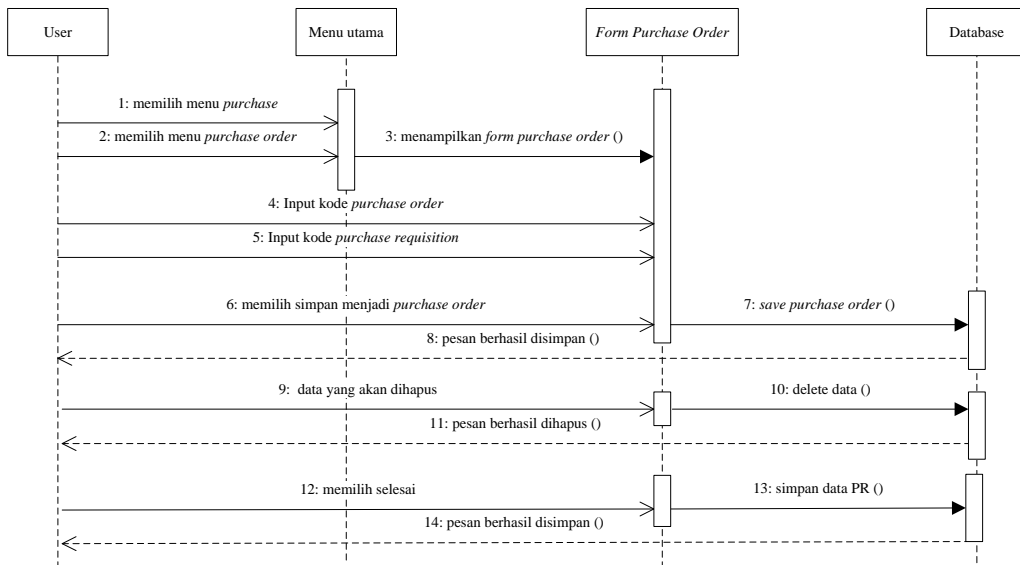
Sequence diagram membuat *purchase requisition* menjelaskan sebuah *sequence diagram* dalam sebuah proses membuat sebuah *purchase requisition* yang dilakukan oleh PPIC berdasarkan dokumen *forecasting*. Adapun *sequence diagram* membuat *purchase requisition* dapat dilihat sebagai berikut.



Gambar V.17 *Sequence Diagram* Membuat *Purchase Requisition*
Sumber: Hasil Analisa Data (2016)

6. Sequence Diagram Membuat Purchase Order

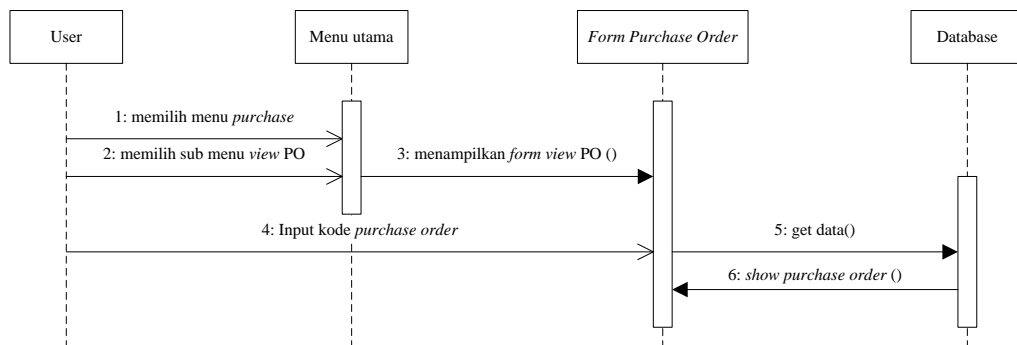
Sequence diagram membuat *Purchase Order* menjelaskan sebuah *sequence diagram* dalam sebuah proses membuat sebuah *Purchase Order* yang dilakukan oleh *Procurement* berdasarkan dokumen *purchase requisition*. Adapun *sequence diagram* membuat *Purchase Order* dapat dilihat sebagai berikut.



Gambar V.18 Sequence Diagram Membuat Purchase Order
 Sumber: Hasil Analisa Data (2016)

7. Sequence Diagram View Purchase Order

Sequence diagram view purchase order menjelaskan sebuah sequence diagram dalam sebuah proses penampilan data purchase order. Adapun sequence diagram view purchase order dapat dilihat sebagai berikut.

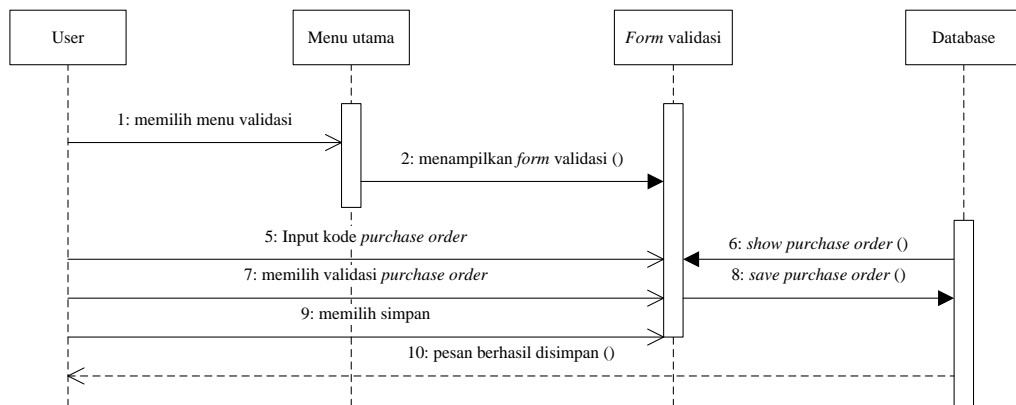


Gambar V.19 Sequence Diagram View Purchase Order
 Sumber: Hasil Analisa Data (2016)

8. Sequence Diagram Menyetujui Purchase Order

Sequence diagram menyetujui Purchase Order menjelaskan sebuah sequence diagram dalam sebuah proses disetujui atau tidaknya Purchase Order

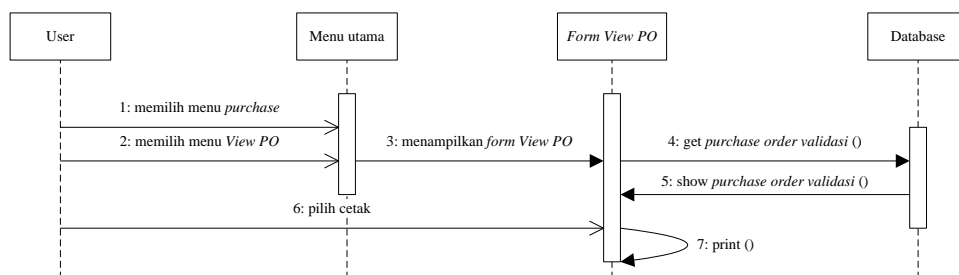
yang dilakukan oleh *Dept Head* berdasarkan anggaran belanja yang sudah ditentukan. Adapun *sequence diagram* menyetujui *Purchase Order* dapat dilihat sebagai berikut.



Gambar V.20 *Sequence Diagram* Menyetujui *Purchase Order*
 Sumber: Hasil Analisa Data (2016)

9. *Sequence Diagram* Melakukan Pemesanan

Sequence diagram melakukan pemesanan menjelaskan sebuah *sequence diagram* dalam sebuah proses melakukan pemesanan yang dilakukan oleh *Procurement* berdasarkan *Purchase Order* yang sudah divalidasi sebelumnya. Adapun *sequence diagram* melakukan pemesanan dapat dilihat sebagai berikut.

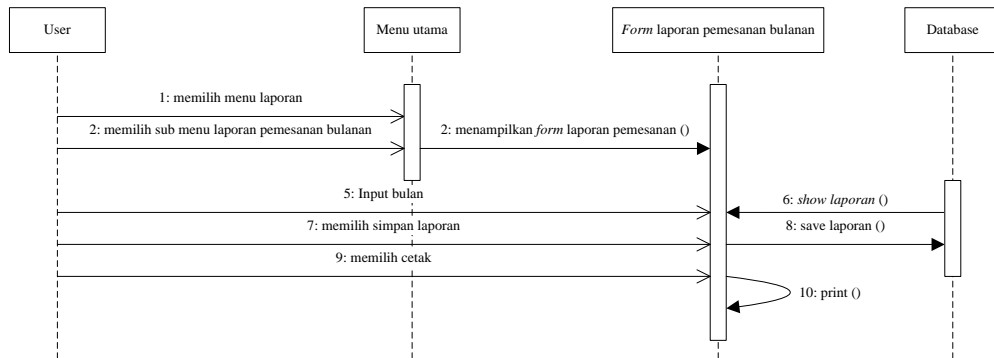


Gambar V.21 *Sequence Diagram* Melakukan Pemesanan
 Sumber: Hasil Analisa Data (2016)

10. *Sequence Diagram* Membuat Laporan Pemesanan Bulanan

Sequence diagram membuat laporan pemesanan bulanan menjelaskan sebuah *sequence diagram* dalam sebuah proses pembuatan laporan yang dilakukan oleh *Procurement* berdasarkan *Purchase Order* yang sudah

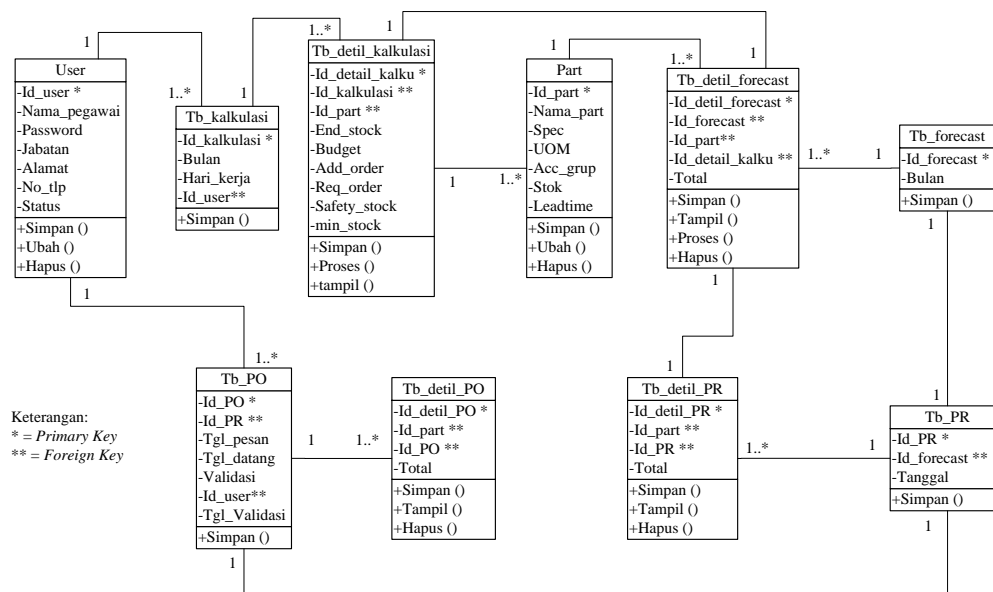
dilakukan pemesanan sebelumnya. Adapun *sequence diagram* Membuat Laporan Pemesanan Bulanan dapat dilihat sebagai berikut.



Gambar V.22 *Sequence Diagram* Membuat Laporan Bulanan
Sumber: Hasil Analisa Data (2016)

5.2.5. *Class Diagram* Usulan

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan rinci tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* aplikasi perencanaan *indirect material* yang diusulkan dapat dilihat sebagai berikut:



Gambar V.23 *Class Diagram* Aplikasi Perencanaan *Indirect Material*
Sumber: Hasil Analisa Data (2016)

5.2.6. Kamus Data

Kamus data adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem, sehingga *user* dan analisis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan dan basis data. Berikut adalah kamus data aplikasi perencanaan *indirect material*:

1. Spesifikasi Tabel *User*

Nama Tabel : *User*

Fungsi : Untuk menyimpan data *user*

Tipe : File data master

Tabel V.14. Tabel *User*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id <i>User</i>	id_user	<i>Varchar</i>	10	<i>Primary Key</i>
2.	Nama Pegawai	Nama_pegawai	<i>Varchar</i>	25	
3.	<i>Password</i>	Password	<i>Varchar</i>	8	
4.	Jabatan	Jabatan	<i>Varchar</i>	20	
5.	Alamat	Alamat	<i>Varchar</i>	50	
6.	No Telepon	No_tlp	<i>Varchar</i>	12	
7.	Status	status	<i>varchar</i>	3	

Sumber: Hasil Analisa Data, 2016

2. Spesifikasi Tabel *Part*

Nama Tabel : *Part*

Fungsi : Untuk menyimpan data *part*

Tipe : File data master

Tabel V.15. Tabel *Part*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id <i>part</i>	Id_part	<i>Varchar</i>	10	<i>Primary Key</i>
2.	Nama part	Nama_part	<i>Varchar</i>	25	
3.	Spesifikasi	Spec	<i>Varchar</i>	25	
4.	Satuan	UOM	<i>Varchar</i>	15	

5.	Grup barang	Acc_grup	<i>Char</i>	2	
6.	Stok	Stok	<i>Smallint</i>		
7.	Lead time	leadtime	<i>Tinyint</i>		

Sumber: Hasil Analisa Data, 2016

3. Spesifikasi Tabel Kalkulasi

Nama Tabel : Kalkulasi

Fungsi : Untuk menampilkan

Tipe : File data kalkulasi

Tabel V.16. Tabel Kalkulasi

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id Kalkulasi	Id_kalkulasi	<i>Varchar</i>	10	<i>Primary Key</i>
2	Hari Kerja	Hari_kerja	<i>Tinyint</i>		
3	Bulan	Bulan	<i>Date</i>		
4	Id user	Id_user**	<i>Varchar</i>	10	<i>Foreign Key</i>

Sumber: Hasil Analisa Data, 2016

4. Spesifikasi Tabel Detail Kalkulasi

Nama Tabel : Detail Kalkulasi

Fungsi : Untuk menyimpan data detail kalkulasi

Tipe : File data *forecasting*

Tabel V.17 Tabel Detail Kalkulasi

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id detail kalkulasi	Id_detail_kalku *	<i>Varchar</i>	10	<i>Primary Key</i>
2.	Id Kalkulasi	Id_kalkulasi **	<i>Varchar</i>	10	<i>Foreign key</i>
3.	Id part	Id_part **	<i>Varchar</i>	10	<i>Foreign Key</i>
4.	Stok Terakhir	End_stock	<i>Smallint</i>		
5.	Anggaran	Budget	<i>Smallint</i>		
6.	Order tambahan	Add_order	<i>Smallint</i>		
7.	Order reguler	Req_order	<i>Smallint</i>		
8.	Safety stok	Safety_stok	<i>Smallint</i>		
9.	Minimum Stok	Min_stok	<i>smallint</i>		

Sumber: Hasil Analisa Data, 2016

5. Spesifikasi Tabel *Forecasting*

Nama Tabel : *Forecasting*

Fungsi : Untuk menyimpan data *forecasting*

Tipe : File data *forecasting*

Tabel V.18. Tabel *Forecasting*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	<i>Id forecasting</i>	<i>Id_forecast</i>	<i>Varchar</i>	10	<i>Primary Key</i>
2	Bulan	Bulan	<i>Date</i>		

Sumber: Hasil Analisa Data, 2016

6. Spesifikasi Tabel Detail *Forecasting*

Nama Tabel : *Detail Forecasting*

Fungsi : Untuk menyimpan data detail *forecasting*

Tipe : File data detail *forecasting*

Tabel V.19. Tabel Detail *Forecasting*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	<i>Id detail forecasting</i>	<i>Id_detail_forecast</i>	<i>Varchar</i>	10	<i>Primary Key</i>
2.	<i>Id forecasting</i>	<i>Id_forecast</i>	<i>Varchar</i>	10	<i>Foreign Key</i>
3.	Id Part	<i>Id_part</i>	<i>Varchar</i>	10	<i>Foreign Key</i>
4	Id Kalkulasi	<i>Id_kalkulasi</i>	<i>Varchar</i>	10	<i>Foreign Key</i>
5	Total	<i>total</i>	<i>Smallint</i>		

Sumber: Hasil Analisa Data, 2016

7. Spesifikasi Tabel *Purchase Requisition*

Nama Tabel : *Purchase Requisition*

Fungsi : Untuk menyimpan data *Purchase Requisition*

Tipe : File data *Purchase Requisition*

Tabel V.22. Tabel *Purchase*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	<i>Id Purchase Requisition</i>	<i>Id_PR *</i>	<i>Varchar</i>	10	<i>Primary Key</i>
3	<i>Id Forecasting</i>	<i>Id_forecasting **</i>	<i>Varchar</i>	10	<i>Foreign key</i>
4	Tanggal	Tanggal	<i>Date</i>		

Sumber: Hasil Analisa Data, 2016

8. Spesifikasi Tabel Detail *Purchase Requisition*

Nama Tabel : Detail *Purchase Requisition*

Fungsi : Untuk menyimpan data detail *Purchase Requisition*

Tipe : File data detail *Purchase Requisition*

Tabel V.23. Tabel Detail *Purchase Requisition*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id detail <i>Purchase Requisition</i>	id_detil_PR	<i>Varchar</i>	10	<i>Primary Key</i>
2	Id <i>Forecasting</i>	Id_forecasting *	<i>Varchar</i>	10	<i>Foreign key</i>
3	Id <i>part</i>	Id_part **	<i>Varchar</i>	10	<i>Foreign key</i>
4	Total	Total	<i>Smallint</i>		

Sumber: Hasil Analisa Data, 2016

9. Spesifikasi Tabel *Purchase Order*

Nama Tabel : *Purchase Order*

Fungsi : Untuk menyimpan data *Purchase Order*

Tipe : File data *Purchase Order*

Tabel V.22. Tabel *Purchase Order*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id <i>Purchase Order</i>	Id_PO *	<i>Varchar</i>	10	<i>Primary Key</i>
2	Id <i>Purchase Requisition</i>	Id_PR **	<i>Varchar</i>	10	<i>Foreign key</i>
3	Tanggal Pesan	Tgl_pesanan	<i>Date</i>		
4	Tanggal Datang	Tgl_datang	<i>Date</i>		
5	Status	Validasi	<i>Varchar</i>	10	
6	Id user	Id_user**	<i>Varchar</i>	10	<i>Foreign key</i>

Sumber: Hasil Analisa Data, 2016

10. Spesifikasi Tabel Detail *Purchase Order*

Nama Tabel : Detail *Purchase Order*

Fungsi : Untuk menyimpan data detail *Purchase Order*

Tipe : File data detail *Purchase Order*

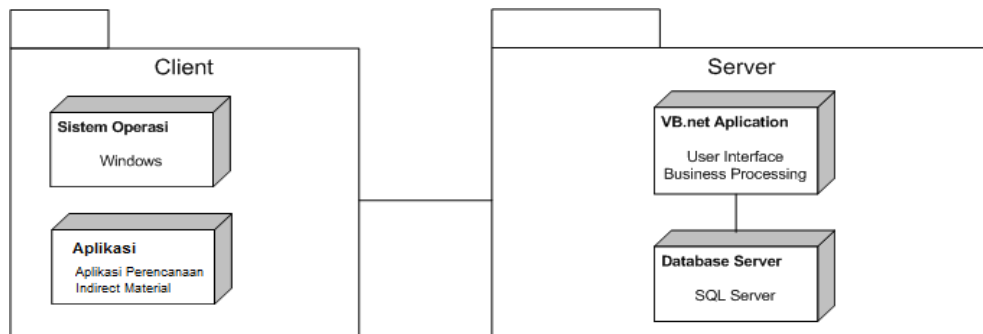
Tabel V.23. Tabel Detail *Purchase Order*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id detail <i>Purchase Order</i>	id_detil_PO	<i>Varchar</i>	10	<i>Primary Key</i>
2	Id <i>PR</i>	Id_PR *	<i>Varchar</i>	10	<i>Foreign key</i>
3	Id <i>part</i>	Id_part **	<i>Varchar</i>	10	<i>Foreign key</i>
4	Total	Total	<i>Smallint</i>		

Sumber: Hasil Analisa Data, 2016

5.2.7. *Deployment Diagram Usulan*

Deployment diagram digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur yang menggambarkan arsitektur fisik dari system, seperti sistem operasi dan sistem yang menjadi pendukung. *Deployment diagram* aplikasi perencanaan *indirect material* yang diusulkan dapat dilihat sebagai berikut.



Gambar V.24 *Deployment Diagram* Aplikasi Perencanaan *Indirect Material*

Sumber: Hasil Analisa Data (2016)

Berikut adalah penjelasan gambar V.23 *deployment diagram* aplikasi perencanaan *indirect material*:

1. Visual Basic

Visual Basic adalah bahasa pemrograman yang digunakan pada aplikasi Microsoft Visual Basic.net untuk pembuatan dan pengembangan aplikasi berbasis desktop.

2. *Database*

Database adalah program komputer yang menyediakan layanan data lainnya ke komputer atau program komputer, seperti yang ditetapkan oleh model

client-server. Istilah ini juga merujuk kepada sebuah komputer yang didedikasikan untuk menjalankan program *server database*.

3. Sistem Operasi

Sistem operasi adalah perangkat lunak komputer atau *software* yang bertugas untuk melakukan kontrol dan manajemen perangkat keras dan juga operasi-operasi dasar sistem. Sistem operasi yang digunakan adalah *windows*.

4. Server

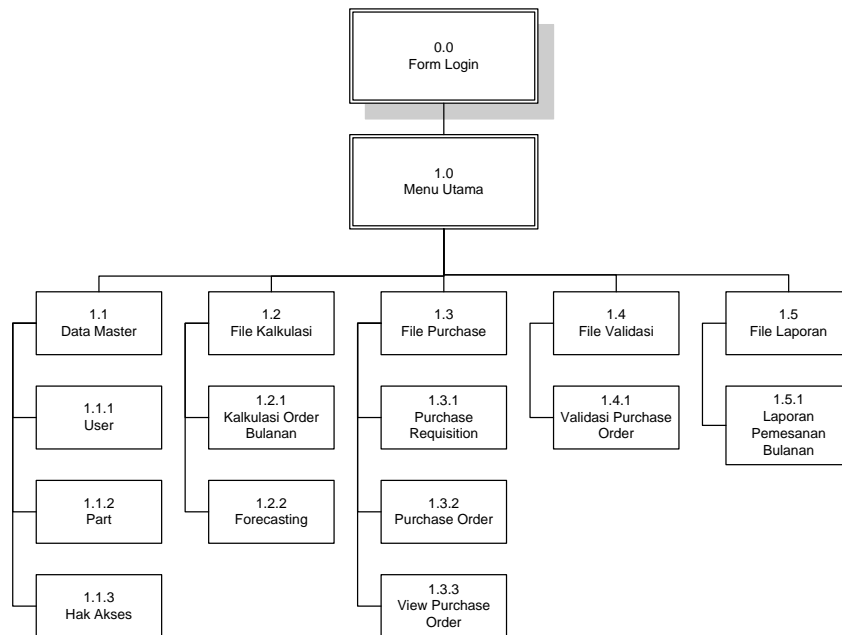
Suatu program sekaligus mesin yang menjalankan program komputer yang dapat menanggapi permintaan-permintaan dari *client*.

5.3. Analisis Desain Program

Tahap ini merupakan tahap kedua dalam metodologi *prototype evolutioner*, yaitu tahap membuat sebuah prototipe dari program/aplikasi. Dimulai dengan analisis desain usulan meliputi pembuatan struktur menu program, *flowchart* program, dan *interface* program sampai dengan program dapat dijalankan. Pada tahap ini dilakukan upaya untuk memperbaiki sistem ataupun membangun dan menghasilkan sistem yang baru dengan memanfaatkan teknologi terbaru dan fasilitas yang tersedia untuk mengurangi dan mengatasi berbagai permasalahan yang telah terjadi pada sistem.

5.3.1. HIPO

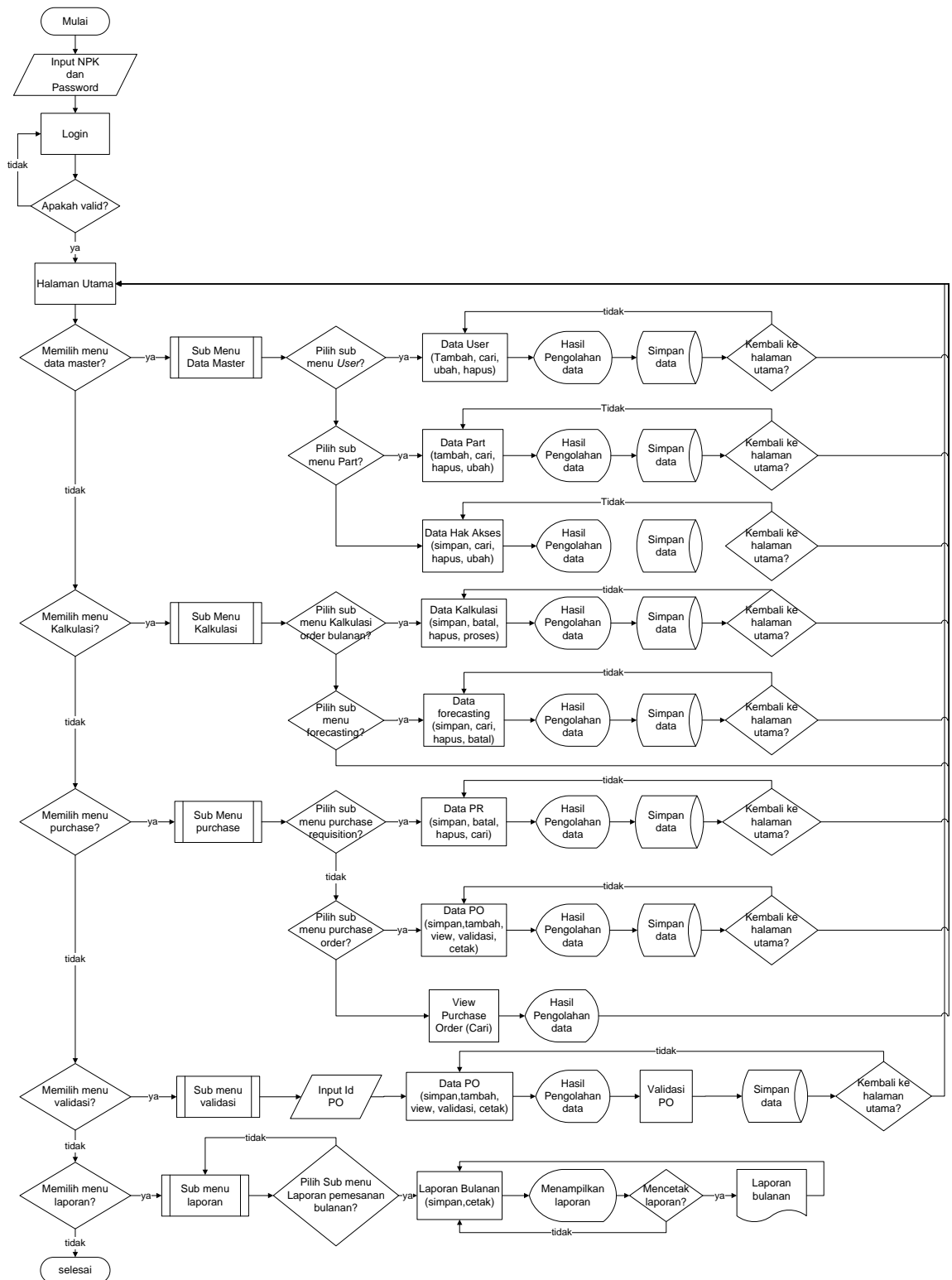
Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hierarkis. Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya. Perancangan HIPO aplikasi usulan aplikasi perencanaan *indirect material* dapat dilihat pada Gambar V.25 sebagai berikut:



Gambar V.25 HIPO Aplikasi Perencanaan *Indirect Material*
 Sumber: Hasil Analisis Data, 2016

5.3.2. Flowchart Program

Flowchart yang digunakan untuk mendokumentasikan aplikasi perencanaan *indirect material* ini menggunakan bagan alir logika program (*program logic flowchart*). Bagan alir ini digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. (Lihat Gambar V.21):



Gambar V.26 Flowchart Aplikasi Perencanaan Indirect Material

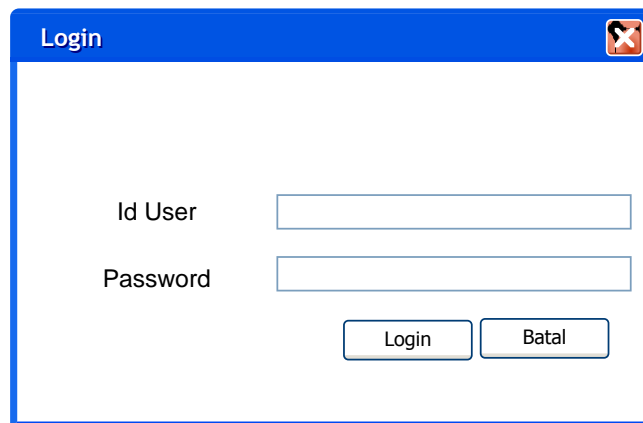
Sumber: Hasil Analisis Data, 2016

5.3.3. Perancangan *Interface Program*

Rancangan interface dari program perencanaan *indirect material* ini adalah sebagai berikut:

1. Tampilan *Form Login*

Form login adalah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk, pengguna harus memasukkan NPK dan sandi milik pengguna secara benar. Rancangan form login dapat dilihat sebagai berikut.



Gambar V.27 *Form Login*
Sumber: Hasil Analisis Data, 2016

2. Tampilan *Form Menu Utama*

Form menu utama adalah tampilan awal pada aplikasi yang berisi enam menu dan sub menu pada menu utama. Berikut adalah rincian menu dan sub menu pada *interface* menu utama:

1. File Data Master
 - a. User
 - b. Part
 - c. Hak Akses
2. File Kalkulasi
 - a. Kalkulasi Order
 - b. *Forecasting*
3. File Purchase
 - a. *Purchase Requisition*
 - b. *Purchase Order*

c. *View PO*

- 4. File validasi
- 5. File Laporan

a. Laporan Pemesanan Bulanan



Gambar V.28 Menu Utama Aplikasi Perencanaan *Indirect Material*
Sumber: Hasil Analisis Data, 2016

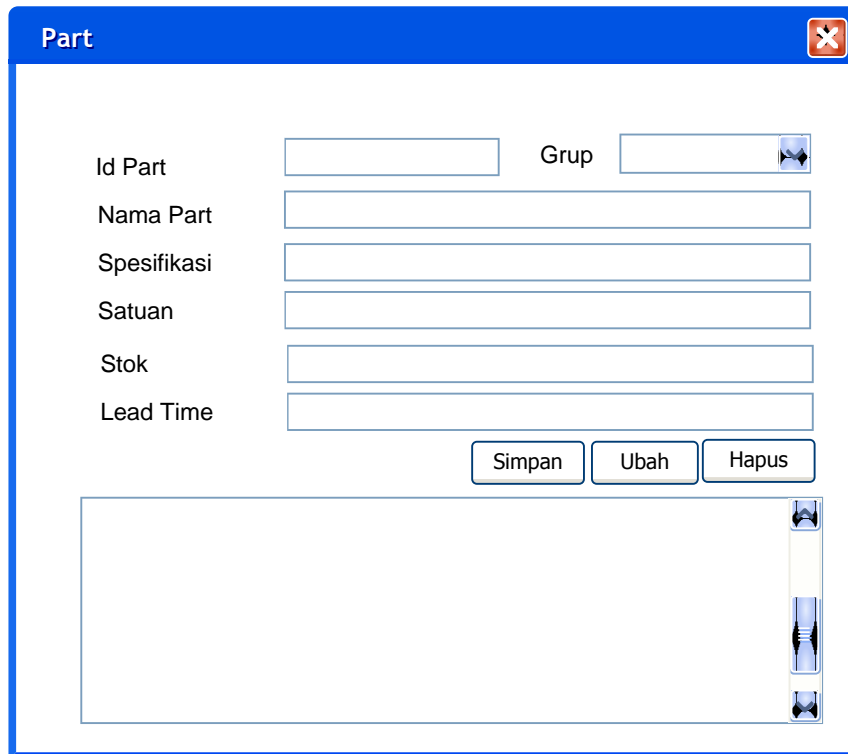
3. Tampilan *Form Data Master User*

Data User Berfungsi untuk menginput data Id User, Nama Pegawai, Password, alamat, jabatan, dan nomor telepon. Tombol yang tersedia adalah tambah, cari, hapus, dan batal. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan data tersebut.

Gambar V.29 Form User
Sumber: Hasil Analisis Data, 2016

4. Tampilan *Form* Data Master Part

Data Part Berfungsi untuk menginput data Id Part, Nama Part, Spesifikasi, satuan, grup part, dan harga. Tombol yang tersedia adalah tambah, cari, hapus, dan batal. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan data tersebut.



Gambar V.29 Form Part
Sumber: Hasil Analisis Data, 2016

5. Tampilan Form Hak Akses

Form Hak Akses digunakan untuk memberikan hak akses kepada user sesuai dengan jabatan dan posisinya. Form ini terdiri dari id user, nama user, dan *password*, serta pilihan form-form yang akan diberikan hak aksesnya. Tombol yang tersedia adalah tambah, proses, hapus, dan batal. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan hak akses tersebut.

Hak Akses

Form Hak Akses

Id User

Nama User

Password

PPIC Procurement Dept Head Procurement

Gambar V.31 Form Hak Akses
Sumber: Hasil Analisis Data, 2016

6. Tampilan *Form* Kalkulasi Order

Form Kalkulasi Order digunakan untuk menginput atau mengolah data untuk pemesanan barang. Form kalkulasi terdiri dari id kalkulasi, id part, anggaran belanja, id pegawai, end stock, jumlah hari kerja, order reguler, dan order tambahan. Tombol yang tersedia adalah tambah, proses, hapus, dan batal. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan data tersebut.

Gambar V.32 Form Kalkulasi Order
 Sumber: Hasil Analisis Data, 2016

7. Tampilan *Form Forecasting*

Form Forecasting digunakan untuk menginput data untuk pemesanan barang yang sudah dihitung. Form kalkulasi terdiri dari Id forecasting, id kalkulasi, id part, bulan dan total. Tombol yang tersedia adalah simpan, cari, hapus, dan batal. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan data tersebut.

Gambar V.33 Form Forecasting
 Sumber: Hasil Analisis Data, 2016

8. Tampilan *Form Purchase Requisition*

Form Purchase Requisition digunakan untuk menginput data Forecasting menjadi purchase requisition. *Form Purchase Requisition* terdiri dari id PR, id part, Id Forecasting, tanggal dan total. Tombol yang tersedia adalah simpan, cari, hapus, dan batal. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan data tersebut.

The screenshot shows a software window titled "Purchase Requisition". The window has a blue title bar with standard Windows window controls (minimize, maximize, close). The main content area contains the following elements:

- Input fields for "Id PR", "Tanggal", "Id Forecasting", "Id Detail PR", "Id Part", and "Total".
- Three buttons: "Simpan", "Hapus", and "Selesai".
- A large empty rectangular area at the bottom, likely a list or table, with a vertical scrollbar on the right side.

Gambar V.34 *Form Purchase Requisition*
Sumber: Hasil Analisis Data, 2016

9. Tampilan *Form Purchase Order*

Form Purchase Order digunakan untuk mengubah data Purchase requisition menjadi *Purchase Order*. *Form Purchase Order* terdiri dari id PR, id part, Id Forecasting, tanggal dan total. Tombol yang tersedia adalah simpan, cari, hapus, dan batal. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan data tersebut.

Gambar V.35 Form *Purchase Order*
 Sumber: Hasil Analisis Data, 2016

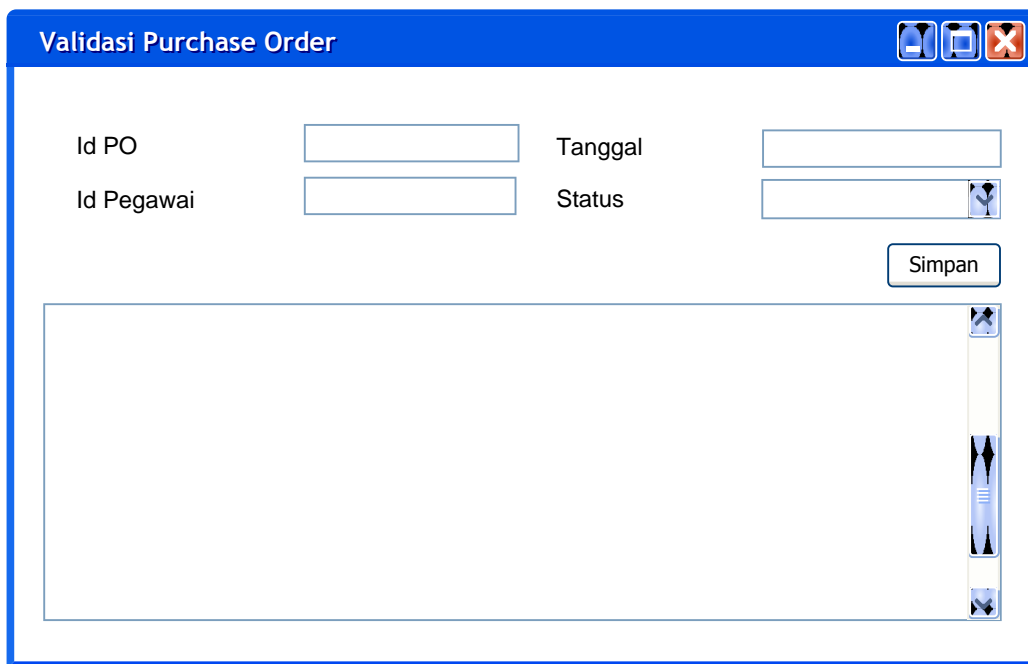
10. Tampilan Form View PO

Form *view* PO digunakan untuk memperlihatkan dokumen PO yang sudah dibuat oleh *procurement*. Form *view* PO terdiri dari id PO dan tombol cari yang berfungsi mencari data PO yang sudah dibuat

Gambar V.36 Form View PO
 Sumber: Hasil Analisis Data, 2016

11. Tampilan *Form Validasi*

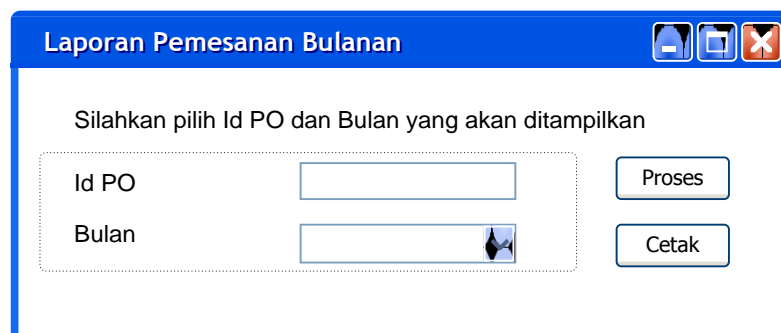
Form validasi digunakan untuk menyetujui *Purchase Order* untuk dilakukan pemesanan. *Form Purchase Order* terdiri dari id PO, id pegawai, tanggal, dan status. Tombol yang tersedia adalah simpan, dan cetak. Dimana masing-masing memiliki tugas menyimpan, mengubah, menghapus penyimpanan data tersebut serta mencetak data menjadi laporan.



Gambar V.37 Form Validasi
Sumber: Hasil Analisis Data, 2016

12. Tampilan *Form Laporan Pemesanan Bulanan*

Form Laporan digunakan untuk menampilkan data untuk pelaporan pemesanan bulanan.



Gambar V.38 Form Laporan Pemesanan Bulanan
Sumber: Hasil Analisis Data, 2016

Laporan Pemesanan Bulanan

Id PO :

Bulan :

Id Pegawai

Id Barang	Anggaran Belanja	Order Reguler	Order Tambahan	Total	End Stock

Gambar V.39 Rancangan Tampilan Laporan Pemesanan Bulanan
Sumber: Hasil Analisis Data, 2016

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

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

1. Aplikasi Perencanaan *Indirect Material* berbasis desktop ini melakukan perhitungan berdasarkan rumus *agregat planning* dalam mengelola perencanaan *indirect material* dan pengawasan keadaan stok sehingga keadaan stok aman dan terkendali.
2. Membuat proses validasi *purchase order* menjadi lebih cepat yang sebelumnya membutuhkan waktu menunggu satu hari untuk validasi. Sistem peringatan yang baru pada aplikasi ini membuat proses validasi bisa dilakukan dengan waktu beberapa menit saja.

6.2 Saran

Saran yang dapat diberikan untuk pengembangan Aplikasi Perencanaan *Indirect Material* ini selanjutnya adalah sebagai berikut:

1. Melakukan pembekalan dasar kepada staf PPIC, *Procurement* dan *Dept Head Procurement* tentang aplikasi perencanaan *indirect material*.
2. Aplikasi dilakukan pemeliharaan secara berkala agar tetap berjalan dengan baik.

DAFTAR PUSTAKA

- Arikunto, Suharsimi. 2006. *Prosedur Penelitian: Suatu Pendekatan Praktek*, Edisi Revisi. PT Rineka Cipta, Jakarta.
- AS, Rosa dan Shalahuddin, M. 2011. *Rekayasa Perangkat Lunak*. Bandung, Modula.
- Dennis, Alan, Wixom, Barbara H., Tegarden, David. 2005. *System Analysis and Design with UML Version 2.0*. John Wiley and Sons, USA.
- Fowler, Martin. 2004. *UML Distilled: A Brief Guide to Standard Modelling Language. 3th Edition*. Pearson Education, Boston.
- Hayder, Hasin. 2007. *Object Oriented Programming with PHP5*. Packt Publishing, Birmingham.
- Jogiyanto, HM. 2005. *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Edisi Ketiga. Andi Offset, Yogyakarta.
- Lethbridge, Timothy C., Laganier, Robert. 2005. *Object-Oriented Software Engineering. 2nd Edition*. McGraw-Hill, New York.
- McLeod, Raymond. 2004. *Sistem Informasi Manajemen*. PT Indeks, Jakarta
- O'Brien, James A. 2005. *Introduction to Information System. 12th Edition*. McGraw-Hill, Northern Arizona.
- Pressman, Roger S. 2005. *Software Engineering A Practitioner's Approach. 6th Edition*. McGraw Hill, New York.
- PT Showa Indonesia Manufacturing, Pemasaran SIM. <http://www.showa.co.id/customer.php/customer-sim/>, 2012. (Tanggal Akses; 21 Januari 2015)
- PT Showa Indonesia Manufacturing, Profil SIM, http://www.showa.co.id/s_contents.php/profil-showa/, 2012. (Tanggal Akses; 25 November 2015)
- PT Showa Indonesia Manufacturing, Visi-Misi SIM, <http://www.showa.co.id/index.php/visi-misi/>, 2012. (Tanggal Akses; 25 September 2015)
- Schmuller, Joseph. 2004. *Teach Yourself UML in 24 Hours. 3th Edition*. Sams Publishing, United States.

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

Sofyan, Diana Khairani. 2013. *Perencanaan dan Pengendalian Produksi*, Penerbit Graha Ilmu, Yogyakarta

Turban, Efraim, Raimer, J.R., R. Kelly, dan Potter, Richard E. 2006. *Pengantar teknologi informasi*. Edisi Ketiga. Salemba Infotek, Jakarta.

Wahana Komputer. 2010. *Panduan Aplikatif dan Solusi (PAS) Membuat Aplikasi Client Server dengan Visual Basic 2008*. Andi offset, Yogyakarta.

Welling, Luke, dan Thomson, Laura. 2003. *PHP and MySQL Web Development*. Sams Publishing, USA.

Whitten, Jeffrey L., Lonnie D. Bentley. 2004. *Metode desain dan analisis sistem*. Edisi Ketujuh. Andi Offset, Yogyakarta.

1. Menu Utama

```
Imports System.Windows.Forms
Imports System.Data.SqlClient

Public Class home

    Private Sub ShowNewForm(ByVal sender As Object, ByVal e As
EventArgs)
        ' Create a new instance of the child form.
        Dim ChildForm As New System.Windows.Forms.Form
        ' Make it a child of this MDI form before showing it.
        ChildForm.MdiParent = Me

        m_ChildFormNumber += 1
        ChildForm.Text = "Window " & m_ChildFormNumber

        ChildForm.Show()
    End Sub

    Private Sub OpenFile(ByVal sender As Object, ByVal e As EventArgs)
        Dim OpenFileDialog As New OpenFileDialog
        OpenFileDialog.InitialDirectory =
My.Computer.FileSystem.SpecialDirectories.MyDocuments
        OpenFileDialog.Filter = "Text Files (*.txt)|*.txt|All Files
(*.*)|*.*"
        If (OpenFileDialog.ShowDialog(Me) =
System.Windows.Forms.DialogResult.OK) Then
            Dim FileName As String = OpenFileDialog.FileName
            ' TODO: Add code here to open the file.
        End If
    End Sub

    Private Sub SaveAsToolStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        Dim SaveFileDialog As New SaveFileDialog
        SaveFileDialog.InitialDirectory =
My.Computer.FileSystem.SpecialDirectories.MyDocuments
        SaveFileDialog.Filter = "Text Files (*.txt)|*.txt|All Files
(*.*)|*.*"
        If (SaveFileDialog.ShowDialog(Me) =
System.Windows.Forms.DialogResult.OK) Then
            Dim FileName As String = SaveFileDialog.FileName
            ' TODO: Add code here to save the current contents of the
form to a file.
        End If
    End Sub

    Private Sub ExitToolsStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        Me.Close()
    End Sub

    Private Sub CutToolStripMenuItem_Click(ByVal sender As Object, ByVal
e As EventArgs)
```

```

        ' Use My.Computer.Clipboard to insert the selected text or
images into the clipboard
    End Sub

    Private Sub CopyToolStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        ' Use My.Computer.Clipboard to insert the selected text or
images into the clipboard
    End Sub

    Private Sub PasteToolStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        'Use My.Computer.Clipboard.GetText() or
My.Computer.Clipboard.GetData to retrieve information from the
clipboard.
    End Sub

    Private Sub ToolBarToolStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        PR.Show()
    End Sub

    Private Sub StatusBarToolStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        PO.Show()
    End Sub

    Private Sub CascadeToolStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        Me.LayoutMdi(MdiLayout.Cascade)
    End Sub

    Private Sub TileVerticalToolStripMenuItem_Click(ByVal sender As
Object, ByVal e As EventArgs)
        Me.LayoutMdi(MdiLayout.TileVertical)
    End Sub

    Private Sub TileHorizontalToolStripMenuItem_Click(ByVal sender As
Object, ByVal e As EventArgs)
        Me.LayoutMdi(MdiLayout.TileHorizontal)
    End Sub

    Private Sub ArrangeIconsToolStripMenuItem_Click(ByVal sender As
Object, ByVal e As EventArgs)
        Me.LayoutMdi(MdiLayout.ArrangeIcons)
    End Sub

    Private Sub CloseAllToolStripMenuItem_Click(ByVal sender As Object,
ByVal e As EventArgs)
        ' Close all child forms of the parent.
        For Each ChildForm As Form In Me.MdiChildren
            ChildForm.Close()
        Next
    End Sub

    Private m_ChildFormNumber As Integer

```

```

Private Function ToolStrip() As Object
    Throw New NotImplementedException
End Function

Private Sub UserToolStripMenuItem_Click(sender As System.Object, e
As System.EventArgs) Handles UserToolStripMenuItem.Click
    'Try
    Call koneksisub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
        User.MdiParent = Me
        User.Focus()
        User.Show()
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
    Exit Sub
    End If
End Sub

Private Sub PartToolStripMenuItem_Click(sender As System.Object, e
As System.EventArgs) Handles PartToolStripMenuItem.Click
    Call koneksisub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
        Part.MdiParent = Me
        Part.Focus()
        Part.Show()
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
    Exit Sub
    End If
End Sub

Private Sub HakAksesToolStripMenuItem_Click(sender As System.Object,
e As System.EventArgs) Handles HakAksesToolStripMenuItem.Click
    Call koneksisub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
        Hak_Akses.MdiParent = Me
        Hak_Akses.Focus()

```

```

        Hak_Akses.Show()
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
        Exit Sub
    End If
End Sub

Private Sub KalkulasiOrderToolStripMenuItem_Click(sender As
System.Object, e As System.EventArgs) Handles
KalkulasiOrderToolStripMenuItem.Click
    Call koneksisub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
        Kalku_Order.MdiParent = Me
        Kalku_Order.Focus()
        Kalku_Order.Show()
        'ID PEGAWAI
        Call koneksisub()
        Dim str1 As String
        str1 = "select * FROM dt_user3 where id_user = '" &
login.txtusername.Text & "'"
        Cmd = New SqlCommand(str, konek)
        DR = Cmd.ExecuteReader
        DR.Read()
        If DR.HasRows Then
            Kalku_Order.txtiduser3.Text = DR.Item("id_user")
        End If
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
        Exit Sub
    End If
End Sub

Private Sub ForecastingToolStripMenuItem_Click(sender As
System.Object, e As System.EventArgs) Handles
ForecastingToolStripMenuItem.Click
    Call koneksisub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
        Forecasting.MdiParent = Me
        Forecasting.Focus()
        Forecasting.Show()
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
    End If
End Sub

```

```

        Exit Sub
    End If
End Sub

Private Sub ViewPOToolStripMenuItem_Click(sender As System.Object, e
As System.EventArgs)
    Call koneksisub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Mid(DR(6), 2, 1) = "1" Then
        View_PO.MdiParent = Me
        View_PO.Focus()
        View_PO.Show()
    ElseIf Microsoft.VisualBasic.Right(DR(2), 1) = "1" Then
        View_PO.MdiParent = Me
        View_PO.Focus()
        View_PO.Show()
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
    Exit Sub
    End If
End Sub

Private Sub OptionsToolStripMenuItem_Click(sender As System.Object,
e As System.EventArgs) Handles OptionsToolStripMenuItem.Click
    Call koneksisub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Right(DR(6), 1) = "1" Then
        Validasi.MdiParent = Me
        Validasi.Focus()
        Validasi.Show()
        Call koneksisub()
        'id boss aktif
        Dim str1 As String
        str1 = "select * FROM dt_user3 where id_user = '" &
login.txtusername.Text & "'"
        Cmd = New SqlCommand(str, konek)
        DR = Cmd.ExecuteReader
        DR.Read()
        If DR.HasRows Then
            Validasi.txtidpeg.Text = DR.Item("id_user")
        End If

    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
    Exit Sub

```

```

        End If
    End Sub

    Private Sub LaporanPemesananBulananToolStripMenuItem_Click(sender As
System.Object, e As System.EventArgs) Handles
LaporanPemesananBulananToolStripMenuItem.Click
        Call koneksiSub()
        Dim str As String
        str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
        Cmd = New SqlCommand(str, konek)
        DR = Cmd.ExecuteReader
        DR.Read()
        If Microsoft.VisualBasic.Mid(DR(6), 2, 1) = "1" Then
            Lap_Bulanan.MdiParent = Me
            Lap_Bulanan.Focus()
            Lap_Bulanan.Show()
        Else
            MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
            Exit Sub
        End If
    End Sub

    Private Sub PurchaseRequisitionToolStripMenuItem_Click(sender As
System.Object, e As System.EventArgs) Handles
PurchaseRequisitionToolStripMenuItem.Click
        Call koneksiSub()
        Dim str As String
        str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
        Cmd = New SqlCommand(str, konek)
        DR = Cmd.ExecuteReader
        DR.Read()
        If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
            PR.MdiParent = Me
            PR.Focus()
            PR.Show()
        Else
            MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
            Exit Sub
        End If
    End Sub

    Private Sub PurchaseOrderToolStripMenuItem_Click(sender As
System.Object, e As System.EventArgs) Handles
PurchaseOrderToolStripMenuItem.Click
        Call koneksiSub()
        Dim str As String
        str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
        Cmd = New SqlCommand(str, konek)
        DR = Cmd.ExecuteReader
        DR.Read()
        If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
            PO.MdiParent = Me

```

```

        PO.Focus()
        PO.Show()
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
        Exit Sub
    End If
End Sub

Private Sub ViewPOToolStripMenuItem_Click_1(sender As System.Object,
e As System.EventArgs) Handles ViewPOToolStripMenuItem.Click
    Call koneksiSub()
    Dim str As String
    str = "SELECT * FROM dt_user3 WHERE id_user = '" &
login.txtusername.Text & "'"
    Cmd = New SqlCommand(str, konek)
    DR = Cmd.ExecuteReader
    DR.Read()
    If Microsoft.VisualBasic.Left(DR(6), 1) = "1" Then
        View_PO.MdiParent = Me
        View_PO.Focus()
        View_PO.Show()
    ElseIf Microsoft.VisualBasic.Mid(DR(6), 2, 1) = "1" Then
        View_PO.MdiParent = Me
        View_PO.Focus()
        View_PO.Show()
    Else
        MsgBox("Maaf Anda Tidak Punya Akses", vbOKOnly,
"Information")
        Exit Sub
    End If
End Sub

Private Sub LogoutToolStripMenuItem_Click(sender As System.Object, e
As System.EventArgs) Handles LogoutToolStripMenuItem.Click
    Dim popup As DialogResult

    popup = MessageBox.Show("Apakah Anda yakin ingin keluar dari
aplikasi ini?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question)
    If popup = Windows.Forms.DialogResult.Yes Then
        Me.Close()
        login.Show()
    Else
        MessageBox.Show("Selesaikan pekerjaan Anda terlebih
dahulu!")
    End If
End Sub
End Class

```

2. Login

```
Imports System.Data.SqlClient
```

```
Public Class login
```

```

Private Sub login_Load(sender As System.Object, e As System.EventArgs)
Handles MyBase.Load

End Sub
Sub bersih()
txtusername.Text = ""
txtpw.Text = ""
End Sub

Private Sub btnkeluar_Click(sender As System.Object, e As
System.EventArgs) Handles btnkeluar.Click
Me.Close()
End Sub

Private Sub btnlogin_Click(sender As System.Object, e As
System.EventArgs) Handles btnlogin.Click
If txtusername.Text = "" Or txtpw.Text = "" Then
MsgBox("Username & Password tidak boleh kosong!!", vbCritical,
"Info")
txtusername.Focus() : Exit Sub
End If
Call koneksisub()
Cmd = New SqlCommand("select * from dt_user3 where id_user='" &
txtusername.Text & "' and password='" & txtpw.Text & "'", konek)
DR = Cmd.ExecuteReader
DR.Read()
If DR.HasRows Then
MsgBox("Login Berhasil", MsgBoxStyle.Information)
home.Show() 'Menu yang akan ditampilkan setelah berhasil login
Me.Hide()
Else
DR.Close()
MsgBox("Username/Password salah!", MsgBoxStyle.Critical)
txtusername.Focus()
End If
'Call bersihlogin()
konek.Close()
End Sub
End Class

```

3. User

```
Imports System.Data.SqlClient
```

```
Public Class User
```

```

Private Sub load_data()
Call koneksisub()
Dim dSet As New DataSet
Dim strSelect As String = "select id_user, nama_peg, password,
jabatan, alamat, no_tlp from dt_user3"
Dim comm As New SqlCommand(strSelect, konek)
Dim dAdapter As SqlDataAdapter
dAdapter = New SqlDataAdapter(comm)
dSet.Clear()
dAdapter.Fill(dSet, "dt_user3")
dgv_user.DataSource = dSet

```

```

        dgv_user.DataMember = "dt_user3"
        dgv_user.Refresh()
    End Sub
    Private Sub aturdgvuser()
        dgv_user.Columns(0).HeaderText = "Id User"
        dgv_user.Columns(1).HeaderText = "Nama"
        dgv_user.Columns(2).HeaderText = "Password"
        dgv_user.Columns(3).HeaderText = "Jabatan"
        dgv_user.Columns(4).HeaderText = "Alamat"
        dgv_user.Columns(5).HeaderText = "No Telp"
    End Sub

    Private Sub Button4_Click(sender As System.Object, e As
System.EventArgs) Handles ubah.Click
        'ubah
        Dim sql As String
        sql = "UPDATE dt_user3 SET nama_peg='" & txtnmpg.Text & "'" & _
            ", password='" & txtpw2.Text & "'" & _
            ", jabatan='" & txtjabatan.Text & "'" & _
            ", alamat='" & txtalamat.Text & "'" & _
            ", no_tlp='" & txtnotlp.Text & "' WHERE
Id_user='" & txtiduser2.Text & "'"
        Call koneksisub()
        Cmd = New SqlCommand(sql, konek)
        Dim x As String = Cmd.ExecuteNonQuery
        If x = 1 Then
            MessageBox.Show("Data Berhasil diubah")
            txtiduser2.Focus()
            Call bersih()
            Me.load_data()
        Else
            MessageBox.Show("Data Gagal Disimpan")
            txtiduser2.Focus()
            Exit Sub
        End If
        simpan.Enabled = True
    End Sub

    Private Sub Button1_Click(sender As System.Object, e As
System.EventArgs) Handles simpan.Click
        Dim sql As String
        sql = "insert into dt_user3 values ('" & txtiduser2.Text & "','" &
txtnmpg.Text & "','" & txtpw2.Text & "','" & txtjabatan.Text & "','" &
txtalamat.Text & "','" & txtnotlp.Text & "','" )"
        Call koneksisub()
        Cmd = New SqlCommand(sql, konek)
        Dim x As String = Cmd.ExecuteNonQuery
        If x = 1 Then
            MessageBox.Show("Data Berhasil Disimpan")
            txtiduser2.Focus()
            Me.bersih()
            Me.load_data()
        Else
            MessageBox.Show("Gagal Menyimpan")
            txtiduser2.Focus()
            Call bersih()
            Exit Sub
        End If
    End Sub

```

```

        End If
    End Sub
    Public Sub bersih()
        txtiduser2.Text = ""
        txtpw2.Text = ""
        txtnmpg.Text = ""
        txtjabatan.Text = ""
        txtalamat.Text = ""
        txtnotlp.Text = ""
        txtiduser2.Focus()
    End Sub

    Private Sub User_Load(sender As System.Object, e As System.EventArgs)
    Handles MyBase.Load
        'TODO: This line of code loads data into the
        'Tugas_akhirDataSet.dt_user3' table. You can move, or remove it, as needed.
        Me.Dt_user3TableAdapter.Fill(Me.Tugas_akhirDataSet.dt_user3)
        Me.load_data()
        Me.aturdgvuser()
    End Sub

    Private Sub hapus_Click(sender As System.Object, e As System.EventArgs)
    Handles hapus.Click
        Dim sql As String
        Dim popup As DialogResult
        popup = MessageBox.Show("Menghapus Data?", "konfirmasi",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question)
        If popup = Windows.Forms.DialogResult.Yes Then
            sql = "delete from dt_user3 where id_user = '" &
            txtiduser2.Text & "'"
            Call koneksisub()
            Cmd = New SqlCommand(sql, konek)
            Dim x As Integer = Cmd.ExecuteNonQuery()
            If x = 1 Then
                MessageBox.Show("Data Berhasil DiHapus")
                txtiduser2.Focus()
                Call bersih()
                Me.load_data()
            Else
                MessageBox.Show("Gagal Menghapus")
                txtiduser2.Focus()
                'Call bersih()
            End If
        End If
    End Sub
    Private Sub cari_Click(sender As System.Object, e As System.EventArgs)
        'CodeForm = ""
        'Call Tampil_Data()
    End Sub

    Private Sub dgv_user_CellMouseClick(sender As Object, e As
    System.Windows.Forms.DataGridViewCellMouseEventArgs) Handles
    dgv_user.CellMouseClick
        On Error Resume Next
        txtiduser2.Text = dgv_user.Rows(e.RowIndex).Cells(0).Value
        txtpw2.Text = dgv_user.Rows(e.RowIndex).Cells(2).Value
        txtnmpg.Text = dgv_user.Rows(e.RowIndex).Cells(1).Value

```

```

txtjabatan.Text = dgv_user.Rows(e.RowIndex).Cells(3).Value
txtalamat.Text = dgv_user.Rows(e.RowIndex).Cells(4).Value
txtnotlp.Text = dgv_user.Rows(e.RowIndex).Cells(5).Value
simpan.Enabled = False
End Sub
End Class

```

4. Part

```
Imports System.Data.SqlClient
```

```
Public Class Part
```

```

Private Sub load_data()
    'UNTUK MELOAD DATA DI DATA GRID VIEW PART
    Call koneksisub()
    Dim dSet As New DataSet
    Dim strSelect As String = "select * from dt_part2"
    Dim comm As New SqlCommand(strSelect, konek)
    Dim dAdapter As SqlDataAdapter
    dAdapter = New SqlDataAdapter(comm)
    dSet.Clear()
    dAdapter.Fill(dSet, "dt_part2")
    dgv_part.DataSource = dSet
    dgv_part.DataMember = "dt_part2"
    dgv_part.Refresh()
End Sub

```

```
End Sub
```

```
Private Sub aturdgvpart()
```

```

'UNTUK MENGATUR DATA DI DGV
dgv_part.Columns(0).HeaderText = "Id Part"
dgv_part.Columns(1).HeaderText = "Nama"
dgv_part.Columns(2).HeaderText = "Spesifikasi"
dgv_part.Columns(3).HeaderText = "Satuan"
dgv_part.Columns(4).HeaderText = "Grup"
dgv_part.Columns(5).HeaderText = "Stok"
dgv_part.Columns(6).HeaderText = "Lead Time"

```

```
End Sub
```

```
Private Sub Button1_Click(sender As System.Object, e As System.EventArgs) Handles simpan.Click
```

```
'TOMBOL SIMPAN
```

```
Dim sql As String
```

```

sql = "insert into dt_part2 values ('" & txtidpart.Text & "','" &
txtnmpart.Text & "','" & txtspec.Text & "','" & txtuom.Text & "','" &
cbogrup.Text & "','" & txtstok.Text & "','" & txtlead.Text & "')"

```

```
Call koneksisub()
```

```
Cmd = New SqlCommand(sql, konek)
```

```
Dim x As String = Cmd.ExecuteNonQuery
```

```
If x = 1 Then
```

```
    MessageBox.Show("Data Berhasil disimpan")
```

```
    Call bersih()
```

```
    Call load_data()
```

```
Else
```

```
    MessageBox.Show("Data Gagal Disimpan")
```

```
    txtidpart.Focus()
```

```
Exit Sub
```

```
End If
```

```
dgv_part.Refresh()
```

```

End Sub
Private Sub bersih()
    txtidpart.Text = ""
    txtnmpart.Text = ""
    txtspec.Text = ""
    txtstok.Text = ""
    txtuom.Text = ""
    cbogrup.Text = ""
    txtlead.Text = ""
    txtidpart.Focus()
End Sub

Private Sub Button2_Click(sender As System.Object, e As
System.EventArgs) Handles hapus.Click
    'MENGHAPUS DATA
    Dim sql As String
    Dim popup As DialogResult
    'HAPUS DATA
    popup = MessageBox.Show("Menghapus Data?", "konfirmasi",
MessageBoxButtons.YesNo, MessageBoxIcon.Question)

    If popup = Windows.Forms.DialogResult.Yes Then
        sql = "delete from dt_part2 where id_part = '" & txtidpart.Text
& "'"

        Call koneksisub()

        Cmd = New SqlCommand(sql, konek)

        Dim x As Integer = Cmd.ExecuteNonQuery
        If x = 1 Then
            MessageBox.Show("Data Berhasil Dihapus")
            txtidpart.Focus()
            Call bersih()
            Call load_data()
        Else
            MessageBox.Show("Gagal Menghapus")
            txtidpart.Focus()
        End If
    End If
End Sub

Private Sub Part_Load(sender As System.Object, e As System.EventArgs)
Handles MyBase.Load
    'TODO: This line of code loads data into the
'Tugas_akhirDataSet1.dt_part2' table. You can move, or remove it, as
needed.

    Me.Dt_part2TableAdapter.Fill(Me.Tugas_akhirDataSet1.dt_part2)
    'CODE FORM
    Call load_data()
    Call aturdgvpert()
End Sub

Private Sub Button4_Click(sender As System.Object, e As
System.EventArgs) Handles ubah.Click
    'MENGUBAH DATA
    Dim sql As String
    sql = "UPDATE dt_part2 SET nama_part='" & txtnmpart.Text & "'" & _

```

```

                                ", spec='" & txtspec.Text & "'" & _
                                ", uom='" & txtuom.Text & "'" & _
                                ", acc_grup='" & cbogrup.Text & "'" & _
                                ", stok='" & txtstok.Text & "'" & _
                                ", leadtime='" & txtlead.Text & "'" WHERE
Id_Part='" & txtidpart.Text & "'"
    Call koneksisub()
    Cmd = New SqlCommand(sql, konek)
    Dim x As String = Cmd.ExecuteNonQuery
    If x = 1 Then
        MessageBox.Show("Data Berhasil disimpan")
        Call bersih()
        Call load_data()
    Else
        MessageBox.Show("Data Gagal Disimpan")
        txtidpart.Focus()
        Exit Sub
    End If
    simpan.Enabled = True
End Sub

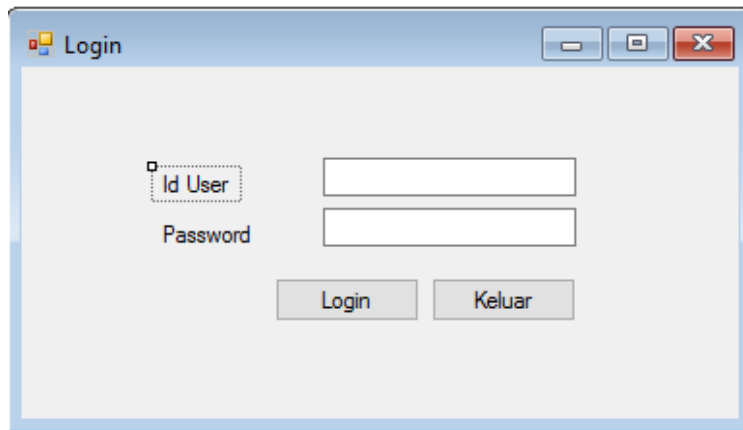
Private Sub DataGridView1_CellContentClick(sender As System.Object, e
As System.Windows.Forms.DataGridViewCellEventArgs) Handles
dgv_part.CellContentClick
    End Sub

Private Sub dgv_part_CellMouseClicked(sender As Object, e As
System.Windows.Forms.DataGridViewCellEventArgs) Handles
dgv_part.CellMouseClicked
    'UNTUK MEMASUKAN DATA KE TXTBOX DARI DGV
    On Error Resume Next
    txtidpart.Text = dgv_part.Rows(e.RowIndex).Cells(0).Value
    txtnmpart.Text = dgv_part.Rows(e.RowIndex).Cells(1).Value
    txtspec.Text = dgv_part.Rows(e.RowIndex).Cells(2).Value
    txtuom.Text = dgv_part.Rows(e.RowIndex).Cells(3).Value
    cbogrup.Text = dgv_part.Rows(e.RowIndex).Cells(4).Value
    txtstok.Text = dgv_part.Rows(e.RowIndex).Cells(5).Value
    txtlead.Text = dgv_part.Rows(e.RowIndex).Cells(6).Value
    simpan.Enabled = False
End Sub
End Class

```

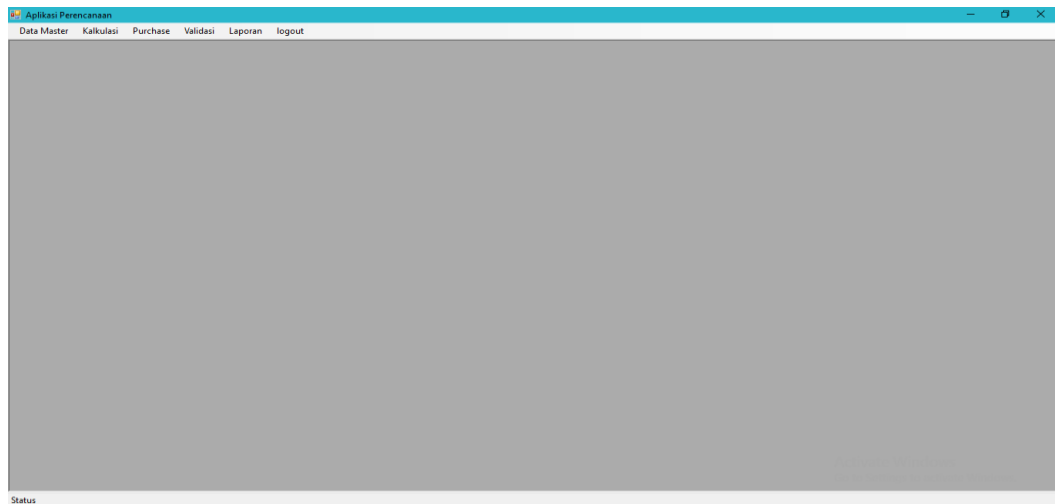
LAMPIRAN

1. Login

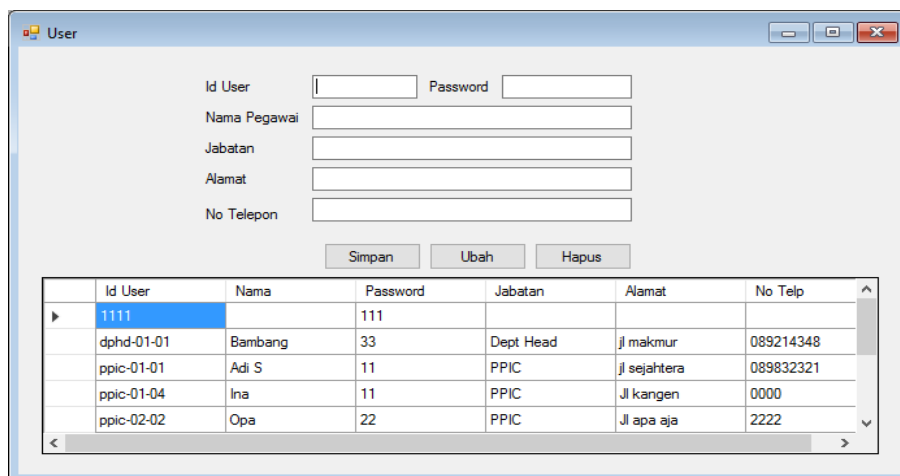


A screenshot of a Windows-style window titled "Login". The window has a light blue title bar with standard minimize, maximize, and close buttons. The main area is light gray and contains two text input fields. The first is labeled "Id User" and the second is labeled "Password". Below the input fields are two buttons: "Login" and "Keluar".

2. Menu Utama



3. User



A screenshot of a Windows-style window titled "User". The window has a light blue title bar with standard window controls. The main area is light gray and contains several text input fields for user information: "Id User", "Password", "Nama Pegawai", "Jabatan", "Alamat", and "No Telepon". Below these fields are three buttons: "Simpan", "Ubah", and "Hapus". At the bottom of the window is a table with the following columns: "Id User", "Nama", "Password", "Jabatan", "Alamat", and "No Telp".

	Id User	Nama	Password	Jabatan	Alamat	No Telp
▶	1111		111			
	dphd-01-01	Bambang	33	Dept Head	jl makmur	089214348
	ppic-01-01	Adi S	11	PPIC	jl sejahtera	089832321
	ppic-01-04	Ina	11	PPIC	Jl kangen	0000
	ppic-02-02	Opa	22	PPIC	Jl apa aja	2222

4. Part

The 'Part' application window contains the following elements:

- Input fields for: Id Part, Nama Part, Spesifikasi, Satuan, Stok, and Lead Time.
- A dropdown menu for Grup.
- Buttons: Simpan, Ubah, Hapus.
- A table with the following data:

	Id Part	Nama	Spesifikasi	Satuan	Grup	Stok	Lead Tim ^
▶	CT-001	Spring	d=30cm	pcs	CT	100	3
	CT-002	Screw	tcgf	pcs	CT	30	3
	CT-003	Milling	RM8ACM	pcs	CT	30	3
	CT-004	Knife	p=50cm	pcs	CT	40	3
	CT-005	Cutter	tcgh	pcs	CT	30	3

5. Hak akses

The 'Hak Akses' application window contains the following elements:

- Input fields for: Id User, Nama Pegawai, and Password.
- Checkboxes for: PPIC, Procurement, and Dept Head Procurement.
- Buttons: Cari, Simpan, Ubah, Hapus.

6. Kalkulasi Order

Kalku_Order

Id Kalkulasi: Id Pegawai:
 Hari Kerja: Bulan:

Id Detail Kalkulasi: Id Part: ...
 Stok Terakhir: Lead Time:

Anggaran:
 Safety Stok: Minimum Stok:
 Reguler Order: Order Tambahan:

Id Part	Stok
CT-001	100
CT-002	30
CT-003	30
CT-004	40
CT-005	30
*	

Proses Simpan Selesai

7. Forecasting

Forecasting

Id Forecasting: Bulan: Id Kalkulasi:
 Id Detail Forecasting: Id Detail Kalkulasi: ...
 Id Part: Order Tambahan:
 Reguler Order: Total:

Proses Simpan Hapus Selesai

Id Detail	Id Forecasting	Id Part	Total	Id Det Kalkulasi
DF-001	FO-003	CT-002	30	DK-006
DF-011	FO-11	CT-003	108	DK-999
DF101601	F101601	CT-001	167	DK-010
DF1116001	FO11160001	CT-001	167	DK-001
*				

Id Part	Order Tambahan
*	

8. Purchase Requisition

Purchase Requisition

Id Purchase Requisition: Id Forecasting:

Tanggal: 03-Nov-16

Id Detail PR: Id Part: ...

Total:

	acc_grup	nama_part	spec	total	uom	tanggal
▶	CT	Milling	RMSACM	108	pcs	03-Nov-16
	CT	Spring	d=30cm	167	pcs	01-Oct-16
	CT	Spring	d=30cm	167	pcs	03-Nov-16
*						

9. Purchase Order

Purchase Order

Id Purchase Order: Id Purchase Requisition:

Tanggal Pesan: 31-Oct-16 Tanggal Datang: 31-Oct-16

Id Detail PO: Id Part: ...

Total:

	Id Det PO	Id PO	Id Part	Stok
▶				0
	DPO-001	PR-001	CT-002	30
	DPO-003	PO-010	CT-002	30
	DPO-11	PO-11	CT-003	108
	DPO101601	PR1016001	CT-001	167
	DPO1116001	PR11160001	CT-001	167
*				

10. View Purchase Order

View_PO

Id Purchase Order:

	id_detailPO	id_PO	id_part	total
▶				0
*				

11. Validasi Purchase Order

Validasi

Id Purchase Order Tanggal 11-Nov-16

Id Pegawai Status

Simpan Cetak

	id_PO	id_PR	validasi	id_user	tgl_psn	tg
*						