

**RANCANG BANGUN SISTEM INFORMASI PACKAGING
PLAN BEARING MENGGUNAKAN PHP 5.6.3 DAN MYSQL
5.6.21 PADA PT SKF INDONESIA**

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Penyelesaian
Jenjang Diploma Empat (D-4) Program Studi Sistem Informasi Industri Otomotif
Pada Politeknik STMI Jakarta

**OLEH
ANDRI SEFRIYAN
1312057**



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2017**

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

TANDA PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI *PACKAGING*
PLAN BEARING MENGGUNAKAN PHP 5.6.3 DAN
MYSQL 5.6.21 PADA PT SKF INDONESIA

Disusun Oleh

Nama : Andri Sefriyan
Nim : 1312057
Program Studi : Sistem Informasi Industri Otomotif

Jakarta, 26 Oktober 2017

Menyetujui
Pembimbing

Dedy Trisanto, S.Kom, MMSI
NIP. 197805052005021002

**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA**

TANDA PERSETUJUAN ASISTEN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI *PACKAGING*
PLAN BEARING MENGGUNAKAN PHP 5.6.3 DAN
MYSQL 5.6.21 PADA PT SKF INDONESIA

Disusun Oleh

Nama : Andri Sefriyan
Nim : 1312057
Program Studi : Sistem Informasi Industri Otomotif

Jakarta, 1 November 2017

Menyetujui
Pembimbing

Triana Fatmawati, ST, MT
NIP. 198005142005022001

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PACKAGING PLAN BEARING MENGGUNAKAN
PHP 5.6.3 DAN MYSQL 5.6.21 PADA PT SKF
INDONESIA

Disusun Oleh :
Nama : Andri Sefriyan
Nim : 1312057
Program Studi : Sistem Informasi Industri Otomotif

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem
Informasi Industri Otomotif Politeknik STMI Jakarta Kementerian Perindustrian
Republik Indonesia pada hari Selasa tanggal 21 November 2017

Jakarta, 27 November 2017

Dosen Pembimbing

Ketua Penguji

Dedy Trisanto, S.Kom, MMSI
NIP. 197805052005021002

Ahmad Juniar, S.kom, MT
NIP. 197906052006041002

Dosen Penguji

Dosen Penguji

Noveriza Yuliasari, Ssi, MT
NIP. 197811212009012003

Ulil Hamida, ST, MT
NIP. 198103272005022001

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

Judul Tugas Akhir : RANCANG BANGUN SISTEM INFORMASI
PACKAGING PLAN BEARING MENGGUNAKAN
PHP 5.6.3 DAN MYSQL 5.6.21 PADA PT SKF
INDONESIA

Disusun Oleh :

Nama : Andri Sefriyan

Nim : 1312057

Program Studi : Sistem Informasi Industri Otomotif

Tanggal Seminar : 07 November 2017

Tanggal Sidang : 21 November 2017

Tanggal Lulus : 21 November 2017

Jakarta, 27 November 2017

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 SISTEM INFORMASI
PACKAGING PLAN BEARING MENGGUNAKAN
PHP 5.6.3 DAN MYSQL 5.6.21 PADA PT SKF
INDONESIA

Disusun Oleh :
Nama : Andri Sefriyan
Nim : 1312057
Program Studi : Sistem Informasi Industri Otomotif
Tanggal Seminar : 07 November 2017
Tanggal Sidang : 21 November 2017
Tanggal Lulus : 21 November 2017

Jakarta, 27 November 2017

Menyetujui
Asisten Dosen Pembimbing

Triana Fatmawati, ST, MT
NIP. 198005142005022001

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Andri Sefriyan

NIM : 1312057

Judul Tugas Akhir : Rancang Bangun Sistem Informasi Packaging Plan Bearing Menggunakan PHP 5.6.3 Dan MySQL 5.6.21 Pada PT SKF Indonesia

Pembimbing : Dedy Trisanto, S.Kom.MMSI

Asisten Pembimbing : Triana Fatmawati, ST, MT

Tanggal	BAB	Keterangan	Paraf
28/7/2017	Bab I,II dan	Menyerahkan Proposal TA	
07/8/2017	III	Revisi Bab I, II dan III	
010/8/2017	Bab I dan II	Menyerahkan revisi Bab I dan Bab II	
	Bab I, II, III	Menyerahkan Bab III dan Bab IV	
15/8/2017	dan IV	Menyerahkan revisi bab III dan IV	
24/8/2017	Bab III dan	Menyerahkan Bab V (<i>flowmap</i> dan <i>use case</i>)	
28/8/2017	IV		
05/9/2017	Bab V	Menyerahkan revisi <i>flowmap</i> dan <i>use case</i>	
11/9/2017	Bab V		
17/10/2017	Bab V	Menyerahkan revisi <i>flowmap</i> dan <i>use case</i>	
26/10/2017	Bab V		
	Bab V dan	Menyerahkan Bab V (keseluruhan) dan	
	VI	Bab VI	
	Bab I -	Revisi Bab VI	
	BabVI	Menyerahkan keseluruhan laporan	

Mengetahui, Ketua Program Studi Sistem Informasi Industri Otomotif	Dosen Pembimbing
<u>Drs. Jacob Saragih, MM</u> NIP. 195404281986031002	<u>Dedy Trisanto,S.Kom,MMSI</u> NIP. 197805052005021002

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Andri Sefriyan

Nim : 1312057

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

“RANCANG BANGUN SISTEM INFORMASI PACKAGING PLAN PADA PT SKF INDONESIA DENGAN MENGGUNAKAN PHP 5.6.3 DAN MYSQL 5.6.21”

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

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

Jakarta, 20 Oktober 2017

Andri Sefriyan

ABSTRAK

PT SKF Indonesia adalah perusahaan yang bergerak di bidang manufaktur, yang kegiatannya adalah memproduksi *bearing* untuk kendaraan roda dua, kendaraan roda empat dan mesin. PT SKF Indonesia memiliki beberapa departemen yang masing-masing menentukan keberhasilan produksi. Salah satu departemen yang dimiliki adalah Departemen *Shipping* yang bertanggung jawab mulai dari menerima produk jadi dari Departemen Produksi, pengemasan produk, pengiriman produk, hingga mengeluarkan surat jalan untuk pengiriman ke pelanggan. Kegiatan ini disebut sebagai *Packaging Plan*, dimana proses tersebut dilakukan untuk menjadwalkan jumlah produk yang akan dikemas sebelum dikirim ke pelanggan. Selama ini, *Packaging Plan* pada Departemen *Shipping* masih menggunakan *Microsoft Excel* yang beresiko terjadi kesalahan dalam menginput jadwal pengemasan produk sehingga sering terjadi keterlambatan pengiriman produk ke pelanggan. Namun kendala tersebut dapat diatasi dengan membangun sebuah sistem informasi yang dapat membantu proses *Packaging Plan* menjadi lebih mudah, tepat dan cepat. Sistem informasi tersebut dinamakan sebagai Sistem Informasi *Packaging Plan* dan dibangun dengan menggunakan metode pengembangan sistem *evolutionary prototype*. Sistem informasi ini dirancang dengan menggunakan *Unified Modelling Language* (UML) untuk pemodelan sistem, PHP 5.6.3 sebagai bahasa pemrograman dan MySQL 5.6.21 sebagai basis datanya. Dengan adanya Sistem Informasi *Packaging Plan* ini, maka tidak akan terjadi kesalahan penginputan yang mengakibatkan keterlambatan pengiriman produk. Selain itu, pembuatan laporan pengemasan produk pada Departemen *Shipping* menjadi lebih berdaya guna dan tepat guna.

Kata Kunci: Sistem Informasi, *Packaging Plan*, *Schedule Delivery*, *Evolutionary Prototype*, *Unified Modeling Language*, PHP 5.6.3, MySQL 5.6.21.

KATA PENGANTAR

Alhamdulillah segala puji kepada Allah Subhanahu wa Ta'ala atas segala nikmat dan pertolongan yang telah Ia berikan kepada penulis sehingga penulis dapat menyelesaikan Laporan Tugas Akhir dengan judul “RANCANG BANGUN SISTEM INFORMASI *PACKAGING PLAN BEARING* MENGGUNAKAN PHP 5.6.3 DAN MYSQL 5.6.21 PADA PT SKF INDONESIA”. Laporan Tugas Akhir ini diajukan untuk memenuhi salah satu syarat penyelesaian Jenjang Studi D-4 Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

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

1. Abati Syofyan, Umi tercinta Priyanti, Kakak Rani dan Rinda yang selalu mendoakan kebaikan, selalu mendukung dalam materi serta moral, dan memberikan kasih sayang dalam keadaan senang ataupun sulit selama menyelesaikan Tugas Akhir.
2. Dr. Mustofa, ST, MT, selaku Direktur Politeknik STMI.
3. Bapak Drs. Jacob Saragih, MM dan Ibu Noveriza Yuliasari, S.Si, MT selaku Ketua dan Sekretaris Program Studi Sistem Informasi Industri Otomotif.
4. Bapak Dedy Trisanto, S.Kom. MMSI selaku dosen Pembimbing yang telah memberikan ilmunya dan membimbing penulis selama penyelesaian Laporan Tugas Akhir.
5. Ibu Triana Fatmawati, ST, MT, selaku asisten dosen Pembimbing yang telah membimbing dan memberikan ilmunya selama penyelesaian Laporan Tugas Akhir.
6. Bapak Hendri Lispriyogi selaku Supervisor di Bagian *Shipping* Pada PT SKF Indonesia yang telah mengizinkan penulis melakukan penelitian dan penulisan Praktik Kerja Lapangan di perusahaan tersebut.

7. Bapak Heri Sisoko yang telah membimbing dan membantu dalam pengumpulan data dalam menjalankan Praktik Kerja Lapangan.
8. Seluruh karyawan di PT SKF Indonesia atas bantuan yang telah diberikan selama berlangsungnya kegiatan Praktik Kerja Lapangan.
9. Seluruh dosen Politeknik STMI yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
10. Kepada teman-teman “Group SiJakRang” Feri, Narto Arief, Gifari atas dukungan dan kebersamaannya selama masa praktek kerja kuliah di PT SKF Indonesia.
11. Seluruh rekan-rekan mahasiswa Politeknik STMI Jakarta Program Studi Sistem Informasi 2012 dan Fitri yang telah memberikan semangat, motivasi dan ilmu kepada penulis sehingga Tugas Akhir ini dapat diselesaikan
12. Serta semua pihak yang baik langsung maupun tidak langsung memberikan kritik, saran dan bantuan dalam pembuatan laporan ini yang tidak bisa penulis sebutkan satu per satu.

Demikianlah penulis berharap semoga laporan ini dapat dijadikan bahan kajian, walaupun dari pemikiran ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran guna perbaikan yang bersifat membangun sangat penulis harapkan. Akhir kata, penulis berharap semoga tulisan ini dapat bermanfaat.

Jakarta, 26 Oktober 2017

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN TUGAS AKHIR	
LEMBAR BIMBINGAN TUGAS AKHIR	
LEMBAR PERNYATAAN KEASLIAN	
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan	2
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat Tugas Akhir	3
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Pengertian sistem	6
2.1.1 Karakteristik Sistem	7
2.1.2 Klasifikasi Sistem	9
2.2 Pengertian Informasi	11
2.2.1 Siklus Informasi	11
2.2.2 Kualitas Informasi	12
2.3 Pengertian Sistem Informasi	12
2.3.1 Komponen sistem informasi	13

2.4	Pengertian packaging	14	
2.5	Pengertian Analisis sistem	15	
2.6	<i>System Development Life Cycle (SDLC)</i>	16	
2.6.1	<i>Model Prototype</i>	18	
2.6.2	<i>Evolutionary Prototype</i>	19	
2.7	Bagan Alir (<i>Flowchart</i>)	20	
2.8	Analisis dan Perancangan Berorientasi Objek	23	
2.9	<i>Unified Modeling Language</i>	23	
2.9.1	<i>Use Case Diagram</i>	25	
2.9.2	<i>Class Diagram</i>	26	
2.9.3	<i>Sequence Diagram</i>	27	
2.9.4	<i>Activity Diagram</i>	29	
2.9.5	<i>Deployment Diagram</i>	29	
2.10	Kamus Data	30	
2.11	Database.....	31	2.12 Hi
2.13	XAMPP	32	
2.14	PHP	33	
2.15	MySQL	34	
BAB III	METODOLOGI PENELITIAN	37	
3.1	Metodologi Penelitian	37	
3.2	Jenis Dan Sumber Data	38	
3.3	Metode Pengumpulan Data	38	
3.4	Metode Pengembangan Sistem	39	
3.5	Kerangka Penelitian	40	
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA	44	
4.1	Pengumpulan Data	44	
4.1.1	Sejarah Dan Profil Perusahaan	44	
4.2	Produk yang dihasilkan	50	

4.3	Daftar Pelanggan	51
4.4	Sistem Packaging Plan yang sedang berjalan	52
4.5	Sistem <i>Packaging Plan</i> dengan <i>Unified Modeling Language</i> ..	57
BAB V	ANALISIS DAN PEMBAHASAN	60
5.1	Analisis Kebutuhan Sistem	60
5.2	Analisis Kebutuhan Rinci Sistem	61
5.3	Prosedur Sistem Informasi <i>Packaging Plan</i> Usulan	63
5.4	Model Sistem yang Digunakan	66
5.4.1	<i>Use Case Diagram</i>	66
5.4.2	<i>Activity Diagram</i>	72
5.4.3	<i>Sequence Diagram</i>	81
5.4.4	<i>Class Diagram</i>	87
5.4.5	Kamus Data	88
5.4.6	<i>Deployment Diagram</i>	92
5.5	Analisis Desain Program	93
5.5.1	HIPO (<i>Hierarchy plus Input-Process-Output</i>).....	93
5.5.2	<i>Flowchart</i> Program.....	94
5.5.3	Perancangan <i>Interface</i> Program.....	96
5.6	Implementasi Sistem <i>Software</i> dan <i>Hardware</i>	111
BAB VI	KESIMPULAN DAN SARAN	113
6.1	Kesimpulan	113
6.2	Saran	113
	DAFTAR PUSTAKA	114
	LAMPIRAN	L-1

DAFTAR GAMBAR

	Halaman
Gambar II.1 Siklus Informasi	11
Gambar II.2 Ilustrasi Model <i>Prototype</i>	18
Gambar II.3 <i>Evolutionary Prototype Model</i>.....	20
Gambar II.4 <i>Deployment Diagram</i>.....	30
Gambar III.1 .. Kerangka Penelitian.....	43
Gambar IV.1 Struktur Organisasi SKF	48
Gambar IV.2 Struktur Organisasi Departemen	49
Gambar IV.3 Prosedur <i>Packaging Plan</i>	54
Gambar IV.4 <i>Purchase Order</i>	55
Gambar IV.5 <i>Schedule Delivery</i>	56
Gambar IV.6 <i>Document PP</i>	57
Gambar IV.7 <i>Usecase Diagram Sistem Packaging Plan</i>	58
Gambar V.1 Prosedur <i>Packaging Plan</i>	65
Gambar V.2 <i>Usecase Diagram Sistem Packaging Plan</i>	66
Gambar V.3 <i>Activity Digram Mengelola Data Master</i>	73
Gambar V.4 <i>Activity Digram Mengelola Data Purchase Order</i>	75
Gambar V.5 <i>Activity Digram Membuat Schedule Delivery</i>	77
Gambar V.6 <i>Activity Digram Membuat Packaging Plan</i>	79
Gambar V.7 <i>Activity Digram Menyetujui Packaging Plan</i>	80
Gambar V.8 <i>Activity Digram Mencetak Packaging Plan</i>	81
Gambar V.9 <i>Sequence Diagram Mengelola Data Master</i>	82
Gambar V.10 <i>Sequence Diagram Mengelola Data PO</i>	83
Gambar V.11 <i>Sequence Diagram Membuat Data Schedule Delivery</i>	84
Gambar V.12 <i>Sequence Diagram Membuat Data Packaging Plan</i>	85
Gambar V.13 <i>Sequence Diagram Menyetujui Data Packaging Plan</i>	86
Gambar V.14 <i>Sequence Diagram Mencetak Data Packaging Plan</i>	86
Gambar V.15 <i>Class Diagram Usulan</i>	87

Gambar V.16	<i>Deployment Diagram</i>	93
Gambar V.17	<i>HIPO Sistem Informasi Packaging Plan</i>	94
Gambar V.18	<i>Flowchart Perancangan Sistem Packaging plan</i>	95
Gambar V.19	<i>Interface Form Login</i>	96
Gambar V.20	<i>Interface Form Menu Utama</i>	97
Gambar V.21	<i>Interface Form Data Master User</i>	97
Gambar V.22	<i>Interface Form Input Data Master User</i>	98
Gambar V.23	<i>Interface Form Data Master Product</i>	99
Gambar V.24	<i>Interface Form Input Data Master Product</i>	100
Gambar V.25	<i>Interface Form Data Customer</i>	101
Gambar V.26	<i>Interface Form Input Data Customer</i>	102
Gambar V.27	<i>Interface Form Data Operator</i>	103
Gambar V.28	<i>Interface Form Input Data Operator</i>	104
Gambar V.29	<i>Interface Form Data Purchase Order</i>	105
Gambar V.30	<i>Interface Form Data Purchase Order Detail</i>	106
Gambar V.31	<i>Interface Form Data Schedule Delivery</i>	107
Gambar V.32	<i>Interface Form Data Schedule Delivery Detail</i>	108
Gambar V.33	<i>Interface Form Data Packaging Plan</i>	109
Gambar V.34	<i>Interface Form Data Packaging Plan Detail</i>	110
Gambar V.35	<i>Interface Form Data Persetujuan</i>	111

DAFTAR TABEL

	Halaman
Tabel II.1 Simbol-Simbol <i>Flowchart</i>	21
Tabel II.2 Simbol-Simbol <i>Flowchart (lanjutan)</i>	22
Tabel II.3 Konsepsi Dasar UML.....	24
Tabel II.4 Notasi <i>Usecase</i> Diagram	25
Tabel II.5 Notasi <i>Usecase</i> Diagram (<i>Lanjutan</i>).....	26
Tabel II.6 Notasi <i>Class</i> Diagram	27
Tabel II.7 Notasi <i>Sequence</i> Diagram	27
Tabel II.8 Notasi <i>Sequence</i> Diagram (Lanjutan)	28
Tabel II.9 Notasi <i>Activity</i> Diagram	28
Tabel II.10 Beberapa Tipe Pada MySQL	36
Tabel IV.1 Daftar Pelanggan PT SKF Indonesia	52
Tabel IV.2 Prosedur <i>Packaging Plan</i>	54
Tabel IV.3 Definisi Aktor Dalam Sitem <i>Packaging Plan</i>	58
Tabel IV.4 Definisi Aktor Dalam Sitem <i>Packaging Plan (lanjutan)</i>	58
Tabel IV.5 Definisi <i>Usecase</i> Dalam Sistem <i>Packaging Plan</i>	59
Tabel V.1 Identifikasi Kebutuhan Sistem	60
Tabel V.2 Kebutuhan Rincian Sistem	62
Tabel V.3 Definisi Aktor <i>Usecase</i> Diagram Sistem Usulan.....	67
Tabel V.4 Definsi <i>Usecase</i> Diagram Sistem Usulan	67
Tabel V.5 Skenario <i>Usecase</i> Mengelola Data Master	68
Tabel V.6 Skenario <i>Usecase</i> Menginput <i>Purchase Order</i>	69
Tabel V.7 Skenario <i>Usecase</i> Membuat <i>Schedule Delivery</i>	69
Tabel V.8 Skenario <i>Usecase</i> Membuat <i>Packaging Plan</i>	70
Tabel V.9 Skenario <i>Usecase</i> Menyetujui <i>Packaging Plan</i>	71
Tabel V.10 Skenarior <i>Usecase</i> Menetak <i>Packaging Plan</i>	71
Tabel V.11 Tabel <i>User</i>	88
Tabel V.12 Tabel Produk	88

Tabel V.13	Tabel <i>Customer</i>	89
Tabel V.14	Tabel Operator	89
Tabel V.15	Tabel <i>Purchase Order</i>	89
Tabel V.16	Tabel <i>Purchase Order Detail</i>	90
Tabel V.17	Tabel <i>Schedule Delivery</i>	91
Tabel V.18	Tabel <i>Schedule Delivery Detail</i>	91
Tabel V.19	Tabel <i>Packaging Plan</i>	92
Tabel V.20	Tabel <i>Packaging Plan Detail</i>	92

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dewasa ini perkembangan teknologi komputer mengalami kemajuan yang demikian pesat. Hal itu dapat dilihat dari mengecilnya ukuran komputer, cepatnya operasi komputer dan mudahnya pengoperasian komputer. Perkembangan teknologi tersebut memicu pengembangan aplikasi-aplikasi baru untuk memudahkan pekerjaan manusia. Salah satu aplikasi yang dikembangkan adalah aplikasi sistem informasi.

Sistem informasi merupakan suatu aplikasi yang dibangun untuk mempermudah penyajian informasi kepada pemakainya. Sistem informasi banyak dimanfaatkan oleh perusahaan-perusahaan, baik perusahaan industri manufaktur ataupun jasa. Sistem informasi digunakan sebagai alat untuk mendukung pengambilan keputusan dan membantu membuat sebuah produk dengan lebih berdaya guna dan tepat guna di dalam perusahaan.

PT SKF Indonesia adalah perusahaan yang bergerak di bidang manufaktur, yang kegiatannya adalah memproduksi *bearing* untuk kendaraan roda dua, kendaraan roda empat dan mesin. PT SKF Indonesia terbagi ke dalam beberapa departemen. Masing-masing departemen akan menentukan keberhasilan proses produksi. Salah satu departemen yang dimiliki adalah Departemen *Shipping* yang bertanggung jawab mulai dari menerima produk jadi dari Departemen Produksi, pengemasan produk, pengiriman produk, hingga mengeluarkan surat jalan untuk pengiriman ke pelanggan.

Pada Departemen *Shipping* PT SKF Indonesia terdapat sistem pengolahan data. Pengolahan data pada *Shipping*, meliputi pengemasan produk dan pengiriman produk namun dalam pembahasan ini hanya melakukan penelitian terhadap pengemasan produk khususnya untuk *schedule delivery* dan *packaging plan*.

Dalam menjalankan kegiatannya, Departemen *Shipping* tidak menggunakan aplikasi yang khusus untuk *schedule delivery*. Selama ini, pembuatan *packaging plan* pada Departemen *Shipping* masih manual yaitu dengan melakukan pencatatan data pada lembar *packaging plan* sehingga sering terjadi kesalahan dalam melakukan pencatatan data. Penginputan *schedule delivery* masih menggunakan *microsoft excel* sehingga sering terjadi kesalahan penginputan yang mengakibatkan kekurangan jumlah produk yang akan dikirim ke pihak pelanggan. Hal ini menyebabkan terlambatnya pengiriman produk ke pelanggan. Penyimpanan dokumen diletakkan pada rak dengan cara ditumpuk dan dicampur dengan dokumen lain sehingga masih sering terjadinya kehilangan data.

Untuk mengatasi masalah tersebut, diperlukan pengembangan sistem informasi dengan membuat *tools* untuk membantu perusahaan dalam memperbaiki masalah tersebut. Adapun judul

Tugas Akhir ini adalah “ RANCANG BANGUN SISTEM INFORMASI *PACKAGING PLAN BEARING* PADA PT SKF INDONESIA DENGAN MENGGUNAKAN PHP 5.6.3 DAN MYSQL 5.6.21”.

1.2 Pokok Permasalahan

Permasalahan yang dapat diidentifikasi dari sistem *packaging plan* yang sedang berjalan pada PT SKF Indonesia adalah:

1. Penginputan *schedule delivery* masih menggunakan *microsoft excel* sehingga sering terjadi kesalahan penginputan yang mengakibatkan kekurangan jumlah produk yang akan dikirim ke pihak pelanggan. Kekurangan jumlah produk ini menyebabkan terlambatnya pengiriman produk ke pelanggan karena tidak sesuai dengan jumlah yang dipesan pelanggan.
2. Pembuatan *packaging plan* masih manual yaitu dengan melakukan pencatatan data pada lembar *packaging plan*, sehingga sering terjadi kesalahan penulisan dalam melakukan pencatatan data.
3. Penyimpanan dokumen diletakkan pada rak dengan cara ditumpuk dan dicampur dengan dokumen lain sehingga masih sering terjadinya kehilangan data.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini pada Departemen *Shipping* PT SKF Indonesia adalah sebagai berikut:

1. Membuat sistem informasi dimana penginputan data *schedule delivery* dapat masuk ke dalam sistem untuk mempermudah melakukan penginputan sehingga dapat mengurangi terjadinya kesalahan penginputan jumlah produk.
2. Membuat sistem informasi *packaging plan* dimana pencatatan data perencanaan pengemasan dapat diinput ke dalam sistem agar lebih cepat dan tepat dalam proses pembuatannya.
3. Membuat sistem basis data untuk *packaging plan* yang dapat mempermudah proses penyimpanan data dan dapat mencegah terjadinya kehilangan data karena setiap data yang dimasukkan ke dalam sistem telah tersimpan dengan aman menggunakan *database MySQL*.

1.4 Batasan Masalah

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

1. Penelitian dilakukan di Departemen *Shipping* pada PT SKF Indonesia, Cakung, Jakarta Timur.
2. Melakukan penelitian terhadap pembuatan *Packaging Plan* dan *Schedule Delivery*
3. Sistem informasi yang dibangun hanya mencakup pada Departemen *Shipping* dan *Packing*.

1.5 Manfaat Tugas Akhir

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

1. Bagi perusahaan
 - a. Untuk menyelesaikan permasalahan yg terjadi di perusahaan tersebut dengan merancang dan membangun aplikasi *Packaging Plan*.
 - b. Memberikan pengambilan keputusan, sehingga dapat lebih berdaya guna dan tepat guna.
2. Bagi mahasiswa
 - a. Memberikan kemampuan dalam mengaplikasikan teori secara jelas terhadap masalah yang diamati.
 - b. Memberikan wawasan dan pengalaman kepada mahasiswa dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi permasalahan.
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

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

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Bab ini menjelaskan tentang berbagai teori yang diperoleh dari buku-buku literatur ataupun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan pada laporan ini adalah seputar analisis, sistem, analisis sistem, *packaging*, *flowchart*, analisis dan perancangan berorientasi objek, *Unified Modeling Language* (UML), HIPO, Kamus Data, MySQL, dan *Personal Home Page* (PHP).

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metode ilmiah dalam mencari, mengembangkan, dan menguji kebenaran tentang suatu pengetahuan. Selain itu dijelaskan pula kerangka pemecahan masalah yang menguraikan tahap-tahap untuk mengatasi permasalahan yang telah diidentifikasi.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini menguraikan data tentang profil perusahaan, produk yang dihasilkan, data pelanggan dan sistem yang sedang berjalan.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini merancang bangun sistem informasi dan menganalisis aplikasi sistem *Packaging Plan* yang digunakan di PT SKF Indonesia.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari hasil rancang bangun sistem informasi *Packaging Plan* dan saran-saran sebagai masukan bagi perusahaan.

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Sistem berasal dari bahasa Latin “*systēma*” atau bahasa Yunani “*sustēma*” yang berarti suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi, atau energi. Istilah sistem merupakan istilah dari bahasa Yunani, *system* yang artinya adalah himpunan bagian atau unsur yang saling berhubungan secara teratur untuk mencapai tujuan bersama. Pengertian sistem menurut para ahli dapat didefinisikan sebagai berikut:

1. Sistem adalah kumpulan elemen yang saling berhubungan dan saling berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama (Sutarman, 2010).
2. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi (Jogiyanto, 2012).
3. Dikutip dari buku Bambang Hartono (2013), menurut Bonita J. Campbel (1979) dalam buku *Understanding Information System: Foundations for Control* menegaskan bahwa sistem adalah “*any group of interrelated components or part which function together to achieve goal*” berarti sistem adalah sehimpunan bagian-bagian atau komponen yang saling berkaitan dan secara bersama-sama berfungsi atau bergerak untuk mencapai suatu tujuan.
4. Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama-sama. Secara garis besar, sebuah sistem informasi terdiri atas tiga komponen utama. Ketiga komponen tersebut mencakup *software*, *hardware* dan *brainware*. Ketiga komponen ini saling berkaitan satu sama lain (Pratama, 2014).

Terdapat dua kelompok pendekatan dalam mendefinisikan sistem, yaitu:

- a. Pendekatan sistem yang lebih menekankan pada prosedur, mendefinisikan sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan. Berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.
- b. Pendekatan yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai suatu kumpulan dari elemen-elemen yang saling berinteraksi untuk mencapai suatu tujuan tertentu.

Berdasarkan beberapa pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa sistem adalah kumpulan elemen-elemen yang terhubung, berinteraksi dan saling mempengaruhi untuk mencapai suatu tujuan.

2.1.1 Karakteristik Sistem

Menurut Sutabri (2012), model umum sebuah sistem adalah input, process, 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. Adapun karakteristik yang dimaksud adalah sebagai berikut:

1. **Komponen Sistem (*Components*)**
 Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu subsistem. Setiap subsistem memiliki sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar, yang disebut dengan Supra Sistem.
2. **Batasan Sistem (*Boundary*)**
 Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan yang tidak dapat dipisah-pisahkan.
3. **Lingkungan Luar Sistem (*Environtment*)**
 Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem.

Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Dengan demikian, lingkungan luar tersebut harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus dikendalikan, jika tidak akan mengganggu kelangsungan hidup sistem tersebut.

4. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

5. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer dan “data” adalah *signal input* yang akan diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal yang menjadi input bagi subsistem lain.

7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Jika suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.2 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara suatu bentuk integrasi antara satu komponen dengan komponen lainnya karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandang, diantaranya (Sutabri, 2012):

1. Sistem Abstrak (*Abstract System*) dan Sistem Fisik (*Physical System*)
 Sistem abstrak (*abstract system*) adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan, sedangkan sistem fisik (*physical system*) merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penggajian, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.
2. Sistem Alamiah (*Natural System*) dan Sistem Buatan Manusia (*Human Made System*)
 Sistem alamiah (*natural system*) adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sedangkan sistem buatan manusia (*human made system*) merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contoh *human machine* karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.
3. Sistem Deterministik (*Deterministic System*) dan Sistem Probabilistik (*Probabilistic System*)
 Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang

dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilistik.

4. Sistem Terbuka (*Closed System*) dan Sistem Tertutup (*Open System*)

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

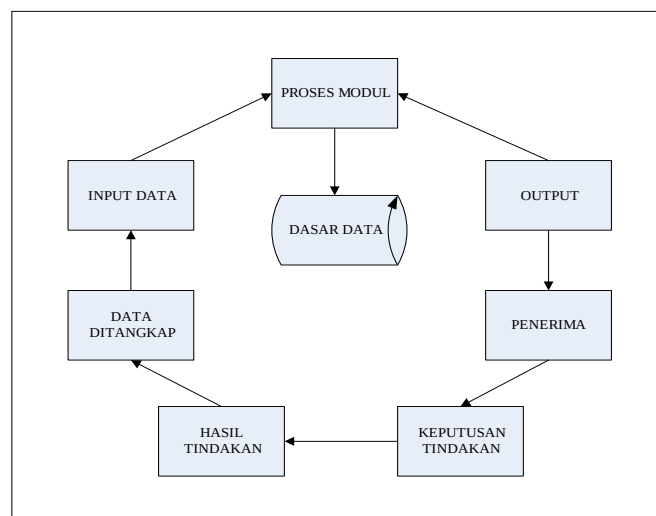
2.2 Pengertian Informasi

Menurut Sutabri (2012), informasi adalah sebuah istilah yang tepat dalam pemakaian umum, mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi, dan lain sebagainya. Informasi juga mencakup mengenai data yang telah diklasifikasikan atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

Menurut Bodnar dan Hopwood (2010) informasi adalah data yang berguna dan telah diolah sehingga dapat dijadikan dasar untuk mengambil keputusan yang tepat. Menurut Raymond Mc.leod informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi penerima dan bermanfaat bagi pengambilan keputusan saat ini atau mendatang. Jadi informasi adalah data yang telah diolah, sehingga lebih berguna, berarti, dan dapat dijadikan dasar untuk mengambil keputusan yang tepat oleh orang yang menerimanya.

2.2.1 Siklus Informasi

Data diolah melalui suatu model informasi, kemudian si penerima akan menangkap informasi tersebut untuk membuat suatu keputusan dan melakukan tindakan. Tindakan si penerima menjadi sebuah data baru. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model, dan seterusnya sehingga membentuk suatu siklus. Siklus inilah yang disebut Siklus Informasi (*Information Cycle*) yang dapat dijelaskan pada gambar II.1 (Sutabri, 2012).



Gambar II.1 Siklus Informasi
(Sumber: Sutabri, 2012)

2.2.2 Kualitas informasi

Menurut Sutabri (2012), kualitas dari suatu informasi (*quality of information*) tergantung dari tiga hal, yaitu informasi harus akurat (*accurate*), tepat pada waktunya (*timeliness*) dan relevan (*relevance*).

1. Akurat (*Accurate*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Informasi harus akurat karena biasanya dari sumber informasi sampai penerima informasi ada kemungkinan terjadi gangguan (*noise*) yang dapat mengubah atau merusak informasi tersebut.

2. Tepat Waktu (*Timeliness*)

Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan di dalam pengambilan keputusan. Bila pengambilan keputusan tersebut terlambat, maka dapat berakibat fatal untuk organisasi.

3. Relevan (*Relevance*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi untuk tiap-tiap orang satu dengan lainnya berbeda.

2.3 Pengertian Sistem Informasi

Menurut Sutabri (2012), suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. Jadi, Sistem Informasi adalah suatu sistem atau proses kumpulan fakta yang dilihat, yang kemudian dicatat dalam suatu data, kemudian data tersebut diolah menjadi suatu informasi yang bermanfaat bagi yang membutuhkan.

Sebagai tambahan terhadap pendukung pengambilan keputusan, koordinasi, dan kendali, sistem informasi dapat juga membantu para manajer dan karyawan untuk meneliti permasalahan, memvisualisasikan pokok-pokok yang kompleks, dan menciptakan produk-produk baru.

2.3.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, komponen teknologi, blok basis data, dan blok kendali. Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran. Berikut ini adalah penjelasan dari masing-masing komponen yang terlibat didalamnya, (Sutabri, 2012) yaitu:

1. Blok Masukan (*Input Block*)

Input mewakili data yang masuk ke dalam sistem informasi. Input yang dimaksudkan adalah metode dan media untuk menangkap data yang akan dimasukkan yang berupa dokumen-dokumen dasar.

2. Blok Model (*Model Block*)

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

3. Blok Keluaran (*Output Block*)

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

4. Blok Teknologi (*Technology Block*)

Teknologi merupakan "*tool box*" dalam sistem informasi, teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu teknis (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data (*Database Block*)

Blok basis data (*database*) merupakan data yang saling berkaitan dan berhubungan satu sama lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data di dalam basis

data perlu diorganisasikan sedemikian rupa agar informasi yang dihasilkan berkualitas.

6. Blok Kontrol (*Control Block*)

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

2.4 Pengertian *Packaging*

Packaging (pengemasan) adalah teknik industri dan pemasaran yang digunakan untuk melindungi, mengidentifikasi dan menyegel produk konsumen yang didistribusikan/dipasarkan (*Encarta Encyclopedia*, 2012).

Kemasan yang baik adalah kemasan yang bisa melindungi isi produk tersebut terhadap cuaca dan proses alam lainnya. Kemasan juga digunakan sebagai wadah agar barang mudah dibawa, tetapi harus juga bisa berkomunikasi agar bisa menerangkan dan mereflesikan produk, citra, *brand* atau merk yang merupakan bagian yang tidak terpisah dari promosi dan pemasaran, tentunya dengan pertimbangan mudah untuk dikenali, dilihat, dipahami, dan diingat (Wirya, 2011).

Kemudian bahwa daya tarik suatu kemasan sangatlah penting untuk menarik minat konsumen dan mempengaruhi tindakan konsumen baik secara sadar maupun tanpa disadari. Selain itu desain suatu kemasan yang optimal mampu memberikan impresi spontan dan langsung atas tindakan konsumen di tempat penjualan, karena tujuan akhir dari desain kemasan adalah menciptakan penjualan.

Maka dari itu sebuah kemasan produk, selain harus dapat melindungi isi produk terhadap cuaca dan proses alam lainnya, juga harus mampu mereflesikan citra dari produk itu sendiri yang mampu dilihat, dikenali, dan diingat oleh

konsumen dengan mudah, sehingga akhirnya mampu membujuk konsumen untuk membeli produk tersebut.

Menurut jenisnya packaging/kemasan dibagi menjadi tiga bagian:

1. Kemasan *Flexible*
2. Kemasan Kaku (Rigid)
3. Kemasan *Hybrid* (perpaduan dua jenis kemasan di atas)

Berdasarkan dengan keterkaitan produk:

1. Kemasan yang kontak langsung dengan makanan (kemasan primer)
2. Kemasan yang memprotek kemasan primer (kemasan sekunder)
3. Kemasan jenis biasanya jenis kemasan *multilayer*.

2.5 Pengertian Analisis Sistem

Menurut Yakub (2012), Analisis sistem dapat diartikan sebagai suatu proses untuk memahami sistem yang ada, dengan menganalisis jabatan dan uraian tugas (*business users*), proses bisnis (*business proses*), ketentuan atau aturan (*business rule*), masalah dan mencari solusinya (*business problem and business soulution*), dan rencana-rencana perusahaan (*business plan*).

Menurut Mulyanto (2010), Analisis sistem adalah teori sistem umum yang sebagai sebuah landasan konseptual yang mempunyai tujuan untuk memperbaiki berbagai fungsi didalam sistem yang sedang berjalan agar menjadi lebih efisien, mengubah sasaran sistem yang sedang berjalan, merancang/mengganti output yang sedang digunakan, untuk mencapai tujuan yang sama dengan seperangkat input yang lain (biasa jadi lebih sederhana dan lebih interatif) atau melakukan beberapa perbaikan serupa.

Tahap analisis dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem. Tahap ini merupakan tahap yang kritis karena kesalahan dalam tahap ini menyebabkan kesalahan pada tahap selanjutnya. Misalnya jika terjadi masalah untuk menentukan seberapa jauh sistem tersebut telah mencapai sasaran. Jika sistem mempunyai beberapa kelemahan, anda harus dapat menemukannya. Tugas ini disebut sebagai analisis sistem.

Dalam tahap analisis sistem, terdapat langkah-langkah dasar yang harus dilakukan oleh sistem analis, yaitu (Mulyanto, 2010):

- a. *Identify*, mengidentifikasi masalah.

- b. *Understand*, memahami kerja sistem yang ada .
- c. *Analyze*, menganalisis sistem yang ada.
- d. *Report*, membuat laporan dari hasil analisis.

2.6 **System Development Life Cycle (SDLC)**

System Development Life Cycle (SDLC) adalah proses 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).

Tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut (Rosa dan Shalahuddin, 2014):

1. Inisiasi (*initiation*)
Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.
2. Pengembangan konsep sistem (*system concept development*)
Menentukan lingkup konsep termasuk dokumen lingkup sistem analisis manfaat biaya, manajemen rencana dan pembelajaran kemudahan sistem.
3. Perencanaan (*planning*)
Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.
4. Analisis kebutuhan (*requirements analysis*)
Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.
5. Desain (*design*)
Mentransformasikan kebutuhan *detail* menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
6. Pengembangan (*development*)
Mengkonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan, membuat basis data dan mempersiapkan prosedur kasus pengujian, mempersiapkan berkas atau *file* pengujian, pengkodean, pengkompilasian, memperbaiki dan membersihkan program serta peninjauan pengujian.
7. Integrasi dan pengujian (*integration and test*)
Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf

penjamin kualitas (*quality assurance*) dan *user* sehingga menghasilkan laporan analisis pengujian.

8. Implementasi (*implementation*)

Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan luar *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9. Operasi dan pemeliharaan (*operations and maintenance*)

Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

10. Disposisi (*disposition*)

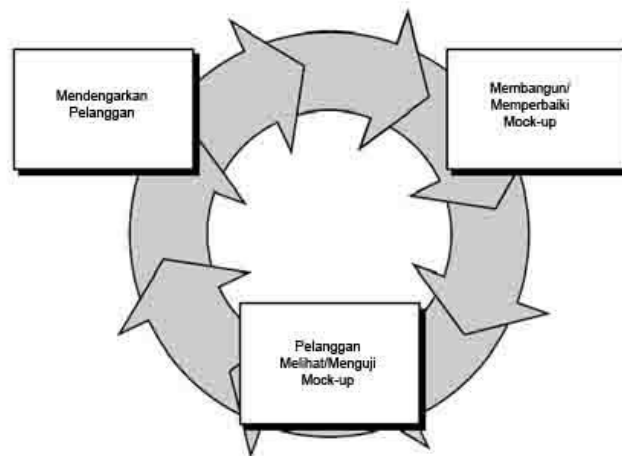
Mendeskripsikan aktivitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktivitas *user*.

2.6.1 Model *Prototype*

Model *prototype* dapat digunakan untuk menyambungkan ketidakpahaman pelanggan mengenai hal teknis yang memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak (Rosa dan Shalahuddin, 2011).

Tahapan-tahapan pada model *prototype* adalah sebagai berikut (Rosa dan Shalahuddin, 2011):

1. Mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat.
2. Membuat *prototype* agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program *prototype* biasanya merupakan program yang belum jadi.
3. Program *prototype* selanjutnya dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*.



Gambar II.2 Ilustrasi Model *Prototype*
 Sumber: Rosa dan Shalahuddin (2011)

Mock-up adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi atau keperluan lain. Sebuah *mock-up* disebut sebagai *prototype* perangkat lunak jika menyediakan atau mampu mendemonstrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Iterasi terjadi pada pembuatan *prototype* sampai sesuai dengan keinginan pelanggan atau *user* (Rosa dan Shalahuddin, 2011).

Kelemahan model *prototype* adalah sebagai berikut (Rosa dan Shalahuddin, 2011):

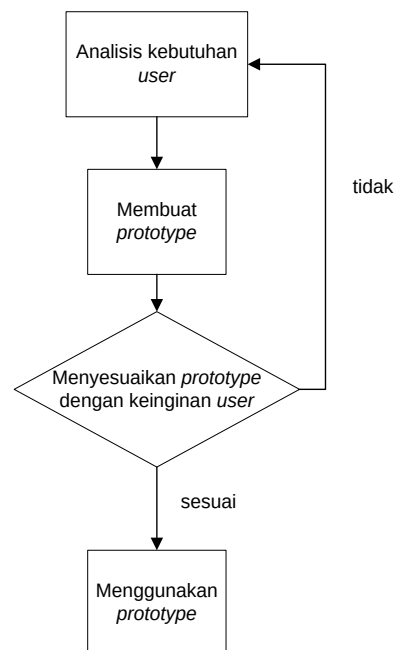
1. Mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat.
2. Membuat *prototype* agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program *prototype* biasanya merupakan program yang belum jadi.
3. Program *prototype* selanjutnya dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*.

Pengembang lebih sering mengambil kompromi dengan pelanggan untuk mendapatkan *prototype* dengan waktu yang cepat sehingga pengembang lebih sering melakukan segala cara (tanpa idealis) guna menghasilkan *prototype* untuk

didemonstrasikan. Hal ini dapat menyebabkan kualitas perangkat lunak yang kurang baik atau bahkan menyebabkan iteratif tanpa akhir.

2.6.2 *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* (McLeod, 2011).



Gambar II.3 *Evolutionary Prototype Model*
Sumber: McLeod (2011)

1. Analisis kebutuhan *user*, pengembang dan *user* atau pemilik sistem melakukan diskusi dimana *user* atau pemilik sistem menjelaskan kepada pengembang tentang kebutuhan sistem yang mereka inginkan.
2. Membuat *prototype*, pengembang membuat *prototype* dari sistem yang telah dijelaskan oleh *user* atau pemilik sistem.

3. Menyesuaikan *prototype* dengan keinginan *user* atau pemilik sistem, pengembang menanyakan kepada *user* atau pemilik sistem tentang *prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.
4. Menggunakan *prototype*, sistem mulai dikembangkan dengan *prototype* yang sudah dibuat.

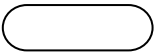
2.7 Bagan Alir (*Flowchart*)

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

1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhirnya.
4. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan, misalnya:
 - a. "Persiapkan" dokumen
 - b. "Hitung" gaji
5. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung di tempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakanlah simbol-simbol bagan alir yang standar.

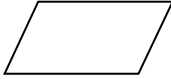
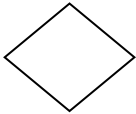
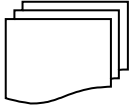
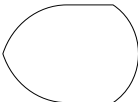
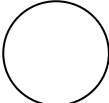
Simbol-simbol standar yang digunakan oleh analis sistem untuk membuat bagan alir dokumen yang menggambarkan sistem tertentu dapat dilihat pada tabel berikut ini:

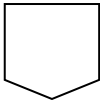
Tabel II.1 Simbol-Simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1.		Terminal	Awal dan akhir dari suatu proses.

No.	Simbol	Nama	Fungsi
2.		Garis akhir	Arus dari suatu proses.
3.		Proses	Proses pengolahan data.

Tabel II.2 Simbol-Simbol *Flowchart* (Lanjutan)

4.		Proses terdistribusi	menjalankan program.
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.

No.	Simbol	Nama	Fungsi
13.		<i>Off page connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda.

(Sumber: Febriani, 2015)

2.8 Analisis dan Perancangan Berorientasi Objek

Analisis dan perancangan berbasis objek (OOAD) adalah rekayasa pendekatan software yang model sistemnya sebagai sekelompok objek yang saling berinteraksi. Setiap objek terdiri dari beberapa entitas yang mempunyai kepentingan dalam sistem yang dimodelkan, dan ditandai oleh class-classnya, elemen data dan perilakunya. Berbagai model dapat dibuat untuk menunjukkan struktur yang bersifat statis, perilaku dinamis dan run-time penyebaran objek yang berkolaborasi (Sutabri,2012).

2.9 Unified Modeling Language (UML)

Menurut Rosa dan Shalahuddin (2014) “*Unified Modeling Language (UML)* adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML merupakan sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek”.

UML terdiri atas banyak elemen-elemen grafis yang digabungkan membentuk diagram. Dengan mrnggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Konsepsi dasar UML dapat dilihat pada tabel II.2.

Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek seperti C++, Java, C#, atau VB.Net. Walaupun demikian, UML tetap dapat digunakan untuk *modeling* aplikasi prosedural dalam VB atau C.

Tabel II.3 Konsepsi Dasar UML

<i>Major Area</i>	<i>View</i>	<i>Diagrams</i>	<i>Main Concepts</i>
<i>Structural</i>	<i>Static view</i>	<i>Class diagram</i>	<i>Class, association, generalization, dependency, realization, interface.</i>
	<i>Use case view</i>	<i>Use case diagram</i>	<i>Use case, actor, association, extend, include, use case generalization.</i>
	<i>Implementation view</i>	<i>Component diagram</i>	<i>Component, interface, dependency, realization.</i>
	<i>Deployment view</i>	<i>Deployment diagram</i>	<i>Node, component, dependency, location.</i>
<i>Dynamic</i>	<i>State machine view</i>	<i>Statechart diagram</i>	<i>State, event, transaction, action.</i>
	<i>Activity view</i>	<i>Activity diagram</i>	<i>State, activity, completion transition, fork, join.</i>
	<i>Interaction view</i>	<i>Sequence diagram</i>	<i>Interaction, object, message, activation.</i>
		<i>Collaboration diagram</i>	<i>Collaboration, interaction, collaboration rule, message.</i>
<i>Model Management</i>	<i>Model management view</i>	<i>Class diagram</i>	<i>Package, subsystem, model.</i>

<i>Extensibility</i>	<i>All</i>	<i>All</i>	<i>Constraint, stereotype, tagged values.</i>
----------------------	------------	------------	---

(Sumber: Rosa dan Shalahuddin, 2014)

2.9.1 Use Case Diagram

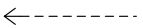
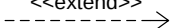
Use case adalah deskripsi fungsi dari sebuah sistem dari sudut pandang para pengguna sistem. *Use case* mendefinisikan “apa” yang dilakukan oleh sistem dan elemen-elemennya, bukan “bagaimana” sistem dan elemen-elemen saling berinteraksi.

Use case bekerja dengan menggunakan “*scenario*”, yaitu deskripsi urutan-urutan langkah yang menerangkan apa yang dilakukan pengguna terhadap sistem atau sebaliknya. *Use case diagram* mengidentifikasi fungsionalitas yang dimiliki oleh sistem (*use case*), *user* yang berinteraksi dengan sistem (*actor*), dan keterhubungan antar *user* dengan fungsionalitas sistem. Berikut adalah notasi yang dimiliki oleh *use case diagram*:

Tabel II.4 Notasi Use Case Diagram

<u>Actor</u> - Seseorang atau sebuah sistem yang mendapatkan keuntungan dari subjek internal. - Dilabelkan berdasarkan perannya.	
<u>Use Case</u> <u>Merepresentasikan sistem fungsionalitas terbesar.</u> <u>Dilabelkan dengan gambaran kata benda.</u>	
<u>Association</u> <u>Menghubungkan aktor dengan use case dengan interaksinya.</u>	

Tabel II.4 Notasi Use Case Diagram (Lanjutan)

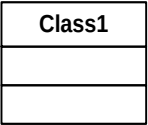
<u><i>Include Relationship</i></u> • <u><i>Merepresentasikan inklusi fungsionalitas antara use case dengan use case lainnya.</i></u> • <u><i>Mempunyai anak panah yang digambarkan dari use case dasar ke use case selanjutnya.</i></u>	<pre><<include>></pre> 
<u><i>Extend Relationship</i></u> • <u><i>Merepresentasikan perpanjangan use case untuk memasukkan pilihan yang sama.</i></u>	<pre><<extend>></pre> 
<u><i>Generalization Relationship</i></u> • <u><i>Merepresentasikan use case khusus ke umum.</i></u>	

(Sumber: Rosa dan Shalahuddin, 2014)

2.9.2 Class Diagram

Class diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). *Class diagram* menggambarkan struktur dan deskripsi *class*, *package*, dan *object* beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. Berikut adalah notasi-notasi yang ada pada *class diagram*:

Tabel II.6 Notasi Class Diagram

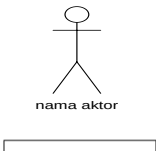
<u>Class</u> • Merepresentasikan orang, tempat, atau hal tentang sistem mana yang akan dibutuhkan dan menyimpan informasi. • Mempunyai nama yang ditulis tebal dan tengah dan memiliki atribut.	
<u>Association</u> • Merepresentasikan hubungan antara <i>class</i>.	
<u>Composition</u> • Menyajikan hubungan fisik antara <i>multiple class</i> dengan <i>class</i> itu sendiri.	
<u>Generalization</u> • Merepresentasikan satu macam hubungan antara <i>multiple class</i>.	

(Sumber: Rosa dan Shalahuddin, 2014)

2.9.3 Sequence Diagram

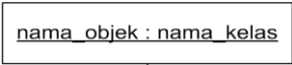
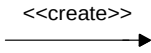
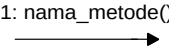
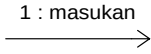
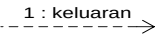
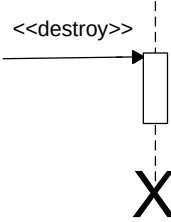
Sequence diagram mendokumentasikan komunikasi atau interaksi antar kelas-kelas. Diagram ini menunjukkan sejumlah obyek dan pesan yang diletakkan diantara obyek-obyek di dalam *use case*. Berikut adalah notasi-notasi yang terdapat pada *sequence diagram*:

Tabel II.7 Simbol-Simbol *Sequence Diagram*

No.	Simbol	Deskripsi
1.	Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi

Tabel II.7 Simbol-Simbol *Sequence Diagram* (Lanjutan)

No.	Simbol	Deskripsi
2.	Garis Hidup/<i>lifeline</i> 	Menyatakan kehidupan suatu objek.

3.	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 di dalamnya.
.	Pesan tipe create 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
.	Pesan tipe call 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode.
.	Pesan tipe send 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
.	Pesan tipe return 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9.	Pesan Tipe Destroy 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

(Sumber: Rosa dan Shalahuddin, 2014)

2.9.4 Activity Diagram

Activity diagram digunakan untuk mendokumentasikan alur kerja pada sebuah sistem, yang dimulai dari pandangan *business level* hingga ke *operational level*. Pada dasarnya *activity diagram* merupakan variasi dari *statechart diagram*.

Activity diagram bisa mendukung perilaku *parallel*. Berikut adalah notasi *activity diagram*:

Tabel II.8 Simbol-Simbol Activity Diagram

No.	Simbol	Deskripsi
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan/decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan/join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

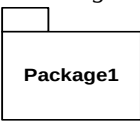
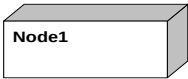
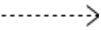
(Sumber: Rosa dan Shalahuddin, 2014)

2.9.5 Deployment Diagram

Deployment atau *physical diagram* menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisik. Sebuah *node* adalah server, *workstation*, atau piranti keras lain yang digunakan untuk men-deploy komponen dalam lingkungan sebenarnya. Hubungan antar *node* (misalnya TCP/IP) dan *requirement* dapat juga didefinisikan dalam diagram ini. Di bawah ini adalah contoh *Deployment Diagram* :

Tabel II.9 Simbol-simbol Deployment Diagram

No.	Simbol	Deskripsi
-----	--------	-----------

1.	<i>Package</i> 	Package merupakan sebuah bungkus dari satu atau lebih node.
2.	<i>Node</i> 	Biasanya mengacu pada perangkat keras (hardware), perangkat lunak yang tidak dibuat sendiri (software), jika di dalam node disertakan komponen untuk mengkonsistenkan rancangan maka, komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.
3.	<i>Dependency</i> 	Kebergantungan antar node, arah panah mengarah pada node yang dipakai.
4.	<i>Link</i> 	Relasi antar node.

(Sumber: Rosa dan Shalahuddin, 2014)

2.10 Kamus Data

Kamus data adalah kumpulan fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data, analis sistem dapat mendefinisikan data yang mengalir dari sistem dengan lengkap. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem.

Pada tahap analisis, kamus data dapat digunakan sebagai alat komunikasi antara analis sistem dengan pemakai sistem yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan-laporan dan *database* (Jogiyanto, 2010).

2.11 Database

Database (basis data) merupakan kumpulan dari *file-file* yang saling berelasi, dimana relasi tersebut ditunjang dengan kunci dari setiap *file* yang ada (Pahlevy, 2010).

Database dapat didefinisikan dalam sejumlah sudut pandang seperti:

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

2.12 HIPO (Hierarchy plus Input-Process-Output)

HIPO (*Hierarchy plus Input-Process-Output*) merupakan metodologi yang dikembangkan dan didukung oleh IBM. HIPO sebenarnya adalah alat dokumentasi program. Akan tetapi sekarang HIPO juga banyak digunakan sebagai alat desain dan teknik dokumentasi dalam siklus pengembangan sistem. HIPO berbasis pada fungsi, yaitu tiap-tiap modul di dalam sistem digambarkan oleh fungsi utamanya (Jogiyanto, 2010).

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

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

2.13 XAMPP

XAMPP adalah perangkat lunak yang berfungsi sebagai *server* yang dapat berdiri sendiri (*localhost*) yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. XAMPP merupakan *web server* yang mudah digunakan dan dapat melayani tampilan halaman *web* yang dinamis, perangkat lunak ini tersedia dalam versi *GNU (General Public License)* yaitu bebas didapatkan tanpa harus membayar (Wikipedia, 2015).

Ada beberapa bagian XAMPP yang biasa digunakan pada umumnya yaitu sebagai berikut:

1. *htdocs*, adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas *PHP*, *HTML* dan *script* lainnya.
2. *phpMyAdmin*, merupakan bagian untuk mengelola basis data *MySQL* yang ada di komputer.
3. *Control Panel*, yang berfungsi untuk mengelola layanan (*service*) XAMPP. Layanan (*service*) ini seperti menghentikan (*stop*) layanan, ataupun memulai (*start*) layanan.

2.14 PHP

Menurut Anhar (2010), *PHP* singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. *PHP* merupakan *script* yang terintegrasi dengan *HTML* dan berada pada *server (server side HTML embedded scripting)*. *PHP* adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*. Mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru atau *up to date*. Semua *script PHP* dieksekusi pada *server* dimana *script* tersebut dijalankan.

PHP merupakan sebuah bahasa pemrograman *web* yang memiliki sintak atau aturan dalam menuliskan *script* atau kode-kodenya. Salah satu contoh penulisan kode *PHP* dapat dilihat sebagai berikut ini:

```
<?php
echo("Tes Pakai PHP");
?>
```

Kode-kode PHP memiliki tata aturan, yaitu diawali dengan tanda `<?php` dan diakhiri dengan tanda `?>`. Tiap akhir baris harus selalu diberi tanda titik koma (;). PHP bersifat *Case Sensitive*, artinya penulisan huruf besar dan kecil pada kode PHP sangat berpengaruh.

Menurut Anhar (2010), beberapa kelebihan PHP dari bahasa pemrograman lain adalah sebagai berikut:

1. PHP adalah bahasa *scripting* yang memiliki referensi yang banyak dan sederhana sehingga mudah untuk dimengerti pemula.
2. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin seperti Linux, Unix, Macintosh, dan Windows serta dapat dijalankan secara *runtime* melalui *console* dan dapat menjalankan perintah-perintah sistem.
6. PHP juga dilengkapi dengan berbagai macam pendukung lain seperti *support* langsung ke berbagai macam *database* yang populer, misal: Oracle, PostgreSQL, dan lain-lain.

2.15 MySQL

MySQL adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS (*Database Management System*), sifat dari DBMS ini adalah *open source*. MySQL sebenarnya produk yang berjalan pada *platform* Linux, dengan adanya perkembangan dan banyaknya pengguna serta lisensi dari *database* ini adalah *open source*, maka para pengembang kemudian merilis versi Windows. (Nugroho, 2010)

MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam basis data sejak lama, yaitu SQL (*Structure Query Language*). SQL adalah sebuah konsep pengoperasian

basis data terutama untuk proses seleksi, pemasukan, pengubahan, dan penghapusan data yang dimungkinkan dapat dikerjakan dengan mudah dan otomatis (Sutaji, 2012).

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

1. *Portability*
Dapat berjalan stabil pada berbagai sistem operasi, di antaranya: Windows, Linux, FreeBSD, MacOS X Server, Solaris, Asigma.
2. *Open source*
Didistribusikan secara gratis di bawah lisensi dari *General Public License* (GPL), dimana setiap orang bebas untuk menggunakannya tetapi tidak boleh untuk dijadikan program induk turunan bersifat *close source* (komersial).
3. *Multi User*
Dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan.
4. *Performance tuning*
Memiliki kecepatan yang tinggi dalam menangani *query*.
5. *Column types*
Memiliki tipe data yang sangat kompleks, seperti: *signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year* dan *enum*.
6. *Command dan function*
Memiliki operator dan fungsi penuh yang mendukung *select* dan *where* dalam *query*.
7. *Security*
Memiliki beberapa lapisan sekuritas seperti tingkat *subnet mask, hostname, privilege user* dengan sistem perijinan yang mendetail serta *password* yang ter-enkripsi.
8. *Scalability dan limits*
Mampu menangani basis data dalam jumlah besar, dengan jumlah *field* lebih dari 50 juta, 60 ribu tabel, dan 5 miliar *record*. Batas indeks mencapai 32 buah per tabel.
9. *Localization*
Dapat mendeteksi pesan kesalahan (*error code*) pada klien dengan menggunakan lebih dari 20 bahasa.
10. *Connectivity*
Dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix Socket, Named Pipes*.
11. *Interface*
Memiliki antarmuka terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan API.
12. *Client dan tools*
Dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi basis data sekaligus dokumen petunjuk *online*.

13. Struktur tabel

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

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

Tabel II.10 Beberapa Tipe Data Pada MySQL

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

Tipe Data	Ukuran	Keterangan
BLOB	-	Untuk menyimpan data biner (misalnya gambar atau suara).

(Sumber: Sutaji, 2012)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian yaitu suatu usaha proses mencari jawaban atas pertanyaan atau masalah dengan cara sabar, hati-hati, terencana, sistematis atas fakta atau prinsip-prinsip, mengembangkan dan menguji kebenaran ilmiah suatu pengetahuan (Soewardi, 2012). Menurut Sugiyono (2013) metodologi penelitian merupakan cara ilmiah untuk mendapatkan data dengan tujuan dan kegunaan tertentu.

Teknik pengumpulan data merupakan langkah yang paling strategis dalam melakukan penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Berikut metode yang digunakan dalam melakukan penelitian (Sugiyono, 2013):

1. Teknik Wawancara
Merupakan pertemuan dua orang untuk bertukar informasi dan ide melalui tanya jawab, sehingga dapat dikonstruksikan makna dalam suatu topik tertentu.
2. Teknik Observasi
Merupakan suatu proses yang kompleks dan tersusun dari berbagai proses. Dua diantaranya yang terpenting adalah proses pengamatan dan ingatan.
3. Teknik Dokumen
Merupakan pengumpulan dokumentasi dengan bentuk berkas, tulisan, ataupun gambar.
4. Teknik Literatur
Merupakan teknik pengumpulan data yang bersifat menggabungkan berbagai sumber daya yang telah ada berdasarkan bahan atau literatur ilmiah sebagai penunjang kelengkapan data sesuai dengan aplikasi pelaksanaan di lapangan.

3.2 Jenis dan Sumber Data

Sumber data dalam penelitian adalah subjek dari mana data dapat diperoleh. Terdapat dua macam sumber data, yaitu:

1. Data primer, yaitu data yang berasal dari sumber secara langsung, tanpa melalui media perantara. Data primer yang didapatkan dalam penelitian ini diantaranya adalah:
 - a. Data yang digunakan dalam proses *packaging plan* pada PT SKF Indonesia
 - b. Proses bisnis sistem saat ini dan sistem yang akan diusulkan.
2. Data sekunder, yaitu data penelitian yang diperoleh secara tidak langsung melalui media perantara atau data yang diperoleh dan dicatat oleh pihak lain yang telah disusun dan

dipublikasikan. Data sekunder yang digunakan dalam penelitian ini laporan-laporan, buku-buku dan kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian.

3.3 Metode Pengumpulan Data

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

1. Studi Lapangan

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

- a. Pengamatan, tahap ini dilakukan secara langsung di Bagian *Shipping* PT SKF Indonesia dengan mengamati prosedur *packaging plan* yang sedang berjalan mulai dari penerimaan *purchase order* oleh *marketing* sampai dengan memberikan produk kepada pihak pelanggan. Melalui teknik ini data yang dibutuhkan untuk diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.
- b. Wawancara, yaitu mencari data yang dibutuhkan secara langsung melalui memberikan pertanyaan spontan terhadap segala hal yang diperlukan pada penyusunan Tugas Akhir ini. Wawancara ini dilakukan kepada karyawan pada Bagian *Shipping* di PT SKF Indonesia.

2. Studi Kepustakaan

Studi kepustakaan dilakukan dengan membaca buku dan literatur dalam lingkup perkuliahan maupun di luar lingkup perkuliahan yang berhubungan dengan judul dan permasalahan, sehingga dapat menunjang dalam penulisan Tugas Akhir ini. Studi kepustakaan yang dilakukan adalah dengan menggunakan buku yang dimiliki, buku yang dipinjam dari perpustakaan dan mencari data yang diperlukan melalui Internet.

3.4 Metode Pengembangan Sistem

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

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

- a. Mengidentifikasi kebutuhan pengguna

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

b. Membuat satu prototipe

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

c. Menentukan apakah prototipe dapat diterima

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

d. Menggunakan prototipe

Menjadikan prototipe sebagai sistem yang dijalankan.

3.5 Kerangka Penelitian

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

1. Pendahuluan

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

2. Identifikasi Masalah

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

3. Merumuskan Masalah

Tahap ini bertujuan untuk merumuskan masalah dalam hal *penginputan schedule delivery* yang masih menggunakan *microsoft excel* dan pembuatan *packaging plan* yang masih manual melakukan pencatatan data dengan penulisan tangan. Cara menentukan masalah yaitu kita harus mengetahui terlebih dahulu pokok permasalahan. Sehingga penelitian yang dilakukan dapat terstruktur dan sesuai dengan kebutuhan pengguna.

4. Menetapkan Tujuan Penelitian

Menetapkan tujuan penelitian dari tugas akhir ini yaitu membangun sistem informasi *packaging plan* yang mampu menyimpan, mendata, menjaga isi data dan mempersingkat

waktu dalam proses pengiriman. Sehingga para pengguna akan mendapatkan informasi yang lebih akurat dan bermanfaat.

5. Batasan Masalah

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

6. Penerapan Metode *Prototype Evolusioner*

a. Mengidentifikasi kebutuhan pelanggan

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

b. Membuat prototipe

Membuat sebuah prototipe sesuai dengan analisis yang dilakukan. Pada tahap ini akan dirancang aplikasi tentang sistem *packaging plan*. Membuat prototipe nantinya akan memuat tentang:

- Memodelkan sistem menggunakan *Unified Modeling Language* (UML).
- Memodelkan basis data dengan menggunakan *class diagram* dan kamus data.
- Merancang sistem dengan menggunakan HIPO, *Flowchart*, dan *Interface*.
- Membuat prototipe dengan menggunakan PHP 5.6.3 sebagai *software* dan MySQL 5.6.21 sebagai *databasenya*.

c. Mengevaluasi prototipe

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

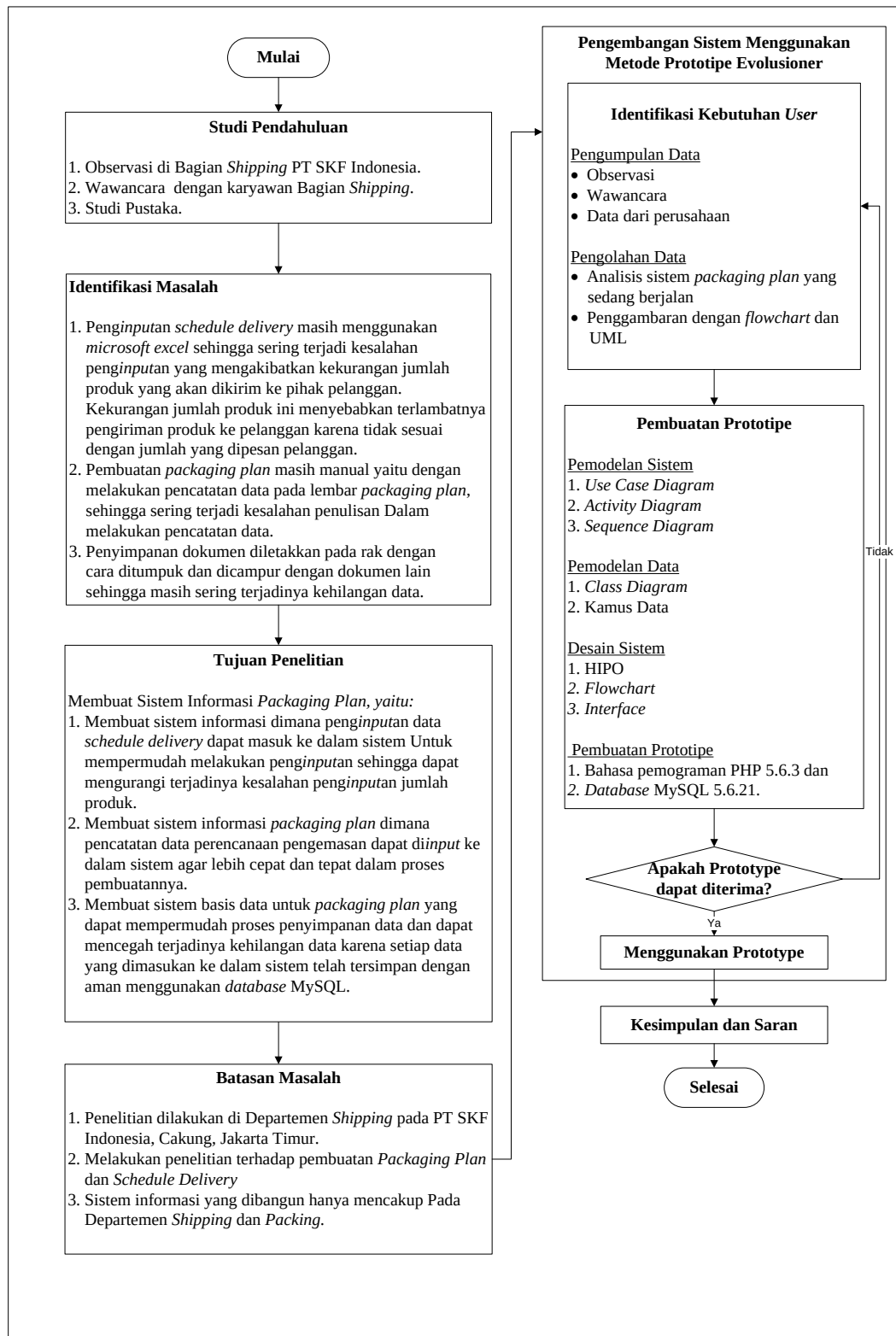
d. Menggunakan prototipe

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

7. Kesimpulan dan Saran

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

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



Gambar III.1 Kerangka Penelitian
(Sumber: Hasil Analisis, 2017)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Pengumpulan Data

Pada penelitian ini difokuskan terlebih dahulu pada sejarah dan profil perusahaan, produk yang dihasilkan, daftar pelanggan, serta sistem yang sedang berjalan di Departemen *Shipping*.

4.1.1 Sejarah dan Profil Perusahaan

PT SKF Indonesia pada mulanya bernama PT Logam Sari Bearindo, didirikan pabrik *bearing* pada tahun 1984 ini oleh seorang yang bernama Wirontono, sampai sekarang masih dapat memenuhi permintaan para konsumen, meskipun telah melewati masa krisis moneter yang terjadi pada tahun 1998. Perusahaan ini memulai produksi komersial pada Januari 1986 dan mempunyai merk pasaran yaitu BI (*Bearing* Indonesia) pada tahun 1988 PT Logam Sari Bearindo mengikat perjanjian dengan SKF Swedia yang merupakan produsen *bearing* terbesar di dunia. Perjanjian tersebut berupa kerja sama teknik yang berguna untuk meningkatkan mutu produksi, sehingga BI dapat bersaing dengan produk-produk impor. Pada September 1988, perusahaan Astra Internasional melalui PT Federal Motor menjadi salah satu pemegang saham PT Logam Sari Bearindo.

Hal itu agar memperkuat struktur permodalan, pemasaran dan menjadikan struktur manajemen menjadi lebih baik. Pada Oktober 1992 perusahaan ini mendirikan *Channel* 5 dan 6 dan bersamaan dengan mendapatkan izin usaha tetap dari Departemen Perindustrian Republik Indonesia. Lisensi SKF diberikan untuk semua tipe *bearing* yang diproduksi. Pada akhir tahun 1992 merk FMB (*Federal Motor Bearing*) diperkenalkan dan menggantikan merk BI. Setelah sekian tahun semenjak produksi pertamanya, perusahaan ini mendapatkan sertifikasi ISO 9002 dari lembaga sertifikasi *Llyod's Register Quality Assurance* (LRQA) pada Januari 1996. Hal ini membuktikan dedikasi pihak manajemen terhadap kualitas manajemen perusahaan yang sudah diakui oleh internasional. Pada Mei 1997 *Channel* 7 selesai dibangun dan pada September 1997, PT Logam Sari Bearindo sudah resmi menjadi bagian dari SKF Swedia, kemudian berganti nama menjadi PT SKF Indonesia. Tentunya hal ini menjadikan produk-produk yang diproduksi memiliki standar dengan SKF Swedia, dan juga tentunya kualitas dan spesifikasi produk akan sama dengan produk yang berasal dari PT SKF di seluruh negara. Demi memenuhi standar manajemen lingkungan maka perusahaan mendapatkan sertifikat ISO 14001 pada Desember 1999, dan juga pada bulan yang sama mendapatkan sertifikat QS 9000. Pada Januari tahun 2000, diperkenalkan sebuah merk dagang baru, yaitu SKF Enduro dan SKF Genio. Desember tahun 2000, Suzuki

sebagai salah satu pelanggan utama memberikan penghargaan sebagai produk *bearing* yang memiliki kualitas terbaik.

Produk PT SKF Indonesia memiliki beberapa sertifikasi serta mampu memasuki pasar internasional, yang dimulai pada Oktober 2001 dengan mengeksport produknya ke Malaysia, dan juga pada bulan yang sama *Channel 8* selesai dibangun dan siap untuk dioperasikan. Pada Maret 2004 PT SKF Indonesia mendapatkan sertifikasi ISO/TS 16949. Tahun 2006 *Channel 9, 10 dan 11* telah selesai dibangun. Dari tahun ketahun perkembangan perusahaan ini semakin meningkat, hal itu ditunjukkan dengan memperluas area pabrik dalam rangka memenuhi banyaknya permintaan pelanggan yang datang baik dari lokal maupun internasional.

Saat ini PT SKF Indonesia yang berlokasi di Cakung, Jakarta Timur memiliki luas tanah 53.000 m² dan luas pabrik 23.500 m² dengan memiliki 11 jalur produksi dan memiliki 440 karyawan. PT SKF Indonesia terdiri atas pabrik dan bisnis unit, yang bergerak di bidang produksi *bearings*, penyediaan dukungan teknis dan solusi yang berfokus pada pelanggan.

Dengan teknologi SKF, perusahaan ini mampu memenuhi tuntutan pasar, khususnya aplikasi dua-roda (sepeda motor), baik bagi pengguna maupun pabrik-pabrik sepeda motor di seluruh Indonesia. Ini merupakan wujud dari keberhasilan jaringan distribusi dan dealer yang menjangkau seluruh Indonesia. Komitmen PT SKF Indonesia adalah memastikan kualitas setiap produk, layanan dan dukungan teknis untuk memenuhi kebutuhan pelanggan, sesuai standar SKF. PT SKF Indonesia juga berkomitmen untuk membangun kerja sama yang erat, melalui kinerja yang efektif dan efisien, untuk mempertahankan pandangan yang tinggi terhadap kualitas SKF.

Sementara itu dari dalam tubuh PT SKF Indonesia terdapat pula para tenaga kerja yang berjasa dalam pemenuhan kebutuhan *bearing* di pasar Indonesia, oleh karena itu untuk meningkatkan kedisiplinan semua tenaga kerja, maka PT SKF Indonesia selalu menekankan program budaya 5R, yaitu:

1. Ringkas: suatu usaha dalam memisahkan barang yang diperlukan dengan yang tidak diperlukan.
2. Rapi: usaha yang selalu memastikan segala sesuatu pada tempatnya dan juga pemberian label agar udah mengenali suatu objek.
3. Resik: usaha membersihkan tempat kerja dan selalu mempertahankan kebersihan.
4. Rawat: usaha yang selalu memelihara keadaan ringkas, resik dan rapih.
5. Rajin: usaha yang menjadikan ringkas, rapi dan rajin menjadi kebiasaan.

Adapun manajemen PT SKF Indonesia memegang teguh kebijakan kualitas SKF, yaitu:

1. SKF hanya akan memasarkan produk layanan dan solusi yang akan menjamin kepuasan pelanggan.
2. Menjalankan proses bisnis yang kompeten, dapat diandalkan dan efisien.
3. Menerapkan perbaikan berkelanjutan di seluruh organisasi demi mencapai kualitas tanpa cacat (*Zero Defect*).

Selain kebijakan kualitas, PT SKF Indonesia dalam aktifitasnya juga menerapkan kebijakan lingkungan, keselamatan dan kesehatan kerja. Hal tersebut diupayakan agar meraih keuntungan yang dapat bertahan dalam jangka panjang. Adapun tugas utama untuk mencapai tujuan tersebut dengan mengembangkan, membuat dan memasarkan produk dan jasa yang dapat memuaskan kebutuhan pelanggan secara aman dalam penggunaannya, energi yang digunakan harus efisien, tidak mencemari lingkungan dan dapat di daur ulang atau dibuang secara aman.

Visi dari perusahaan PT SKF Indonesia adalah melengkapi kebutuhan *bearing* dunia dengan produk SKF.

Misi PT SKF Indonesia adalah :

1. Menjadi perusahaan pilihan bagi para pelanggan, distributor dan pemasok perusahaan.
2. Menjadi perusahaan pilihan bagi karyawan.
3. Menjadi perusahaan pilihan bagi pemegang saham.

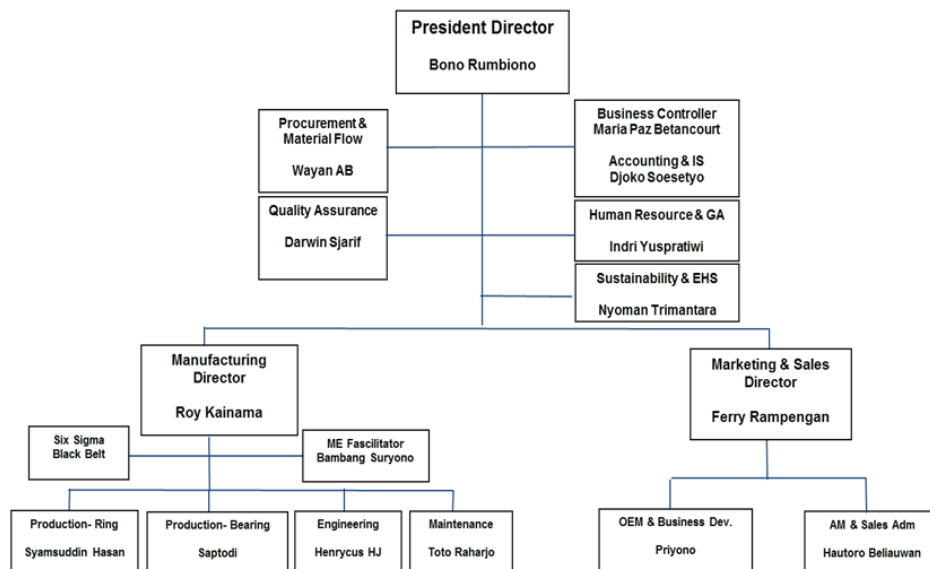
Agar organisasi dapat bekerja dengan baik maka diperlukan suatu struktur organisasi. Struktur organisasi merupakan suatu tindakan dari suatu fungsi dan tanggung jawab dalam sebuah instansi, dimana memperlihatkan hubungan garis wewenang dan pertanggung jawaban antara jabatan dan peranan dalam suatu organisasi khususnya di bagian *general store*.

Oleh karena itu, pada dasarnya suatu perusahaan perlu membuat struktur organisasi dimana menjelaskan secara rinci mengenai tugas, tanggung jawab dan wewenang masing-masing jabatan yang terdapat di struktur organisasi tersebut, sehingga kerja sama antara masing-masing jabatan dapat terjalin secara harmonis dan menguntungkan perusahaan. Struktur organisasi yang *simple* dan tidak rumit akan dapat menunjang dan mewujudkan perusahaan dalam pencapaian visi dan misinya. Oleh karena itu perusahaan perlu merancang struktur organisasi seefektif dan seefisien mungkin yang mampu mengkoordinasikan semua jabatan sehingga perusahaan mampu mewujudkan tujuan dan sasarannya.

Struktur organisasi bertindak sebagai pekerjaan dan tanggung jawab, aturan kerja dan hubungan serta jalur komunikasi dan pembentukan struktur organisasi berfungsi untuk membagi pekerjaan anggota-anggota organisasi dan dan mengkoordinasikan aktivitas-aktivitas yang dilakukan sehingga semua anggota organisasi dapat diarahkan untuk mencapai tujuan organisasi. Oleh sebab itu struktur organisasi dan *jobdescription* pada PT SKF Indonesia disusun berdasarkan fungsi-fungsi yang dibutuhkan di dalam perusahaan. Struktur organisasi PT SKF Indonesia dapat dilihat pada Gambar IV.1.

STRUKTUR ORGANISASI PT.SKF INDONESIA

As per Nov 15th, 2010



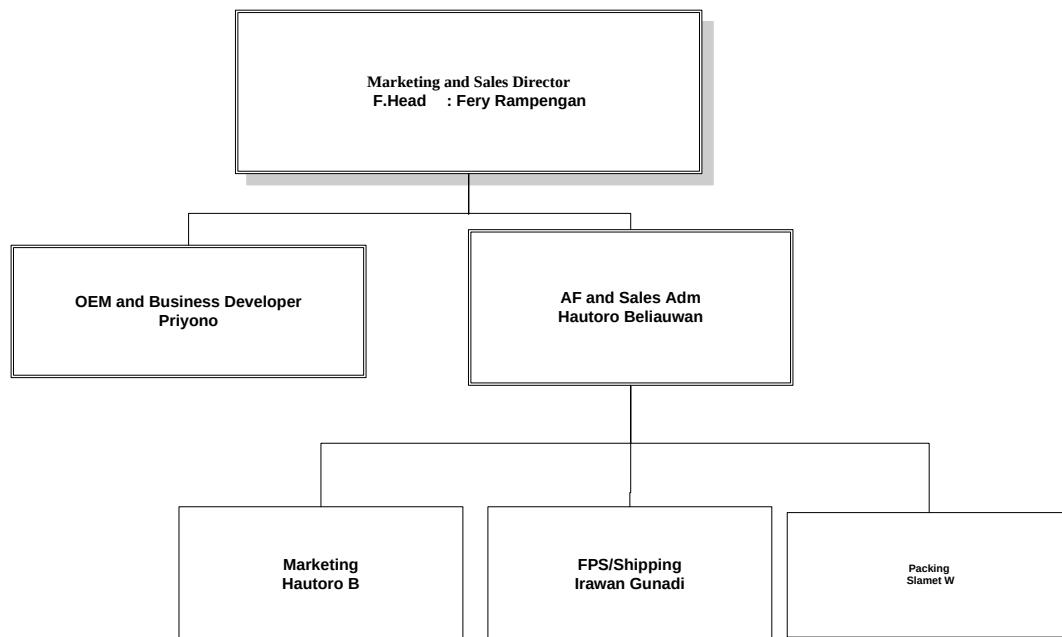
© SKF Group Slide 12 Strictly Confidential

SKF

Gambar IV.1 Struktur Organisasi SKF
(Sumber: PT SKF Indonesia, 2016)

Departemen *Marketing and Sales Director* bertugas menjualkan dan memasarkan produk-produk yang telah diproduksi dan siap dijual dan dipasarkan. Departemen ini disebut juga sebagai ujung tombaknya suatu perusahaan karena departemen ini dapat memberikan sebuah keuntungan ataupun kerugian bagi perusahaan tersebut.

Departemen *Marketing and Sales Director* membawahi divisi *OEM and Business Developer*, dan *After Market and Sales Adm* bertugas memasarkan produk dan menangani semua keluhan dari pelanggan setelah penjualan dilakukan. *After Market and Sales Adm* dipimpin oleh seorang Manajer. Departemen ini mempunyai tiga divisi yaitu *After Market*, *Shipping* dan *Sales Adm*. Divisi *Shipping* dipimpin oleh seorang *supervisor* yang bertugas menangani produk jadi, pengemasan produk, pengiriman produk dan membuat dokumen yang berkaitan dengan produk jadi sebelum sampai ke pelanggan. Sedangkan divisi *After Market* dan divisi *Sales Adm* dipimpin langsung oleh manajer *Marketing and Sales Adm* karena lingkup divisi yang sangat kecil dan sudah kebijakan dari PT SKF Indonesia. Struktur organisasi Departemen *Marketing and Sales Director* dapat dilihat pada Gambar IV.2.



Gambar IV.2 Struktur Organisasi Departemen *Marketing and Sales Director*

(Sumber: PT SKF Indonesia, 2016)

Tugas dari *Marketing* sebagai berikut:

1. Menerima PO (*purchase order*) dari *customer*.
2. Mengcopy PO (*purchase order*) dan menyimpan sebagian arsip PO.
3. Memberikan PO yang telah dicopy ke *shipping*.

Tugas dari *shipping* sebagai berikut:

1. Menerima PO yang telah dicopy dari *marketing*.
2. Membuat SD (*schedule delivery*).
3. Menyimpan arsip SD (*schedule delivery*).
4. Membuat PP (*packaging plan*).
5. Menyimpan sebagian arsip PP dan menyerahkan arsip PP ke *packing*.

Tugas dari *packing* sebagai berikut:

1. Menerima arsip PP dari *shipping*.
2. Melakukan proses *packaging*.
3. Memberikan produk ke pelanggan.

4.2 Produk yang Dihasilkan

SKF mempunyai reputasi unggul dalam memproduksi *bearing* berkualitas tinggi. Selain itu, SKF menyediakan aneka jenis produk yang digunakan dalam berbagai aplikasi industri. Selain produk-produk tradisional, SKF juga memproduksi produk-produk lain yang mencakup teknologi

baru untuk menghasilkan kinerja yang semakin baik dalam dunia industri baru yang semakin menantang ini.

1. *Bearing* Roda Dua (Sepeda motor)

Salah satu produk unggulan PT SKF Indonesia yang memiliki daya jual tinggi di Indonesia adalah *Bearing* untuk sepeda motor. Tingkat permintaan untuk *bearing* jenis ini relatif naik di tiap periode. SKF di Indonesia memproduksi *bearing* untuk beberapa merk sepeda motor ternama, seperti:

a. Honda

Digunakan untuk sepeda motor honda merk supra fit, astrea grand, astrea star, karisma, karisma 125D, karisma X 125D, kirana, legenda, supra, supra V, supra XX, supra X 125, supra X 125DGL max, GL 100 K2, GL pro, GL pro neo tech, mega pro, GL max neo tech, NSR-150 R, tiger 2000, tiger 200 C/W, win, dan scooter vario.

b. Yamaha

Digunakan untuk sepeda motor yamaha merk F-1 Z, F-1 ZR, F-1 ZR C/W, sigma, mio, nouvo, jupiter Z, jupiter MX, scorio, crypton, vega, vega R, dan RX king 150.

c. Suzuki

Digunakan untuk sepeda motor suzuki merk bravo, shogun 110, shogun 110 disk brake, shogun 125, satria R, satria C/W, satria S, smash, smash full chain case, smash DB and full chain, tornado GS, tornado GX, thunder, thunder 125, scooter spin, dan satria F 150.

d. Kawasaki

Digunakan untuk sepeda motor kawasaki merk kaze E, kaze R, kaze VR, blitz, blitz R (DB), sport ninja, sport ninja R C/W, dan ZX 130.

e. Piaggio (vespa).

Digunakan untuk sepeda motor piaggio merk vespa excel, vespa exclusive 2, dan corsa.

2. *Bearing* Roda Empat (Mobil)

Salah satu produk unggulan PT SKF Indonesia yang memiliki daya jual tinggi di Indonesia adalah *bearing* untuk mobil. Tingkat permintaan untuk *bearing* jenis ini relatif naik di tiap periode. SKF di Indonesia memproduksi *bearing* untuk beberapa merk mobil ternama, seperti:

a. Toyota

Digunakan untuk mobil Camry, Corolla, Land Cruiser, Yaris, dan Tacoma.

b. Honda

Digunakan untuk mobil Civic, Cr-v, Odyssey, Accord, dan Ridgeline.

c. Nissan

Digunakan untuk mobil Frontier, Juke, March, X-trail, Titan, Versa, dan X-terra.

d. Mazda

Digunakan untuk mobil Mazda 2, Mazda 3, Mazda 5, Mazda 6, dan cx-9

4.3 Daftar Pelanggan

Dibawah ini adalah rincian pelanggan dari PT SKF Indonesia:

Tabel IV.1 Daftar Pelanggan PT SKF Indonesia

No.	Nama Perusahaan	Alamat
1.	PT Astra Honda Motor	Jl. Laksda Yos Sudarso, Sunter I, Jakarta
2.	PT Suzuki Indomobil Motor	Jl. Diponegoro KM 38,2 Tambun,Bekasi
3.	PT Yamaha Motor Indonesia	Jl. DR. KRT. Rajiman Widyodiningrat - (Jl. Raya Bekasi Km 23) Pulo Gadung, Jakarta
4.	PT Kawasaki Motor Indonesia	Jl. Printis Kemerdekaan Km 20,Pegangsaan Dua Kelapa Gading,Jakarta Utara
5.	PT Astra Daihatsu Motor	Jl. Gaya Motor III No.5 Sunter II Jakrta Utara
6.	PT Isuzu Astra Motor Indonesia	Jl. Danau Sunter Utara Blok O-3 Kav 30 Sunter Jakarta
7.	PT Krama Yudha Tiga Berlian Motors (Mitsubishi Motors)	Jl. Jend A Yani,Proyek Pulo Mas, Jakarta
8.	PT Piaggio Indonesia	Jl. Jendral Sudirman Kav.45-46,Jakarta
9.	PT Astra Otoparts	Jl. Raya Pegangsaan Dua KM 2.2. Kelapa Gading Jakarta Utara
10.	PT Musashi Auto Part Indonesia	Kawasan Industry Ejip Industrial Park Kav 3/J-2 Cikarang Barat,Bekasi
11.	PT Akashi Wahana Indonesia	Jl. Pegangsaan Dua Bl A-1 Pegangsaan Dua, Kelapa Gading Jakarta Utara
12.	PT Mitsuba Indonesia Pipe Parts	Kawasan Industri MM-2100 Blok NN-12 Jatiwangi Cikarang Barat,Bekasi

No.	Nama Perusahaan	Alamat
13.	PT Chemco Harapan Nusantara	Jl. Jababeka Raya Blok F No. 19 – 28 Cikarang, Bekasi

(Sumber: PT SKF Indonesia, 2016)

4.4 Sistem *Packaging Plan* yang Sedang Berjalan

Sistem informasi *shipping* merupakan sistem yang membahas bagian pengemasan, pengiriman, hingga mengeluarkan surat jalan untuk pengiriman ke pelanggan, khususnya dalam *packaging plan* yang dibuat sebelum produk dikirim ke pelanggan. Analisis sistem merupakan tahap awal sebelum tahap perancangan, hal ini dilakukan untuk mengidentifikasi masalah dan kekurangan dari sistem yang ada.

Sistem *packaging plan* adalah sistem yang digunakan dalam menentukan tanggal mulai sampai selesai pengemasan hingga menentukan nama operator yang dibutuhkan dalam mengemasi produk. Adapun proses-prosesnya sebagai berikut :

1. Pelanggan (*customer*) membuat PO untuk diserahkan ke bagian *marketing*.
2. *Marketing* menerima PO dari Pelanggan lalu PO di *fotocopy* dan diserahkan ke bagian *shipping*.
3. *Shipping* membuat *schedule delivery* (SD) setelah SD dibuat selanjutnya membuat *packaging plan* (PP) untuk diserahkan ke bagian *packing*.
4. *Packing* menerima *packaging plan* (PP) dari *shipping*.
5. *Packing* melakukan proses pengemasan (*packaging*).
6. *Packing* melakukan proses memberikan *product* ke customer.
7. *Customer* menerima produk dari bagian *packing*. Hal ini dapat dilihat pada gambar IV.3 Prosedur *Packaging Plan*.

Adapun contoh-contoh dokumen dari prosedur *Packaging Plan* diatas sebagai berikut:

1. Dokumen *Purchase Order*

Dokumen *Purchase Order* (PO) adalah surat yang di berikan oleh pelanggan untuk pemesanan produk *bearing*. Data yang terdapat pada dokumen ini yaitu nomor PO, tanggal PO, keterangan tentang vendor, nomor *part*, deskripsi part, jumlah dan harga. Dokumen PO dilengkapi dengan tandatangan dari pihak yang menerima, yang menyerahkan, yang mengeluarkan dan yang mengetahui. Contoh dokumen dapat dilihat pada Gambar IV.4

No.	Nomor Part	Deskripsi Part	Jumlah	Price
1.	62015-8701-0001-0001	BRO. BALL. RADIAL 6204	1000	27.0000

Gambar IV.4 *Purchase Order* (PO)
(Sumber: PT SKF Indonesia, 2016)

2. Dokumen *Schedule Delivery*

Schedule Delivery (SD) adalah dokumen penjadwalan pengiriman yang dibuat oleh bagian *Shipping* setelah PO diterima dari bagian *Marketing*. Data yang terdapat pada dokumen ini yaitu *id customer*, *name customer*, nomor PO, tanggal PO, *final variant*, jumlah PO dan *Plan* berdasarkan tanggal. Contoh dokumen *schedule delivery* (SD) dapat dilihat pada Gambar IV.5.

S/D

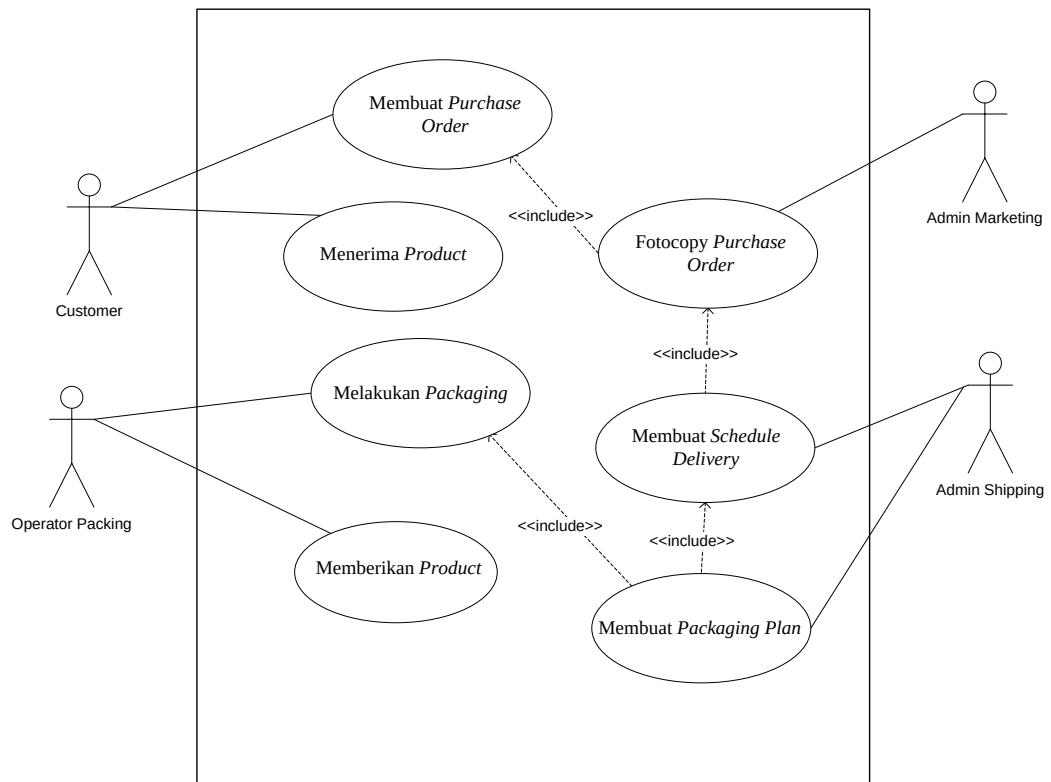


SKF		SALES Apr-14													
Cus 500 PT. Astra Honda Motor		PO # 00-Jan-00													
No.	Final Variant	PO	4th Plan	8th Plan	10th Plan	11th Plan	15th Plan	16th Plan	17th Plan	22nd Plan	23rd Plan	24th Plan	25th Plan	29th Plan	Act
1	6000/VK315	350													
2	6001/VK315	1.000													
10	6201/VCS26	42.000			4.500	4.500	4.500	4.500	4.500	6.000	4.500	4.500	4.500		
11	6201-RS1/MTVC526A	70.000		10.000	10.000	10.000	10.000	10.000	10.000	5.000	5.000	5.000	5.000		
12	6201-RS1/MTVC526	60.000			10.000	10.000	10.000		5.000	5.000	10.000	5.000			
14	6201-2RS1/VCS26	200								200					
16	6202/VCS26	4.000								4.000					
18	6203/VCS26	13.000		13.000											
22	6203-RS1/MTVC526	24.000	10.000	10.000	4.000										
23	6203-2RS1/VCS26	100		100											
25	6204/VK315	3.000											3.000		
30	6300-RS1/MTVC526	15.000											15.000		
32	6301-RS1/MTVC526A	30.000	10.000	10.000	10.000										
33	6301-RS1/MTVC526	50.000			10.000	10.000	10.000	10.000	10.000						
35	6301-2RS1/VCS26	100	100												
37	6302-RS1/MTVC526A	6.000					5.000	1.000							
38	6302-RS1/MTVC526	6.000					5.000	1.000							
41	6303-2RS1/VK315	1.100						1.000						1.000	1.000
42	6304/DBV315A	5.000			5.000				1.000	100					
43	6304/DBV315	10.000					5.000	5.000				7.350			
45	BB1-3260A/VK315	200				200						200			
46	63/22/HN3C40SMVK315	4.000							4.000				4.000		
47	63/22/HN3C40SMVK315	7.000										7.000		500	500
51	BB1-4708/VK315	1.200													
52	BB1-4724/VK315	3.000													
53	BB1-4710/VK315	1.000													
		357.250	20.200	43.000	53.500	34.700	44.500	40.500	35.500	25.500	19.500	25.050	26.500	1.500	1.500

Gambar IV.5 *Schedule Delivery*
(Sumber: PT SKF Indonesia, 2016)

3. Dokumen *Packaging Plan*

Packaging Plan (PP) adalah dokumen perencanaan pengemasan produk *bearing* yang akan dikirim ke pelanggan. Data yang terdapat pada dokumen ini yaitu tanggal pembuatan dokumen *packaging plan*, *type*, *pack*, *customer*, *qty*, tanggal mulai, tanggal selesai dan operator packing. Contoh dokumen *packaging plan* (PP) dapat dilihat pada gambar IV.6.



Gambar IV.7 Use Case Diagram Sistem Packaging Plan
(Sumber: Pengolahan Data, 2016)

Sistem *packaging plan* pada PT SKF Indonesia dapat digambarkan dalam bentuk *actor* dan *use case*. Berikut ini adalah tabel definisi *actor* dan definisi *use case* pada sistem *packaging plan*.

Tabel IV.2 Definisi Actor dalam Sistem Packaging Plan

No	Actor	Keterangan
1.	Customer	Melakukan kegiatan pemesanan produk dan pengiriman PO untuk diserahkan ke bagian <i>marketing</i> .
2.	Admin Marketing	Melakukan kegiatan <i>fotocopy</i> PO yang telah diterima dari <i>customer</i> dan menyerahkan ke bagian <i>shipping</i> .
3.	Admin Shipping	Melakukan kegiatan membuat <i>schedule delivery</i> (SD) berdasarkan <i>fotocopy</i> PO yang diberikan oleh bagian <i>marketing</i> . Selain itu, bagian <i>shipping</i> juga melakukan kegiatan membuat <i>packaging plan</i> (PP) dan mengirim ke bagian <i>packing</i> .

Tabel IV.2 Definisi Actor dalam Sistem *Packaging Plan* (Lanjutan)

No.	Actor	Keterangan
4.	Operator <i>Packing</i>	Melakukan kegiatan penerimaan <i>packaging plan</i> (PP) dari bagian <i>shipping</i> . Setelah itu melakukan proses <i>packaging</i> dan memberikan produk ke <i>customer</i> .

(Sumber: Pengolahan Data, 2017)

Tabel IV.4 Definisi use case dalam sistem *Packaging Plan*

No.	Use Case	Keterangan
1.	Membuat <i>Purchase Order</i>	Melakukan kegiatan pembuatan PO yang dilakukan oleh <i>customer</i> untuk pemesanan produk.
2.	<i>Fotocopy Purchase Order</i>	Melakukan kegiatan di bagian Marketing untuk memfotokopi PO yang telah diterima dari <i>customer</i> .
3.	Membuat <i>Schedule Delivery</i>	Melakukan kegiatan di bagian <i>Shipping</i> untuk membuat <i>schedule delivery</i> (SD) setelah <i>fotocopy</i> PO diterima dari bagian Marketing.
4.	Membuat PP	Melakukan kegiatan di bagian <i>shipping</i> untuk membuat <i>packaging plan</i> (PP) setelah <i>schedule delivery</i> (SD) dibuat.
5.	Proses <i>Packaging</i>	Melakukan kegiatan proses <i>packaging</i> dari bagian <i>shipping</i> untuk segera ditindaklanjuti.
6.	Memberikan <i>Product</i>	Melakukan kegiatan proses memberikan <i>product</i> untuk diberikan ke <i>customer</i> .
7.	Menerima <i>Product</i>	Melakukan kegiatan proses menerima <i>product</i> dari operator <i>packing</i> .

(Sumber: Pengolahan Data, 2017)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mendapatkan informasi kebutuhan sistem yang diharapkan dapat diusulkan perbaikan terhadap sistem yang diamati. Berdasarkan hasil pengolahan data dan analisis sistem, dibutuhkan perancangan sistem informasi *packaging plan* pada Divisi *Shipping* PT SKF Indonesia yang dapat mempermudah dalam mengolah dan memproses data *packaging plan*. Berikut adalah daftar kebutuhan sistem informasi *packaging plan*:

Tabel V.1 Identifikasi Kebutuhan Sistem

Kebutuhan Sistem	
<i>Project Name</i>	Sistem persediaan <i>packaging plan</i> pada Divisi <i>Shipping</i> PT SKF Indonesia
<i>Business Need</i>	1. Terwujudnya sebuah sistem informasi yang dapat memudahkan bagian <i>packing</i> untuk dapat menginput data <i>packaging plan</i> ke dalam sistem, sehingga lebih cepat dan tepat dalam proses pembuatannya. 2. Terwujudnya sebuah sistem informasi yang dapat mempermudah proses penyimpanan data dan dapat mencegah terjadinya kehilangan data karena setiap data yang dimasukan telah tersimpan dengan aman di dalam database.
<i>Business Requirement</i>	Memberikan sistem informasi <i>packaging plan</i> yang dapat membantu Bagian <i>Shipping</i> .

Tabel V.1 Identifikasi Kebutuhan Sistem (Lanjutan)

Kebutuhan Sistem	
<i>Business Value</i>	1. Memudahkan dalam pembuatan laporan <i>packaging plan</i> sehingga menjadi lebih informatif dan mudah dimengerti. 2. Mampu menyimpan data di <i>Shipping</i> dengan baik dan

	aman di basis data.
<i>Special Issues Or Constrains</i>	Pengendalian data <i>packaging plan</i> hanya pada Divisi <i>Shipping</i> .

Sumber: Pengolahan Data (2017)

Dalam pembangunan Sistem Informasi *Packaging Plan* pada Divisi *Shipping* Berbasis Web pada PT SKF Indonesia, ada tiga tahapan yang digunakan yaitu:

1. Pembuatan model sistem digunakan untuk mengetahui alur sistem yang diusulkan, menggunakan *tools* pemodelan *Unified Modelling Language* (UML).
2. Pembuatan model data dengan Kamus Data.
3. Perancangan program yang diusulkan dengan pembuatan *Hierarchy Plus Input-Proses-Output* (HIPO), *flowchart* program dan pembuatan aplikasi dengan menggunakan bahasa pemrograman PHP, dan perangkat lunak basis data MySQL.

5.2 Analisis Kebutuhan Rinci Sistem

Analisis kebutuhan rinci sistem untuk sistem informasi *packaging plan* pada PT SKF Indonesia dijelaskan pada Tabel V.2

Tabel V.2 Kebutuhan Rinci Sistem

No.	Kebutuhan Rinci Sistem	Uraian
1	Mengelola data <i>user</i>	Proses yang dapat dilakukan oleh kepala <i>shipping</i> pada mengelola data <i>user</i> yaitu: a. Kepala <i>shipping</i> dapat menambah data <i>user</i>. b. Kepala <i>shipping</i> dapat mencari, mengubah dan menghapus data <i>user</i>.
2	Mengelola data produk	Proses yang dapat dilakukan oleh kepala <i>shipping</i>

		pada mengelola data produk yaitu: - Kepala <i>shipping</i> dapat menambah data produk. - Kepala <i>shipping</i> dapat mencari, mengubah dan menghapus data produk.
3	Mengelola data <i>customer</i>	Proses yang dapat dilakukan oleh kepala <i>shipping</i> pada mengelola data <i>customer</i> yaitu: - Kepala <i>shipping</i> dapat menambah data <i>customer</i>. - Kepala <i>shipping</i> dapat mencari, mengubah dan menghapus data <i>customer</i>.
4	Mengelola data operator	Proses yang dapat dilakukan oleh kepala <i>shipping</i> dalam mengelola data operator yaitu: - Kepala <i>shipping</i> dapat menambah data operator. - Kepala <i>shipping</i> dapat mencari, mengubah dan menghapus data operator.
5	Menginput data <i>purchase order</i>	Proses yang dapat dilakukan oleh admin <i>shipping</i> dalam menginput data <i>purchase order</i> yaitu admin

Tabel V.2 Kebutuhan Rinci Sistem

No.	Kebutuhan Rinci Sistem	Uraian
6	Membuat data <i>schedule delivery</i>	Proses yang dapat dilakukan oleh admin <i>shipping</i> dalam membuat data <i>schedule delivery</i> yaitu: - Admin <i>shipping</i> dapat menambah data <i>schedule delivery</i>. - Admin <i>shipping</i> dapat mencari, mengubah dan menghapus data <i>schedule delivery</i>.
7	Membuat data <i>packaging plan</i>	Proses yang dapat dilakukan oleh admin <i>shipping</i> dalam membuat data <i>packaging plan</i> yaitu: - Admin <i>shipping</i> dapat menambah data <i>packaging plan</i>. - Admin <i>shipping</i> dapat mencari, mengubah dan menghapus data <i>packaging plan</i>.

8	Menyetujui data <i>packaging plan</i>	Proses yang dapat dilakukan oleh kepala <i>shipping</i> dalam menyetujui data <i>packaging plan</i> yaitu melihat data <i>packaging plan</i> terlebih dahulu sebelum menyetujui lalu apabila sudah sesuai kepala <i>shipping</i> segera menyetujui data <i>packaging plan</i> .
9	Mencetak data <i>packaging plan</i>	Proses yang dilakukan oleh admin <i>shipping</i> selanjutnya yaitu mencetak data <i>packaging plan</i> .

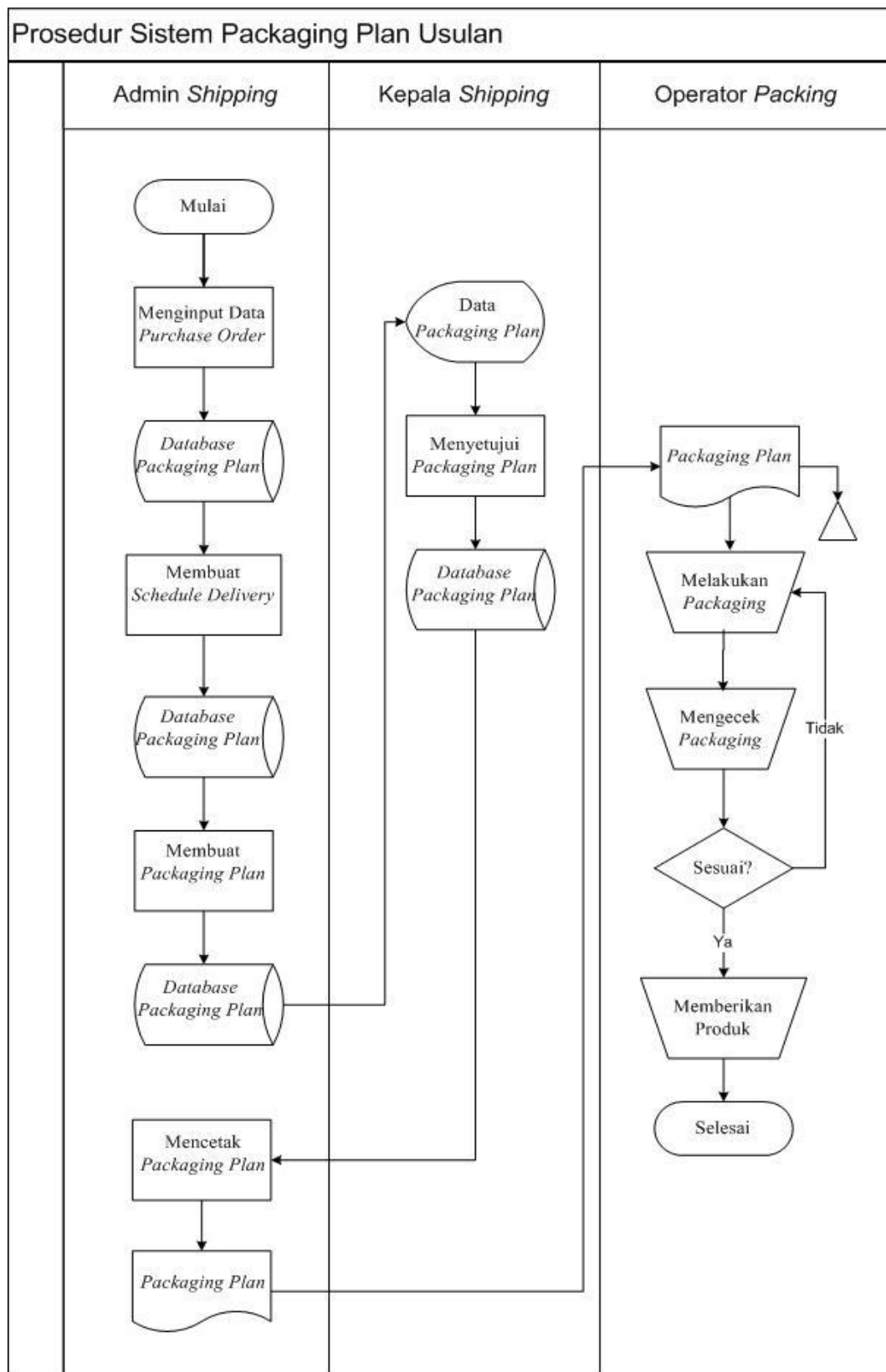
Sumber: Pengolahan Data (2017)

5.3 Prosedur Sistem Informasi *Packaging Plan* Usulan

Pada sistem informasi *packaging plan* yang diusulkan ini yang terlibat hanya pada bagian Departemen *Shipping* dan *Packing*. Sistem *Packaging Plan* digunakan untuk menentukan tanggal mulai dan tanggal selesai pengemasan hingga menentukan nama operator yang dibutuhkan dalam mengemasi produk. Adapun Proses-prosesnya sebagai berikut :

8. Bagian admin *shipping* menerima *purchase order* (PO) dari *customer*.
9. Lalu bagian admin *shipping* menginput data *purchase order* dan masuk ke *database*.
10. Bagian admin *shipping* membuat *schedule delivery* (SD).
11. Bagian admin *shipping* memasukkan data *schedule delivery* (SD) ke *database*.
12. Bagian admin *shipping* membuat *packaging plan* (PP) dan menyimpan pada sistem informasi *packaging plan*.
13. Kepala *shipping* melihat *packaging plan* (PP).
14. Kepala *shipping* menyetujui *packaging plan* (PP) dan menyimpan pada sistem informasi *packaging plan*.
15. Kemudian admin *shipping* mencetak *packaging plan* (PP).
16. Bagian operator *packing* menerima *packaging plan* dan melakukan proses *packaging*.
17. Operator *packing* melakukan pengecekan setelah melakukan proses *packaging*.

18. Operator *packing* setelah melakukan *packaging* lalu menentukan keputusan apabila *packaging* sesuai maka tahap selanjutnya memberikan produk ke *customer* namun bila tidak sesuai maka kembali melakukan proses *packaging*. Hal ini dapat dilihat pada gambar V.1 Prosedur *Packaging Plan* Usulan.



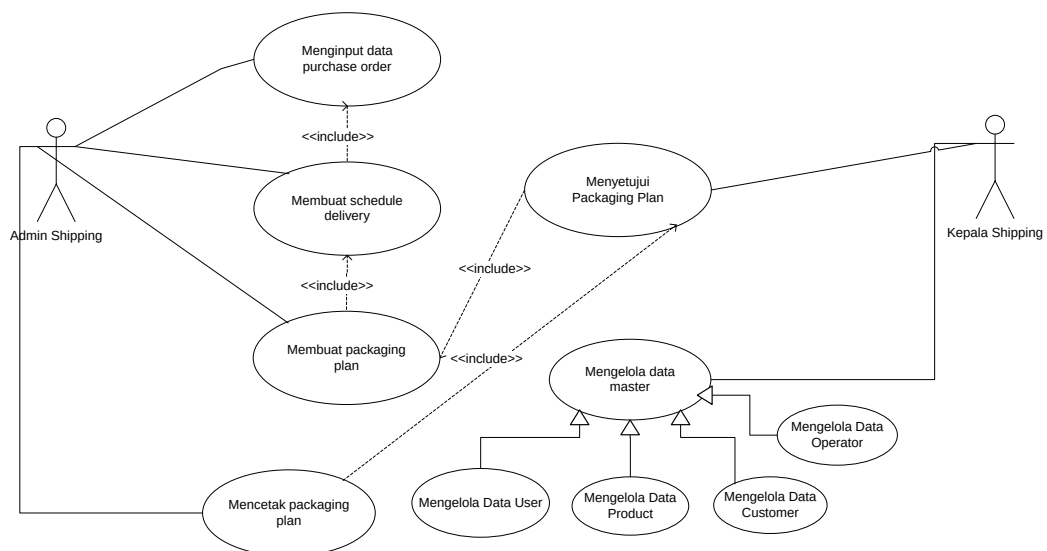
Gambar V.1 Prosedur Sistem *Packaging Plan* Usulan
(Sumber: Hasil Analisis Data, 2017)

5.4 Model Sistem yang digunakan

Adapun model sistem yang digunakan adalah dengan *Unified Modeling Language* (UML) dalam hal ini suatu rancangan sistem *Packaging Plan* yang antara lain dibuat suatu *use case diagram*, *use case description*, *activity diagram*, *sequence diagram*, *class diagram* dan *deployment diagram*.

5.4.1 Use Case Diagram

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Rancangan *use case diagram* sistem informasi *packaging plan* pada Divisi *packing* yang diusulkan dapat dilihat pada Gambar V.2



Gambar V.2 Use Case Diagram Sistem Informasi *Packaging Plan* Usulan
(Sumber: Hasil Analisis Data, 2017)

Perancangan *use case diagram* sistem informasi *packaging plan* usulan dapat dijelaskan sebagai berikut:

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi *packaging plan* pada Divisi *Shipping* di PT SKF Indonesia dapat dilihat pada Tabel V.3 berikut:

Tabel V.3 Definisi Aktor Use Case Diagram Sistem Usulan

No.	Aktor	Deskripsi
1.	Admin Shipping	Admin Shipping adalah aktor yang memiliki hak untuk mengelola <i>purchase order</i> , membuat <i>schedule delivery</i> dan

		membuat <i>packaging plan</i> hingga mencetak <i>packaging plan</i> .
2.	Kepala <i>Shipping</i>	Kepala <i>Shipping</i> adalah aktor yang bertugas untuk menyetujui <i>packaging plan</i> dan mengelola data master.

Sumber: Pengolahan Data (2017)

2. Definisi *Use Case*

Pendefinisian *use case* pada sistem informasi *packaging plan* usulan dapat dilihat pada Tabel V.4:

Tabel V.4 Definisi *Use Case Diagram* Sistem Usulan

No.	<i>Use Case</i>	Deskripsi
1.	Mengelola Data Master	<i>Use case</i> ini menggambarkan proses pengelolaan data <i>master</i> yang terdiri dari data <i>user</i> , <i>product</i> , <i>customer</i> , dan <i>operator</i> .
2.	Menginput <i>purchase order</i>	Proses mengelola <i>purchase order</i> dilakukan oleh bagian admin <i>shipping</i> .
3.	Membuat <i>schedule delivery</i>	Proses membuat <i>schedule delivery</i> dilakukan oleh bagian admin <i>shipping</i> .
4.	Membuat <i>packaging plan</i>	Proses membuat <i>packaging plan</i> dilakukan oleh bagian admin <i>shipping</i> .
5.	Menyetujui <i>packaging plan</i>	Proses menyetujui <i>packaging plan</i> dilakukan oleh kepala <i>shipping</i> .
6.	Mencetak <i>packaging plan</i>	Proses mencetak <i>packaging plan</i> dilakukan oleh bagian admin <i>shipping</i> .

3. Sumber: Pengolahan Data (2017)

Skenario jalannya masing-masing *use case* pada sistem informasi *packaging plan* dapat dilihat pada *point* berikut:

a. *Use Case* Mengelola Data Master

Berikut skenario *use case* mengelola data *master* yang terdapat pada Tabel V.5:

Nama <i>Use Case</i>	Mengelola Data Master
<i>Primary Actor</i>	Kepala <i>Shipping</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses

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

	pengolahan data master yang terdiri dari data <i>user</i> , data <i>product</i> , data <i>customer</i> dan data <i>operator</i> .
<i>Relationship</i>	Generalisasi: Data <i>user</i> , data <i>product</i> , data <i>customer</i> dan data <i>operator</i> .
<i>Normal Flow of Events</i>	1. Setelah melakukan <i>login</i>, maka kepala <i>shipping</i> masuk ke tampilan menu utama. 2. Kepala <i>shipping</i> memilih menu data master. 3. Kepala <i>shipping</i> memilih sub menu yang terdiri dari data <i>user</i>, data <i>product</i>, data <i>customer</i> dan data <i>operator</i>. 4. Kepala <i>shipping</i> melakukan proses tambah, ubah, cari dan hapus data di basis data masing-masing pada data master.

Sumber: Pengolahan Data (2017)

b. *Use Case* Mengelola *Purchase Order* (PO)

Berikut adalah skenario *use case* menginput *purchase order* yang terdapat pada Tabel V.6:

Tabel V.6 Skenario *Use Case* Menginput *Purchase Order*

Nama <i>Use Case</i>	Menginput <i>Purchase Order</i>
<i>Primary Actor</i>	Admin <i>Shipping</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses memasukkan data <i>purchase order</i> .
<i>Relationship</i>	<i>Association</i> : Admin <i>Shipping</i>
<i>Normal Flow of Events</i>	1. Setelah melakukan <i>login</i>, maka admin <i>shipping</i> masuk ke tampilan menu utama. 2. Admin <i>shipping</i> memilih menu transaksi, lalu memilih <i>sub menu Purchase Order</i> (PO).

	3. Selanjutnya sistem menampilkan <i>form</i> transaksi <i>purchase order</i> .. 4. Admin <i>shipping</i> memasukkan data <i>purchase order</i> ke dalam <i>form purchase order</i> .
--	--

Sumber: Pengolahan Data (2017)

c. *Use Case Membuat Schedule Delivery (SD)*

Berikut adalah skenario *use case* membuat *schedule delivery* (SD) yang terdapat pada Tabel V.7:

Tabel V.7 Skenario *Use Case* Membuat *Schedule Delivery* (Lanjutan)

Nama Use Case	Membuat <i>Schedule Delivery</i>
<i>Primary Actor</i>	Admin <i>Shipping</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses membuat <i>schedule delivery</i> .
<i>Relationship</i>	<i>Association</i> : Admin <i>Shipping</i>
<i>Normal Flow of Events</i>	1. Setelah melakukan <i>login</i> , maka admin <i>shipping</i> masuk ke tampilan menu utama. 2. Sistem menampilkan <i>form schedule delivery</i> . 3. Admin <i>shipping</i> kemudian memasukkan <i>schedule delivery</i> dan menyimpan ke basis data.

Sumber: Pengolahan Data (2017)

d. *Use Case Membuat Packaging Plan (PP)*

Berikut adalah skenario *use case* membuat *packaging Plan* (PP) yang terdapat pada Tabel V.8:

Tl V.108Skenario *Use Case* Membuat *Packaging Plan*

Nama Use Case	Membuat <i>Packaging Plan</i>
<i>Primary Actor</i>	Admin <i>Shipping</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses membuat <i>packaging plan</i> .

Tabel V.8 Skenario *Use Case* Membuat *Packaging Plan*

<i>Relationship</i>	<i>Association: Admin Shipping</i>
<i>Normal Flow of Events</i>	1. Setelah melakukan <i>login</i>, maka admin <i>shipping</i> masuk ke tampilan menu utama. 2. Sistem menampilkan <i>form packaging plan</i>. 3. Admin <i>shipping</i> kemudian memasukan data <i>packaging plan</i> dan menyimpan ke basis data.

Sumber: Pengolahan Data (2017)

e. *Use Case* Menyetujui *Packaging Plan* (PP)

Berikut adalah skenario *use case* menyetujui *packaging Plan* (PP) yang terdapat pada Tabel V.9:

Tabel V.9 Skenario *Use Case* Menyetujui *Packaging Plan*

Nama Use Case	Mencetak <i>Packaging Plan</i>
<i>Primary Actor</i>	Kepala <i>Shipping</i>
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan proses menyetujui <i>packaging plan</i> .
<i>Relationship</i>	<i>Association: Kepala Shipping</i>
<i>Normal Flow of Events</i>	1. Setelah melakukan <i>login</i>, maka kepala <i>shipping</i> masuk ke tampilan menu utama. 2. Sistem menampilkan <i>form packaging plan</i>. 3. Kepala <i>shipping</i> menyetujui data <i>packaging plan</i>.

Sumber: Pengolahan Data (2017)

f. *Use Case* Mencetak *Packaging Plan* (PP)

Berikut adalah skenario *use case* membuat *packaging Plan* (PP) yang terdapat pada Tabel V.10:

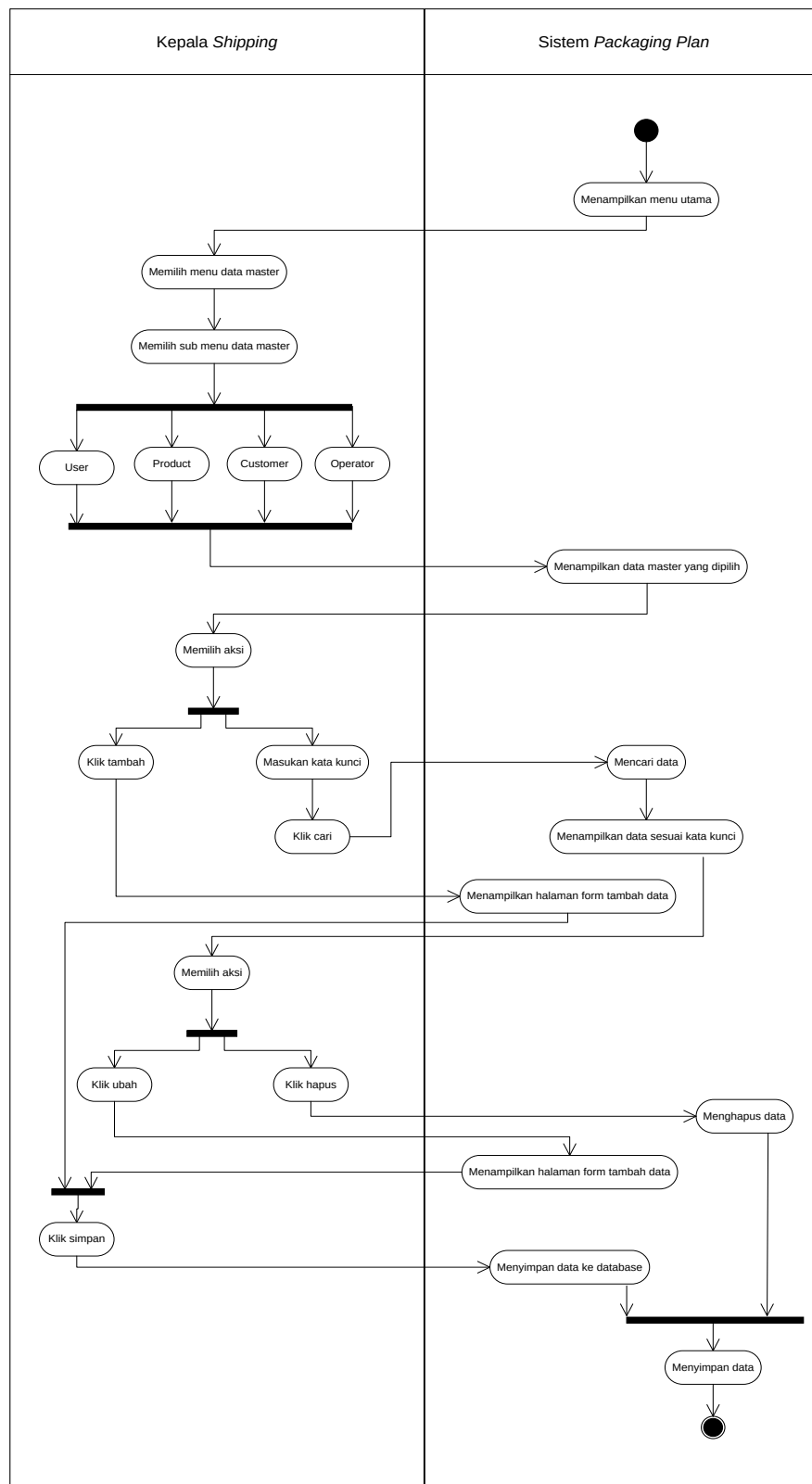
Nama Use Case	Mencetak Packaging Plan
<i>Primary Actor</i>	<i>Admin Shipping</i>
<i>Use Case Description</i>	<i>Use case ini menggambarkan proses mencetak packaging plan.</i>
<i>Relationship</i>	<i>Association: Admin Shipping</i>
<i>Normal Flow of Events</i>	1. Setelah melakukan <i>login</i>, maka admin <i>shipping</i> masuk ke tampilan menu utama. 2. Sistem menampilkan <i>form packaging plan</i>. 3. Admin <i>shipping</i> mencetak data <i>packaging plan</i>.

5.4.2 Sumber: Pengolahan Data (2017)

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 *packaging plan* ini.

1. Activity Diagram Megelola Data Master

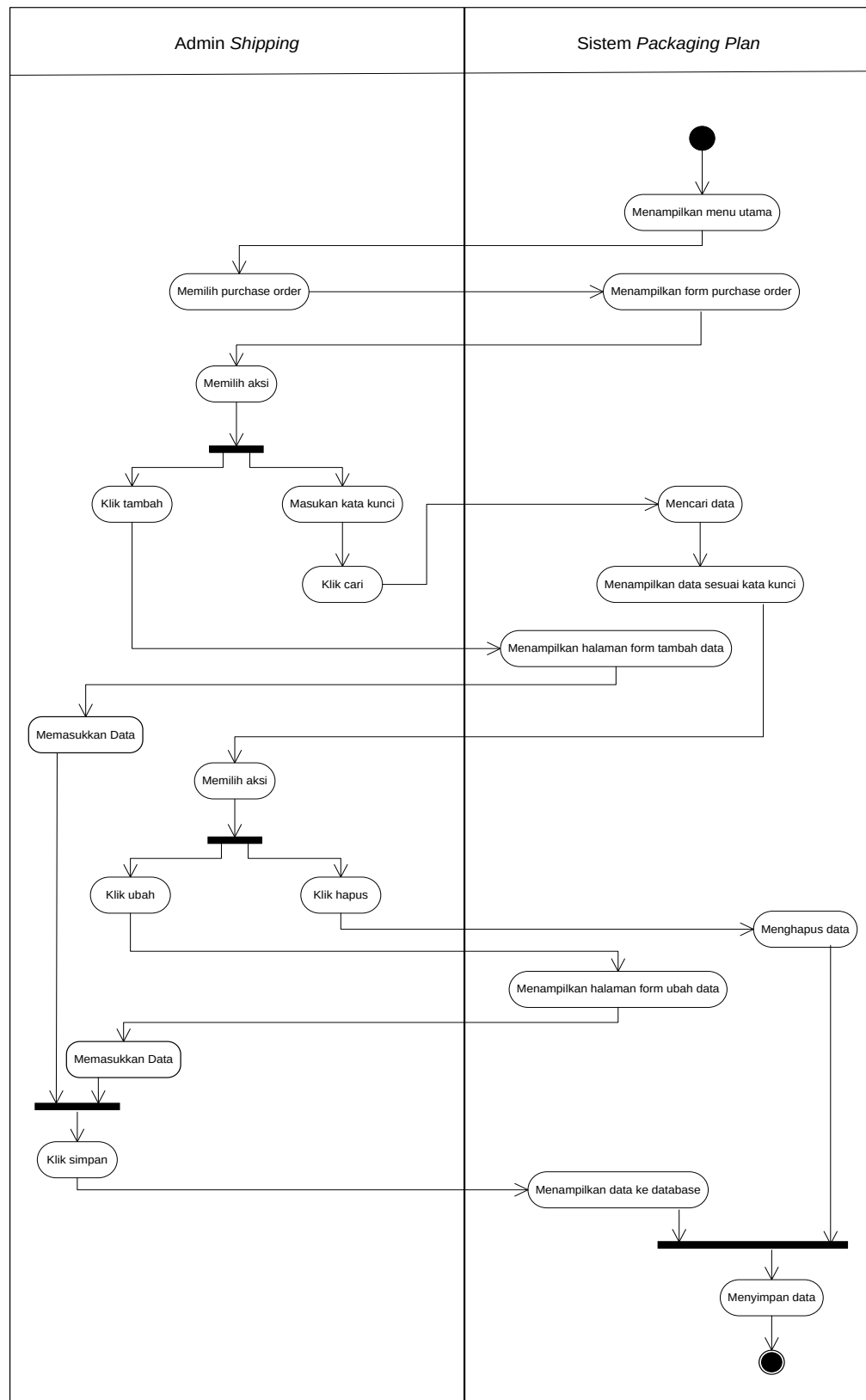
Activity diagram berikut ini menunjukkan aktivitas yang dilakukan dalam mengelola data master, dapat dilihat pada Gambar V.3:



Gambar V.3 Activity Diagram Mengelola Data Master
Sumber: Pengolahan Data (2017)

2. *Activity Diagram Mengelola Data Purchase Order*

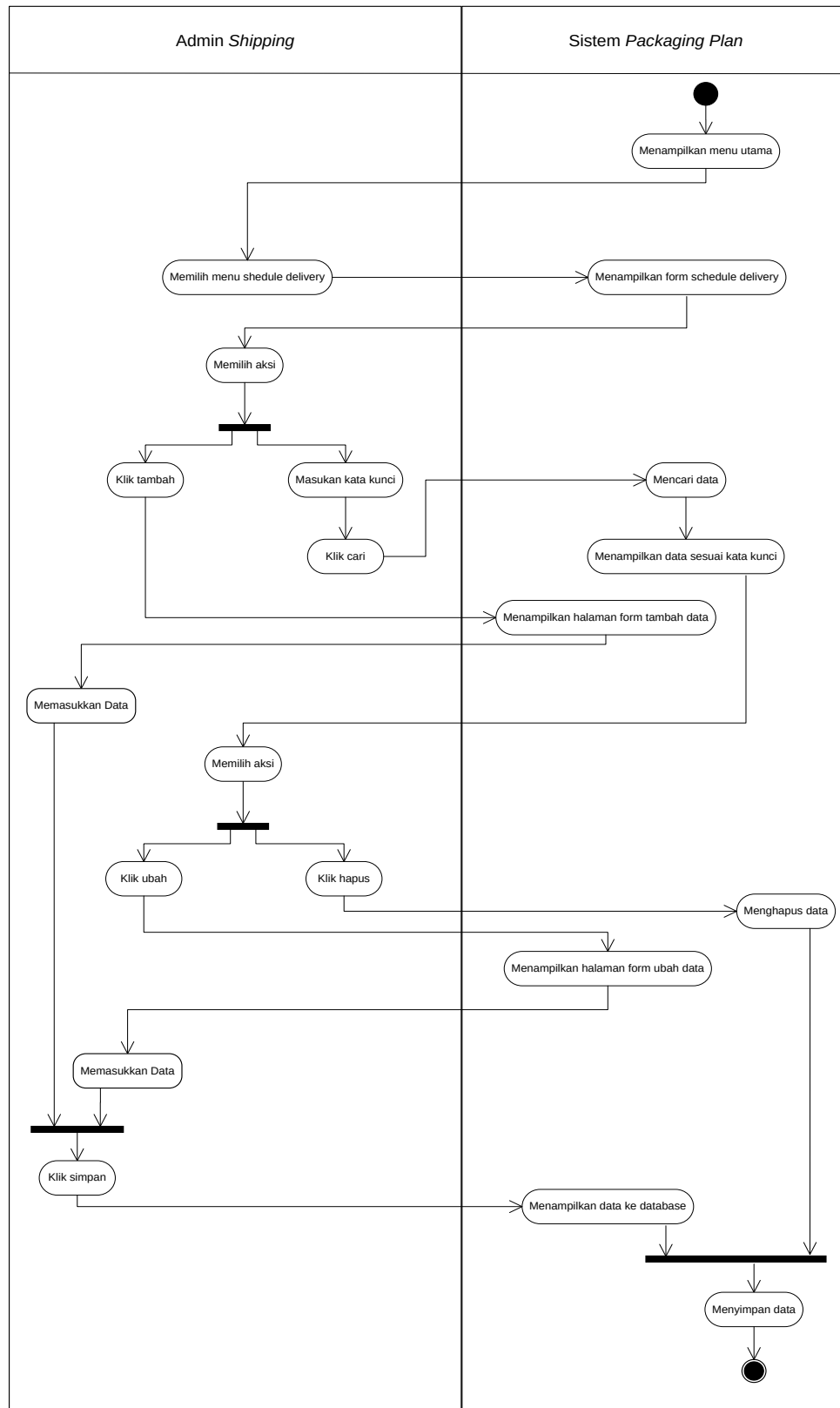
Activity diagram mengelola data *purchase order* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* mengelola data *purchase order* dapat dilihat pada Gambar V.4 berikut ini:



Gambar V.4 Activity Diagram Mengelola Data Purchase Order
Sumber: Pengolahan Data (2017)

3. *Activity Diagram Membuat Schedule Delivery*

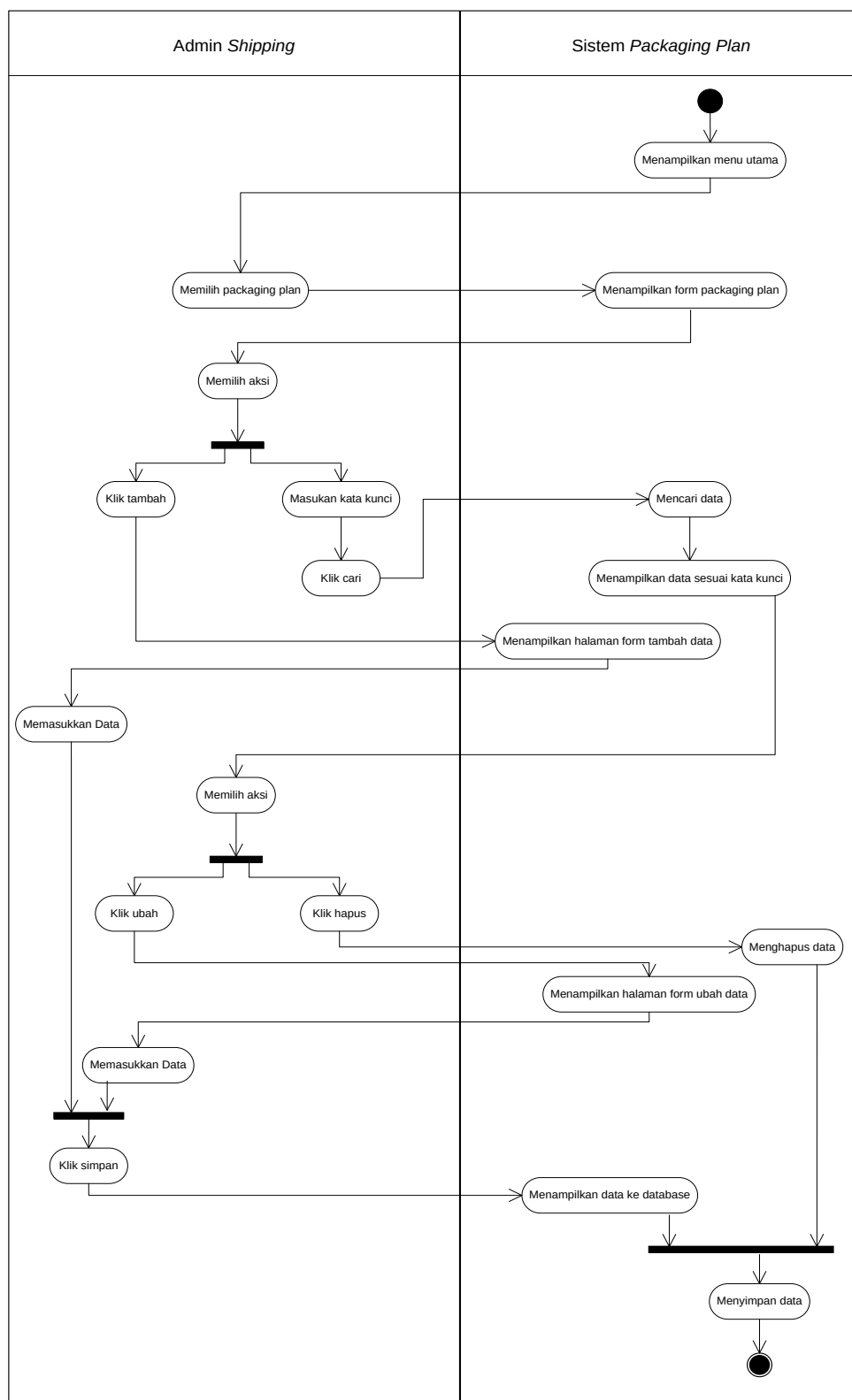
Activity diagram mencetak *schedule delivery* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* mencetak *schedule delivery* dapat dilihat pada Gambar V.5 berikut ini:



Gambar V.5 Activity Diagram Membuat Schedule Delivery
Sumber: Pengolahan Data (2017)

4. *Activity Diagram Membuat Packaging Plan*

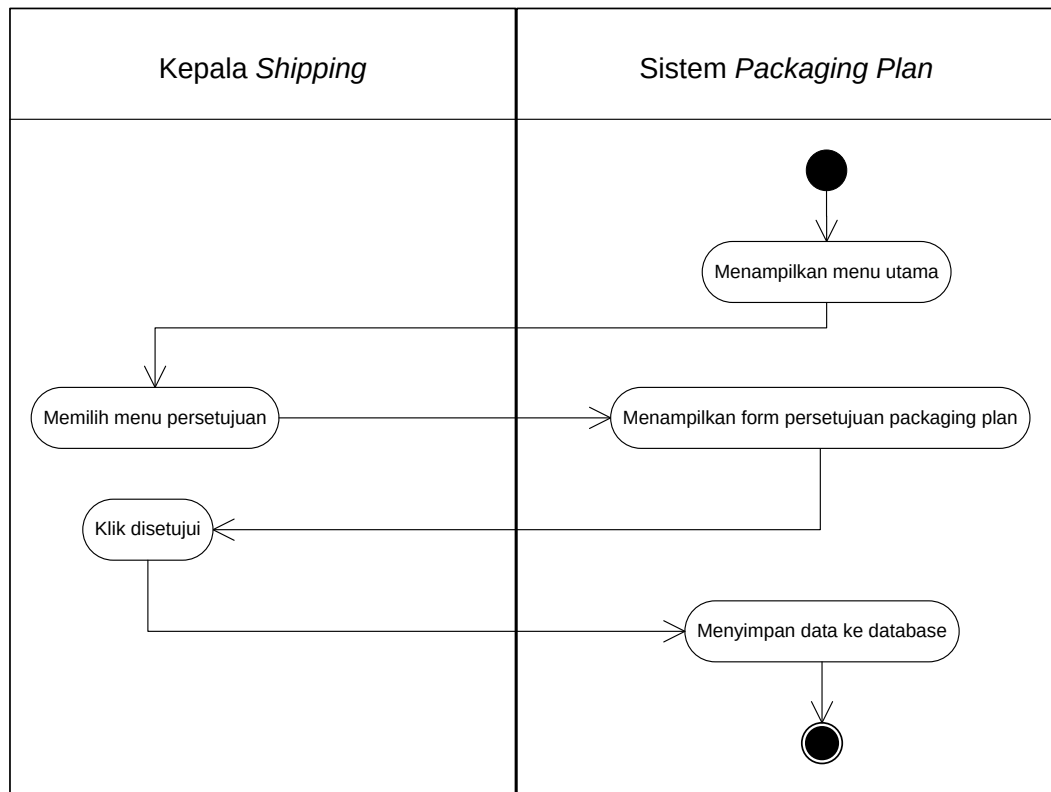
Activity diagram membuat *packaging plan* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* membuat *packaging plan* dapat dilihat pada Gambar V.6 berikut ini:



Gambar V.6 *Activity Diagram* Membuat *Packaging Plan*
Sumber: Pengolahan Data (2017)

5. *Activity Diagram Menyetujui Packaging Plan*

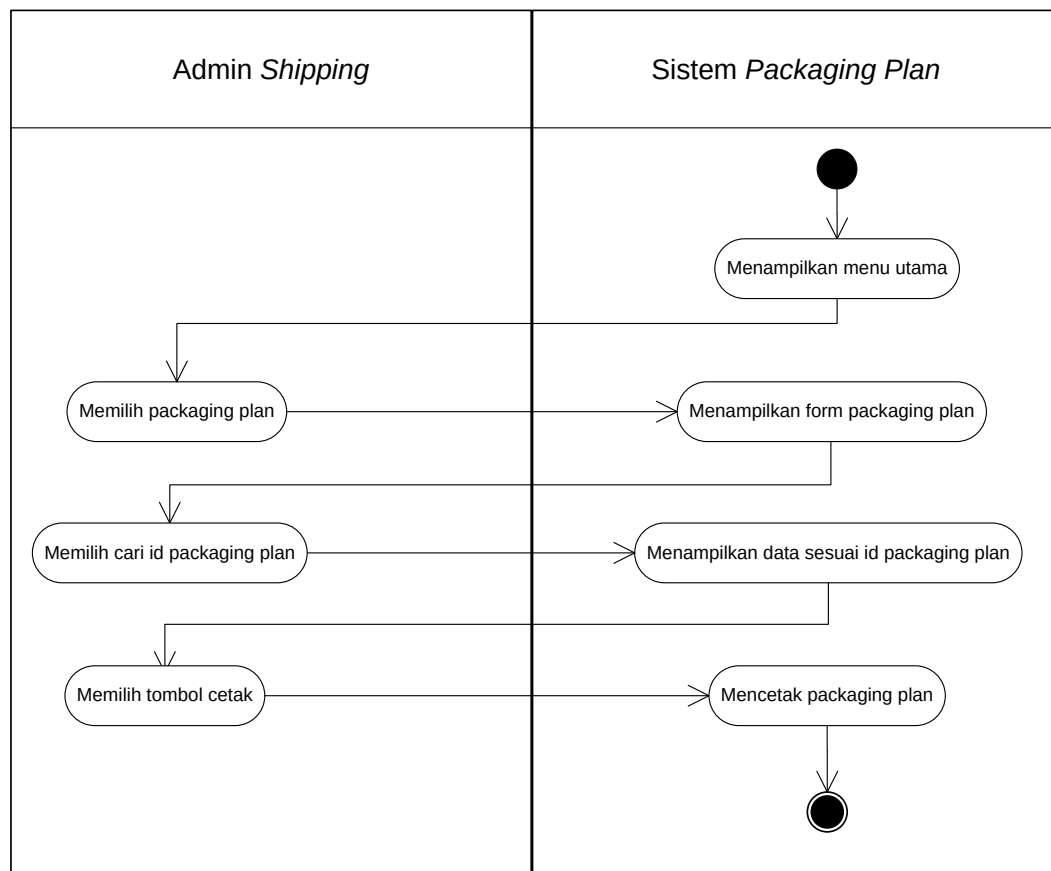
Activity diagram menyetujui *packaging plan* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* menyetujui *packaging plan* dapat dilihat pada Gambar V.7 berikut ini:



Gambar V.7 *Activity Diagram Menyetujui Packaging Plan*
Sumber: Pengolahan Data (2017)

6. *Activity Diagram Mencetak Packaging Plan*

Activity diagram mencetak *packaging plan* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada aplikasi pengolahan data. *Activity diagram* mencetak *packaging plan* dapat dilihat pada Gambar V.8 berikut ini:



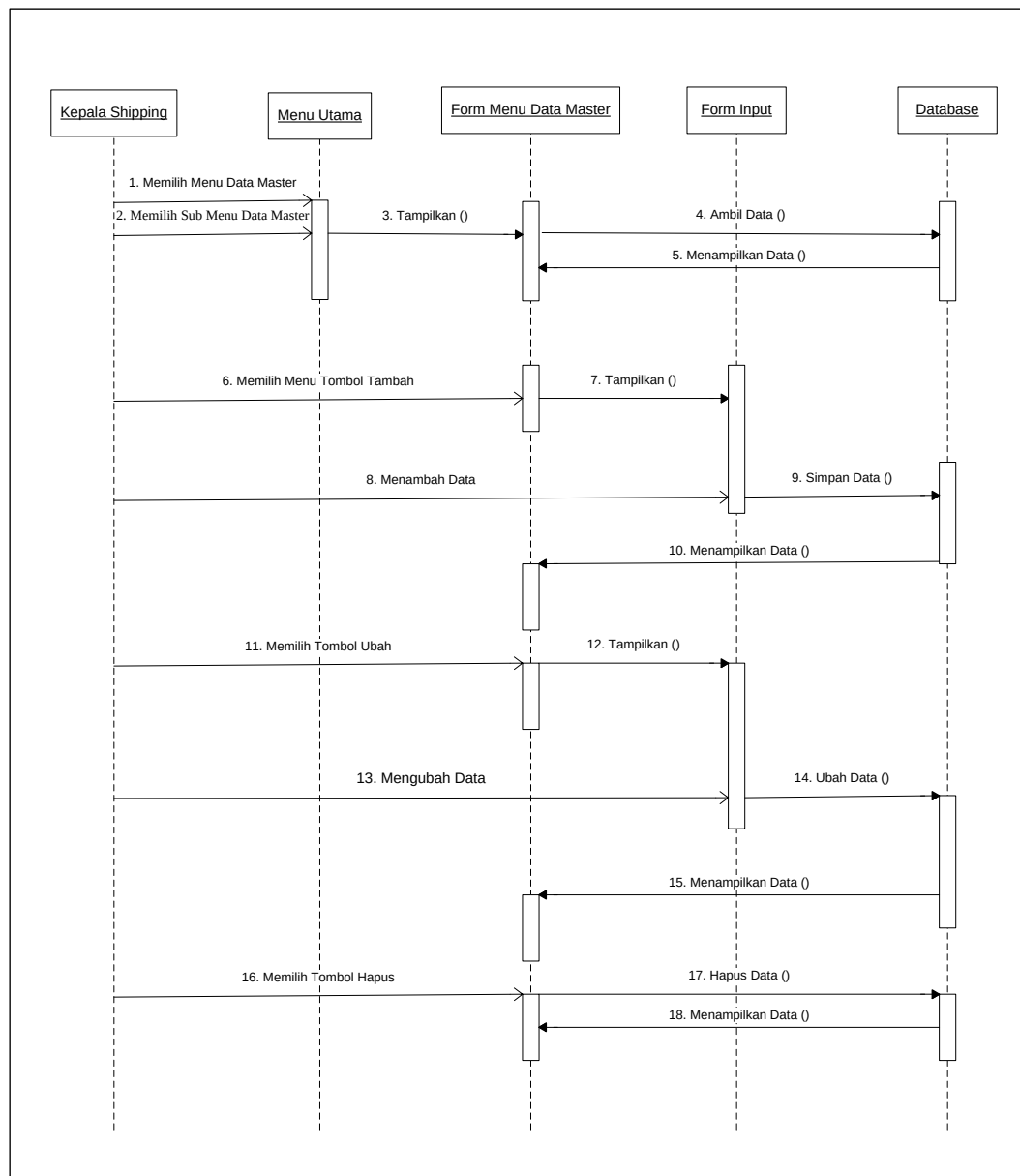
Gambar V.8 Activity Diagram Mencetak Packaging Plan
Sumber: Pengolahan Data (2017)

5.4.3 Sequence Diagram

Sequence diagram menggambarkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu, dimana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada diagram. Hubungan yang ada pada gambar 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 sistem informasi *packaging plan*:

1. Sequence Diagram Mengelola Data Master

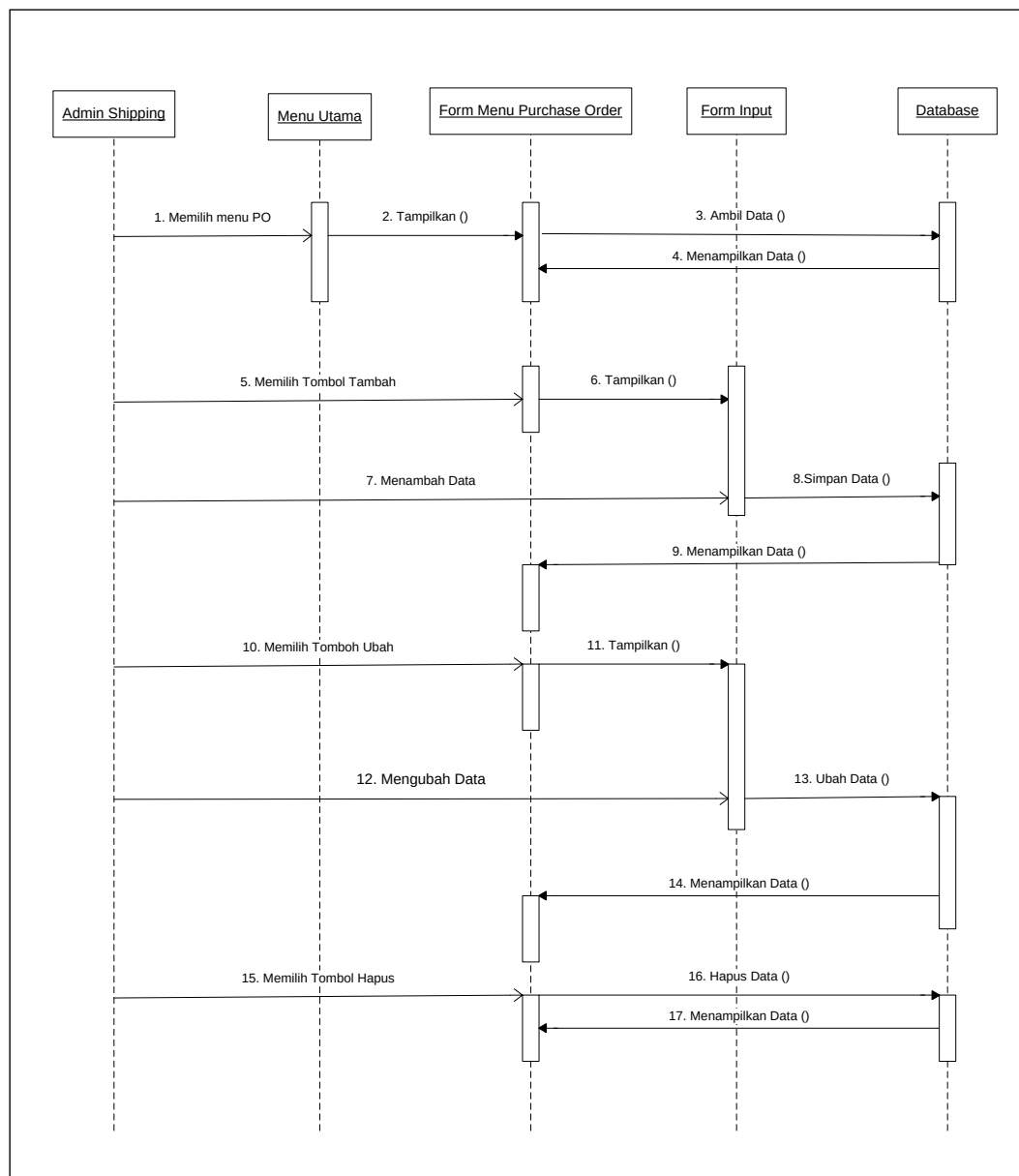
Sequence diagram mengelola data *master* menjelaskan sebuah *sequence diagram* dalam proses transaksi menambah, menghapus, mencari dan mengubah data *master*. Adapun *sequence diagram* dari *use case* mengelola data *master* dapat dilihat pada Gambar V.9.



Gambar V.9 *Sequence Diagram* Mengelola Data Master
Sumber: Hasil Analisis Data (2017)

2. *Sequence Diagram* Mengelola Data Purchase Order

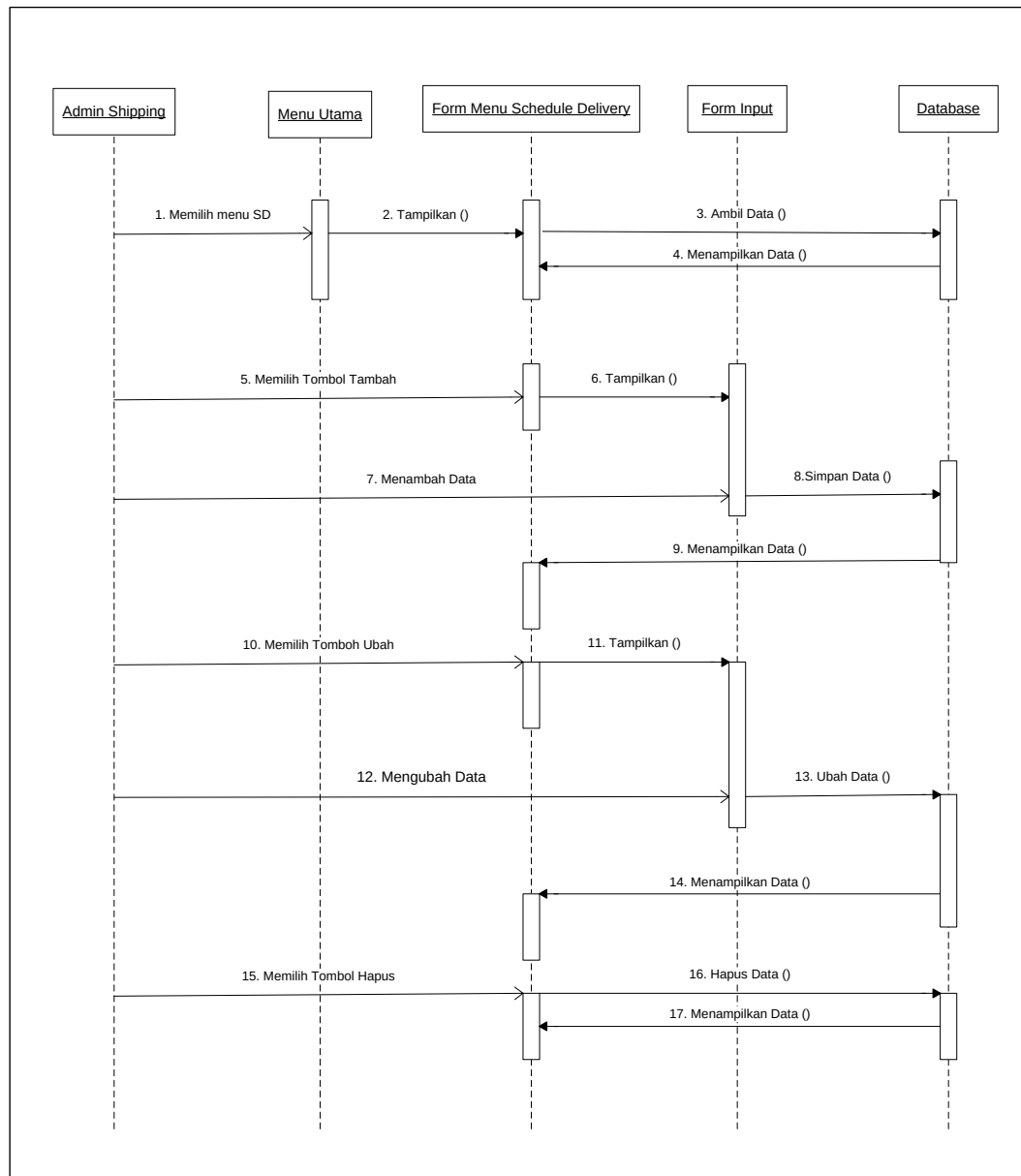
Sequence diagram mengelola data purchase order menjelaskan sebuah *sequence diagram* dalam proses transaksi menambah, menghapus, dan mengubah data purchase order. Adapun *sequence diagram* dari use case mengelola data purchase order dapat dilihat pada Gambar V.10.



Gambar V.10 Sequence Diagram Mengelola Data PO
Sumber: Hasil Analisis Data (2017)

3. Sequence Diagram Membuat Data Schedule Delivery

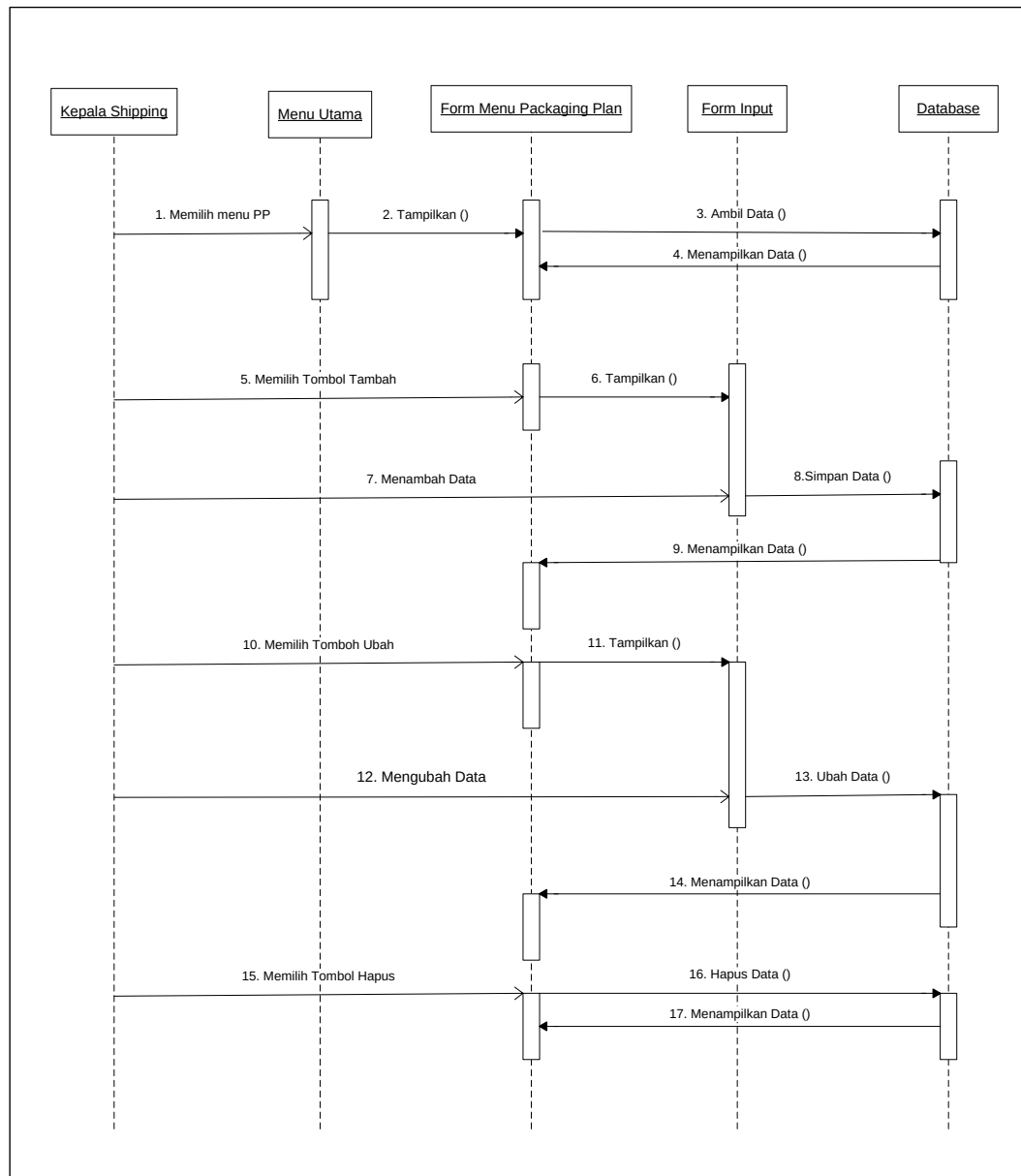
Sequence diagram membuat data *schedule delivery* menjelaskan sebuah sequence diagram dalam proses transaksi menambah, menghapus, mencari dan mengubah data *schedule delivery*. Adapun sequence diagram dari use case membuat data *schedule delivery* dapat dilihat pada Gambar V.11.



Gambar V.11 *Sequence Diagram* Membuat Data *Schedule Delivery*
 Sumber: Hasil Analisis Data (2017)

4. *Sequence Diagram* Membuat Data *Packaging Plan*

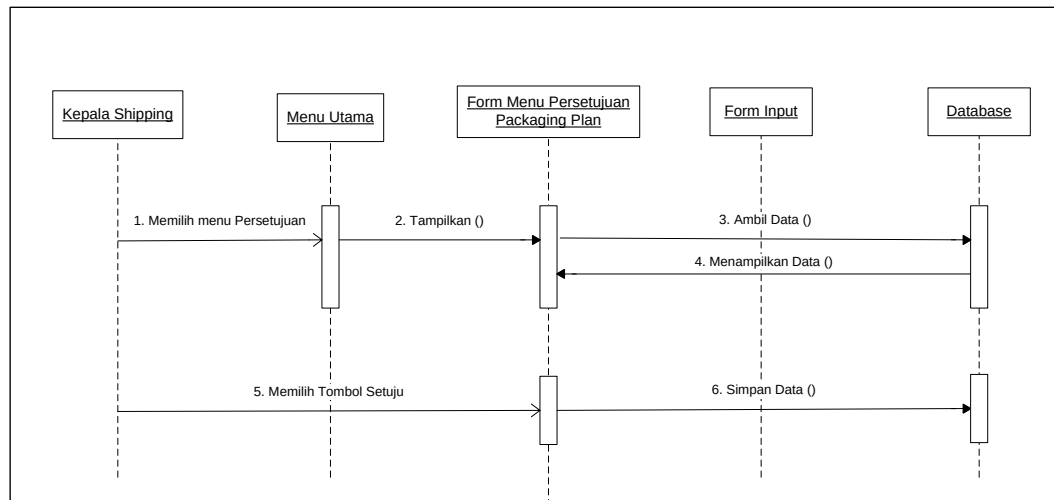
Sequence diagram membuat data *packaging plan* menjelaskan sebuah *sequence diagram* dalam proses transaksi menambah, menghapus, mencari dan mengubah data *packaging plan*. Adapun *sequence diagram* dari *use case* membuat data *packaging plan* dapat dilihat pada Gambar V.12.



Gambar V.12 *Sequence Diagram* Membuat Data *Packaging Plan*
 Sumber: Hasil Analisis Data (2017)

5. *Sequence Diagram* Menyetujui Data *Packaging Plan*

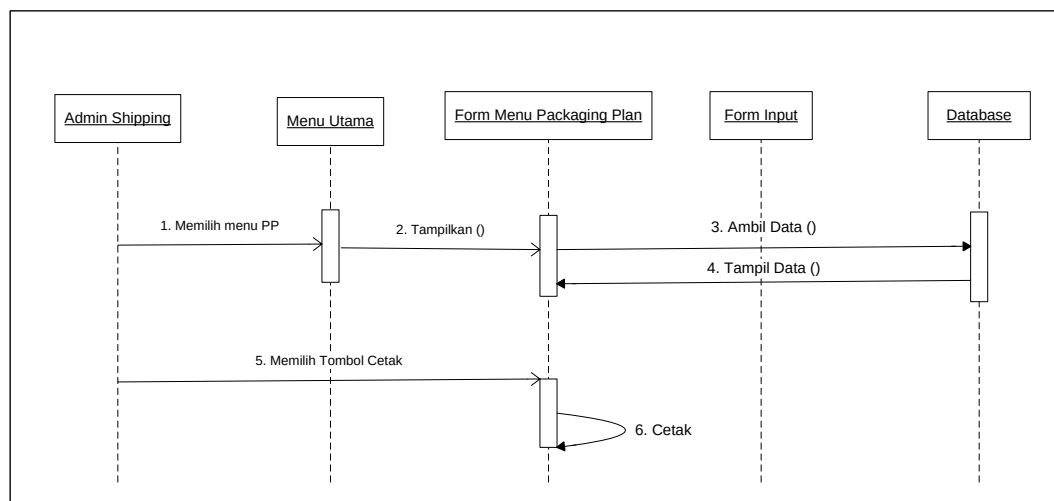
Sequence diagram menyetujui data *packaging plan* menjelaskan sebuah *sequence diagram* dalam proses menambah, menghapus, mencari dan mengubah data *packaging plan*. Adapun *sequence diagram* dari *use case* menyetujui data *packaging plan* dapat dilihat pada Gambar V.13.



Gambar V.13 *Sequence Diagram* Menyetujui Data *Packaging Plan*
 Sumber: Hasil Analisis Data (2017)

6. *Sequence Diagram* Mencetak Data *Packaging Plan*

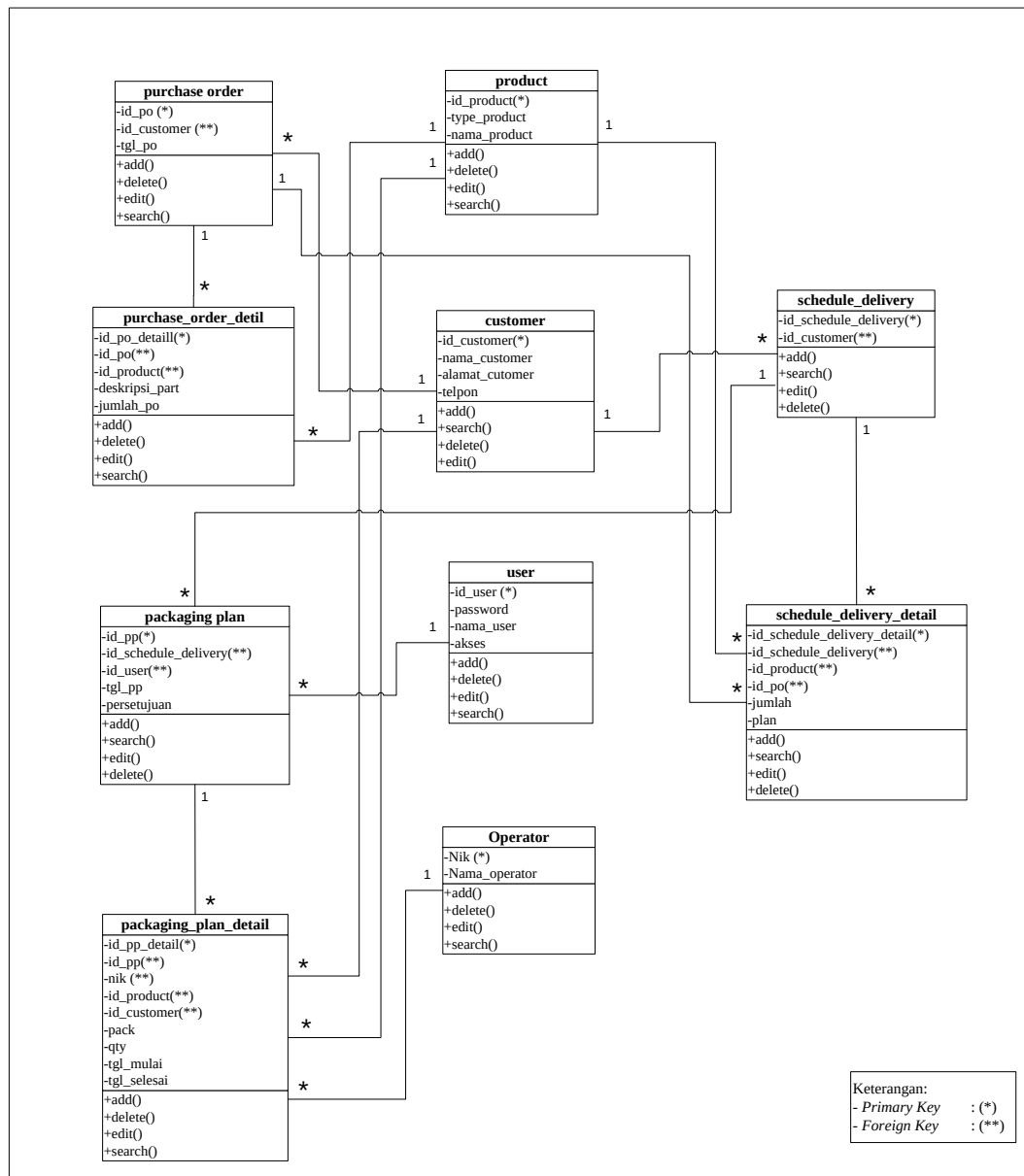
Sequence diagram mencetak data *packaging plan* menjelaskan sebuah *sequence diagram* dalam proses transaksi *packaging plan*. Adapun *sequence diagram* dari use case mencetak data *packaging plan* dapat dilihat pada Gambar V.14.



Gambar V.14 *Sequence Diagram* Mencetak Data *Packaging Plan*
 Sumber: Hasil Analisis Data (2017)

5.4.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan *detail* tiap-tiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi *packaging plan* yang diusulkan dapat dilihat pada Gambar V.15 sebagai berikut:



Gambar V.15 *Class Diagram* Usulan
 Sumber: Hasil Analisis Data (2017)

5.4.5 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 sistem informasi *packaging plan*:

1. Spesifikasi Tabel *User*

Nama Tabel : *User*

Fungsi : Untuk menyimpan data *user*

Tipe : File data *master*

Tabel V.11 Tabel *User*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id <i>User</i>	<i>id_user</i>	<i>Char</i>	7	<i>Primary key</i>
2.	<i>Password</i>	<i>Password</i>	<i>Varchar</i>	5	
3.	Nama <i>User</i>	<i>nama_user</i>	<i>Varchar</i>	25	
4.	Akses	Akses	<i>Char</i>	35	

Sumber: Hasil Analisis Data (2017)

2. Spesifikasi Tabel *Product*

Nama Tabel : *Product*

Fungsi : Untuk menyimpan data *product*

Tipe : File data *master*

Tabel V.12 Tabel *Product*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id <i>Product</i>	<i>id_product</i>	<i>Char</i>	7	<i>Primary key</i>
2.	Type <i>Product</i>	<i>type_product</i>	<i>Varchar</i>	35	
3.	Nama <i>Product</i>	<i>nama_product</i>	<i>Varchar</i>	50	

Sumber: Hasil Analisis Data (2017)

3. Spesifikasi Tabel *Customer*

Nama Tabel : *Customer*

Fungsi : Untuk menyimpan data bahan baku

Tipe : File data *master*

Tabel V.13 Tabel *Customer*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id <i>Customer</i>	id_customer	Char	7	Primary key
2.	Nama <i>Customer</i>	nama_customer	Varchar	25	
3.	Alamat	Alamat	Varchar	25	
4.	Telpon	Telpon	Integer	10	

Sumber: Hasil Analisis Data (2017)

4. Spesifikasi Tabel *Operator*

Nama Tabel : *Operator*

Fungsi : Untuk menyimpan data operator

Tipe : File data operator

Tabel V.14 Tabel *Operator*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Nik	Nik	Char	16	Primary Key
2.	Nama Operator	Nama_operator	Varchar	25	

Sumber: Hasil Analisis Data (2017)

5. Spesifikasi Tabel *Purchase Order*

Nama Tabel : *Purchase Order*

Fungsi : Untuk menyimpan data *purchase order*

Tipe : File data *purchase order*

Tabel V.15 Tabel *Purchase Order*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
-----	-------------	---------	------	---------	------------

1.	Id PO	id_po	Char	10	Primary key
2.	Id Customer	id_customer	Char	10	
3.	Tgl_PO	tgl_po	Varchar	15	

Sumber: Hasil Analisis Data (2017)

6. Spesifikasi Tabel *Purchase Order Detail*

Nama Tabel : *Purchase Order Detail*

Fungsi : Untuk menyimpan data *purchase order detail*

Tipe : File data *purchase order detail*

Tabel V.16 Tabel *purchase order detail*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id PO Detail	id_po_detail	Char	7	Primary key
2.	Id PO	id_po	Char	7	Foreign Key
3.	Id Product	id_product	Char	7	Foreign Key
4.	Deskripsi Part	deskripsi_part	Varchar	25	
5.	Jumlah PO	jumlah_po	Integer	10	

Sumber: Hasil Analisis Data (2017)

7. Spesifikasi Tabel *Schedule Delivery*

Nama Tabel : *Schedule Delivery*

Fungsi : Untuk menyimpan data *schedule delivery*

Tipe : File data *schedule delivery*

Tabel V.17 Tabel *Schedule Delivery*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id Schedule Delivery	id_sd	Char	7	Primary key
2.	Id Customer	id_customer	Char	7	Foreign Key

Sumber: Hasil Analisis Data (2017)

8. Spesifikasi Tabel *Schedule Delivery Detail*Nama Tabel : *Schedule Delivery Detail*Fungsi : Untuk menyimpan data *schedule delivery detail*Tipe : File data *schedule delivery detail*Tabel V.18 Tabel *Schedule Delivery Detail*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id SD Detail	id_sd_detail	Char	7	Primary key
2.	Id SD	id_sd	Char	7	Foreign Key
3.	Id Product	id_product	Char	7	Foreign Key
4.	Id PO	id_po	Char	7	Foreign Key
5.	Jumlah	jumlah	Varchar	10	

Sumber: Hasil Analisis Data (2017)

9. Spesifikasi Tabel *Packaging Plan*Nama Tabel : *Packaging Plan*Fungsi : Untuk menyimpan data *packaging plan*Tipe : File data *packaging plan*Tabel V.19 Tabel *Packaging Plan*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	id PP	id_pp	Char	7	Primary key
2.	Id SD	id_sd	Char	7	Foreign Key
3.	Id User	id_user	Char	10	Foreign Key
4.	Tgl PP	tgl_pp	Date		
5.	Persetujuan	persetujuan	Int	1	

Sumber: Hasil Analisis Data (2017)

10. Spesifikasi Tabel *Packaging Plan Detail*Nama Tabel : *Packaging Plan Detail*Fungsi : Untuk menyimpan data *packaging plan detail*Tipe : File data *packaging plan detail*

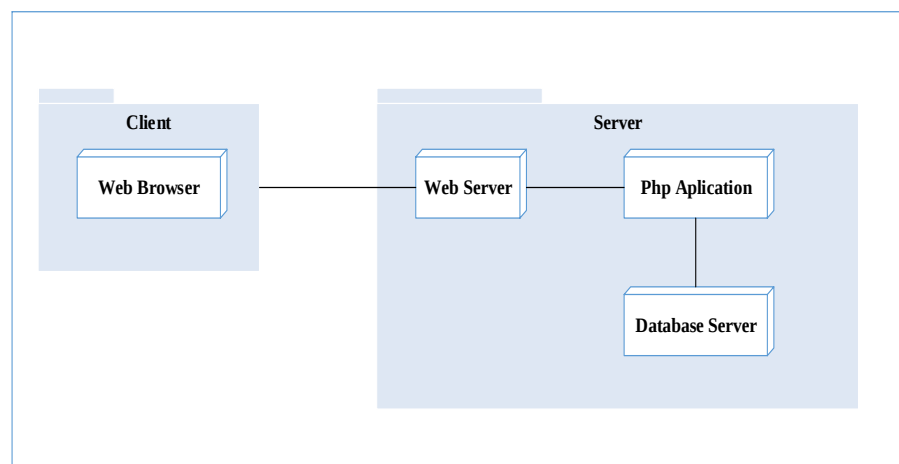
Tabel V.20 Tabel *Packaging Plan Detail*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Id PP Detail	id_pp_detail	Char	7	Primary key
2.	Id PP	id_pp	Char	7	Foreign Key
3.	Nik	Nik	Char	16	Foreign Key
4.	Id_Product	id_product	Char	7	Foreign Key
5.	Id_Customer	id_customer	Char	7	Foreign Key
6.	Pack	Pack	Varchar	10	
7.	Qty	Qty	Integer	30	
8.	Tgl Mulai	tgl_mulai	Date		
9.	Tgl Selesai	tgl_selesai	Date		

Sumber: Hasil Analisis Data (2017)

5.4.6 *Deployment Diagram*

Deployment diagram digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur yang menggambarkan arsitektur fisik dari sistem, seperti *web server* dan semua perangkat lunak tambahan pendukung. *Deployment diagram* sistem informasi persediaan bahan baku yang diusulkan dapat dilihat pada Gambar V.16.

Gambar V.16 *Deployment Diagram*

Sumber: Hasil Analisis Data (2017)

Berikut adalah penjelasan Gambar V.16 *deployment diagram* sistem informasi *packaging plan*:

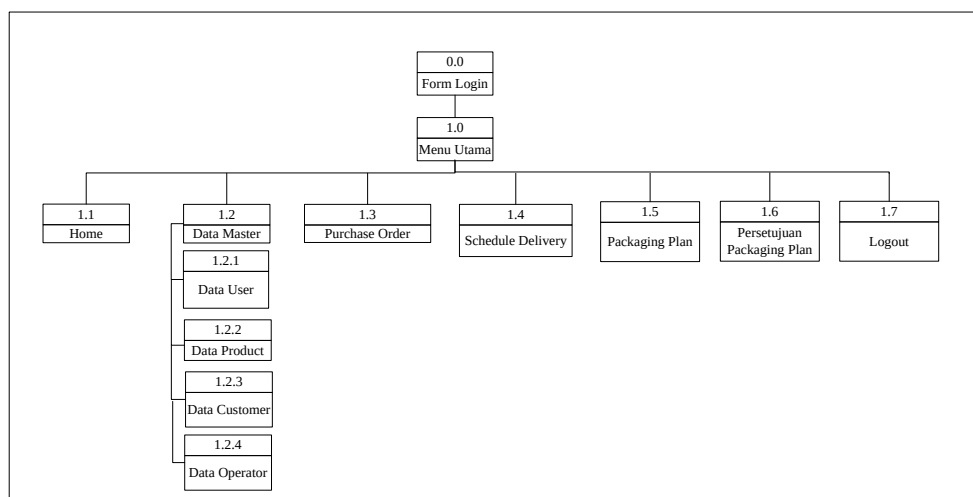
1. *Client* adalah komputer *client* yang harus terinstal sebuah *web browser* untuk menjalankan aplikasi sistem informasi *packaging plan* dan terhubung dengan *server*.
2. *Server* aplikasi sistem *packaging plan* terdiri dari *web server* berupa *apache*, bahasa pemrograman PHP berbasis *web*, dan *database MySQL*.

5.5 Desain Program

Tahap ini merupakan tahap kedua dalam metodologi *prototype evolutioner*, yaitu tahap membuat sebuah prototipe dari program atau aplikasi. Dimulai dengan desain usulan meliputi pembuatan struktur menu program, *flowchart* program, dan *interface* program sampai dengan program dapat dijalankan.

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

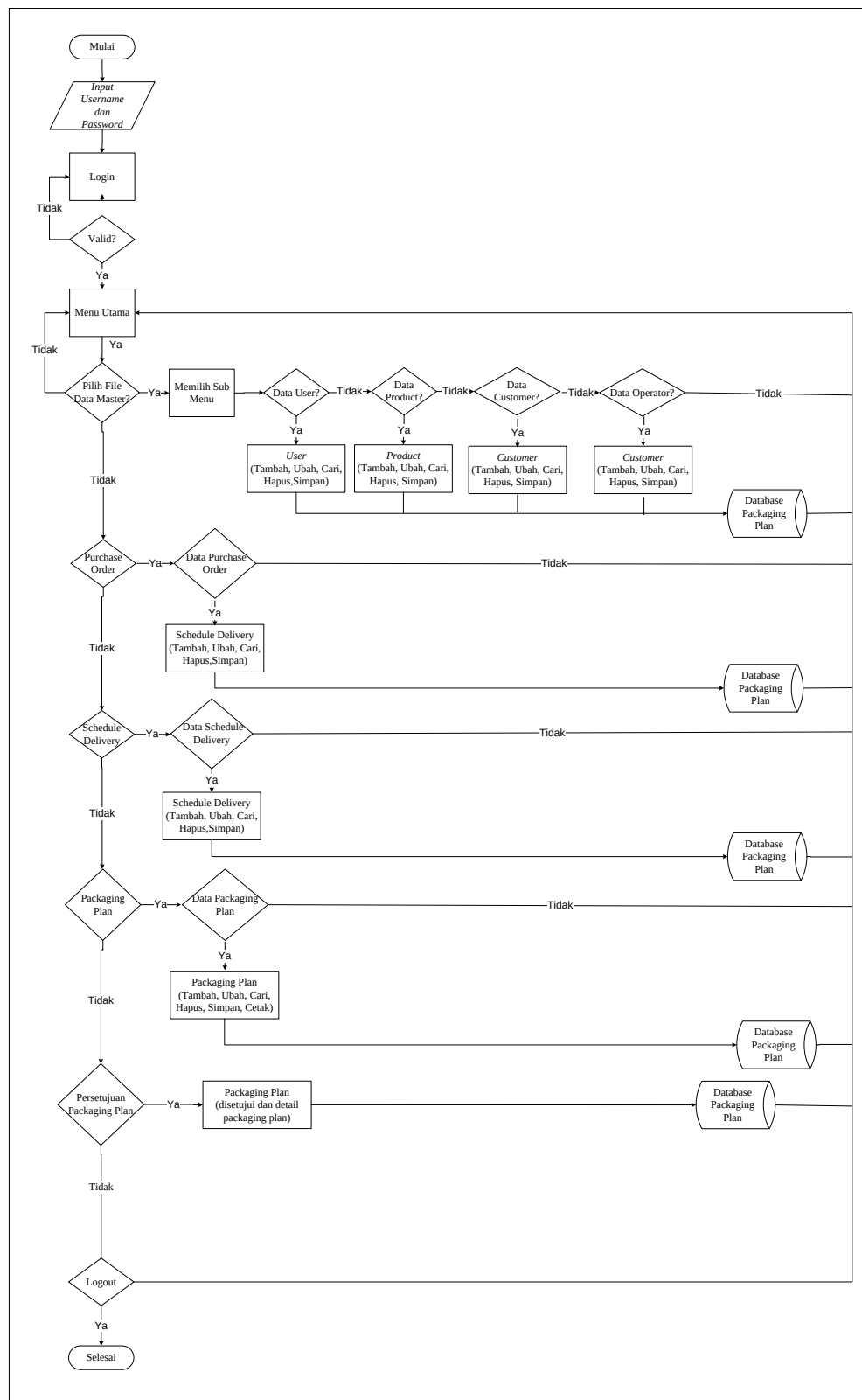
Hierarchy plus Input-Process-Output (HIPO) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hierarkis. Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya. Perancangan HIPO aplikasi usulan sistem *packaging plan* dapat dilihat pada Gambar V.17 sebagai berikut:



Gambar V.17 HIPO Sistem Informasi *Packaging Plan*
Sumber: Hasil Analisis Data (2017)

5.5.2 Flowchart Program

Flowchart yang digunakan untuk mendokumentasikan aplikasi sistem informasi *packaging plan* ini menggunakan bagan alir logika program (*program logic flowchart*). Bagan alir ini digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika, bukan untuk menggambarkan instruksi-instruksi program komputer secara terinci. Gambar V.18 menunjukkan *program logic flowchart* aplikasi sistem informasi *packaging plan*:



Gambar V.18 Flowchart Perancangan Sistem *Packaging Plan*
 Sumber: Hasil Analisis Data (2017)

5.5.3 Perancangan *Interface* Program

Rancangan *interface* dari program pengendalian kualitas bahan baku ini adalah sebagai berikut:

1. *Form Login*

Form login adalah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan id user dan password yang benar. Rancangan *form login* dapat dilihat pada Gambar V.19 berikut:

Aplikasi Sistem Packaging Plan
ID
Password
Login

Gambar V.19 *Interface Form Login*
Sumber: Hasil Analisis Data (2017)

2. *Form Menu Utama*

Form menu utama adalah tampilan awal pada aplikasi yang berisi enam menu. Menu tersebut adalah Home, Data Master, Transaksi, Menginput Spesifikasi BB, Validasi, dan Laporan. Rancangan menu utama dapat dilihat pada Gambar V.20 berikut:

Gambar V.20 *Interface Form Menu Utama*
Sumber: Hasil Analisis Data (2017)

3. *Form Data Master User*

Form Data Master User adalah *form* yang digunakan untuk mengelola data *user*. Rancangan *interface* dari *form* data pengguna dapat dilihat pada Gambar V.21 berikut:

Gambar V.21 *Interface Form Data Master User*
Sumber: Hasil Analisis Data (2017)

Pada *interface form* data master user terdapat empat tombol, yaitu cari, tambah, ubah dan hapus. Tombol cari digunakan untuk mencari data *user* yang diinginkan. Jika tombol tambah atau ubah ditekan maka akan tampil *form input user*. Sedangkan tombol hapus digunakan untuk menghapus data *user* yang dipilih. *Interface form input user* dapat dilihat pada Gambar V.21 berikut:

Gambar V.22 *Interface Form Input Data Master User*
Sumber: Hasil Analisis Data (2017)

Pada *form input user* terdapat dua tombol, yaitu tombol simpan dan batal. Tombol simpan digunakan untuk menyimpan data *user* ke dalam *database*. Sedangkan tombol batal digunakan untuk membatalkan penginputan data *user*.

4. *Form Data Master Product*

Form data supplier adalah *form* yang digunakan untuk mengelola data *supplier*. Rancangan *interface* dari *form data supplier* dapat dilihat pada Gambar V.23 berikut:

PT SKF Indonesia Logout

Home
Data Master ▾
Data User
Data Product
Data Customer
Data Operator
Purchase Order
Schedule Delivery
Packaging Plan
Persetujuan

Data Master Product

Tambah

Cari

Id Product	Type Product	Nama Product	Aksi
			Ubah Hapus

Footer

Gambar V.23 *Interface Form Data Master Product*

Sumber: Hasil Analisis Data (2017)

Pada *interface form* data master *product* terdapat empat tombol, yaitu cari, tambah, ubah dan hapus. Tombol cari digunakan untuk mencari data *product* yang diinginkan. Jika tombol tambah atau ubah ditekan maka akan tampil *form input product*. Sedangkan tombol hapus digunakan untuk menghapus data *product* yang dipilih. *Interface form input product* dapat dilihat pada Gambar V.24 berikut:

PT SKF Indonesia Logout

Home
Data Master ▾
Data User
Data Product
Data Customer
Data Operator
Purchase Order
Schedule Delivery
Packaging Plan
Persetujuan

Input Data Product

Id Product

Type Product

Nama Product

Footer

Gambar V.24 Interface Form Input Data Master Product
Sumber: Hasil Analisis Data (2017)

Pada *form input product* terdapat dua tombol, yaitu tombol simpan dan batal. Tombol simpan digunakan untuk menyimpan data *product* ke dalam *database*. Sedangkan tombol batal digunakan untuk membatalkan penginputan data *product*.

5. *Form Data Master Customer*

Form data master customer adalah *form* yang digunakan untuk mengelola data *customer*. Rancangan *interface* dari *form data customer* dapat dilihat pada Gambar V.25 berikut:

PT SKF Indonesia		Logout
Home ▼		
Data Master ▼		
Data User ▼		
Data Product ▼		
Data Customer ▼		
Data Operator ▼		
Purchase Order ▼		
Schedule Delivery ▼		
Packaging Plan ▼		
Persetujuan ▼		

Data Master Customer

Tambah

Cari

Id Customer	Nama Customer	Alamat Customer	Telpon	Aksi
				Ubah Hapus

Footer

Gambar V.25 *Interface Form Data Customer*
Sumber: Hasil Analisis Data (2017)

Pada *interface form data master customer* terdapat empat tombol, yaitu cari, tambah, ubah dan hapus. Tombol cari digunakan untuk mencari data *customer* yang diinginkan. Jika tombol tambah atau ubah ditekan maka akan tampil *form input customer*. Sedangkan tombol hapus digunakan untuk menghapus data *customer* yang dipilih. *Interface form input customer* dapat dilihat pada Gambar V.26 berikut:

Gambar V.26 *Interface Form Input Data Customer*
 Sumber: Hasil Analisis Data (2017)

Pada *form input customer* terdapat dua tombol, yaitu tombol *simpan* dan *batal*. Tombol *simpan* digunakan untuk menyimpan data *customer* ke dalam *database*. Sedangkan tombol *batal* digunakan untuk membatalkan penginputan data *customer*.

6. *Form Data Master Operator*

Form data master operator adalah *form* yang digunakan untuk mengelola data *operator*. Rancangan *interface* dari *form data operator* dapat dilihat pada Gambar V.27 berikut:

PT SKF Indonesia		Logout																															
<table border="1"> <tr><td>Home</td></tr> <tr><td>Data Master ▼</td></tr> <tr><td> Data User</td></tr> <tr><td> Data Product</td></tr> <tr><td> Data Customer</td></tr> <tr><td> Data Operator</td></tr> <tr><td>Purchase Order</td></tr> <tr><td>Schedule Delivery</td></tr> <tr><td>Packaging Plan</td></tr> <tr><td>Persetujuan</td></tr> </table>		Home	Data Master ▼	Data User	Data Product	Data Customer	Data Operator	Purchase Order	Schedule Delivery	Packaging Plan	Persetujuan	<table border="1"> <tr><td colspan="4">Data Operator</td></tr> <tr><td colspan="4">Tambah</td></tr> <tr> <td colspan="2"> </td> <td colspan="2">Cari</td> </tr> <tr> <td>Nik</td> <td>Nama Operator</td> <td colspan="2">Aksi</td> </tr> <tr> <td></td> <td></td> <td>Ubah</td> <td>Hapus</td> </tr> </table>		Data Operator				Tambah						Cari		Nik	Nama Operator	Aksi				Ubah	Hapus
Home																																	
Data Master ▼																																	
Data User																																	
Data Product																																	
Data Customer																																	
Data Operator																																	
Purchase Order																																	
Schedule Delivery																																	
Packaging Plan																																	
Persetujuan																																	
Data Operator																																	
Tambah																																	
		Cari																															
Nik	Nama Operator	Aksi																															
		Ubah	Hapus																														
		Footer																															

Gambar V.27 *Interface Form Data Operator*

Sumber: Hasil Analisis Data (2017)

Pada *interface form* data master operator terdapat empat tombol, yaitu cari, tambah, ubah dan hapus. Tombol cari digunakan untuk mencari data operator yang diinginkan. Jika tombol tambah atau ubah ditekan maka akan tampil *form input* operator. Sedangkan tombol hapus digunakan untuk menghapus data operator yang dipilih. *Interface form input* operator dapat dilihat pada Gambar V.28 berikut:

PT SKF Indonesia		Logout										
<table border="1"> <tr><td>Home</td></tr> <tr><td>Data Master ▼</td></tr> <tr><td> Data User</td></tr> <tr><td> Data Product</td></tr> <tr><td> Data Customer</td></tr> <tr><td> Data Operator</td></tr> <tr><td>Purchase Order</td></tr> <tr><td>Schedule Delivery</td></tr> <tr><td>Packaging Plan</td></tr> <tr><td>Persetujuan</td></tr> </table>			Home	Data Master ▼	Data User	Data Product	Data Customer	Data Operator	Purchase Order	Schedule Delivery	Packaging Plan	Persetujuan
Home												
Data Master ▼												
Data User												
Data Product												
Data Customer												
Data Operator												
Purchase Order												
Schedule Delivery												
Packaging Plan												
Persetujuan												
Input Data Operator Nik Nama Operator Simpan												
		Footer										

Gambar V.28 *Interface Form Input Data Operator*
 Sumber: Hasil Analisis Data (2017)

7. *Form Data Purchase Order*

Form data purchase order adalah *form* yang digunakan untuk mengelola data purchase order. Rancangan *interface* dari *form data purchase order* dapat dilihat pada Gambar V.29 berikut:

PT SKF Indonesia Logout

Home
Data Master
Purchase Order ▾
Schedule Delivery
Packaging Plan
Persetujuan

Input Purchase Order

Id PO

Nama Customer

Tgl PO

Simpan

-Purchase Order

Tambah

Cari

Id PO	Nama Customer	Tanggal PO	Aksi
			Ubah Hapus Detail

Footer

Gambar V.29 *Interface Form Data Purchase Order*
Sumber: Hasil Analisis Data (2017)

8. *Form Data Purchase Order Detail*

Form data purchase order detail adalah *form* yang digunakan untuk mengelola data purchase order detail. Rancangan *interface* dari *form data purchase order detail* dapat dilihat pada Gambar V.30 berikut:

PT SKF Indonesia

Logout

Home

Data Master ▾

Purchase Order

Schedule Delivery

Packaging Plan

Persetujuan

Input Purchase Order Detail

Id PO Detail

Id PO

Id Product

Deskripsi Part

Jumlah PO

Simpan

Purchase Order Detail

Tambah

Cari

ID PO Detail	ID PO	Nama Product	Deskripsi Part	Jumlah	Aksi
					Ubah Hapus

Footer

Gambar V.30 *Interface Form Data Purchase Order Detail*
 Sumber: Hasil Analisis Data (2017)

9. *Form Data Schedule Delivery*

Form data Schedule Delivery adalah *form* yang digunakan untuk menyimpan data *Schedule Delivery*. Rancangan *interface* dari *form data Schedule Delivery* dapat dilihat pada Gambar V.31 berikut:

The interface is titled "PT SKF Indonesia" and includes a "Logout" button in the top right corner. The sidebar menu on the left contains the following items: Home, Data Master (with a dropdown arrow), Purchase Order, Schedule Delivery (highlighted), Packaging Plan, and Persetujuan. The main content area is divided into two sections. The top section, titled "Schedule Delivery", contains a form with two input fields: "Id Schedule Delivery" and "Nama Customer", followed by a "Simpan" button. The bottom section, also titled "Schedule Delivery", contains a "Tambah" button, a search bar with a "Cari" button, and a table. The table has three columns: "ID SD", "Nama Customer", and "Aksi". The "Aksi" column contains four buttons: "Ubah", "Hapus", "Detail", and "Lihat". A "Footer" section is located at the bottom of the main content area.

Gambar V.31 *Interface Form Data Schedule Delivery*
Sumber: Hasil Analisis Data (2017)

10. *Form Schedule Delivery Detail*

Form schedule delivery detail adalah *form* yang digunakan untuk mengelola data *schedule delivery detail*. Rancangan *interface* dari *form schedule delivery detail* dapat dilihat pada Gambar V.29 berikut:

PT SKF Indonesia
Logout

Home
Data Master ▾
Purchase Order
Schedule Delivery
Packaging Plan
Persetujuan

Input Schedule Delivery Detail

Id Schedule Delivery Detail

Id Schedule Delivery

Id PO

Nama Product

Plan

Jumlah

Simpan

Schedule Delivery Detail

Tambah

Cari

ID SD Detail	ID SD	Nama Product	PO	Plan	Jumlah Plan	AKsi
						Ubah Hapus

Footer

Gambar V.32 Interface Form Data Schedule Delivery Detail
Sumber: Hasil Analisis Data (2017)

11. Form Data Packaging Plan

Form data *packaging plan* adalah form yang digunakan untuk mengelola data *packaging plan*. Rancangan *interface* dari form *packaging plan* dapat dilihat pada Gambar V.33 berikut:

PT SKF Indonesia
Logout

Home
Data Master ▾
Purchase Order
Schedule Delivery
Packaging Plan
Persetujuan

Packaging Plan

Id PP

Id SD

Id User

Tgl PP

Validasi

Simpan

Packaging Plan

Tambah

 Cari

ID PP	ID SD	Petugas Input	Tanggal PP	Validasi	Aksi
					Ubah Hapus Detail Cetak

Footer

Gambar V.33 Interface Form Data Packaging Plan
Sumber: Hasil Analisis Data (2017)

12. Form Data Packaging Plan Detail

Form data packaging plan detail adalah form yang digunakan untuk mengelola data packaging plan detail. Rancangan interface dari form packaging plan detail dapat dilihat pada Gambar V.34 berikut:

PT SKF Indonesia
Logout

Home
Data Master ▾
Purchase Order
Schedule Delivery
Packaging Plan
Persetujuan

Input Packaging Plan Detail

Id PP Detail

Id PP

Nik

Id Product

Id Customer

Pack

Qty

Tgl Mulai

Tgl Selesai

Simpan

Packaging Plan Detail

Tambah

Cari

ID PP Detail	ID PP	Operator	Nama Product	Customer	Pack	Quantity	Tgl Mulai	Tgl Selesai	Aksi
									Ubah Hapus

Footer

Gambar V.34 Interface Form Data Packaging Plan Detail
Sumber: Hasil Analisis Data (2017)

13. Form Data Persetujuan *Packaging Plan*

Form data persetujuan *packaging plan* adalah form yang digunakan untuk mengelola data persetujuan *packaging plan*. Rancangan *interface* dari form persetujuan *packaging plan* dapat dilihat pada Gambar V.35 berikut:

PT SKF Indonesia		Logout													
Home	Persetujuan Packaging Plan <table border="1"> <thead> <tr> <th>Id PP</th> <th>Id PO</th> <th>Id User</th> <th>Tgl PP</th> <th>Status</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td> Detail Disetujui </td> </tr> </tbody> </table>			Id PP	Id PO	Id User	Tgl PP	Status	Aksi						Detail Disetujui
Id PP				Id PO	Id User	Tgl PP	Status	Aksi							
								Detail Disetujui							
Data Master ▾															
Purchase Order															
Schedule Delivery															
Packaging Plan															
Persetujuan	Footer														

Gambar V.35 *Interface Form Data Persetujuan*
Sumber: Hasil Analisis Data (2017)

5.6 Implementasi Sistem *Software* dan *Hardware*

Tahap ini adalah tahap pengkodean program menggunakan bahasa pemrograman PHP dengan *Macromedia Dreamweaver 8* dan *Sublime Text* dan *Notepad++* sebagai aplikasi editor. Setiap *interface* berisikan kode program agar program dapat dijalankan sesuai dengan fungsinya. Untuk mendukung kebutuhan implementasi sistem diperlukan suatu spesifikasi perangkat lunak (*software*) maupun perangkat keras (*hardware*). Adapun spesifikasinya sebagai berikut:

1. Analisis Kebutuhan *Software*

- a. Sistem Operasi : *Microsoft Windows 7*
- b. Database Server : *MySQL versi 5.6.21*
- c. Bahasa Pemrograman : *PHP 5.6.3*
- d. Web Browser : *Mozilla Firefox, Google Chrome*

2. Analisis Kebutuhan *Hardware*

- a. *Processor* : Minimal *Processor Pentium IV*
- b. *RAM* : Minimal *RAM 512 MB*
- c. *Harddisk* : Minimal *Harddisk 64 GB*
- d. *Brainware* : Kepala *Shipping* dan Admin *Shipping*
- e. *Mouse, Keyboard, Monitor* sebagai peralatan antar muka.
- f. *Printer* sebagai media pencetakan dokumen.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil pengumpulan, pengolahan dan analisis data yang telah dilakukan dalam penelitian ini dapat disimpulkan bahwa:

1. Dengan adanya aplikasi *packaging plan* ini memudahkan bagian Departemen *Shipping* melakukan penginputan data *schedule delivery* dapat masuk ke dalam sistem untuk mempermudah melakukan penginputan sehingga dapat mengurangi terjadinya ketidaksesuaian dan keterlambatan pengiriman jumlah produk ke pelanggan.
2. Membuat sistem informasi *packaging plan* dimana pencatatan data perencanaan pengemasan dapat diinput ke dalam sistem agar lebih cepat dan tepat dalam proses pembuatannya.
3. Membuat sistem basis data untuk *packaging plan* yang dapat mempermudah proses penyimpanan data dan dapat mencegah terjadinya kehilangan data karena setiap data yang dimasukkan ke dalam sistem telah tersimpan dengan aman menggunakan *database MySQL*.

6.2 Saran

Saran yang dapat dipertimbangkan untuk pengembangan sistem informasi *packaging plan* ini:

1. Dalam penerapan sistem baru terhadap sistem lama sebaiknya dilakukan secara bertahap, untuk menghindari terjadinya kegagalan terhadap sistem yang baru dan diperlukan adanya sosialisai serta pengenalan untuk penggunaan sistem ini kepada bagian yang menerapkannya.
2. Sebaiknya dilakukan pemeliharaan aplikasi secara berkala, sehingga aplikasi dapat berjalan dengan baik.
3. Untuk pengembangan selanjutnya dapat ditambahkan pembuatan laporan *packaging report* yang dilakukan oleh operator *packing*.

DAFTAR PUSTAKA

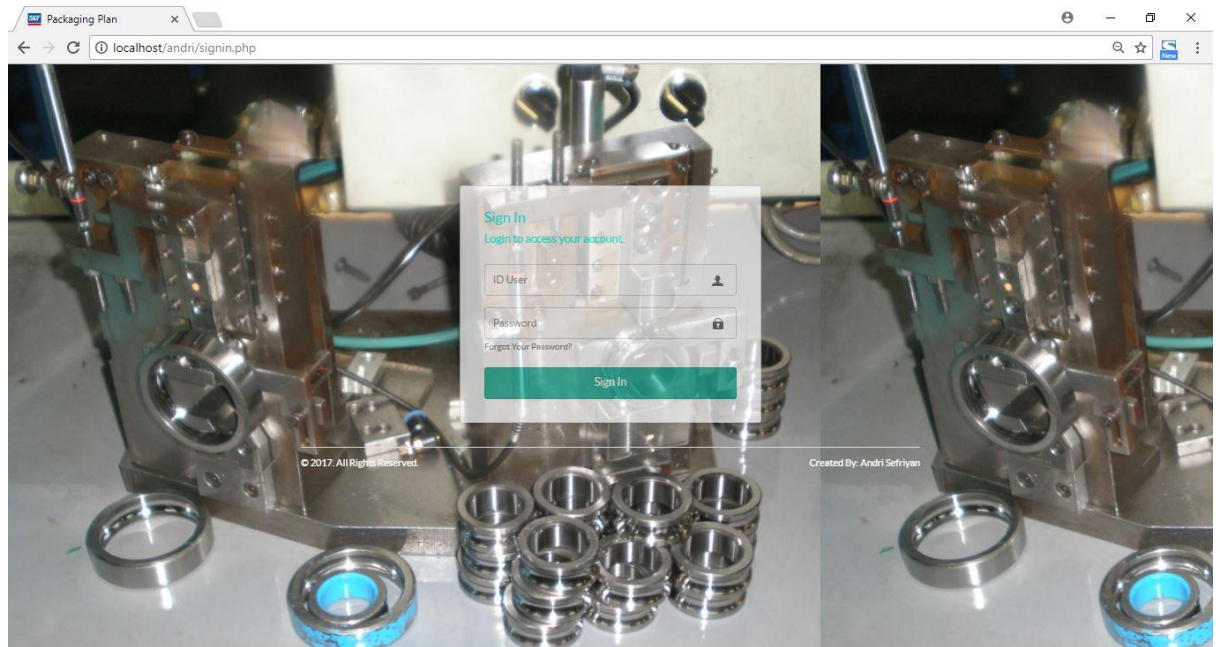
- Anhar. 2010. PHP dan MySQL Secara Otodidak. Jakarta: PT Trans Media.
- Assauri, S .2010. Manajemen Produksi dan Operasi. Edisi Revisi. Jakarta : Lembaga Penerbit Fakultas Ekonomi Universitas Indonesia.
- Bodnar, George dan William, Hopwood. 2012. Sistem Informasi Akuntansi. Jakarta : Salemba Empat.
- Hartono. Bambang. 2013. Sistem Informasi Manajemen Berbasis Komputer. Jakarta: Rineka Cipta.
- Herjanto, Eddy, 2011. Manajemen Operasi Edisi Ketiga. Jakarta: Grasindo.
- Jogiyanto, HM. 2010. *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis. Edisi Ketiga*. Yogyakarta: Andi Offset.
- Jogiyanto, HM. 2010. Sistem Informasi Teknologi. Yogyakarta: Andi Offset.
- McLeod, R. J, dan Scholl, G. 2011. Sistem Informasi Manajemen Edisi 9. Jakarta: PT Indeks.
- Mulyanto, Agus. 2010. *Sistem Informasi Konsep dan Aplikasi*. Yogyakarta : Pustaka Pelajar.
- Munawar. 2012. Pemodelan dengan UML. Yogyakarta: Graha Ilmu.
- Mulyadi. 2012. Sistem Akuntansi. Jakarta : Salemba Empat.
- Rangkuti, Freddy. 2010. Manajemen Persediaan Aplikasi. Jakarta: Raja Grafindo Persada.
- Rosa, A.S dan Shalahudin, M. 2016. *Modul Pembelajaran Rekayasa Perangkat Lunak. Cetakan kedua*. Bandung : Modula.
- Solihin, Achmad. 2010. MySQL 5 dari pemula hingga mahir. Jakarta : Universitas Budi Luhur.
- Stice, Earl K; Stice, James D; Skousen, K Fred. (2010). Akuntansi Intermediate. Jakarta: Salemba Empat.
- Sutabri, Tata. 2012. *Analisa Sistem Informasi*. Yogyakarta: Andi Offset.

Sutarman. 2010. Pengantar teknologi Informasi. Jakarta : Bumi Aksara.

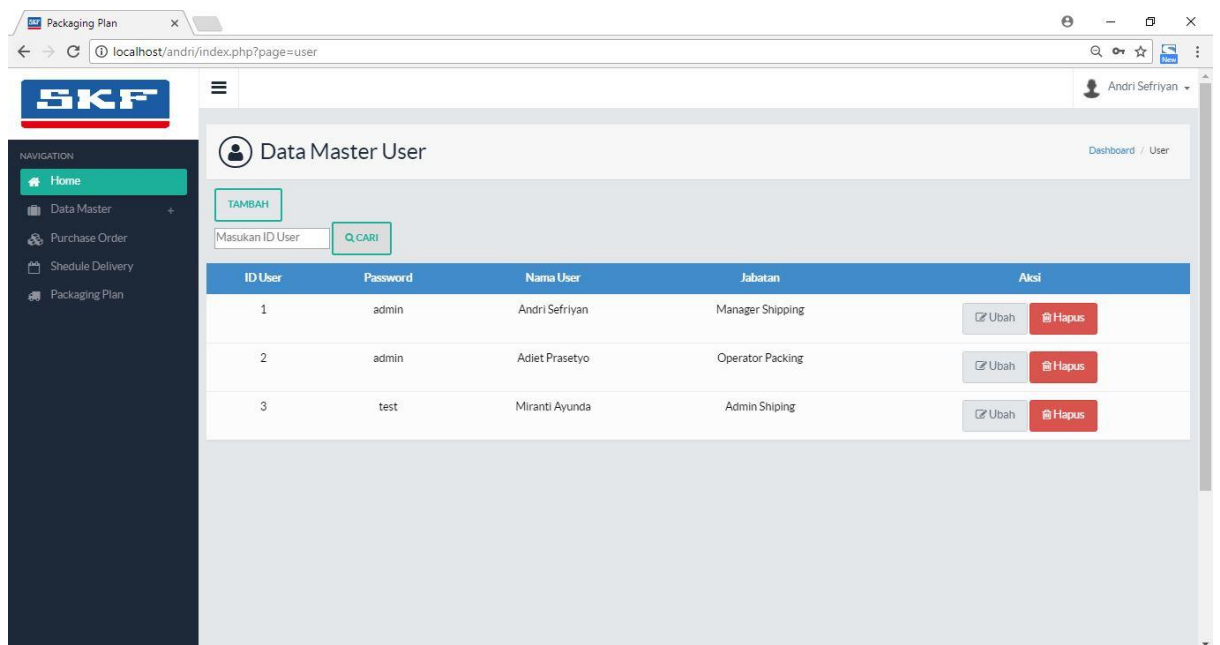
Yakub. 2012. Pengantar Sistem Informasi. Yogyakarta: Graha Ilmu.

Lampiran A

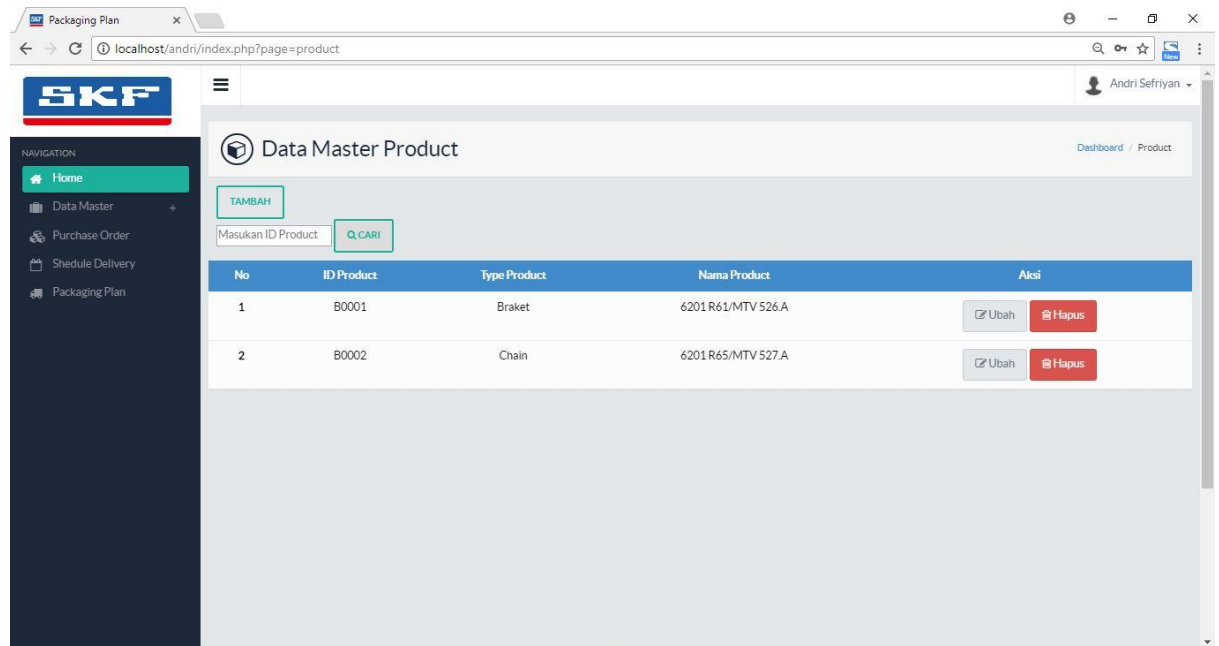
Tampilan Form Login



Tampilan Form User



Tampilan Form Product



Lampir
an B

Login

```
<?php
include 'class.php';
include 'config.php';
?>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0,
maximum-scale=1.0">
  <meta name="description" content="">
  <meta name="author" content="">
  <link rel="shortcut icon" href="images/skf.png" type="image/png">
  <style type="text/css">
    body {background: url(images/pairing1.jpg) !important;}
  </style>

  <title>Packaging Plan</title>

  <link href="css/style.default.css" rel="stylesheet">
```

```

<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media
queries -->
<!--[if lt IE 9]>
<script src="js/html5shiv.js"></script>
<script src="js/respond.min.js"></script>
<![endif]-->
</head>

<body class="signin">

<section>

<div class="signinpanel">

<div class="row">
<div class="col-sm-3"></div>

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

<form method="POST" style="background-color: white; opacity: 0.7">
<h4 class="nomargin">Sign In</h4>
<p class="mt5 mb20">Login to access your account.</p>

<input type="text" name="id_user" class="form-control uname"
placeholder="ID User" />
<input type="password" name="password" class="form-control
pword" placeholder="Password" />
<a href=""><small>Forgot Your Password?</small></a>
<button class="btn btn-success btn-block" name="mlogin">Sign
In</button>

</form>
</div><!-- col-sm-5 -->

</div><!-- row -->

<div class="signup-footer">
<div class="pull-left" style="color: white">
&copy; 2017. All Rights Reserved.
</div>
<div class="pull-right" style="color: white">
Created By: <a href="http://themepixels.com/" target="_blank"
style="color: white ">Andri Sefriyan</a>
</div>
</div>

```

```

</div><!-- signin -->

</section>
<?php
    if(isset($_POST['melogin'])){
        $cobalogin = $pengguna-
>login_pengguna($_POST['id_user'],$_POST['password']);
        if($cobalogin)
        {
            echo"<script>alert('Yes! Login Berhasil');</script>";
            echo"<script>>window.location='index.php';</script>";
        }
        else
        {
            echo"<script>alert('Username atau Password Yang Anda
Masukan Salah!');</script>";
            echo"<script>>window.location='signin.php';</script>";
        }
    }
?>

<script src="js/jquery-1.11.1.min.js"></script>
<script src="js/jquery-migrate-1.2.1.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/modernizr.min.js"></script>
<script src="js/jquery.sparkline.min.js"></script>
<script src="js/jquery.cookies.js"></script>

<script src="js/toggles.min.js"></script>
<script src="js/retina.min.js"></script>

<script src="js/custom.js"></script>
<script>
    jQuery(document).ready(function(){

        // Please do not use the code below
        // This is for demo purposes only
        var c = jQuery.cookie('change-skin');
        if (c && c == 'greyjoy') {
            jQuery('.btn-success').addClass('btn-orange').removeClass('btn-success');
        } else if(c && c == 'dodgerblue') {
            jQuery('.btn-success').addClass('btn-primary').removeClass('btn-success');
        } else if (c && c == 'katniss') {

```

```

        jQuery('.btn-success').addClass('btn-primary').removeClass('btn-success');
    }
});
</script>
</body>
</html>

```

Index

```

<?php
include 'class.php';

if(empty($_SESSION['id_user'])){
    header('location:signin.php');
}

if(isset($_GET['aksi']))
{
    if($_GET['aksi']=='logout')
    {
        $pengguna->logout_pengguna();
        echo "<script>alert('Anda Telah Keluar');</script>";
        echo "<script>window.location='signin.php';</script>";
    }
}

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0,
maximum-scale=1.0">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="shortcut icon" href="images/skf.png" type="image/png">

    <title>Packaging Plan</title>

    <link href="css/style.default.css" rel="stylesheet">

    <!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media
queries -->
    <!--[if lt IE 9]>
    <script src="js/html5shiv.js"></script>
    <script src="js/respond.min.js"></script>
    <![endif]-->

```

```

</head>

<body>
<!-- Preloader -->
<div id="preloader">
    <div id="status"><i class="fa fa-spinner fa-spin"></i></div>
</div>

<section>

    <div class="leftpanel">

        <div class="logopanel">
            <h1></h1>
        </div><!-- logopanel -->

        <div class="leftpanelinner">

            <!-- This is only visible to small devices -->
            <div class="visible-xs hidden-sm hidden-md hidden-lg">
                <div class="media userlogged">
                    
                    <div class="media-body">
                        <h4><?php echo $_SESSION['nm_user'];?></h4>
                        <span>"Life is so..."</span>
                    </div>
                </div>

                <h5 class="sidebararticle actitle">Account</h5>
                <ul class="nav nav-pills nav-stacked nav-bracket mb30">
                    <li><a href="profile.html"><i class="fa fa-user"></i>
<span>Profile</span></a></li>
                    <li><a href=""><i class="fa fa-cog"></i> <span>Account
Settings</span></a></li>
                    <li><a href=""><i class="fa fa-question-circle"></i>
<span>Help</span></a></li>
                    <li><a href="index.php?aksi=logout"><i class="fa fa-sign-out"></i>
<span>Sign Out</span></a></li>
                </ul>
            </div>

            <h5 class="sidebararticle">Navigation</h5>
            <ul class="nav nav-pills nav-stacked nav-bracket">
                <?php if($_SESSION['akses'] == 'Manager Shipping') { ?>

```

```

        <li class="active"><a href="index.php"><i class="fa fa-home"></i>
<span>Home</span></a></li>
        <li class="nav-parent"><a href=""><i class="fa fa-suitcase"></i>
<span>Data Master</span></a>
        <ul class="children">
            <li><a href="index.php?page=user"><i class="fa fa-caret-right"></i>
User</a></li>
            <li><a href="index.php?page=product"><i class="fa fa-caret-right"></i>
Product</a></li>
            <li><a href="index.php?page=customers"><i class="fa fa-caret-
right"></i> Customers</a></li>
            <li><a href="index.php?page=operator"><i class="fa fa-caret-right"></i>
Operator</a></li>
        </ul>
        </li>
        <li><a href="index.php?page=po"><i class="fa fa-cubes"></i>
<span>Purchase Order</span></a></li>
        <li><a href="index.php?page=sd"><i class="fa fa-calendar-o"></i>
<span>Shedule Delivery</span></a></li>
        <li><a href="tables.html"><i class="fa fa-truck"></i> <span>Packaging
Plan</span></a></li>
        <?php } else if($_SESSION['akses'] == 'Admin Shipping') { ?>
        <li class="active"><a href="index.php"><i class="fa fa-home"></i>
<span>Dashboard</span></a></li>
        <li><a href="tables.html"><i class="fa fa-th-list"></i> <span>Shedule
Delivery</span></a></li>
        <li><a href="tables.html"><i class="fa fa-th-list"></i> <span>Packaging
Plan</span></a></li>
        <li class="active"><a href="index.php"><i class="fa fa-home"></i>
<span>Dashboard</span></a></li>
        <li><a href="tables.html"><i class="fa fa-th-list"></i> <span>Shedule
Delivery</span></a></li>
        <li><a href="tables.html"><i class="fa fa-th-list"></i> <span>Packaging
Plan</span></a></li>
        <?php } ?>
    </ul>
</div><!-- leftpanelinner -->
</div><!-- leftpanel -->

<div class="mainpanel">

<div class="headerbar">

    <a class="menutoggle"><i class="fa fa-bars"></i></a>

    <div class="header-right">

```

```

<ul class="headermenu">
  <li>
    <div class="btn-group">
      <button type="button" class="btn btn-default dropdown-toggle" data-
toggle="dropdown">
        
        <?php echo $_SESSION['nama_user'];?>
        <span class="caret"></span>
      </button>
      <ul class="dropdown-menu dropdown-menu-usermenu pull-right">
        <li><a href="index.php?aksi=logout"><i class="glyphicon glyphicon-
log-out"></i> Log Out</a></li>
      </ul>
    </div>
  </li>
</ul>
</div><!-- header-right -->

</div><!-- headerbar -->

<div class="contentpanel">

  <div class="row">
    <?php
include "config.php";

if(isset($_GET['page'])){
  if($_GET['page'] == 'user'){
    include 'user.php';
  }else if($_GET['page'] == 'tambahuser'){
    include 'tambahuser.php';
  }else if ($_GET['page']=='home') {
    include 'signin.php';
  }else if($_GET['page'] == 'aksi'){
    include 'aksi.php';
  }else if($_GET['page'] == 'ubahuser'){
    include 'ubahuser.php';
  }else if($_GET['page'] == 'product'){
    include 'product.php';
  } else if($_GET['page'] == 'tambahproduct'){
    include 'tambahproduct.php';
  }else if($_GET['page'] == 'ubahproduct'){
    include 'ubahproduct.php';
  }else if($_GET['page'] == 'customers'){
    include 'customers.php';
  }else if($_GET['page'] == 'tambahcustomer'){

```

```

        include 'tambahcustomer.php';
    }else if($_GET['page'] == 'ubahcustomer'){
        include 'ubahcustomer.php';
    }else if($_GET['page'] == 'operator'){
        include 'operator/operator.php';
    }else if($_GET['page'] == 'tambahoperator'){
        include 'operator/tambahoperator.php';
    }else if($_GET['page'] == 'ubahoperator'){
        include 'operator/ubahoperator.php';
    }else if($_GET['page'] == 'po'){
        include 'po/po.php';
    }else if($_GET['page'] == 'tambahpo'){
        include 'po/tambahpo.php';
    }else if($_GET['page'] == 'ubahpo'){
        include 'po/ubahpo.php';
    }else if($_GET['page'] == 'po_detail'){
        include 'po_detail/po_detail.php';
    }else if($_GET['page'] == 'ubahpo_detail'){
        include 'po_detail/ubahpo_detail.php';
    }else if($_GET['page'] == 'sd'){
        include 'sd/sd.php';
    }else if($_GET['page'] == 'tambahsd'){
        include 'sd/tambahsd.php';
    }else if($_GET['page'] == 'ubahsd'){
        include 'sd/ubahsd.php';
    }else if($_GET['page'] == 'sd_detail'){
        include 'sd_detail/sd_detail.php';
    }else if($_GET['page'] == 'ubahsd_detail'){
        include 'sd_detail/ubahsd_detail.php';
    }else if($_GET['page'] == 'index'){
        include 'index.php';
    }else{
        include 'index.php';
    }
}
?>

```

```

</div><!-- tab-pane -->

```

```

</div><!-- tab-content -->

```

```

</div><!-- rightpanel -->

```

```

</section>

```

```
<script src="js/jquery-1.11.1.min.js"></script>
<script src="js/jquery-migrate-1.2.1.min.js"></script>
<script src="js/jquery-ui-1.10.3.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/modernizr.min.js"></script>
<script src="js/jquery.sparkline.min.js"></script>
<script src="js/toggles.min.js"></script>
<script src="js/retina.min.js"></script>
<script src="js/jquery.cookies.js"></script>

<script src="js/flot/jquery.flot.min.js"></script>
<script src="js/flot/jquery.flot.resize.min.js"></script>
<script src="js/flot/jquery.flot.spline.min.js"></script>
<script src="js/morris.min.js"></script>
<script src="js/raphael-2.1.0.min.js"></script>

<script src="js/custom.js"></script>
<script src="js/dashboard.js"></script>

</body>
</html>
```