

**RANCANG BANGUN APLIKASI *PROCUREMENT CONTROL SYSTEM*  
TERHADAP *ASSET UNDER CONSTRUCTION* DENGAN METODE *SIMPLE  
ADDICTIVE WEIGHTING* MENGGUNAKAN *CODEIGNITER FRAMEWORK*  
2.1.0 DAN *MYSQL 5.6.21* PADA PT ASTRA HONDA MOTOR**

**TUGAS AKHIR**

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

**DISUSUN OLEH**

**Riyan Alfian**

**1412005**



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

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

**LEMBAR PENGESAHAN**

Judul Tugas Akhir : RANCANG BANGUN APLIKASI *PROCUREMENT CONTROL SYSTEM* TERHADAP *ASSET UNDER CONSTRUCTION* DENGAN METODE *SIMPLE ADDICTIVE WEIGHTING* MENGGUNAKAN CODEIGNITER FRAMEWORK 2.1.0 DAN MYSQL 5.6.21 PADA PT ASTRA HONDA MOTOR

Disusun Oleh :  
Nama : Riyan Alfian  
Nim : 1412005  
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, 24 November 2017

Dosen Pembimbing

Ketua Penguji

**Dedy Trisanto S.Kom, MMSI**  
NIP.197805052005021002

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

Dosen Penguji

Dosen Penguji

**Drs. Jacob Saragih, MM**  
NIP. 195404281986031002

**Ulil Hamida, ST, MT**  
NIP. 198103272005022001

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

**TANDA PERSETUJUAN DOSEN PEMBIMBING**

Judul Tugas Akhir : RANCANG BANGUN APLIKASI *PROCUREMENT CONTROL SYSTEM* TERHADAP *ASSET UNDER CONSTRUCTION* DENGAN METODE *SIMPLE ADDICTIVE WEIGHTING* MENGGUNAKAN CODEIGNITER FRAMEWORK 2.1.0 DAN MYSQL 5.6.21 PADA PT ASTRA HONDA MOTOR

Disusun Oleh

Nama : Riyan Alfian  
Nim : 1412005  
Program Studi : Sistem Informasi Industri Otomotif  
Tanggal Seminar : 01 November 2017  
Tanggal Sidang : 21 November 2017  
Tanggal Lulus : 21 November 2017

Jakarta, 23 November 2017

Menyetujui  
Dosen 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 APLIKASI *PROCUREMENT CONTROL SYSTEM* TERHADAP *ASSET UNDER CONSTRUCTION* DENGAN METODE *SIMPLE ADDICTIVE WEIGHTING* MENGGUNAKAN CODEIGNITER FRAMEWORK 2.1.0 DAN MYSQL 5.6.21 PADA PT ASTRA HONDA MOTOR

Disusun Oleh

Nama : Riyan Alfian  
Nim : 1412005  
Program Studi : Sistem Informasi Industri Otomotif  
Tanggal Seminar : 01 November 2017  
Tanggal Sidang : 21 November 2017  
Tanggal Lulus : 21 November 2017











Jakarta, 24 November 2017

Menyetujui  
Asisten Dosen Pembimbing

**Triana Fatmawati ST, MT**  
NIP. 198005142005022001

## LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Riyan Alfian  
NIM : 1412005  
Judul TA : Rancang Bangun Aplikasi Procurement Control System Terhadap Asset Under Construction Dengan Metode Simple Addictive Weighting Menggunakan CodeIgniter Framework 2.1.0 Dan MYSQL 5.6.21 Pada PT Astra Honda Motor  
Pembimbing : Dedy Trisanto S.Kom, MMSI  
Asisten Pembimbing : Triana Fatmawati ST, MT

Tanggal	BAB	Keterangan	Paraf
11/05/2016	Bab I	Bimbingan Bab I	
30/05/2016	Bab I, II	Revisi Bab I, Bimbingan Bab II	
08/06/2016	Bab I,II,III	Revisi Bab I,II, Bimbingan Bab III	
20/06/2016	Bab III,IV	Revisi Bab III, Bimbingan Bab IV	
19/08/2016	Bab IV,V	Revisi Bab IV, Bimbingan Bab V	
29/08/2016	Bab V	Revisi Bab V ( <i>Use Case, Activity</i> )	
24/10/2016	Bab V	Revisi Bab V ( <i>Flowmap, Use Case, Activity</i> )	
25/10/2016	Bab V, VI	Revisi Bab V, Bimbingan Bab VI	
09/10/2017	BAB V	Revisi BAB VI, Demo Program	
26/10/2017	Finalisasi	Menyerahkan Tugas Akhir Keseluruhan	

Mengetahui,  
Ketua Program Studi  
Sistem Informasi Industri Otomotif

Dosen Pembimbing



Drs. Jacob Saragih, MM  
NIP : 195404281986031002

Dedy Trisanto S.Kom, MMSI  
NIP : 197805052005021002

**LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR**

Nama : Riyan Alfian  
 NIM : 1412005  
 Judul TA : Rancang Bangun Aplikasi Procurement Control System Terhadap Asset Under Construction Dengan Metode Simple Addictive Weighting Menggunakan CodeIgniter Framework 2.1.0 Dan MYSQL 5.6.21 Pada PT Astra Honda Motor  
 Pembimbing : Dedy Trisanto S.Kom, MMSI  
 Asisten Pembimbing : Triana Fatmawati ST, MT

Tanggal	BAB	Keterangan	Paraf
06/06/2016	Bab I	Bimbingan Bab I	<i>Riyan</i>
01/07/2016	Bab I, II	Revisi Bab I, Bimbingan Bab II	<i>Riyan</i> <i>Dedy</i>
21/07/2016	Bab I,II	Revisi Bab I,II	<i>Riyan</i> <i>Dedy</i>
23/08/2016	Bab II,III	Revisi Bab II, Bimbingan Bab III	<i>Riyan</i> <i>Dedy</i>
26/08/2016	Bab III,IV	Revisi Bab III, Bimbingan Bab IV	<i>Riyan</i> <i>Dedy</i>
21/10/2016	Bab IV, V	Revisi Bab IV, Bimbingan Bab V	<i>Riyan</i> <i>Dedy</i>
24/10/2016	Bab V	Revisi Bab V ( <i>Flowmap, Use Case, Activity</i> )	<i>Riyan</i> <i>Dedy</i>
25/10/2016	Bab V, VI	Revisi Bab V, Bimbingan Bab VI	<i>Riyan</i> <i>Dedy</i>
25/10/2017	BAB VI	Revisi BAB VI, Demo Program	<i>Riyan</i> <i>Dedy</i>
26/10/2017	Finalisasi	Menyerahkan Tugas Akhir Keseluruhan	<i>Riyan</i> <i>Dedy</i>

Mengetahui,  
 Ketua Program Studi  
 Sistem Informasi Industri Otomotif

Drs. Jacob Saragih, MM  
 NIP : 195404281986031002

Asisten Dosen Pembimbing



Triana Fatmawati ST, MT  
 NIP : 1980051420050220

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

**LEMBAR PERNYATAAN KEASLIAN**

Saya yang bertanda tangan di bawah ini:

Nama : Riyan Alfian

Nim : 1412005

Program Studi : Sistem Informasi Industri Otomotif.

Dengan ini menyatakan bahwa karya Tugas Akhir yang saya buat dengan judul:  
**“RANCANG BANGUN APLIKASI PROCUREMENT CONTROL SYSTEM TERHADAP ASSET UNDER CONSTRUCTION DENGAN METODE SIMPLE ADDICTIVE WEIGHTING MENGGUNAKAN CODEIGNITER FRAMEWORK 2.1.0 DAN MYSQL 5.6.21 PADA PT ASTRA HONDA MOTOR”**. Merupakan dibuat dan diselesaikan sendiri, dengan menggunakan literatur hasil kuliah, survei lapangan, dosen pembimbing dan asisten dosen pembimbing, melalui tanya jawab maupun asistensi serta buku-buku acuan yang tertera dalam referensi pada karya Tugas Akhir ini.

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

Jakarta, 26 Oktober 2017  
Yang Membuat Pernyataan,

Riyan Alfian

## KATA PENGANTAR

Puji syukur penulis panjatkan bagi Tuhan Yang Maha Esa, yang atas berkat dan kasih-Nya sehingga penulis mampu menyelesaikan Tugas Akhir dengan judul **“Rancang Bangun Aplikasi *Procurement Control System* Terhadap *Asset Under Construction* Dengan Metode *Simple Addictive Weighting* Menggunakan CodeIgniter Framework 2.1.0 Dan MYSQL 5.6.21 Pada PT Astra Honda Motor”**.

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

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

1. Bapak Dr. Mustofa, ST, MT, Selaku Direktur Politeknik STMI Jakarta.
2. Bapak Drs. Jacob Saragih, MM, Selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
3. Bapak Dedy Trisanto, S.Kom, MMSI, Selaku dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
4. Ibu Triana Fatmawati ST, MT, Selaku asisten dosen pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
5. Ibu Ennit Sutari selaku Orang Tua penulis yang selalu mendoakan, memberikan motivasi dan pengorbanannya baik segi moril, materi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
6. Sdri. Geni Mardyaningtyas yang senantiasa memberikan motivasi serta doa sehingga Tugas Akhir ini dapat diselesaikan dengan baik..

7. Bapak Mizaro Alifsyar dan Bapak Dena Sucianandika selaku pembimbing di PT Astra Honda Motor serta seluruh pekerja khususnya *Asset Control Department* yang telah membimbing dan memberikan informasi yang berguna kepada penulis dalam penyusunan Tugas Akhir.
8. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
9. Seluruh teman-teman program studi Sistem Informasi Industri Otomotif angkatan 2012, khususnya SA21, yang telah memberikan semangat dan motivasi kepada penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
10. Serta semua pihak yang telah membantu penulis dan yang tidak dapat disebutkan satu per satu.

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

Jakarta, 26 Oktober 2017

Penulis,  
Riyan Alfian

## ABSTRAK

Kemajuan dalam hal teknologi informasi dapat meningkatkan produktifitas yang mendukung pertumbuhan ekonomi yang semakin tinggi . Penggunaan sistem yang terkomputerisasi akan memberikan keuntungan bagi perusahaan seperti kegiatan di dalam perusahaan berlangsung lebih cepat dan pengolahan data yang lebih akurat. PT Astra Honda Motor merupakan pelopor industri sepeda motor di Indonesia. Di setiap fasilitas pabrik memiliki *fixed asset* yang bernilai cukup besar sehingga perlu adanya kontrol yang merupakan wewenang dari *Asset Control Department* sebagai pemilik *fixed asset*. Salah satu tugas dari *Asset Control* adalah melakukan kontrol terhadap *Asset Under Construction* (AUC) yang merupakan bentuk *fixed asset* yang belum siap pakai dan belum melalui proses *settlement*. Masalah besar yang dihadapi saat ini adalah proses *settlement* membutuhkan waktu yang lama sedangkan kebutuhan aset untuk segera beroperasi tidak dapat ditunda. Hal tersebut menyebabkan masalah besar bagi perusahaan karena aset yang seharusnya sudah dapat menghasilkan keuntungan belum diijinkan untuk beroperasi. Untuk mengatasi masalah tersebut, diperlukan pembuatan *Procurement Control System* yang merupakan sistem terintegrasi mulai dari proses *order* hingga *settlement* sehingga dapat mempercepat proses. *Procurement Control System* ini dikembangkan dengan metodologi prototipe evolusioner. Analisis dan perancangan sistem yang digunakan adalah analisis dan perancangan berorientasi objek dengan *tools* pemodelan sistem UML (*Unified Modelling Language*). Pemodelan data dengan menggunakan kamus data, HIPO (*Hierarchy plus Input-Process-Output*) dan *flowchart*. Sedangkan untuk *engine* program menggunakan bahasa pemrograman PHP dan *database* menggunakan MySQL dan untuk metode pemecahan masalah menggunakan SAW (*Simple Addictive Weighting*). Sistem yang dibangun dapat melakukan integrasi proses pengadaan, observasi dan *settlement* yang merupakan bentuk control AUC pada bagian *Asset Control* di PT Astra Honda Motor.

**Kata kunci** : *Procurement Control System*, Prototipe Evolusioner, *Unified Modeling Language* (UML), Kamus Data, HIPO, PHP 5.6.3, MySQL 5.6.21, *Simple Addictive Weighting* (SAW).

## DAFTAR ISI

Halaman

HALAMAN JUDUL	
LEMBAR PENGESAHAN	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR PERSETUJUAN ASISTEN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN DENGAN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN DENGAN ASISTEN DOSEN PEMBIMBING	
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 .....	3
1.3. Tujuan Penelitian .....	3
1.4. Batasan Masalah .....	4
1.5. Manfaat Tugas Akhir .....	4
1.6. Sistematika Penulisan .....	5
BAB II LANDASAN TEORI .....	7
2.1. Analisis .....	7
2.2. Konsep Dasar Sistem.....	7
2.2.1. Pengertian Sistem.....	7
2.2.2. Karakteristik Sistem.....	9
2.2.3. Analisis Sistem.....	11
2.2.4. Langkah – Langkah Analisis Sistem.....	12

2.3.	Konsep Dasar Informasi .....	13
2.3.1.	Pengertian Informasi.....	13
2.3.2.	Siklus Informasi.....	13
2.3.3.	Kualitas Informasi .....	14
2.4.	Konsep Dasar Sistem Informasi .....	15
2.4.1.	Pengertian Sistem Informasi .....	15
2.4.2.	Komponen Sistem Informasi .....	16
2.4.3.	Sistem Informasi Manajemen .....	17
2.4.4.	Peranan Sistem Informasi dalam Proses Manajemen	18
2.5.	Konsep Dasar Basis Data.....	19
2.6.	Konsep Dasar Aktiva Tetap.....	23
2.6.1.	Pengertian Aktiva Tetap .....	23
2.6.2.	Karakteristik Aktiva Tetap.....	25
2.6.3.	Klasifikasi Aktiva Tetap .....	26
2.6.4.	Penyajian Aktiva Tetap Di Laporan Keuangan .....	28
2.6.5.	Harga Perolehan Aktiva Tetap.....	30
2.7.	Konsep <i>Simple Additive Weighting</i> .....	31
2.8.	<i>Software Development Life Cycle</i> .....	32
2.8.1.	Model Prototipe Secara Umum.....	34
2.8.2.	<i>Evolutionary Prototype</i> .....	36
2.8.3.	Metodologi <i>Waterfall</i> .....	37
2.9.	Bagan Alir ( <i>Flowchart</i> ) .....	37
2.10.	Konsep Dasar OOAD .....	40
2.10.1.	Pengertian OOAD .....	40
2.10.2.	Metodologi Pengembangan Sistem Berbasis Objek .....	43
2.11.	Unified Modeling Language (UML) .....	44
2.11.1.	<i>Use Case Diagram</i> .....	45
2.11.2.	<i>Activity Diagram</i> .....	46
2.11.3.	<i>Sequence Diagram</i> .....	47
2.11.4.	<i>Class Diagram</i> .....	48

	2.11.5. <i>Component Diagram</i> .....	49
	2.11.6. <i>Deployment Diagram</i> .....	50
	2.12. Kamus Data .....	51
	2.13. <i>Hypertext Preprocessor</i> (PHP).....	52
	2.13.1. Kelebihan PHP .....	53
	2.14. <i>Codeigniter Framework</i> .....	54
	2.14.1. Kelebihan dan Kelemahan <i>Codeigniter</i> .....	54
	2.14.2. Konsep MVC .....	55
	2.15. <i>My SQL</i> .....	56
	2.15.1. Tipe Data MySQL .....	57
BAB III	METODOLOGI PENELITIAN .....	61
	3.1. Jenis dan Sumber Data.....	61
	3.2. Metode Pengumpulan Data .....	62
	3.3. Metode Pendekatan, Pemecahan Masalah dan Pengembangan Sistem .....	62
	3.3.1. Metode Pendekatan Sistem .....	63
	3.3.2. Metode Pemecahan Masalah .....	63
	3.3.3. Metode Pengembangan Sistem.....	63
	3.4. Kerangka Penelitian.....	64
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA .....	70
	4.1. Sejarah Perusahaan .....	70
	4.2. Profil Perusahaan .....	72
	4.3. Visi dan Misi Perusahaan .....	75
	4.4. Struktur Organisasi Perusahaan.....	75
	4.5. <i>Life Cycle Fixed Assets</i> .....	81
	4.6. Proses Kontrol AUC.....	83
	4.6.1. Proses Order .....	83
	4.6.2. Prosedur Observasi AUC.....	86
	4.6.3. <i>Settlement</i> AUC .....	90
	4.7. Klasifikasi AUC.....	91
	4.8. Fungsi – Fungsi Terkait .....	94

	4.9. Fungsi SAP .....	95
	4.10. <i>Flow Map</i> Sistem Berjalan .....	96
	4.11. <i>Use Case Diagram</i> Sistem saat Ini .....	99
	4.12. Analisis System yang Berjalan .....	100
	4.13. Penerapan Metode SAW .....	101
BAB V	ANALISIS DAN PEMBAHASAN .....	103
	5.1. Analisis Kebutuhan Sistem .....	103
	5.2. Analisis Kebutuhan Rinci Sistem .....	104
	5.3. Analisis Sistem Usulan .....	105
	5.3.1. <i>Flow Map</i> Sistem yang Diusulkan .....	105
	5.3.2. <i>Use Case Diagram</i> .....	107
	5.3.3. <i>Activity Diagram</i> .....	114
	5.3.4. <i>Sequence Diagram</i> .....	125
	5.3.5. <i>Deployment Diagram</i> .....	132
	5.3.6. <i>Class Diagram</i> .....	133
	5.4. Perancangan Basis Data .....	134
	5.5. <i>Hierachy plus Input-Process-Output (HIPO) Procurement     Conrol System</i> .....	140
	5.6. <i>Flowchart Program</i> .....	141
	5.7. Perancangan <i>Interface Procurement Control System</i> .....	142
	5.8. Implementasi Sistem .....	146
	5.8.1. Kebutuhan Perangkat Lunak .....	146
	5.8.2. Kebutuhan Perangkat Keras .....	146
BAB VI	KESIMPULAN DAN SARAN .....	148
	6.1. Kesimpulan .....	148
	6.2. Saran .....	148
	DAFTAR PUSTAKA .....	150

## DAFTAR GAMBAR

	Halaman
Gambar II.1.	Karakteristik Sistem..... 9
Gambar II.2.	Siklus Informasi..... 14
Gambar II.3.	Kriteria Kualitas Informasi ..... 15
Gambar II.4.	Ilustrasi Model Prototipe ..... 35
Gambar II.5.	Evolutionary Prototype Model ..... 36
Gambar II.6.	<i>Waterfall Model</i> ..... 37
Gambar II.7.	Klasifikasi Diagram UML ..... 44
Gambar II.8.	Logo <i>Codeigniter Framework</i> ..... 54
Gambar II.9.	Konsep MVC ..... 56
Gambar III.1.	Kerangka Penelitian..... 68
Gambar III.2.	Kerangka Penelitian (Lanjutan)..... 69
Gambar IV.1.	Struktur Organisasi PT Astra Honda Motor ..... 76
Gambar IV.2.	Struktur Organisasi Divisi <i>Business Control</i> ..... 80
Gambar IV.3.	<i>Fixed Assets – Life Cycle</i> ..... 82
Gambar IV.4.	<i>List Data Pengajuan Aset</i> ..... 84
Gambar IV.5.	<i>Flowchart</i> Proses Order ..... 84
Gambar IV.6.	<i>Budget Detail Memo</i> ..... 85
Gambar IV.7.	<i>Flowchart</i> Proses Observasi AUC..... 87
Gambar IV.8.	Dokumen Approval Observasi..... 89
Gambar IV.9.	Data Hasil Observasi ..... 91
Gambar IV.10.	Turning 2 - Turret L MC ..... 92
Gambar IV.11.	Roller Annealing..... 92
Gambar IV.12.	Injection Machine 300A..... 93
Gambar IV.13.	Stamping Press MC 80T ..... 93
Gambar IV.14.	Jig Leak Tesser ..... 93

Gambar IV.15.	Master Data.....	96
Gambar IV.16.	Laporan Hasil Observasi.....	97
Gambar IV.17.	<i>Flow Map</i> Analisis Sistem Berjalan .....	98
Gambar IV.18.	<i>Flow Map</i> Analisis Sistem Berjalan (Lanjutan) .....	99
Gambar IV.19.	<i>Use Case Diagram</i> Saat Ini .....	100
Gambar V.1.	<i>Flowmap</i> Sistem yang Diusulkan .....	107
Gambar V.2.	<i>Use Case Diagram Procurement Control System</i> .....	108
Gambar V.3.	<i>Activity Diagram Login</i> .....	115
Gambar V.4.	<i>Activity Diagram</i> Mengelola Master Data.....	116
Gambar V.5.	<i>Activity Diagram</i> Pengajuan Aset.....	117
Gambar V.6.	<i>Activity Diagram Approval</i> Pengajuan Aset.....	118
Gambar V.7.	<i>Activity Diagram</i> Membuat <i>Purchase Order</i> .....	119
Gambar V.8.	<i>Activity Diagram Update Good Receipt</i> .....	120
Gambar V.9.	<i>Activity Diagram</i> Pelaksanaan Observasi .....	121
Gambar V.10.	<i>Activity Diagram Approval</i> Hasil Observasi .....	122
Gambar V.11.	<i>Activity Diagram</i> Monitor Hasil Approval Observasi .....	123
Gambar V.12.	<i>Activity Diagram Approval Settlement</i> .....	124
Gambar V.13.	<i>Sequence Diagram Login</i> .....	125
Gambar V.14.	<i>Sequence Diagram</i> Pengajuan Aset.....	126
Gambar V.15.	<i>Sequence Diagram Approval</i> Pengajuan Aset.....	127
Gambar V.16.	<i>Sequence Diagram</i> Membuat <i>Purchase Order</i> .....	127
Gambar V.17.	<i>Sequence Diagram Update Good Receipt</i> .....	128
Gambar V.18.	<i>Sequence Diagram Update</i> Data Observasi .....	129
Gambar V.19.	<i>Sequence Diagram Approval</i> Hasil Observasi .....	130
Gambar V.20.	<i>Sequence Diagram Update</i> Hasil Observasi.....	131
Gambar V.21.	<i>Sequence Diagram Approval Settlement</i> .....	132
Gambar V.22.	<i>Deployment Diagram Procurement Control System</i> .....	132
Gambar V.23.	<i>Class Diagram Procurement Control System</i> .....	133
Gambar V.24.	<i>Program Logic Flowchart Procurement Control System</i> ....	142
Gambar V.25.	Rancangan <i>Form Login</i> .....	143
Gambar V.26.	Rancangan <i>Form Menu</i> .....	143

Gambar V.27.	Rancangan Tampilan Untuk Menampilkan Data <i>Supplier</i> ..	144
Gambar V.28.	Rancangan <i>Form</i> Tambah Data <i>Supplier</i> .....	145
Gambar V.29.	Rancangan <i>Form</i> Edit Data <i>Supplier</i> .....	145
Gambar V.30.	Rancangan Tampilan Untuk Menampilkan Data Observasi	146

## DAFTAR TABEL

	Halaman
Tabel II.1.	Elemen-elemen dari Diagram Hubungan Entitas ..... 20
Tabel II.2.	Simbol symbol <i>Flowchart</i> ..... 39
Tabel II.2.	Simbol symbol <i>Flowchart</i> (Lanjutan) ..... 40
Tabel II.3.	Simbol symbol <i>Use Case</i> Diagram..... 45
Tabel II.3.	Simbol symbol <i>Use Case</i> Diagram (Lanjutan )..... 46
Tabel II.4.	Simbol symbol <i>Activity</i> Diagram..... 46
Tabel II.4.	Simbol symbol <i>Activity</i> Diagram (Lanjutan )..... 47
Tabel II.5.	Simbol symbol <i>Sequence</i> Diagram..... 48
Tabel II.6.	Simbol symbol <i>Class</i> Diagram ..... 49
Tabel II.7.	Simbol symbol <i>Component</i> Diagram..... 50
Tabel II.8.	Simbol symbol <i>Deployment</i> Diagram..... 50
Tabel II.8.	Simbol-simbol <i>Deployment</i> Diagram (Lanjutan) ..... 51
Tabel II.9.	Simbol-simbol Kamus Data Untuk Tabel Pemasok ..... 52
Tabel V.1.	Kebutuhan Sistem..... 103
Tabel V.2.	Kebutuhan Rinci Sistem ..... 104
Tabel V.3.	Definisi Aktor <i>Use Case</i> Diagram yang Di Usulkan..... 109
Tabel V.4.	<i>Use Case Description Login</i> ..... 110
Tabel V.5.	<i>Use Case Description</i> Mengelola Data <i>User</i> ..... 110
Tabel V.6.	<i>Use Case Description</i> Mengelola Pengajuan Aset ..... 110
Tabel V.7.	<i>Use Case Description</i> Menyetujui Pengajuan Aset..... 111
Tabel V.8.	<i>Use Case Description</i> Mengelola Data PO ..... 111
Tabel V.9.	<i>Use Case Description</i> Mengirim Aset..... 112
Tabel V.10.	<i>Use Case Description</i> Mengelola Data Good Receipt ..... 112
Tabel V.11.	<i>Use Case Description</i> Melaksanakan Data Observasi..... 113
Tabel V.12.	<i>Use Case Description</i> Menyetujui Hasil Observasi ..... 113
Tabel V.13.	<i>Use Case Description</i> Monitor Approval Hasil Observasi..... 113

Tabel V.14.	<i>Use Case Description</i> Menyetujui Settlement.....	114
Tabel V.15.	Tabel <i>Asset</i> .....	134
Tabel V.16.	Tabel <i>Class</i> .....	135
Tabel V.17.	Tabel <i>Plant</i> .....	135
Tabel V.18.	Tabel <i>Location</i> .....	135
Tabel V.19.	Tabel <i>Position</i> .....	136
Tabel V.20.	Tabel <i>Department</i> .....	136
Tabel V.21.	Tabel <i>Supplier</i> .....	136
Tabel V.22.	Tabel <i>Project</i> .....	137
Tabel V.23.	Tabel <i>User</i> .....	137
Tabel V.24.	Tabel <i>Observasi</i> .....	138
Tabel V.25.	Tabel <i>Approval_Asset</i> .....	138
Tabel V.26.	Tabel <i>Approval_Obsevasi</i> .....	139
Tabel V.27.	Tabel <i>Settlement</i> .....	139
Tabel V.28.	Tabel <i>PO</i> .....	140

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

. Perkembangan dunia industri otomotif sepeda motor saat ini mengalami peningkatan yang cukup pesat akibat dari tingkat kebutuhan masyarakat akan transportasi yang juga cukup tinggi. Perkembangan ini diperkirakan akan bergerak terus secara konsisten dalam beberapa tahun mendatang.

Kemajuan dalam hal teknologi informasi dapat meningkatkan produktifitas yang mendukung pertumbuhan ekonomi yang semakin tinggi . Penggunaan sistem yang terkomputerisasi akan memberikan keuntungan bagi perusahaan seperti kegiatan di dalam perusahaan berlangsung lebih cepat dan pengolahan data yang lebih akurat. Salah satu kegiatan yang dimaksud adalah proses kontrol terhadap pengadaan *Asset Under Construction* (AUC) yang dilakukan di PT Astra Honda Motor. Pemanfaatan teknologi informasi jika dikaitkan dengan proses kontrol pengadaan AUC diharapkan teknologi tersebut mampu mendukung kegiatan perusahaan mulai dari pengajuan hingga *settlement* untuk mendukung kegiatan operasional perusahaan khususnya pada *Asset Control Department*.

PT Astra Honda Motor merupakan pelopor industri sepeda motor di Indonesia. Saat ini PT Astra Honda Motor memiliki tiga fasilitas pabrik perakitan, pabrik pertama berlokasi di Sunter, Jakarta Utara yang juga berfungsi sebagai kantor pusat. Pabrik kedua berlokasi di Pegangsaan Dua, Kelapa Gading, serta pabrik ketiga yang berlokasi di kawasan MM 2100 Cikarang Barat, Bekasi. Di setiap fasilitas pabrik memiliki *fixed asset* yang bernilai cukup besar sehingga perlu adanya kontrol yang merupakan wewenang dari *Asset Control Department* sebagai pemilik *fixed asset*. Salah satu tugas dari *Asset Control* adalah melakukan kontrol terhadap AUC yang merupakan bentuk *fixed asset* yang belum siap pakai dan belum melalui proses *settlement*. *Settlement* merupakan pengakuan aset tetap yang berasal dari AUC pada saat selesai dan siap untuk dipergunakan sesuai dengan tujuan yang dimaksud. Kontrol yang dilakukan terhadap AUC meliputi proses *order*, observasi dan *settlement*.

Dalam proses kontrol AUC, *Asset Control Department* menggunakan bantuan Microsoft Excel dan email internal perusahaan. Sistem ini cukup membantu dalam laporan hasil observasi dan mengalirkan informasi *settlement* namun hal ini memakan waktu yang cukup lama. Selain itu proses *order* yang tidak terintegrasi dengan proses observasi dan *settlement* di dalam satu sistem menyebabkan diperlukan penarikan data *good receipt* 100% per-awal bulan sebelum pelaksanaan observasi. Hal tersebut mengakibatkan data di SAP *pending* terlebih dahulu selama sebulan sehingga kurang efektif. Sedangkan dalam proses pengajuan, *approval* dan daftar AUC untuk observasi masih menggunakan *hardcopy* yang sangat rentan terhadap kehilangan data pada proses tersebut.

Masalah besar yang dihadapi saat ini adalah proses *settlement* membutuhkan waktu yang lama sedangkan kebutuhan aset untuk segera beroperasi tidak dapat ditunda. Hal tersebut menyebabkan masalah besar bagi perusahaan karena aset yang seharusnya sudah dapat menghasilkan keuntungan belum diijinkan untuk beroperasi. Selanjutnya untuk masalah aset yang beroperasi sebelum melalui proses *settlement* juga dapat menimbulkan masalah karena dalam memperhitungkan harga suatu produk yang dihasilkan seharusnya nilai depresiasi aset juga masuk dalam biaya produk yang menentukan harga jual produk sedangkan apabila aset belum melalui proses *settlement* untuk nilai depresiasi belum terhitung.

Untuk mengatasi masalah tersebut, diperlukan pengembangan teknologi informasi dengan membuat sistem terintegrasi mulai dari proses *order* hingga *settlement* berbasis *mobile site*. Untuk itu penulis merencanakan pembuatan suatu sistem yang membantu dalam mengontrol pengadaan *fixed asset*. Oleh karena itu penelitian ini diberi judul “Rancang Bangun Aplikasi *Procurement Control System* Terhadap *Asset Under Construction* Dengan Metode *Simple Addictive Weighting* Menggunakan Codeigniter Framework 2.1.0 Dan MySQL 5.6.21 Pada PT Astra Honda Motor”.

## 1.2 Pokok Permasalahan

Permasalahan yang terjadi pada PT Astra Honda Motor adalah sebagai berikut:

1. Dalam proses pengajuan *fixed asset* masih menggunakan form *hardcopy* dan *approval* secara manual oleh *Asset Controller Department Head* sehingga memperlambat pengajuan *fixed asset* dan proses *approval* yang tidak fleksibel.
2. Proses *settlement* yang lambat dan masih terdapat *fixed asset* yang beroperasi tanpa melalui proses *settlement* yang dapat merugikan perusahaan akibat dari SAP tidak terintegrasi dengan proses observasi dan *settlement*.
3. Penarikan data aset dilakukan per-awal bulan yang menyebabkan data aset dengan *goods receipt* 100% dibiarkan *pending* selama sebulan sehingga memperlambat proses *settlement*.
4. Observasi dilakukan menggunakan berkas *hardcopy* yang berisi daftar aset yang sudah di olah dengan Microsoft Excel dimana dilakukan *filter* tanggal penggunaan, tipe *project* dan harga dari aset tersebut untuk mengetahui aset yang diprioritaskan.
5. Untuk mengetahui *approval settlement*, pihak *Asset Controller* harus menunggu email internal perusahaan terlebih dahulu yang dikirimkan oleh *Asset Control Department Head*.

## 1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah membangun sistem dimana:

1. Pengajuan *fixed asset* tidak memerlukan berkas *hardcopy* dan mempercepat proses *approval* karena dapat termonitor secara *real time*.
2. Pengadaan yang terintegrasi dengan segala kegiatan observasi dan *settlement*.

3. Data aset dengan *goods receipt* 100% dapat termonitor secara real time tanpa harus menarik data per-bulan.
4. Proses observasi dapat secara langsung *update* data aset melalui *mobile* dimana prioritas aset telah terhitung otomatis dengan metode SAW (*Simple Addictive Weighting*) dan hasil observasi dapat langsung dilihat melalui system.
5. Proses *approval settlement* dapat termonitor secara *real time* sehingga *Asset Controller* tidak perlu menunggu email internal terlebih dahulu.

#### **1.4 Batasan Masalah**

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

1. Tempat pengamatan adalah PT Astra Honda Motor, Jakarta.
2. Pembatasan masalah dalam laporan ini yaitu menganalisis sistem kontrol AUC dari suatu aset masih dalam bentuk pengajuan hingga proses *settlement*.
3. AUC yang diamati meliputi aset tidak bergerak dengan nilai di atas Rp.10.000.000 di luar aset bangunan dan gedung.

#### **1.5 Manfaat Tugas Akhir**

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

1. Bagi penulis  
Sebagai alat untuk mempraktekkan teori-teori yang telah diperoleh selama perkuliahan, sehingga penulis dapat menambah pengetahuan secara praktis tentang masalah-masalah yang dihadapi oleh perusahaan khususnya teori tentang analisis dan perancangan sistem informasi pada industri otomotif.

## 2. Bagi perusahaan

Hasil penelitian ini agar dapat diimplementasikan di perusahaan untuk membantu kinerja perusahaan, karena dengan adanya *Procurement Control System (PCS)* dapat mempercepat proses pengajuan aset dan *settlement* sehingga adanya efisiensi terhadap proses control *Asset Under Construction (AUC)*.

## 3. Bagi pihak lain

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

## 1.6 Sistematika Penulisan

Agar lebih mempermudah perumusan dan pemecahan masalah yang akan dibahas pada penelitian Tugas Akhir ini, maka penulis menguraikan tahapan-tahapan dalam penyusunan laporan ini sebagai berikut:

### **BAB I PENDAHULUAN**

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

### **BAB II LANDASAN TEORI**

Bab ini membahas tentang pengertian sistem, jenis sistem, pengertian informasi, jenis-jenis informasi, konsep dasar *fixed assets*, metodologi pemecahan masalah, metodologi pengembangan sistem, *Flowchart*, *Unified Modeling Language (UML)*, Kamus Data, Personal Home Page (PHP), *Codeigniter Framework* dan MySQL.

### **BAB III METODOLOGI PENELITIAN**

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

**BAB IV PENGUMPULAN DAN PENGOLAHAN DATA**

Bab ini membahas tentang sejarah, profil perusahaan, struktur organisasi, produk yang dihasilkan, proses order, proses observasi AUC, proses *settlement* dan fungsi-fungsi terkait pada sistem yang berjalan saat ini serta dokumen-dokumen terkait.

**BAB V ANALISIS DAN PEMBAHASAN**

Bab ini berisi analisis kebutuhan sistem yang ada di lapangan, pemodelan sistem dengan *flowchart* dan diagram-diagram UML, pemodelan data dengan kamus data, perancangan antarmuka, perancangan menu dan perancangan *flowchart* program.

**BAB VI KESIMPULAN DAN SARAN**

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran yang diperlukan perusahaan dan peneliti selanjutnya.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Analisis**

Kata *analysis* berasal dari bahasa Yunani, terdiri dari kata “*ana*” dan “*lysis*”. *Ana* artinya atas (*above*), *lysis* artinya memecahkan atau menghancurkan. Secara definitif ialah: “*Analysis is a process of resolving data into its constituent components to reveal its characteristic elements and structure*” (Dey, 1995). Analisis data merupakan proses paling vital dalam sebuah penelitian. Hal ini berdasarkan argumentasi bahwa dalam analisis inilah data yang diperoleh peneliti bisa diterjemahkan menjadi hasil yang sesuai dengan kaidah ilmiah. Maka dari itu, perlu kerja keras, daya kreatifitas dan kemampuan intelektual yang tinggi agar mendapat hasil yang memuaskan.

Analisis data berasal dari hasil pengumpulan data. Data yang telah terkumpul, bila tidak dianalisis hanya menjadi barang yang tidak bermakna, tidak berarti, menjadi data yang mati, data yang tidak berbunyi. Oleh karena itu, analisis data berfungsi untuk memberi arti, makna dan nilai yang terkandung dalam data itu (Kasiram, 2006).

#### **2.2 Konsep Dasar Sistem**

##### **2.2.1 Pengertian Sistem**

Kata sistem sendiri berasal dari bahasa Latin “*systēma*” dan bahasa Yunani “*sustēma*” adalah suatu kesatuan yang terdiri 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. Untuk beberapa sistem yang diintisarikan dari beberapa sumber referensi dari buku dan *website*, mendefinisikan bahwa sistem:

1. Menurut Indrajit (2001) “Mengemukakan bahwa sistem mengandung arti kumpulan-kumpulan dari komponen-komponen yang dimiliki

unsur keterkaitan antara satu dengan lainnya”.

2. Menurut Jogianto (2005) “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”.
3. Menurut Murdick (1991) “Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur/bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data dan/atau barang pada waktu rujukan tertentu untuk menghasilkan informasi dan/atau energi dan/atau barang”.
4. Menurut Davis (1995) “Sistem secara fisik adalah kumpulan dari elemen-elemen yang beroperasi bersama-sama untuk menyelesaikan suatu sasaran”.
5. Menurut Sidharta (1995), “Sistem adalah himpunan dari bagian-bagian yang saling berhubungan yang secara bersama mencapai tujuan”.

Suatu sistem dapat terdiri dari sistem-sistem bagian (*subsystem*). Sebagai contoh, system computer dapat terdiri dari subsistem perangkat keras (*hardware*) dan perangkat lunak (*software*). Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih kecil atau terdiri dari komponen-komponen. Subsistem perangkat keras (*hardware*) dapat terdiri dari alat masukan, alat pemroses, alat keluaran dan simpanan luar. Subsistem-subsistem saling berinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran system tersebut dapat tercapai. Interaksi dari subsistem-subsistem sedemikian rupa sehingga dicapai suatu kesatuan yang terpadu atau terintegrasi. (Jogiyanto, 2005)

### 2.2.2 Karakteristik Sistem

Menurut Jogiyanto (2005) Karakteristik sistem adalah sistem yang mempunyai komponen (*components*), batas sistem (*boundary*), lingkungan sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran (*objectives*), seperti yang dapat dilihat pada gambar II.1:



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

#### 1. Komponen Sistem (*components*)

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

#### 2. Batasan Sistem (*boundary*)

Batasan sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. batasan suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

### 3. Lingkungan Sistem (*environments*)

Lingkungan luar (*environment*) dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan berupa energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

### 4. Penghubung (*interface*)

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

Sedangkan menurut Burch dan Grundnitski (dalam Jogiyanto, 2005) desain sistem dapat didefinisikan sebagai penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi.

Desain sistem menentukan bagaimana suatu sistem akan menyelesaikan tahap ini menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah instalasi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem. (Jogiyanto, 2005)

### 5. Masukan (*input*)

Masukan (*input*) sistem adalah energi yang masukan kedalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*), dan masukan sinyal (*signal input*). Maintenance input adalah energi yang dimasukkan supaya tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

#### 6. Keluaran (*output*)

Keluaran (*output*) sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

#### 7. Pengolah (*process*)

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

#### 8. Sasaran (*objectives*)

Sebuah sistem sudah tentu mempunyai sasaran ataupun tujuan. Dengan adanya sasaran sistem, maka kita dapat menentukan masukan yang dibutuhkan sistem dan keluaran apa yang akan dihasilkan sistem tersebut dapat dikatakan berhasil apabila mencapai/mengenai sasaran atau pun tujuan.

### 2.2.3 Analisis Sistem

Berikut pengertian analisis sistem menurut beberapa ahli:

1. Menurut McLeod (2004), “Analisis sistem adalah penelitian atas sistem yang telah ada dengan tujuan merancang sistem yang baru.”
2. Menurut Laudon (2007), “Analisis sistem adalah memeriksa sebuah masalah yang ada yang akan diselesaikan oleh perusahaan dengan menggunakan sistem informasi.” Analisis sistem mencakup beberapa langkah yang harus dilakukan, yaitu :
  - a. Menentukan masalah,
  - b. Mengidentifikasi penyebab dari masalah tersebut,
  - c. Menentukan pemecahan masalahnya,
  - d. Mengidentifikasi kebutuhan informasi yang dibutuhkan untuk memecahkan masalah tersebut.

3. Menurut Bentley dan Whitten (2007), “Analisis sistem adalah suatu teknik untuk menyelesaikan suatu masalah yang ada pada suatu sistem dengan cara membagi masalah tersebut ke beberapa bagian dengan maksud agar mudah dicari penyelesaiannya.”
4. Menurut Jogiyanto (2005), “Analisis sistem adalah penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.”

#### **2.2.4 Langkah-Langkah Analisis Sistem**

Menurut Jogiyanto (2005) terdapat langkah-langkah yang harus dilakukan dalam menganalisis sistem, yaitu:

1. *Identify*

Mengidentifikasi masalah merupakan langkah pertama yang dilakukan dalam tahap analisis. Masalah dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan. Masalah inilah yang menyebabkan sasaran dari sistem tidak dapat dicapai. Oleh karena itulah pada tahap analisis sistem, langkah yang pertama dilakukan adalah mengidentifikasi terlebih dahulu masalah-masalah yang terjadi.

2. *Understand*

Langkah kedua dari tahap analisis adalah memahami sistem yang ada. Langkah ini dapat dilakukan dengan mempelajari operasi secara terinci bagaimana sistem yang ada beroperasi. Untuk mempelajari operasi yang berjalan diperlukan data yang dapat diperoleh dengan cara melakukan penelitian.

3. *Analyze*

Langkah ini dilakukan berdasarkan data yang diperoleh dari hasil penelitian yang telah dilakukan.

#### 4. *Report*

Setelah proses analisis sistem selesai dilakukan, langkah terakhir adalah membuat laporan hasil analisis sistem yang menjelaskan temuan-temuan yang dapat dijadikan acuan alternatif penyelesaian masalah.

### **2.3 Konsep Dasar Informasi**

#### **2.3.1 Pengertian Informasi**

Menurut Moeliono (dalam Jimmy dan Gaol, 2008) informasi adalah penerangan, keterangan, pemberitahuan, kabar atau berita. Selanjutnya beliau mengatakan bahwa informasi juga adalah keterangan atau bahan nyata yang dapat dijadikan dasar kajian atau analisis kesimpulan.

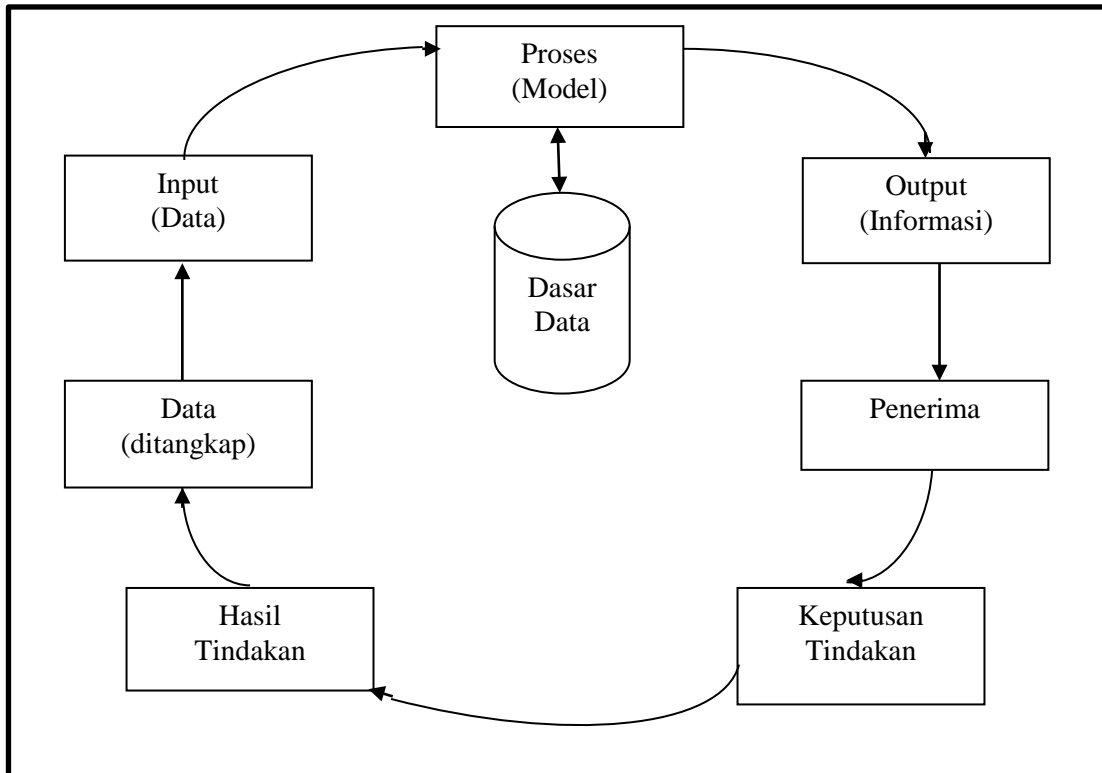
Kualitas dari suatu informasi tergantung dari tiga hal yaitu informasi harus akurat (*accurate*), tepat pada waktunya (*timeliness*), dan relevan (*relevance*). Sedangkan nilai dari informasi dikatakan bernilai bila manfaatnya efektif dibandingkan dengan biaya mendapatkannya (Jogiyanto, 2005).

#### **2.3.2 Siklus Informasi**

Di dalam kegiatan suatu perusahaan, misalnya dari hasil transaksi penjualan oleh sejumlah *salesman*, dihasilkan sejumlah faktor-faktor yang merupakan data dari penjualan pada suatu periode tertentu. Faktor-faktor penjualan tersebut masih belum dapat bercerita banyak kepada manajemen. Untuk pengambilan keputusan, maka faktor-faktor tersebut perlu diolah lebih lanjut untuk menjadi suatu informasi (Jogiyanto, 2005).

Data agar menjadi lebih berarti dan berguna dalam bentuk informasi maka perlu diolah menjadi suatu model tertentu. Data yang telah diolah tersebut kemudian diterima oleh penerima, lalu penerima membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, dan diproses kembali lewat suatu model dan seterusnya sehingga membentuk

suatu siklus. Siklus ini disebut dengan siklus informasi (*information cycle*) atau disebut pula siklus pengolahan data (*processing cycles*) (Jogiyanto, 2005).



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

### 2.3.3 Kualitas informasi

Agar informasi dapat mempunyai manfaat dalam proses pengambilan keputusan, informasi harus mempunyai kualitas dan nilai. Kriteria kualitas informasi adalah (Jogiyanto, 2005):

1. Akurat (*accuracy*)

Yang berarti informasi harus tidak bias atau menyesatkan dan bebas dari kesalahan.

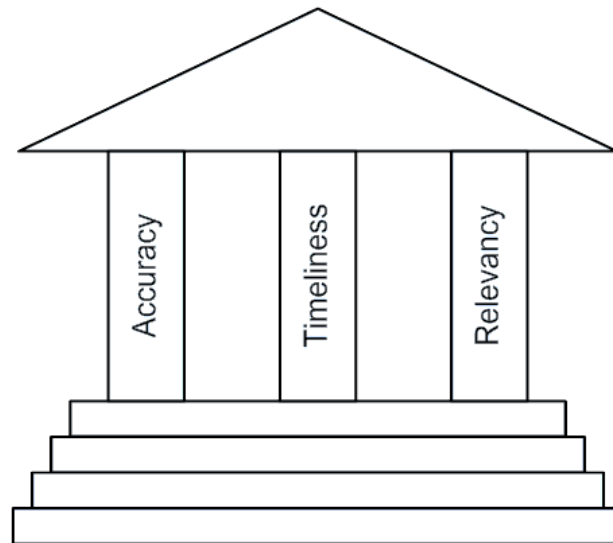
2. Tepat waktu (*timeliness*)

Yang berarti informasi yang sampai kepada penerima tidak boleh terlambat. Mahalnya nilai informasi saat ini adalah karena harus cepatnya informasi

tersebut didapatkan, sehingga diperlukan teknologi mutakhir untuk mendapatkan, mengolah, dan mengirimkan.

### 3. Relevan (*relevancy*)

Yang berarti informasi harus mempunyai manfaat bagi pihak yang menerimanya.



Gambar II.3 Kriteria Kualitas Informasi  
Sumber: Jogiyanto(2005)

## 2.4 Konsep Dasar Sistem Informasi

### 2.4.1 Pengertian Sistem Informasi

Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen dalam pengambilan keputusan. Informasi diperoleh dari sistem informasi (*information systems*) atau disebut juga *processing systems* atau *information processing systems* atau *information-generating systems*. Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Jogiyanto, 2005).

### 2.4.2 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*buildingblock*), yang terdiri dari komponen *input*, komponen *model*, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran (Jogiyanto, 2005).

Berikut ini adalah penjelasan dari masing-masing komponen yang terlibat di dalamnya, yaitu (Jogiyanto, 2005):

1. Komponen *input*. *Input* mewakili data yang masuk ke dalam sistem informasi. *Input* disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.
2. Komponen model. Komponen ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.
3. Komponen *output*. Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.
4. Komponen teknologi. Teknologi merupakan *tool box* dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.
5. Komponen *hardware*. *Hardware* berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung *database* atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.
6. Komponen *software*. *Software* berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari *hardware* untuk menciptakan suatu informasi.

7. Komponen basis data. Merupakan blok yang berisi definisi basis data yang disediakan untuk menyimpan data-data yang akan disimpan dalam media penyimpan. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).
8. Komponen kontrol. Merupakan sebuah komponen yang bertugas mendefinisikan bagaimana kontrol terhadap sistem dilakukan sehingga sistem dapat berjalan dengan baik. Dalam blok kontrol ini misalnya didefinisikan bagaimana melindungi data yang ada di *database* agar selalu sama dengan kenyataan yang dicatat.

### 2.4.3 Sistem Informasi Manajemen

Dalam organisasi bisnis, harus ada hubungan timbal balik dan keterkaitan yang erat antara setiap fungsi manajemen dengan setiap teknik manajemen agar kondisi sinergi bisa tercapai. Untuk mewujudkan keterkaitan antara setiap fungsi manajemen dengan setiap teknik manajemen, dibutuhkan sistem informasi manajemen yang akan melingkupi seluruh fungsi dan teknik manajemen (isjd.pdii.lipi.go.id, 2014).

Sistem Informasi Manajemen ini bertugas mengumpulkan, menyimpan dan mengolah data untuk akhirnya menyajikan informasi kepada semua tingkatan manajemen berkaitan dengan fungsi manajemen dalam pengelolaan sumber daya. Sistem informasi manajemen bertujuan menunjang proses pengambilan keputusan dalam melaksanakan fungsi manajemen pada berbagai tingkatan manajemen, dengan mewujudkan hubungan timbal balik dan keterkaitan informasi antar bagian organisasi sehingga sinergi organisasi dapat tercapai (isjd.pdii.lipi.go.id, 2014).

Menurut beberapa ahli sistem informasi manajemen dapat didefinisikan sebagai berikut (Jogiyanto, 2005):

1. Sistem informasi manajemen adalah kumpulan dari interaksi sistem-sistem informasi yang menyediakan informasi baik untuk kebutuhan manajerial maupun kebutuhan operasi.
2. Sistem informasi manajemen adalah kumpulan dari manusia dan sumber-sumber daya modal di dalam suatu organisasi yang bertanggungjawab mengumpulkan dan mengolah data untuk menghasilkan informasi yang berguna untuk semua tingkatan manajemen dalam kegiatan perencanaan dan pengendalian.
3. Sistem informasi manajemen adalah kumpulan dari sistem-sistem yang menyediakan informasi untuk mendukung manajemen.
4. Sistem informasi manajemen adalah suatu sistem berbasis komputer yang menyediakan informasi bagi beberapa pemakai dengan kebutuhan sistem serupa.

#### **2.4.4 Peranan Sistem Informasi dalam Proses Manajemen**

Peranan sistem informasi dalam kegiatan manajemen adalah menyediakan informasi untuk menunjang proses pengambilan keputusan yang dilakukan manajemen. Informasi yang dipakai untuk membantu pengambilan keputusan dilihat dari asalnya, datang dari luar organisasi (eksternal). Tugas sistem informasi adalah menyediakan informasi yang bersifat internal. Agar informasi yang dihasilkan sistem informasi lebih mengena dan berguna bagi manajemen maka harus dilakukan analisis untuk mengetahui kebutuhan informasi bagi setiap tingkatan manajemen (library.upnvj.ac.id, 2014).

Pada 3 tingkatan manajemen, yaitu manajemen tingkat atas, menengah, dan bawah. Masing-masing tingkatan mempunyai tingkatan yang berbeda dan karena itu pengelolaan informasi yang dibutuhkan untuk pengambilan keputusan juga berbeda. Pembagian kegiatan manajemen menurut tingkatannya adalah sebagai berikut (Jogiyanto, 2005):

1. Perencanaan strategis adalah kegiatan manajemen tingkat atas, yang berupa penetapan tujuan dan penentuan strategi serta kebijaksanaan yang dibutuhkan untuk pencapaian tujuan. Pengambilan keputusan dalam perencanaan strategis banyak dipengaruhi kondisi lingkungan yang dinamis dan serba tak pasti sehingga informasi yang dibutuhkan banyak berupa ringkasan dan bersifat eksternal. Pengalaman dan intuisi banyak berperan dalam pengambilan keputusan.
2. Pengendalian manajemen, adalah kegiatan manajemen tingkat menengah yang dilakukan untuk memastikan bahwa organisasi telah melaksanakan kebijakan yang telah ditetapkan untuk mencapai tujuan organisasi. Informasi yang dibutuhkan pengendalian manajemen berupa laporan-laporan dari pusat pertanggungjawaban yang dilakukan secara terkoordinasi. Dari laporan-laporan ini dibuat suatu analisis untuk membandingkan kinerja sesungguhnya dengan rencananya. Berdasarkan analisis itu, manajemen membuat keputusan-keputusan, misalnya pembuatan sistem operasi baru, pembuatan anggaran, dan lain-lain.
3. Pengendalian operasional, merupakan kegiatan untuk memastikan bahwa tindakan-tindakan operasional telah dijalankan dengan efisien dan efektif. Pengendalian operasional merupakan penerapan keputusan yang telah dihasilkan oleh tingkatan pengendalian manajemen dan menghasilkan informasi hasil pelaksanaan tindakan, menghasilkan informasi hasil pelaksanaan tindakan, pengalokasian sumber daya, dan pengukuran kinerja. Informasi yang dibutuhkan harus mempunyai tingkat ketepatan tinggi dan bersifat sangat baru. Sistem informasi pada tingkatan ini bisa disebut menghasilkan keputusan karena keputusan yang dibuat seringkali berulang rutin dan terstruktur sehingga kebanyakan bisa diotomatisasikan atau diprogramkan.

## **2.5 Konsep Dasar Basis Data**

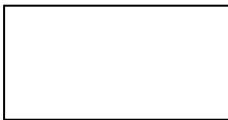
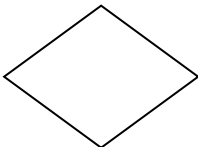
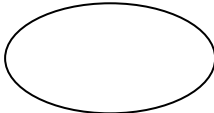

Dalam pengertian/mendefinisikan basis data menurut para ahli berbeda-beda, diantaranya:

1. Ramakrishnan dan Gehrke (2003), menyatakan basis data sebagai kumpulan data, umumnya mendeskripsikan aktivitas satu organisasi atau lebih yang berhubungan.
2. Menurut Silberschatz, dkk. (2002) database adalah kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan.

Elemen–elemen dalam Basis Data diantaranya:

1. Diagram Hubungan Antara Entitas (*Entity Relationship Diagram*)

ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD merupakan model jaringan data yang menekankan pada struktur–struktur dan relationship data (Ladjamudin, 2006).

SIMBOL	NAMA	KETERANGAN
	Entitas	Digambarkan dengan bentuk persegi panjang. Entitas adalah sesuatu apa aja yang ada dalam sistem, nyata maupun abstrak dimana data disimpan atau dimana terdapat data.
	Relasi	Digambarkan dengan sebuah bentuk belah ketupat. Relasi adalah hubungan alamiah yang terjadi antara entitas.
	Atribut	Adalah sifat atau karakteristik dari tiap-tiap entitas dan relasi atau elemen data dari entitas dan relasi. Atribut ini digunakan untuk penamaan dari bagian-bagian yang terdapat dalam entitas.
	Garis Lurus	Menghubungkan antara entitas satu dengan entitas yang lainnya.

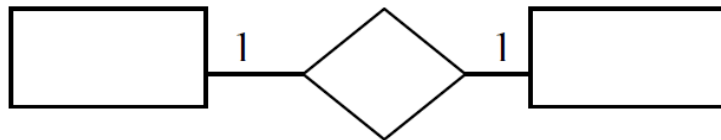
Tabel II.1 Elemen-elemen dari Diagram Hubungan Entitas  
(Sumber: Ladjamudin, 2006)

- a. *Cardinality* adalah tingkat hubungan yang terjadi antara entitas didalam sebuah sistem, dari sejumlah kemungkinan banyaknya

hubungan antar entitas tersebut, terdapat tiga macam *cardinality* yaitu :

1) *One to one*

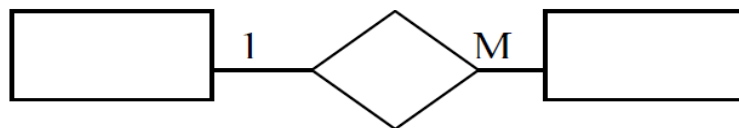
Bentuknya:



Adalah tingkat hubungan satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas kedua atau sebaliknya.

2) *One to Many*

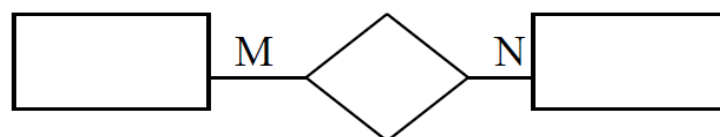
Bentuknya:



Adalah tingkat hubungan satu ke banyak atau sama dengan banyak ke satu. Tergantung pada arah mana hubungan itu dilihat.

3) *Many to Many*

Bentuknya:



Adalah tingkat hubungan banyak ke banyak terjadi jika kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya. Baik dilihat dari sisi entitas pertama maupun dilihat dari sisi entitas ke kedua.

b. Kunci (*Key*) merupakan suatu atribut yang unik yang dapat digunakan untuk membedakan suatu entitas dengan entitas yang lainnya dalam suatu himpunan entitas. Dari sudut pandang basis data, perbedaan diantara mereka harus dicerminkan lewat perbedaan dalam nilai atributnya. Nilai-nilai atribut kunci dapat secara unik mengidentifikasi suatu entitas terhadap entitas yang lainnya. Dengan kata lain, tidak ada lebih dari satu entitas memiliki nilai-nilai yang sama untuk semua atributnya. Macam-macam jenis kunci (*Key*) diantaranya:

1) *Primary Key* (Kunci Primer)

Adalah satu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian spesifik, tapi juga dapat mewakili setiap kejadian dari suatu entitas.

*Primary Key* mempunyai tiga kriteria yakni:

- *Key* tersebut lebih natural digunakan sebagai acuan.
- *Key* tersebut lebih sederhana.
- *Key* tersebut terjamin keunikannya.

2) *Foreign Key* (Kunci Tamu)

Merupakan sembarang atribut yang menunjuk kepada *Primary Key* pada tabel yang lain. *Foreign Key* terjadi pada suatu relasi yang memiliki *Cardinality one to many* atau *many to many*. *Foreign Key* biasanya selalu diletakan pada tabel/relasi yang mengarah ke banyak.

## 2. Normalisasi

Menurut Abdul Kadir (2009), "Normalisasi adalah suatu proses yang digunakan untuk menentukan pengelompokan atribut-atribut dalam sebuah relasi sehingga diperoleh relasi yang berstruktur baik". Tahap-tahap dalam normalisasi yaitu:

a. Bentuk Normal Pertama (1NF)

Bentuk normal pertama biasa dikenakan pada tabel yang belum ternormalisasi. Tabel yang belum ternormalisasi adalah tabel yang memiliki atribut yang berulang.

b. Bentuk Normal Kedua (2NF)

Bentuk normal kedua didefinisikan berdasarkan dependensi fungsional. Dalam ungkapan yang lebih praktis, bentuk normal kedua mensyaratkan setiap atribut bergantung pada kunci primer.

c. Bentuk Normal Ketiga (3NF)

Suatu relasi dikatakan dalam bentuk normal ketiga jika sudah dalam bentuk normal kedua dan setiap atribut bukan kunci tidak memiliki dependensi transitif terhadap kunci primer.

d. Bentuk Normal Boyce-Codd (BCNF)

Suatu relasi disebut memenuhi bentuk normal BCNF jika dan hanya jika semua penentu (determinan) adalah kunci kandidat (atribut yang bersifat unik).

3. Struktur *Database*/Spesifikasi *file*

Spesifikasi proses memberikan suatu penjelasan yang mendasari kebijaksanaan pengolahan suatu fungsional primitif yang mentransformasikan data keluaran. Dapat diartikan pula sebagai penggambaran dari kebijaksanaan dan prosedur yang mentransformasikan data juga sebagai alat yang efisien untuk menggambarkan algoritma (Kadir, 2009).

## 2.6 Konsep Dasar Aktiva Tetap

### 2.6.1 Pengertian Aktiva Tetap

Aktiva tetap merupakan bagian dari neraca yang dilaporkan oleh manajemen dalam setiap periode atau setiap tahun (repository.uin-suska.ac.id, 2015). Menurut Juan (2012), menyatakan bahwa aset tetap adalah aset berwujud yang :

1. Dimiliki untuk digunakan dalam produksi atau penyediaan barang atau jasa untuk direntalkan kepada pihak lain atau untuk tujuan administratif.
2. Diharapkan untuk digunakan selama lebih dari satu periode.

Menurut Standar Akuntansi Keuangan, (2011) aset tetap adalah aset berwujud yang dimiliki untuk digunakan dalam produksi atau penyediaan barangatau jasa, untuk direntalkan kepada pihak lain, atau untuk tujuan administratif dan diharapkan untuk digunakan selama lebih dari satu periode. Menurut Ilahi (2011), kategori aktiva tetap adalah sebagai berikut :

1. Dimiliki dan dikuasai oleh perusahaan.
2. Nilainya cukup material dan bersifat relatif permanen.
3. Digunakan dalam kegiatan normal perusahaan.
4. Mempunyai manfaat dan daya guna lebih dari satu tahun.
5. Tidak diperjualbelikan dalam kegiatan perusahaan.
6. Dapat diobservasi dengan alat perasa fisik.

Aktiva tetap adalah aktiva berwujud yang diperoleh dalam bentuk siap pakai atau dengan dibangun terlebih dahulu, yang digunakan dalam operasi perusahaan, tidak dimaksudkan untuk dijual dalam rangka kegiatan normal perusahaan dan mempunyai masa manfaat lebih dari satu tahun, Waluyo (2010).

Menurut Soepriyanto (2010), aktiva tetap merupakan aset yang bersifat jangka panjang atau secara relatif memiliki sifat permanen serta dapat digunakan dalam jangka panjang. Aset ini dimiliki dan digunakan oleh perusahaan dan tidak dijual sebagai bagian dari kegiatan operasi normal. Menurut Samryn (2011), aktiva tetap merupakan kelompok aktiva perusahaan yang mempunyai kriteria sebagai berikut:

1. Mempunyai masa manfaat, atau umur ekonomi lebih dari 1 tahun.
2. Dimiliki dengan tujuan untuk digunakan dalam membantu aktivitas perusahaan.
3. Fisik barangnya dapat dilihat dan diraba, sehingga biasa juga disebut aktiva tetap berwujud.
4. Biasanya mempunyai nilai perolehan yang relatif besar.

Menurut Rudianto (2009) pengertian aktiva tetap merupakan barang berwujud milik perusahaan yang sifatnya relatif permanen yang digunakan dalam kegiatan normal perusahaan bukan untuk diperjual belikan.

Menurut Kasmir (2008), aktiva tetap merupakan harta atau kekayaan perusahaan yang digunakan dalam jangka panjang lebih dari satu tahun. Secara garis besar, aktiva tetap dibagi dua macam, yaitu: aktiva tetap yang berwujud (tampak fisik) seperti tanah, bangunan, mesin, kendaraan, dan lainnya, dan aktiva tetap yang tidak berwujud (tidak tampak fisik) merupakan hak yang dimiliki perusahaan, contoh hak paten, merek dagang, *goodwill*, lisensi dan lainnya.

Menurut PSAK No. 16 Tahun 2009, aktiva tetap adalah aktiva berwujud yang diperoleh dalam bentuk siap pakai atau dengan dibangun terlebih dahulu, yang digunakan dalam operasi perusahaan, tidak dimaksudkan untuk dijual dalam rangka kegiatan normal perusahaan dan mempunyai masa manfaat lebih dari satu tahun.

### **2.6.2 Karakteristik Aktiva Tetap**

Menurut Juan (2012), suatu aset tetap harus memiliki karakteristik sebagai berikut:

1. Aset tersebut digunakan dalam operasi. Hanya aset yang digunakan dalam operasi normal perusahaan saja yang dapat diklasifikasikan sebagai aset tetap (misalnya kendaraan bermotor yang dimiliki oleh *dealer* mobil untuk dijual kembali harus diperhitungkan sebagai persediaan).
2. Aset tersebut memiliki masa (umur) manfaat yang panjang, lebih dari satu tahun periode.
3. Aset tersebut memiliki substansi fisik. Aset tetap memiliki ciri substansi fisik kasat mata sehingga dibedakan dari aset tak berwujud seperti hak paten dan merek dagang.

Dalam pernyataan Standar Akuntansi Keuangan (2009), suatu aktiva tetap harus memiliki karakteristik-karakteristik berikut:

1. Aktiva yang hanya digunakan dalam operasi normal perusahaan saja yang dapat diklasifikasikan sebagai aktiva tetap (misalnya kendaraan bermotor yang dimiliki oleh *dealer* mobil untuk dijual kembali harus diperhitungkan sebagai persediaan).
2. Aktiva tersebut memiliki masa (umur) manfaat yang panjang atau lebih dari satu periode.
3. Aktiva tersebut memiliki ciri-ciri substansi fisik kasat mata sehingga dibedakan dari aktiva tak berwujud seperti hak paten dan merk dagang.

Menurut Firdaus (2008), beberapa karakteristik dari aset tetap adalah:

1. Aset tetap adalah digunakan dalam kegiatan perusahaan dan bukan untuk dijual belikan dalam kegiatan normal perusahaan.
2. Umur atau jangka waktu pemakaiannya lebih dari satu tahun.
3. Pengeluaran untuk aset tersebut harus merupakan pengeluaran yang nilainya besar atau material bagi perusahaan.

Menurut Kieso dan Weygandt (2007), karakteristik aktiva tetap adalah sebagai berikut:

1. Aktiva tersebut diperoleh untuk digunakan dalam operasi dan bukan untuk dijual kembali.
2. Aktiva tersebut bersifat jangka panjang dan merupakan subjek penyusutan.
3. Aktiva tersebut memiliki substansi fisik.

### **2.6.3 Klasifikasi Aktiva Tetap**

Pengadaan aktiva tetap harus benar-benar sesuai dengan kebutuhan sehingga investasi yang dilakukan terhadap aktiva tetap tidak akan sia-sia. Dalam melaksanakan kegiatan atau aktivitas operasionalnya perusahaan selalumenggunakan sarana-sarana penunjang bagi terlaksananya operasi

perusahaan untuk mencapai tujuan yang telah ditetapkan yaitu mengoptimalkan laba yang akan dihasilkan. Aktiva tetap dapat dikelompokkan dalam berbagai sudut antara lain (repository.uin-suska.ac.id, 2015):

1. Sudut substansi, aktiva tetap dapat dibagi :
  - a. *Tangible assets* atau aktiva berwujud seperti lahan, mesin, gedung dan peralatan.
  - b. *Intangible Assets* atau aktiva yang tidak berwujud seperti Hak Guna Usaha (HGU), Hak Guna Bangunan (HGB), *Goodwill Patents, Copyright, Hakcipta, Franchise*, dan lain-lain.
2. Sudut Disusutkan Atau Tidak
  - a. *Deprciated Plant Assets* yaitu aktiva tetap yang disusutkan seperti *building* (bangunan), *equipment* (peralatan), *machinary* (mesin), inventaris, jalan dan lain-lain.
  - b. *Undepreciated Plant Assets* yaitu aktiva tetap yang tidak disusutkan seperti *land* (tanah).

### 3. Berdasarkan Jenis

Aktiva tetap berdasarkan jenis dapat dibagi sebagai berikut :

#### a. Lahan

Lahan adalah bidang tanah maupun tanah terhampar baik merupakan tempat bangunan maupun yang masih kosong. Dalam akuntansi apabila ada lahan yang didirikan bangunan di atasnya harus dipisahkan pencatatannya dari lahan itu sendiri.

#### b. Bangunan/Gedung

Gedung adalah bangunan yang berdiri di atas bumi ini baik di atas lahan/air. Pencatatannya harus terpisah dari lahan yang menjadi lokasi gedung itu.

#### c. Mesin

Mesin termasuk peralatan-peralatan yang menjadi bagian dari mesin yang bersangkutan.

d. Kendaraan

Semua jenis kendaraan seperti alat pengangkut, *truck*, *grader*, *tractor*, *forklift*, mobil, kendaraan roda dua dan lain-lain.

e. Perabot

Dalam jenis ini termasuk perabot kantor, perabot laboratorium, perabot pabrik yang merupakan isi dari suatu bangunan.

f. Inventaris/Peralatan

Peralatan yang dianggap merupakan alat-alat besar yang digunakan dalam perusahaan seperti inventaris kantor, inventaris pabrik, inventaris laboratorium, inventaris gudang dan lain-lain.

g. Prasarana

Di Indonesia adalah merupakan kebiasaan bahwa perusahaan membuat klasifikasi khusus prasarana seperti : jalan, jembatan, riol dan lain-lain.

Sehingga dapat disimpulkan bahwa secara luas aktiva tetap dapat dikelompokkan sebagai berikut (repository.uin-suska.ac.id, 2015) :

1. Aktiva tetap berwujud yaitu yang mempunyai bentuk fisik dan nyata, dan digunakan dalam operasional seperti tanah, bangunan, peralatan, dan mesin.
2. Aktiva tetap sumber alam seperti tambang, dan hasil hutan.
3. Aktiva tetap tidak berwujud yaitu aktiva yang berupa hak istimewa yang dimiliki perusahaan dan mempunyai nilai seperti hak paten, hak cipta, merk dagang, dan termasuk pula persetujuan dan perjanjian kontrak.

#### **2.6.4 Penyajian Aktiva Tetap Di Laporan Keuangan**

Aktiva tetap disajikan pada neraca disebelah debet secara sistematis sesuai dengan sifat permanennya (urutan kekekalannya) yaitu dimulai dari aktiva yang paling lama umurnya atau masa manfaatnya sampai kepada aktiva tetap yang

paling lama umurnya atau masa manfaatnya sampai pada aktiva tetap yang palingsingkat umurnya (repository.uin-suska.ac.id, 2015).

Akumulasi penyusutan dari aktiva tetap disajikan sebagai pengurangan nilai aktiva tetap baik sendiri-sendiri atau secara keseluruhan. Tujuan penyajian aktiva tetap untuk memberikan gambaran kuantitatif dan jenis-jenis aktiva tetap yang dimiliki oleh perusahaan dan juga memberikan ramalan mengenai arus kas keluar dari aktiva tetap dimasa yang akan datang.

Menurut Standar Akuntansi Keuangan, penyajian aktiva tetap dineraca adalah aktiva tetap disajikan berdasarkan nilai perolehan aktiva tetap tersebut dikurangi akumulasi penyusutan. Dalam PSAK (2011) menyatakan:

Laporan keuangan mengungkapkan, untuk setiap kelompok aset tetap:

1. Dasar pengukuran yang digunakan dalam menentukan jumlah tercatat bruto.
2. Metode penyusutan yang digunakan.
3. Umur manfaat atau tarif penyusutan (dijumlahkan dengan akumulasi rugi penurunan nilai) pada awal dan akhir periode.
4. Rekonsiliasi jumlah tercatat pada awal dan akhir periode yang menunjukkan:
  - a. Penambahan.
  - b. Aset yang diklasifikasi sebagai tersedia untuk dijual atau termasuk dalam kelompok yang akan dilepaskan yang diklasifikasikan sebagai tersedia untuk dijual sesuai dengan paragraf 45 atau pelepasan lainnya.
  - c. Akuisisi melalui penggabungan usaha.
  - d. Peningkatan atau penurunan akibat revaluasi sesuai dengan paragraf 31, 39, dan 40 serta dari segi penurunan nilai yang diakui atau dijurnal baik secara langsung pada ekuitas sesuai PSAK No.48.
  - e. Rugi penurunan nilai yang diakui dalam laporan laba rugi sesuai PSAK No. 48.

- f. Rugi penurunan nilai yang dijurnal balik dalam laporan laba rugi sesuai PSAK No.48, jika ada.
- g. Penyusutan
- h. Selisih nilai tukar neto yang timbul dalam penjabaran laporan keuangan dari mata uang fungsional menjadi mata uang pelaporan yang berbeda, termasuk penjabaran dari kegiatan usaha luar negeri menjadi mata uang pelaporan dari entitas pelapor.
- i. Perubahan lain.

### **2.6.5 Harga Perolehan Aktiva Tetap**

Aktiva tetap harus dicatat sebesar harga perolehan. Harga perolehan meliputi semua pengeluaran yang diperlukan untuk mendapatkan aktiva dan pengeluaran-pengeluaran lain agar aktiva tetap siap untuk digunakan. Pajak penjualan, biaya transportasi, asuransi aktiva tetap selama aktiva dalam perjalanan, pondasi khusus.

Menurut Juan (2012), seluruh pengorbanan ekonomis yang dikeluarkan untuk mendapatkan aset tetap hingga siap untuk dipergunakan dicatat sebagai harga perolehan. Biaya perolehan aset tetap meliputi :

1. Harga perolehannya termasuk bea impor dan pajak pembelian yang tidak boleh dikreditkan setelah dikurangi diskon pembelian dan potongan-potongan lain.
2. Biaya-biaya yang dapat distribusikan secara langsung untuk membawa aset ke lokasi dan kondisi yang diinginkan agar aset siap digunakan sesuai dengan keinginan dan maksud manajemen.
3. Estimasi awal biaya pembongkaran dan pemindahan aset tetap dan restorasi lokasi aset. Kewajiban atas biaya tersebut timbul ketika aset tersebut diperoleh atau karena entitas menggunakan aset tersebut selama periode tertentu untuk tujuan selain untuk menghasilkan persediaan.

Pada prinsip aset tetap yang diperoleh perusahaan dicatat berdasarkan harga perolehannya. Harga perolehan aset tetap meliputi harga faktur dan biaya-biaya yang dikeluarkan dalam memperoleh aset tersebut sampai dapat digunakan dalam hubungannya dengan kepemilikan sampai aset tetap tersebut siap digunakan dicatat sebagai harga perolehan.

## 2.7 Metode *Simple Additive Weighting*

Metode SAW sering dikenal dengan istilah metode penjumlahan terbobot. Konsep dasar metode SAW (*Simple Additive Weighting*) adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (etunas.co.id, 2015).

Metode SAW dapat membantu dalam pengambilan keputusan suatu kasus, akan tetapi perhitungan dengan menggunakan metode SAW ini hanya yang menghasilkan nilai terbesar yang akan terpilih sebagai alternatif yang terbaik. Perhitungan akan sesuai dengan metode ini apabila alternatif yang terpilih memenuhi kriteria yang telah ditentukan. Metode SAW ini lebih efisien karena waktu yang dibutuhkan dalam perhitungan lebih singkat.

Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Metode ini merupakan metode yang paling terkenal dan paling banyak digunakan dalam menghadapi situasi *Multiple Attribute Decision Making* (MADM).

MADM itu sendiri merupakan suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Metode SAW ini mengharuskan pembuat keputusan menentukan bobot bagi setiap atribut. Skor total untuk alternatif diperoleh dengan menjumlahkan seluruh hasil perkalian antara rating (yang dapat dibandingkan lintas atribut) dan bobot tiap atribut. Rating tiap atribut haruslah bebas dimensi dalam arti telah melewati proses normalisasi matriks sebelumnya (etunas.co.id, 2015).

Langkah Penyelesaian *Simple Additive Weighting* (SAW) sebagai berikut:

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu  $C_i$ .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria( $C_i$ ), kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan ataupun atribut biaya) sehingga diperoleh matriks ternormalisasi  $R$ .
4. Hasil akhir diperoleh dari proses perankingan yaitu penjumlahan dari perkalian matriks ternormalisasi  $R$  dengan vektor bobot sehingga
5. diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik ( $A_i$ ) sebagai solusi.

Formula untuk melakukan normalisasi tersebut adalah :

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i x_{ij}} & \text{jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\text{Min}_i x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases}$$

Dimana :

$r_{ij}$  = Nilai rating kinerja

$x_{ij}$  = Nilai kinerja dari setiap rating

$\text{Max } x_{ij}$  = Nilai terbesar dari tiap kriteria

$\text{Min } x_{ij}$  = Nilai terkecil dari tiap kriteria

## 2.8 *Software Development Life Cycle (SDLC)*

*Software Development Life Cycle* atau sering disebut juga *System Development Life Cycle* 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, 2013).

Menurut Rosa dan Shalahuddin (2013), tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut:

1. Inisiasi (*initiation*)  
Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.
2. Pengembangan konsep sistem (*system concept development*)  
Mendefinisikan 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*)  
Mengonversi 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, pengodean, pengompilasian, 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 aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

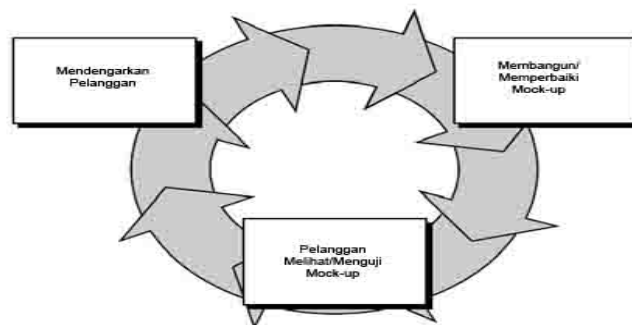
### **2.8.1 Model Prototipe Secara Umum**

Model prototipe dapat digunakan untuk menyambungkan ketidakpahaman pelanggan mengenai hal teknis yang memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak (Rosa dan Shalahuddin, 2013).

Tahapan-tahapan pada model prototipe (*prototype model*) adalah sebagai berikut (Rosa dan Shalahuddin, 2013):

1. Mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat.
2. Membuat prototipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi.

3. Program prototipe selanjutnya dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*.



Gambar II.4 Ilustrasi Model Prototipe  
Sumber: Rosa dan Shalahuddin (2013)

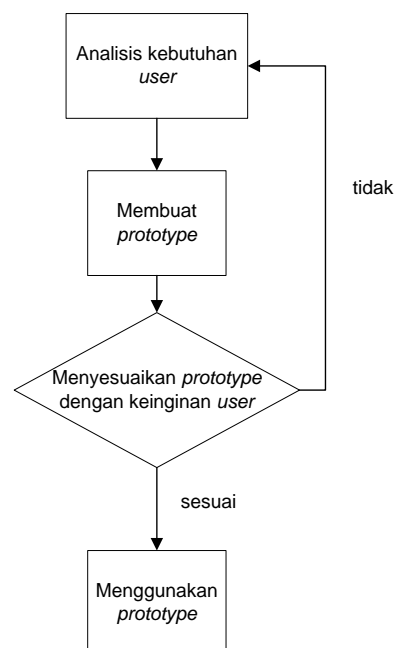
*Mock-up* adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi atau keperluan lain. Sebuah *mock-up* disebut sebagai prototipe perangkat lunak jika menyediakan atau mampu mendemonstrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Iterasi terjadi pada pembuatan prototipe sampai sesuai dengan keinginan pelanggan atau *user* (Rosa dan Shalahuddin, 2013).

Kelemahan model prototipe adalah sebagai berikut (Rosa dan Shalahuddin, 2013):

1. *User* dapat sering mengubah-ubah atau menambah spesifikasi kebutuhan karena menganggap aplikasi sudah dengan cepat dikembangkan, karena adanya iterasi ini dapat menyebabkan pengembang banyak mengalah dengan *user* karena perubahan atau penambahan spesifikasi kebutuhan perangkat lunak.
2. Pengembang lebih sering mengambil kompromi dengan pelanggan untuk mendapatkan prototipe dengan waktu yang cepat sehingga pengembang lebih sering melakukan segala cara (tanpa idealis) guna menghasilkan prototipe untuk didemonstrasikan. Hal ini dapat menyebabkan kualitas perangkat lunak yang kurang baik atau bahkan menyebabkan iteratif tanpa akhir.

### 2.8.2 *Evolutionary Prototype*

Menurut Janner (2010) Metode *Evolutionary Prototype* secara formal telah diperkenalkan pada komunitas sistem informasi pada awal tahun 1980-an. Model ini digunakan untuk mengatasi kelemahan model air terjun tradisional, hal tersebut merupakan sebuah proses iteratif dari pengembangan sistem. Pengerjaannya dilakukan secara tertutup dengan pengguna, perancangan desain, dan pembangunan model fungsional yang diturunkan dari sistem yang diinginkan.



Gambar II.5 *Evolutionary Prototype Model*  
Sumber: Janner (2010)

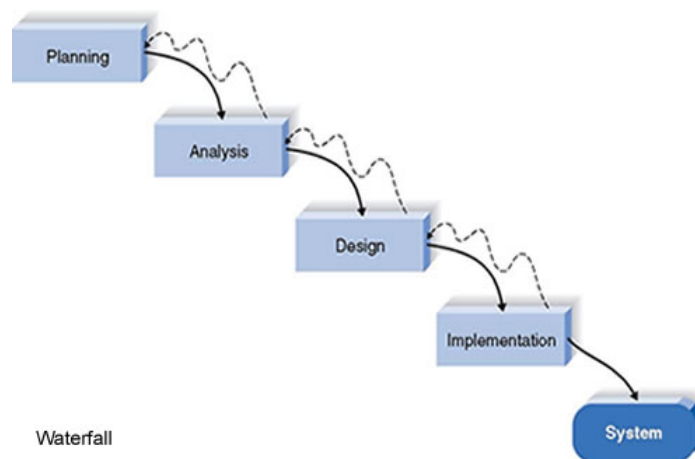
1. Analisis kebutuhan *user*, pengembang dan *user* atau pemilik sistem melakukan diskusi dimana *user* atau pemilik sistem menjelaskan kepada pengembang tentang kebutuhan sistem yang mereka inginkan.
2. Membuat *prototype*, pengembang membuat *prototype* dari sistem yang telah dijelaskan oleh *user*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

### 2.8.3 Metodologi *Waterfall*

Pada tahun 1960–1970an, proyek pengembangan perangkat lunak merupakan pekerjaan yang sangat memakan biaya dan waktu karena pengembangan perangkat lunak ini difokuskan pada perencanaan dan pengendalian. Kemunculan model *waterfall* adalah untuk membantu mengatasi kerumitan yang terjadi akibat proyek-proyek pengembangan perangkat lunak. Model *waterfall* memacu tim pengembang untuk memerinci apa yang seharusnya perangkat lunak lakukan, yaitu mengumpulkan dan menentukan kebutuhan sistem sebelum sistem dikembangkan (Simarmata, 2010).

Gambaran tahap dari metodologi *waterfall* dapat dilihat pada Gambar II.6 (Dennis et al, 2005):



Gambar II.6 *Waterfall Model*  
Sumber: Dennis et al(2005)

## 2.9 Bagan Alir (*Flowchart*)

Bagan alir adalah bagan yang menunjukkan alir didalam program atau prosedur sistem secara logika. Menurut Jogiyanto (2005), terdapat 5 macam bagan alir, yaitu:

1. Bagan alir sistem (*systems flowchart*)

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

2. Bagan alir dokumen (*document flowchart*)

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

3. Bagan alir skematik (*schematic flowchart*)

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

4. Bagan alir program (*program flowchart*)

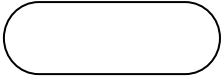





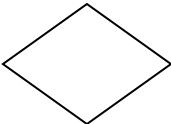
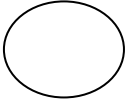
Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derivikasi bagan alir sistem.

5. Bagan alir proses (*process flowchart*)

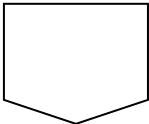
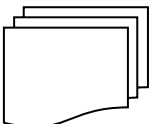
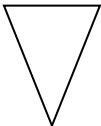
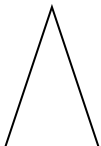
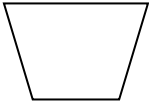
Bagan alir proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri. Bagan alir ini juga berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur.

Berikut ini adalah simbol yang biasa digunakan dalam *flow chart*:

Tabel II.2 Simbol-Simbol *Flow Chart*

Simbol	Nama	Fungsi
	Terminator	Awal dan akhir dari suatu proses.
	Garis Akhir ( <i>Front Line</i> )	Arus dari suatu proses
	<i>Preparation</i>	Proses inisialisasi awal
	Proses	Proses pengolahan data
	<i>Input/Output Data</i>	Mewakili data masukan atau keluaran.
	<i>Predefined Process</i> (Sub Proses)	Permulaan sub proses
	<i>Decision</i>	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	<i>On Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman

Tabel II.2 Simbol-Simbol *Flow Chart*(lanjutan)

Simbol	Nama	Fungsi
	<i>Off Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman berbeda
	Dokumen Rangkap	Menggambarkan dokumen asli dan tembusannya
	Arsip Sementara	Menunjukkan tempat penyimpanan dokumen
	Arsip Permanen	Menunjukkan tempat penyimpanan dokumen secara permanen yang tidak akan diproses lagi
	Proses Manual	Untuk menggambarkan kegiatan manual seperti : menerima <i>order</i> , mengisi formulir, membandingkan, dll.

Sumber: Jogyanto (2005)

## 2.10 Konsep Dasar OOAD (*Object Oriented Analysis and Design*)

### 2.10.1 Pengertian OOAD

OOAD adalah metode analisis yang memeriksa *requirements* dari sudut pandang kelas kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek system atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar

dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas (gunadarma.ac.id, 2016).

OOAD mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD). OOA adalah metode analisis yang memeriksa *requirement* (syarat/keperluan) yang harus dipenuhi sebuah sistem dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup perusahaan. Sedangkan OOD adalah metode untuk mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

Terdapat beberapa konsep dalam OOAD, yaitu :

1. Kelas (*Class*)

Kumpulan objek-objek dengan karakter yang sama. Sebuah kelas mempunyai sifat (atribut), kelakuan (operasi/metode), hubungan (*relationship*) dan arti. Suatu kelas dapat diturunkan dari kelas yang lain, dimana atribut dan kelas semula dapat diwariskan ke kelas yang baru. Kelas adalah sebuah struktur tertentu dalam pembuatan perangkat lunak. Kelas merupakan bentuk struktur pada kode program yang menggunakan metodologi berorientasi objek.

2. Objek (*Object*)

Abstraksi dan sesuatu yang mewakili dunia nyata. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan.

3. Metode (*Method*)

Operasi atau metode pada kelas hampir sama dengan fungsi atau prosedur pada metodologi terstruktur. Operasi merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan objek.

4. Atribut

Variabel global yang dimiliki kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas. Atribut dipunyai secara individu oleh suatu objek misalnya berat, jenis.

#### 5. Abstraksi

Prinsip untuk merepresentasikan dunia nyata yang kompleks menjadi suatu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan masalah.

#### 6. Enkapsulasi

Pembungkusan atribut data dan layanan (operasi-operasi) yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerjanya.

#### 7. Pewarisan

Mekanisme yang memungkinkan satu objek mewarisi sebagian atau seluruh dan objek lain sebagai bagian dari dirinya.

#### 8. Antar muka

Biasanya digunakan agar kelas yang lain tidak mengakses langsung ke suatu kelas.

#### 9. *Reusability*

Pemanfaatan kembali objek yang sudah didefinisikan untuk suatu permasalahan pada permasalahan lainnya yang melibatkan objek tersebut.

#### 10. Generalisasi & spesialisasi

Menunjukkan hubungan antar kelas dan objek yang umum dengan kelas dan objek yang khusus.

#### 11. Komunikasi Antar Objek

Dilakukan lewat pesan (*message*) yang dikirim dari satu objek ke objek lainnya.

#### 12. Polimorfisme

Kemampuan suatu objek untuk digunakan di banyak tujuan yang berbeda dengan nama yang sama sehingga menghemat program.

### 13. *Package*

Sebuah kontainer atau kemasan yang dapat digunakan untuk mengelompok kelas-kelas yang bernama sama disimpan dalam *package* yang berbeda.

#### 2.10.2 Metodologi Pengembangan Sistem Berbasis Objek

Metodologi adalah cara sistematis untuk mengerjakan analisis and desain. Dengan metodologi, pihak yang membangun *system software* dapat merencanakan dan mengulangi pekerjaan dilain waktu. Metodologi juga menghilangkan perbedaan notasi untuk suatu hal yang sama karena setiap orang akan berbicara dalam bahasa yang sama. Metodologi yang paling banyak dalam OOAD, yaitu :*Object Modeling Technique* (OMT) dari Rumbaugh, *Object Oriented Booch*, *Responsibility-Driven Design/Class Responsibility Calloboration* (RDD/CRC) dari Wirf-Broock, Metodologi Coad/ Yourdan dan Jacobson *Object Oriented Software Enginering* (OOSE) (gunadarma.ac.id, 2016).

##### 1. *Object Modeling Technique* (OMT)

Dikembangkan oleh James Rumbaugh sebagai metode untuk mengembangkan sistem berorientasi objek dan untuk mendukung pemograman berorientasi objek

##### 2. *Object Oriented Booch*

Dikembangkan oleh Grady Booch terdiri dari diagram kelas, objek, transisi status, interaksi, modul dan proses.

##### 3. *Class Resposibility Calloboration* (CRC)

Merupakan bagian dari *Object-Oriented Programming, System, Languages And Application* (OOPSLA). Dibuat untuk menjadi kelas yang akan dianalisis.

##### 4. Metodologi Coad/ Yourdan

Menyediakan sebuah diagram kelas, pembuatannya dengan langkah-langkah berikut :

- a. Mendefinisikan kelas dan objek
- b. Mengidentifikasi struktur kelas dan objek.
- c. Mendefenisikan subjek nama kelas.
- d. Mendefenisikan atribut.

e. Mendefinisikan operasi/layanan (*service*).

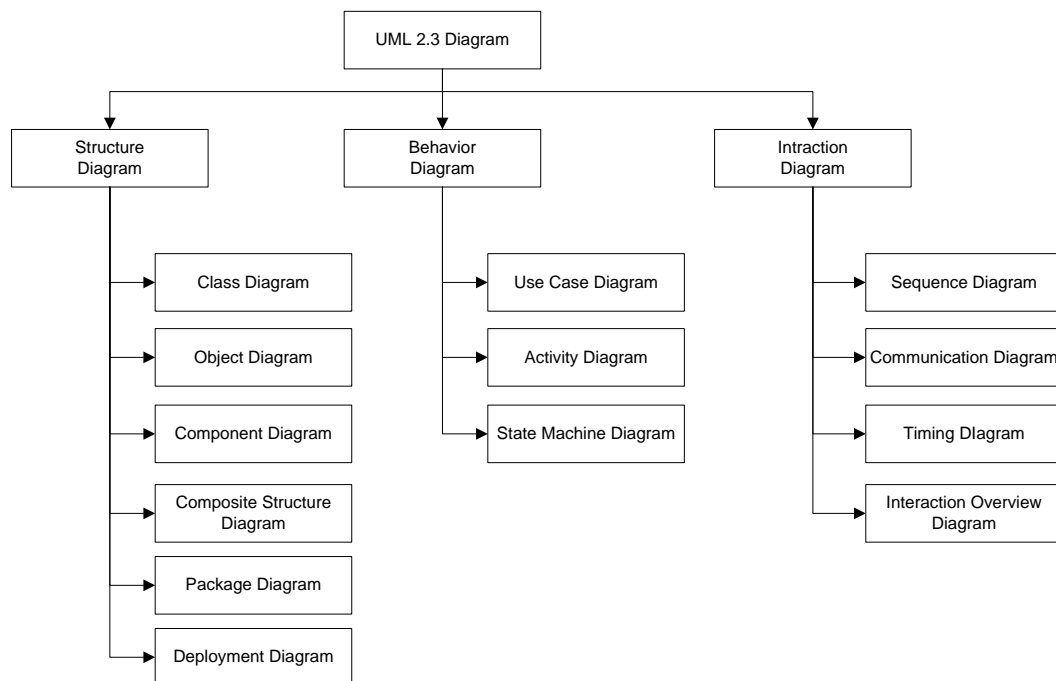
## 5. *Object Oriented Software Engineering* (OOSE)

Dikembangkan oleh Ivar Jacobson adalah metode disain berorientasi objek yang melibatkan *use case*.

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

*Unified Modeling Language* (UML) adalah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek (Rosa dan Shalahuddin, 2011).

Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori tersebut dapat dilihat pada gambar berikut ini (Rosa dan Shalahuddin, 2013) :



Gambar II.7 Klasifikasi Diagram UML

Sumber: Rosa dan Shalahuddin (2013)

Berikut ini penjelasan singkat dari pembagian kategori tersebut (Rosa dan Shalahuddin, 2013) :


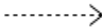



1. *Structure diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.

2. *Behavior diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan proses sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams*, yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.




### 2.11.1 Use Case Diagram

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

Tabel II.3 Simbol-simbol *Use Case Diagram*

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

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

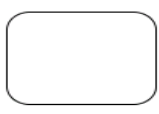



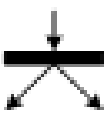
Simbol	Nama	Keterangan
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem.

Sumber: Munawar (2005)


### 2.11.2 Activity Diagram

*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2013).

Tabel II.4 Simbol-simbol *Activity Diagram*

Simbol	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
	<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

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


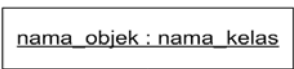

Simbol	Nama	Keterangan
	<i>Join Node</i>	Beberapa aliran yang pada tahap tertentu berubah menjadi satu aliran


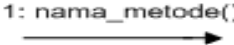
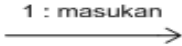
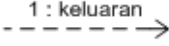
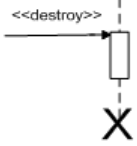
Sumber: Sommerville (2003)

### 2.11.3 *Sequence Diagram*

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat scenario yang ada pada *use case* (Rosa dan Shalahuddin, 2014).

Tabel II.5 Simbol-simbol *Sequence Diagram*

No.	Simbol	Deskripsi
2.	Garis Hidup/ <i>lifeline</i> 	Menyatakan kehidupan suatu objek.
3.	Objek 	Menyatakan objek yang berinteraksi pesan.
4.	Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.



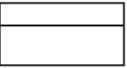


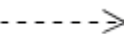

No.	Simbol	Deskripsi
5.	Pesan tipe create 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
6.	 Pesan tipe call	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode.
7.	Pesan tipe send 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8.	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 create maka ada destroy.

Sumber: Rosa dan Shalahuddin (2014)

#### 2.11.4 Class Diagram

*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa dan Shalahuddin, 2013).

Tabel II.6 Simbol-simbol *Class Diagram*




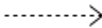

Simbol	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Sumber: Munawar (2005)

### 2.11.5 *Component Diagram*

*Component diagram* mengandung *component*, *interface* dan *relationship*. Hal yang penting pada *component* adalah *component* mewakili potongan-potongan yang independen yang bisa dipesan dan diperbaharui sewaktu-waktu. *Component* dihubungkan melalui *interface* yang diimplementasikan (Munawar, 2005).

Tabel II.7 Simbol-simbol *Component Diagram*



<b>Simbol</b>	<b>Nama</b>	<b>Keterangan</b>
	<i>Package</i>	Merupakan sebuah kumpulan dari satu atau lebih komponen
	<i>Link</i>	Relasi antar objek
	<i>Component</i>	Komponen sistem
	<i>Dependency</i>	Hubungan suatu elemen mandiri( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
	<i>Interface</i>	Sebagai antarmuka komponen

Sumber: Rosa dan Shalahuddin (2013)


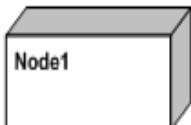
### 2.11.6 *Deployment Diagram*

*Deployment Diagram* menyediakan gambaran bagaimana sistem secara fisik akan terlihat. Sistem terdiri dari *node-node* di mana setiap *node* diwakili oleh sebuah kubus dan garis yang menghubungkan antara kubus tersebut menunjukkan hubungan antara kedua *node* tersebut (Munawar, 2005).

Tabel II.8 Simbol-simbol *Deployment Diagram*

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

Tabel II.8 Simbol-simbol *Deployment Diagram* (Lanjutan)

Simbol	Nama	Keterangan
	<i>Dependency</i>	Hubungan pada suatu elemen mandiri( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ).
	<i>Node</i>	Perangkat keras dan perangkat lunak

Sumber: Rosa dan Shalahuddin (2013)

## 2.12 Kamus Data

Kamus data adalah daftar alfabetis dari nama-nama yang termasuk pada berbagai model sistem. Seperti namanya, kamus harus mencakup deskripsi yang berhubungan dengan entitas bernama tersebut dan jika nama itu merepresentasikan objek komposit, mungkin saja ada deskripsi mengenai komposisinya. Informasi lain seperti tanggal pembuatan, pembuatnya dan representasi entitas juga dapat dimasukkan, tergantung pada tipe model yang dikembangkan (Sommerville, 2003).

Keuntungan penggunaan kamus data adalah (Sommerville, 2003):

1. Kamus data merupakan mekanisme untuk manajemen nama. Banyak orang yang harus menciptakan nama untuk entitas dan relasi ketika mengembangkan model sistem yang besar. Nama-nama ini harus dipakai secara konsisten dan tidak boleh bentrok. Kamus data dapat memeriksa keunikan nama dan memberitahu analis persyaratan sekiranya terjadi duplikasi nama.
2. Kamus data sebagai tempat penyimpanan informasi organisasional yang dapat menghubungkan analis, desain, implementasi dan evolusi. Sementara sistem dikembangkan, informasi diambil untuk memberitahu

perkembangan. Informasi baru ditambahkan pada sistem. Semua informasi mengenai entitas berada pada satu tempat.

Kamus data sangat membantu analis sistem dalam mendefinisikan data yang mengalir di dalam sistem, sehingga pendefinisian data itu dapat dilakukan dengan lengkap dan terstruktur. Pembentukan kamus data dilaksanakan dalam tahap analisis dan perancangan suatu sistem.

Pada tahap analisis, kamus data merupakan alat komunikasi antara *user* dan analis sistem tentang data yang mengalir di dalam sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh user. Sementara itu, pada tahap perancangan sistem kamus data digunakan untuk merancang *input*, laporan dan *database*. Berikut adalah contoh penulisan kamus data:

Spesifikasi tabel pemasok.

Nama tabel : Pemasok

Tipe : File master

Tabel II.9 Kamus Data Untuk Tabel Pemasok

No	Nama Elemen	Akronim	Tipe	Width	Keterangan
1	ID pemasok	ID_pemasok	Char	10	<i>Primary Key</i>
2	Nama pemasok	Nama_pemasok	Varchar	50	
3	Alamat pemasok	Alamat	Varchar	100	
4	Nomor telepon	Telepon	Varchar	12	

(Sumber: Jogiyanto, 2005)

### 2.13 *Hypertext Preprocessor (PHP)*

*Personal Home Page (PHP)* diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, PHP digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Rasmus Lerdorf adalah salah seorang pendukung *open source*. Oleh karena itu, ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis kemudian menambah kemampuan PHP dan meluncurkan PHP 2.0 (Peranginangin, 2006).

Pada tahun 1996, PHP telah banyak digunakan dalam *website* di dunia. Sebuah kelompok pengembang *software* yang terdiri dari Rasmus, Zeew Suraski, Andi Gutman, Stig Bakken, Shane Caraveo dan Jim Winstead bekerja sama untuk menyempurnakan PHP 2.0. Akhirnya, pada tahun 1998, PHP 3.0.diluncurkan. Penyempurnaan terus dilakukan sehingga pada tahun 2000 dikeluarkan PHP 4.0. Tidak berhenti sampai disitu, kemampuan PHP terus bertambah dan versi terbaru yang telah dikeluarkan adalah PHP 5.0.x (Peranginangin, 2006).

### 2.13.1 Kelebihan PHP

PHP memiliki kelebihan yang tidak dimiliki oleh bahasa *script* yang sejenis. PHP difokuskan pada pembuatan *script server-side* yang bisa melakukan apa saja yang dapat dilakukan oleh CGI, seperti mengumpulkan data dari form, menghasilkan isi halaman *web* dinamis dan kemampuan mengirim serta menerima *cookies*, bahkan lebih daripada kemampuan CGI.

PHP dapat digunakan pada semua sistem operasi, antara lain *Linux, Unix* (termasuk variannya *HP-UX, Solaris, dan OpenBSD*), *Microsoft Windows, Mac OS X, RISC OS*. PHP juga mendukung banyak *web server*, seperti Apache, *Microsoft Internet Information Server (MIIS), Personal Web Server (PWS), Netscape and iPlanet servers, Oreillu Website Pro Server, Audium, Xitami, niHTTPd*, dan bahkan PHP bisa bekerja sebagai suatu *CGI processor*.

PHP tidak terbatas pada hasil keluaran *Hypertext Markup Language* (HTML). PHP juga memiliki kemampuan untuk mengolah keluaran gambar, file PDF, dan *movies* Flash. PHP juga dapat menghasilkan teks seperti XHTML dan file XML lainnya. Salah satu fitur yang dapat diandalkan oleh PHP adalah dukungannya terhadap banyak *database* (Peranginangin, 2006). Berikut *database* yang dapat didukung oleh PHP diantaranya: *dBase, FrontBase, IBM DB2, Informix, Ingres, MSOL, MySQL, Oracle (OC17 dan OC18), PostgrSQL, dan Sybase* (Peranginangin, 2006).

## 2.14 *Codeigniter Framework*

*Codeigniter* merupakan *framework* PHP yang diklaim memiliki eksekusi tercepat dibandingkan dengan *framework* lainnya. *Codeigniter* bersifat *open source* dan menggunakan model basis MVC (*Model, View, Controller*), yang merupakan model konsep modern *framework* yang digunakan saat ini (Saputra, 2011).

*Framework* adalah suatu kerangka kerja dalam aplikasi web yang didalamnya memiliki suatu potongan-potongan program yang disusun (modul), sehingga *programmer* tidak perlu membuat kode dari nol, karena *framework* telah menyediakannya (Saputra, 2011).



Gambar II.8 Logo *Codeigniter Framework*  
Sumber: Saputra (2011)

### 2.14.1 Kelebihan dan Kelemahan *Codeigniter*

*Codeigniter Framework* memiliki beberapa kelebihan (versi 2.1.0), diantaranya (Saputra, 2011):

1. Gratis

*Codeigniter* dilisensikan di bawah lisensi dari *Apache/BSD style open source*, sehingga gratis untuk memperolehnya.

2. Mendukung PHP4 dan PHP 5

Karena masih banyak orang yang masih menggunakan PHP4, maka *Codeigniter* dikembangkan agar mampu berjalan baik pada PHP4 maupun PHP5.

3. Berukuran kecil dan cepat

Dikarenakan *Codeigniter* hanya me-load fungsi atau *library* yang digunakan saja, berbeda dengan *framework* lainnya yang menggunakan seluruh *library* walaupun *library* tersebut tidak digunakan.

4. Dokumentasi yang baik

Dikarenakan didukung *user guide* yang mudah dimengerti.

5. Menggunakan konsep MVC

Dengan adanya MVC, pengerjaan antara logika dan *layout* dapat dipisahkan sehingga memudahkan *programmer* dan *designer*.

6. Komunitas yang banyak

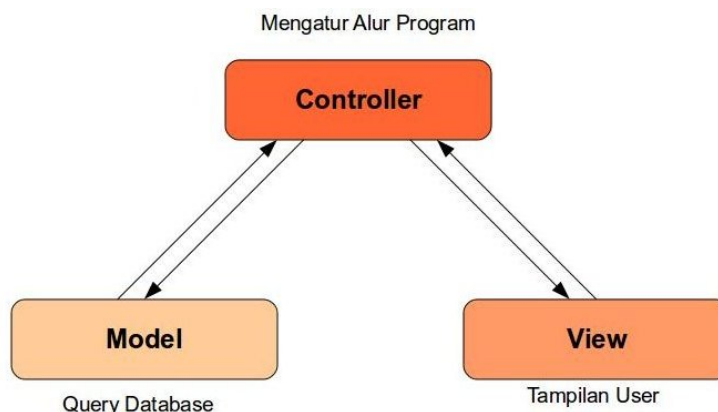
Disamping kelebihanannya, *Codeigniter Framework* juga mempunyai beberapa kelemahan, diantaranya (Saputra, 2011):

1. *Codeigniter* dikembangkan oleh *Ellislab* dan bukan oleh suatu komunitas.
2. Tidak ditujukan untuk pembuatan web dengan skala besar (*enterprise*) walaupun tersedia banyak *library*.
3. Masih banyak kelonggaran dalam hal *coding*, misalnya bebas dalam penamaan *file*.
4. Tidak mencerminkan *Model View Controller* yang sesungguhnya, misalnya penulisan *echo* masih bisa dilakukan pada *controller*.

### 2.14.2 Konsep MVC

MVC memiliki kepanjangan dari *Model View Controller*, merupakan alur kerja dari *Codeigniter Framework*. Dengan konsep MVC ini, berbagai macam logika dan *layout* telah dipisahkan, sehingga *programmer* dan *designer* dapat mengerjakan masing-masing tugasnya secara fokus. Konsep model MVC juga menuntut para pembuat program untuk membangun web dengan cara yang terstruktur (Saputra, 2011).

Alur kerja dari *Codeigniter Framework* dapat dilihat dari gambar berikut (Saputra, 2011) :



Gambar II.9 Konsep MVC  
Sumber: Saputra (2011)

1. *Model*, digunakan sebagai presentasi *database*.
2. *View*, adalah suatu halaman khusus yang digunakan untuk menyajikan informasi kepada *user*.
3. *Controller*, digunakan sebagai pengendali (*control*) antara *view* dan *model* melalui permintaan dari HTTP.

## 2.15 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang *multithread*, *multiuser*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public Licence* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL (Solichin, 2010).

Fitur-fitur MySQL antara lain (Solichin, 2010):

1. *Relational Database System*  
Seperti halnya *software database* lain yang ada di pasaran, MySQL termasuk RDBMS (*Relational DataBase Management System*).
2. *Arsitektur Client-Server*  
MySQL memiliki arsitektur *client-server* dimana *server database* MySQL terinstal di *server*. *Client* MySQL dapat berada di komputer yang sama dengan *server* dan dapat juga di komputer lain yang berkomunikasi dengan server melalui jaringan bahkan internet.

3. Mengenal perintah SQL standar  
SQL (*Structured Query Language*) merupakan suatu bahasa standar yang berlaku di hampir semua *software database*. MySQL mendukung SQL versi SQL:2003.
4. Mendukung *Sub Select*  
Mulai versi 4.1 MySQL telah mendukung *select* dalam *select (sub select)*.
5. Mendukung *Views*  
MySQL mendukung *views* sejak versi 5.0
6. Mendukung *Stored Prosedured (SP)*  
MySQL mendukung SP sejak versi 5.0
7. Mendukung *Triggers*  
MySQL mendukung *trigger* pada versi 5.0 namun masih terbatas. Pengembang MySQL berjanji akan meningkatkan kemampuan trigger pada versi 5.1.
8. Mendukung *replication*.
9. Mendukung transaksi.
10. Mendukung *foreign key*.
11. Tersedia fungsi GIS.
12. *Free* (bebas diunduh)
13. Stabil dan tangguh
14. Fleksibel dengan berbagai pemrograman
15. *Security* yang baik
16. Dukungan dari banyak komunitas
17. Perkembangan *software* yang cukup cepat.

### 2.15.1 Tipe Data MySQL

Secara garis besar, *database* MySQL mempunyai 3 macam tipe data, yaitu (naura-lab.blogspot.co.id, 2016) :

## 1. Tipe Data Numeric

Tipe Data Numeric pada *database* MySQL terbagi atas beberapa macam tipe data, yaitu:

### a. INT

Digunakan untuk menyimpan data yang berupa bilangan bulat positif dan negatif dengan jangkauan antara -2.147.483.648 s/d 2.147.483.647. Tipe data ini mempunyai ukuran 4 byte (32 bit).

### b. TINYINT

Digunakan untuk menyimpan data yang berupa bilangan bulat positif dan negatif dengan jangkauan antara -128 s/d 127. Tipe data ini mempunyai ukuran 1 *byte* (8 bit).

### c. SMALLINT

Digunakan untuk menyimpan data yang berupa bilangan bulat positif dan negatif dengan jangkauan antara -32.768 s/d 32.767. Tipe data ini mempunyai ukuran 2 *byte* (16 bit).

### d. MEDIUMINT

Digunakan untuk menyimpan data yang berupa bilangan bulat positif dan negatif dengan jangkauan antara -8.388.608 s/d 8.388.607. Tipe data ini mempunyai ukuran 3 *byte* (24 bit).

### e. BIGINT

Digunakan untuk menyimpan data yang berupa bilangan bulat positif dan negatif dengan jangkauan antara -8.388.608 s/d 8.388.607. Tipe data ini mempunyai ukuran 8 *byte* (64 bit) .

### f. FLOAT

Digunakan untuk menyimpan data yang berupa bilangan pecahan positif dan negatif presisi tunggal. Tipe data ini mempunyai ukuran 4 *byte* (32 bit).

### g. DOUBLE

Digunakan untuk menyimpan data yang berupa bilangan pecahan

positif dan negatif presisi ganda. Tipe data ini mempunyai ukuran 8 byte (64 bit).

h. DECIMAL

Digunakan untuk menyimpan data yang berupa bilangan pecahan positif dan negatif presisi ganda. Tipe data ini mempunyai ukuran 8 byte (64 bit).

i. REAL

Digunakan untuk menyimpan data yang berupa bilangan pecahan positif dan negatif. Tipe data ini mempunyai ukuran 8 byte (64 bit).

j. NUMERIC

Digunakan untuk menyimpan data yang berupa bilangan pecahan positif dan negatif. Tipe data ini mempunyai ukuran 8 byte (64 bit).

2. Tipe Data *Date & Time*

Tipe Data *Date & Time* pada database MySQL terbagi atas beberapa macam tipe data, yaitu :

a. DATE

Digunakan untuk menyimpan data tanggal dalam format YY:MM:DD

b. DATETIME

Digunakan untuk menyimpan data tanggal dan waktu dalam format YY:MM:DD HH:MM:SS

c. TIME

Digunakan untuk menyimpan data waktu dalam format HH:MM:SS

d. YEAR

Digunakan untuk menyimpan data tahun.

3. Tipe Data String

Tipe Data String pada database MySQL terbagi atas beberapa macam tipe data, yaitu:

a. CHAR

Digunakan untuk menyimpan data karakter/string dengan ukuran

tetap. Tipe data ini mempunyai jangkauan antara 0 sampai dengan 255 karakter.

b. VARCHAR

Digunakan untuk menyimpan data karakter/string dengan ukuran dinamis. Tipe data ini mempunyai jangkauan antara 0 sampai dengan 255 untuk MySQL versi 4.1. Dan mempunyai jangkauan antara 0 s/d 65.535 untuk MySQL versi 5.0.3

c. BLOB

BLOB (*Binary Large Object*) adalah tipe data yang digunakan untuk menyimpan data biner seperti *files*, *images*, suara, dll. Tipe data ini mempunyai jangkauan 2<sup>16</sup>-1 byte.

d. TINYBLOB

Digunakan untuk menyimpan data biner seperti *file*, *image*, dan suara. Tipe data ini mempunyai jangkauan 255 byte.

e. MEDIUMBLOB

Digunakan untuk menyimpan data biner seperti *file*, *image*, dan suara. Tipe data ini mempunyai jangkauan 2<sup>24</sup>-1 byte.

f. LONGBLOB

Digunakan untuk menyimpan data biner seperti *file*, *image*, dan suara. Tipe data ini mempunyai jangkauan 2<sup>32</sup>-1 byte

g. TEXT

Digunakan untuk menyimpan data text. Tipe data ini mempunyai jangkauan antara 0 sampai dengan 65.535 (2<sup>16</sup>-1) karakter.

h. TINYTEXT

Digunakan untuk menyimpan data text. Tipe data ini mempunyai jangkauan antara 0 s/d 255 untuk MySQL versi 4.0, dan mempunyai jangkauan antara 0 s/d 65.535 untuk MySQL versi 5.0.3

i. MEDIUMTEXT

Digunakan untuk menyimpan data text. Tipe data ini mempunyai jangkauan antara 0 sampai dengan 2<sup>24</sup>-1 karakter

## **BAB III METODOLOGI PENELITIAN**

Metodologi penelitian merupakan ilmu mengenai jenjang-jenjang yang harus dilalui dalam suatu proses penelitian (Rianto, 2004). Metode penelitian ini terdiri dari beberapa tahapan, yaitu tahapan pengumpulan data atau informasi dan pengembangan sistem. Dalam tahap pengumpulan data dan informasi dilakukan dengan cara observasi langsung di tempat penelitian, wawancara dengan pengguna sistem yang diamati dan studi kepustakaan. Sedangkan dalam pengembangan sistem menggunakan model *prototype* jenis *evolutionary*.

### **3.1 Jenis dan Sumber Data**

Data yang dibutuhkan dalam penelitian ini dikelompokkan menjadi dua jenis, yaitu sumber data primer dan data sekunder.

#### **1. Data primer**

Data yang diperoleh melalui pengamatan langsung pada PT Astra Honda Motor pada bagian *Asset Control* terhadap sistem yang sedang berjalan dan wawancara dengan pegawai sebagai sumber informasinya. Dalam penelitian ini data tersebut berupa data alur proses order dan pelaksanaan observasi serta dokumen yang terlibat pada proses kontrol AUC diantaranya kebijakan *fixed asset*, intruksi kerja observasi, sampel aset dan laporan hasil observasi.

#### **2. Data Sekunder**

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

### 3.2 Metode Pengumpulan Data

Metode yang digunakan dalam pengumpulan data pada penelitian ini, diantaranya :

1. Studi Pustaka

Tahap ini merupakan tahap awal pengumpulan data dari sumber-sumber seperti buku, jurnal, yang berhubungan dengan masalah yang sedang dianalisis. Studi pustaka yang dilakukan berkaitan dengan analisis dan desain sistem informasi, pemrograman PHP, pemrograman MySQL, *fixed asset*, pengembangan perangkat lunak, dan juga beberapa contoh *coding* program melalui media *internet*.

2. Observasi

Tahap ini merupakan tahap melakukan pengamatan langsung terhadap pengguna sistem dan pengamatan langsung terhadap sistem yang sedang berjalan di PT Astra Honda Motor. Observasi ini dilakukan pada bagian *Asset Control* untuk mengetahui alur proses kontrol *Asset Under Construction* (AUC) yang meliputi proses kegiatan order, observasi dan *settlement*.

3. Wawancara

Pada tahap ini penulis melakukan wawancara dengan para pengguna sistem yang sedang berjalan tersebut untuk memperoleh keterangan mengenai alur proses kegiatan kontrol *Asset Under Construction* (AUC) serta mengenai kebutuhan sistem yang akan dikembangkan selanjutnya. Wawancara ini dilakukan pada karyawan PT Astra Honda Motor pada bagian *Asset Control*.

### 3.3 Metode Pendekatan, Pemecahan Masalah dan Pengembangan Sistem

Metode pendekatan analisis dan desain yang digunakan yaitu analisis dan perancangan berorientasi objek dengan alat bantu analisis perancangan *Unified Modeling Language* (UML). Adapun metode pengembangan perangkat lunak yang digunakan adalah *evolutionary prototype* sedangkan untuk metode pemecahan masalah menggunakan *Simple Additive Weighting*.

### 3.3.1 Metode Pendekatan Sistem

Metode pendekatan yang digunakan adalah metode pendekatan berorientasi objek (*Object Oriented*). Menurut Nugroho (2005) Pendekatan berorientasi objek merupakan cara berfikir baru serta berlogika dalam menghadapi masalah-masalah yang akan di atasi dengan bantuan komputer. OOP mencoba melihat permasalahan lewat pengamatan dunia nyata dimana setiap objek adalah entitas tunggal yang memiliki kombinasi struktur data dan fungsi. Pendekatan berorientasi objek terdiri dari analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD).

Menurut Sumarta, Siswoyo dan Juhana (2004) dalam membangun sebuah sistem yang kompleks memerlukan perancangann model yang sistematis dalam mengerjakan pekerjaan analisis dan desainnya. Dengan *Unified Modeling Language* (UML) pekerjaan analisis dan desain dapat dengan mudah dirancang karena merupakan pemodelan secara visual dan memiliki semantik dan notasi UML yang bekerja dalam OOA dan OOD.

### 3.3.2 Metode Pemecahan Masalah

Metode pemecahan masalah yang digunakan adalah *Simple Additive Weighting* (SAW) untuk menentukan aset mana yang akan menjadi prioritas utama dalam proses order, observasi dan *settlement* di PT Astra Honda Motor. Dalam penerapan metode ini dibagi menjadi tiga atribut sebagai dasar penilaian, yaitu kategori proyek dan harga aset sebagai atribut *benefit* sedangkan tanggal kebutuhan sebagai atribut *cost*. Dengan metode inilah proses kontrol dari pengadaan hingga *settlement* aset yang dilakukan dapat lebih optimal.

### 3.3.3 Metode Pengembangan Sistem

Metode pengembangan *Procurement Control System* (PCS) ini menggunakan model *prototype* jenis *evolutionary*. Model *prototype* cocok digunakan untuk menjabarkan kebutuhan *user* secara lebih detail karena *user* sering kali kesulitan menyampaikan kebutuhannya secara *detail* tanpa melihat gambaran yang jelas (Rosa dan Shalahuddin, 2011).

Menurut Janner tahapan-tahapan pada model *prototype* adalah sebagai berikut:

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.

### **3.4 Kerangka Penelitian**

Beberapa tahapan yang penulis lakukan dalam melakukan penelitian ini adalah sebagai berikut:

#### **1. Studi Pendahuluan**

Tujuan melakukan studi pendahuluan adalah untuk mendapatkan pengetahuan umum mengenai sistem yang sedang diteliti. Pada tahap ini penulis mengumpulkan data yang diperlukan dengan cara observasi langsung terhadap sistem yang berjalan, serta wawancara dengan pihak yang mempunyai hubungan dan pengetahuan terhadap sistem yang berjalan. Data yang dikumpulkan berkaitan dengan kebutuhan *user* terhadap sistem yang akan dikembangkan, seperti alur dokumen sistem yang sedang berjalan, arus informasi yang terdapat dalam sistem yang sedang berjalan serta dokumen-dokumen yang digunakan pada sistem yang sedang berjalan.

#### **2. Identifikasi dan perumusan masalah**

Tahap ini penulis mengidentifikasi dan menetapkan permasalahan yang terjadi selama penelitian pada sistem kontrol *Asset Under Construction* (AUC) yang sedang berjalan pada PT Astra Honda Motor.

3. Tujuan Penelitian

Penulis menetapkan tujuan dari penelitian pada sistem informasi kontrol *Asset Under Construction* (AUC) untuk memberikan solusi dari masalah yang teridentifikasi dalam proses kontrol AUC.

4. Batasan Masalah

Pada tahap ini memberikan batasan terhadap proses yang akan diteliti sehingga tidak terlalu luas dan penelitian lebih terarah.

5. Analisis dan Perancangan Sistem Informasi Kontrol AUC

Kegiatan analisis untuk mengembangkan sistem informasi dilakukan melalui tahapan-tahapan sebagai berikut:

a. Analisis kebutuhan *user*

Melakukan analisis kembali terhadap data yang telah didapat dari hasil observasi dan wawancara, sehingga dapat diketahui kebutuhan *user* terhadap aplikasi yang akan dibuat.

b. Pemodelan sistem

Membuat pemodelan sistem berbasis objek dengan menggunakan *Unified Modeling Language* (UML) *Diagram*, yaitu:

1) *Use Case Diagram*

2) *Activity Diagram*

3) *Sequence Diagram*

c. Pemodelan data

Membuat desain pemodelan data dengan menggunakan *Class Diagram* dan Kamus Data.

d. Desain antarmuka

Membuat desain antar muka menggunakan Microsoft Visio.

e. Desain sistem

Membuat desain sistem dengan menggunakan *Hierarchy plus Input-Proses-Output* (HIPO) dan *Flowchart*.

6. Pembuatan *prototype* sistem

Tahap selanjutnya *prototype* sistem, pengkodean sistem dan penerapan metode pemecahan masalah pada algoritma *prototype*. *Prototype* yang dibuat adalah:

- a. Membuat *Procurement Control System* (PCS) dengan menggunakan *tools* Sublime Text 2. Sublime Text 2 dipilih karena terdapat bermacam-macam fitur ini membuat pengolahan kode menjadi lebih cepat dan mudah (sublimetext.com).
- b. Membuat *database* dengan perangkat lunak basis data MySQL. MySQL dipilih karena dapat berjalan stabil pada berbagai macam sistem operasi (phpmyadmin.net).

7. Menyesuaikan *Prototype* dengan Keinginan *User*

Pada tahap ini, *prototype* yang diusulkan akan diperlihatkan dan diuji coba terlebih dahulu kepada *user* apakah sudah sesuai dengan keinginan *user* atau belum. Jika belum maka dilakukan analisis kembali, tetapi apabila sesuai maka tahap selanjutnya adalah implementasi penggunaan *prototype* tersebut.

8. Implementasi Sistem

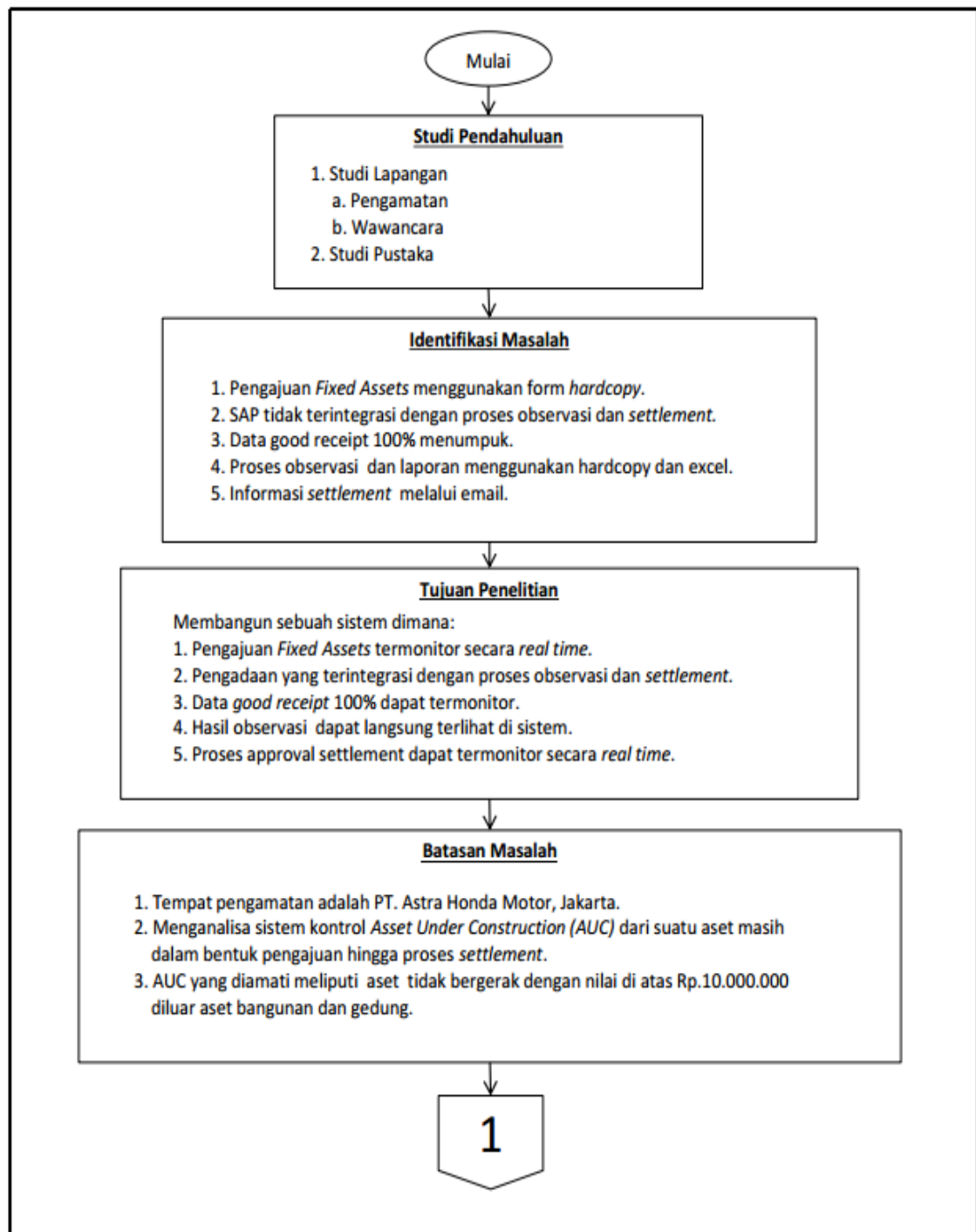
Setelah semua fungsi-fungsi *software* harus diuji coba agar *software* bebas dari kesalahan, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya. Maka proses selanjutnya adalah bagaimana sistem baru akan diinstal dan dijalankan di perusahaan dengan pengoperasian yang dilakukan oleh *user*.

9. Kesimpulan dan Saran

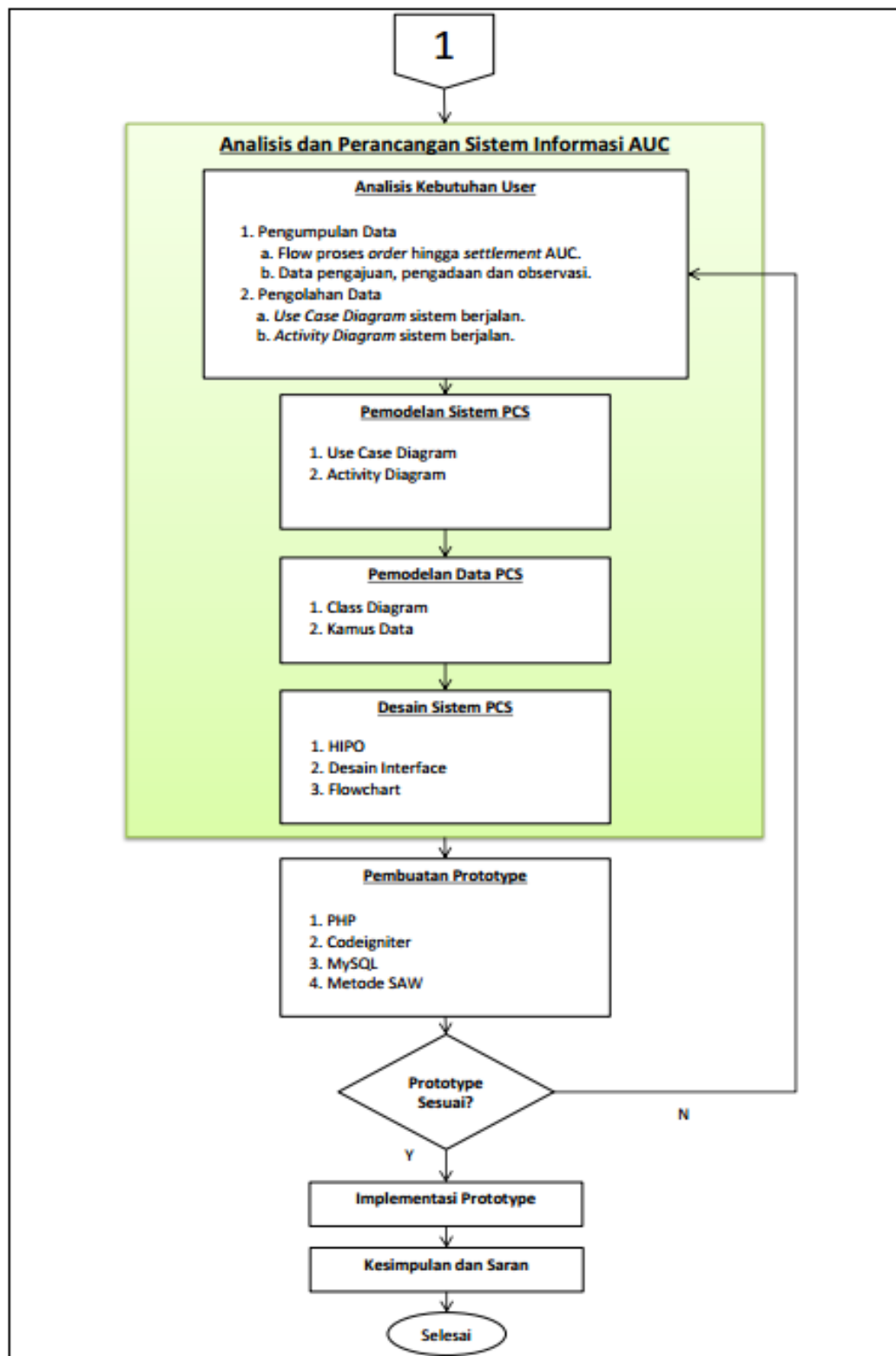
Setelah *prototype* sistem usulan diterima maka tahap selanjutnya adalah menarik kesimpulan dari penelitian yang telah dilakukan di PT Astra Honda Motor dan memberikan saran yang membangun bagi perusahaan

tersebut. Saran untuk penelitian selanjutnya agar dapat mengembangkan *software* ini agar dapat diintegrasikan.

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



Gambar III.1 Kerangka Penelitian (Lanjutan)  
Sumber: Pengolahan Data ( 2016)



Gambar III.2 Kerangka Penelitian (Lanjutan)  
 Sumber: Pengolahan Data (2016)

## **BAB IV**

### **PENGUMPULAN DAN PENGOLAHAN DATA**

#### **4.1 Sejarah Perusahaan**

PT Astra Honda Motor (AHM) merupakan pelopor industri sepeda motor di Indonesia. Didirikan pada 11 Juni 1971 dengan nama awal PT Federal Motor, yang sahamnya secara mayoritas dimiliki oleh PT Astra International. Saat itu, PT Federal Motor hanya merakit, sedangkan komponennya diimpor dari Jepang dalam bentuk CKD (*completely knock down*).

Tipe sepeda motor yang pertama kali di produksi Honda adalah tipe bisnis, S 90 Z bermesin 4 tak dengan kapasitas 90cc. Jumlah produksi pada tahun pertama selama satu tahun hanya 1500 unit, namun melonjak menjadi sekitar 30 ribu pada tahun berikutnya dan terus berkembang hingga saat ini. Sepeda motor terus berkembang dan menjadi salah satu moda transportasi andalan di Indonesia.

Kebijakan pemerintah dalam hal lokalisasi komponen otomotif mendorong PT Federal Motor memproduksi berbagai komponen sepeda motor Honda tahun 2001 di dalam negeri melalui beberapa anak perusahaan, diantaranya PT Honda Federal (1974) yang memproduksi komponen-komponen dasar sepeda motor Honda seperti rangka, roda, knalpot dan sebagainya, PT Showa Manufacturing Indonesia (1979) yang khusus memproduksi peredam kejut, PT Honda Astra Engine Manufacturing (1984) yang memproduksi mesin sepeda motor serta PT Federal Izumi Mfg.(1990) yang khusus memproduksi piston.

Seiring dengan perkembangan kondisi ekonomi serta tumbuhnya pasar sepeda motor terjadi perubahan komposisi kepemilikan saham di pabrikan sepeda motor Honda ini. Pada tahun 2000 PT Federal Motor dan beberapa anak perusahaan di merger menjadi satu dengan nama PT Astra Honda Motor, yang komposisi kepemilikan sahamnya menjadi 50% milik PT Astra International Tbk dan 50% milik Honda Motor Co. Japan.

Saat ini PT Astra Honda Motor memiliki 4 fasilitas pabrik perakitan, pabrik pertama berlokasi Sunter, Jakarta Utara yang juga berfungsi sebagai kantor pusat. Pabrik ke dua berlokasi di Pegangsaan Dua, Kelapa Gading, pabrik ketiga berlokasi di kawasan MM 2100 Cikarang Barat, Bekasi, serta pabrik keempat yang sekaligus pabrik paling mutakhir berlokasi di kawasan industri Indotaisei, Karawang. Pabrik keempat ini merupakan fasilitas pabrik perakitan terbaru yang mulai beroperasi sejak tahun 2014.

Dengan keseluruhan fasilitas ini PT Astra Honda Motor saat ini memiliki kapasitas produksi 5 juta unit sepeda motor per-tahunnya, untuk permintaan pasar sepeda motor di Indonesia yang terus meningkat.

Salah satu puncak prestasi yang berhasil diraih PT Astra Honda Motor adalah pencapaian produksi ke 40 juta pada tahun 2014. Prestasi ini merupakan prestasi pertama yang berhasil diraih oleh industri sepeda motor di Indonesia bahkan untuk tingkat ASEAN. Secara dunia pencapaian produksi sepeda motor Honda 40 juta unit adalah yang kedua, setelah pabrik sepeda motor Honda di India.

Guna menunjang kebutuhan serta kepuasan pelanggan sepeda motor Honda, saat PT Astra Honda Motor di dukung oleh 1.600 *showroom dealer* penjualan yang diberi kode H1, 3.800 layanan *service* atau bengkel AHASS (*Astra Honda Authorized Service Station*) dengan kode H2, serta 6.500 gerai suku cadang atau H, yang siap melayani jutaan penggunaan sepeda motor Honda di seluruh Indonesia.

Industri sepeda motor saat ini merupakan suatu industri yang besar di Indonesia. Karyawan PT Astra Honda Motor saja saat ini berjumlah sekitar 22.000 orang, ditambah 130 *vendor* dan *supplier* serta ribuan jaringan lainnya, yang kesemuanya ini memberikan dampak ekonomi berantai yang luar biasa. Keseluruhan rantai ekonomi tersebut diperkirakan dapat memberika kesempatan kerja kepada sekitar 500 ribu orang. PT Astra Honda Motor akan terus berkarya menghasilkan sarana transportasi roda 2 yang menyenangkan, aman dan ekonomis sesuai dengan harapan dan kebutuhan masyarakat Indonesia.

Prestasi PT Astra Honda Motor mendapat kehormatan meraih sejumlah penghargaan dari berbagai organisasi dan bangga akan apa yang telah tercapai. PT Astra Honda Motor percaya kesuksesan ini akan menjadi motivasi untuk melakukan yang lebih baik lagi di masa yang akan datang. Penghargaan-penghargaan yang telah diterima antara lain:

1. *Motorcycle Indonesia WOW Brand* 2014.
2. Penghargaan Evaluasi Baku Mutu Emisi Gas Buang Kendaraan Bermotor Tipe Baru 2013 dari Kementerian Lingkungan Hidup.
3. *Digital Marketing Award* 2014.
4. *Hai Youth Brand Award* 2014.
5. *Indonesia Most Favorite Netizen Brand Award* 2014.
6. Pelopor *Link and Match* Bidang Keahlian Sepeda Motor di Sekolah Menengah Kejuruan 2013.
7. Penghargaan Kepedulian dan Partisipasi dalam Pelaksanaan Pendidikan Etika Berlalu Lintas di Satuan Pendidikan 2013.
8. *Words of Mouth Award* 2014.
9. *Indonesia's Most Favorite Woman Brand* 2014.
10. *Indonesia's Most Admired Companies* 2014.
11. *Indonesia's Most Favorite Youth Brand* 2014.
12. *Otomotif Award* 2014.
13. *Top Brand Award* 2015.
14. *Excellent Service Experience Award* 2015.
15. *Motorplus Award* 2014.

#### **4.2 Profil Perusahaan**

Nama Perusahaan : PT Astra Honda Motor  
 Status Perusahaan : Perseroan Terbatas  
 Status Investasi : PMA (Penanaman Modal Asing)

Alamat : Kantor Pusat & Pabrik Perakitan Sunter  
JL Laksda Yos Sudarso, Sunter I  
Jakarta 14350  
Telp 021 6518080  
Fax 021 6521889

Pabrik Perakitan Cikarang  
Kawasan Industri MM2100, Jl Raya Kalimantan  
Blok AA-1  
Cikarang Barat  
Telp 021 89981818  
Fax 021 8980859

AHM Parts Center  
Jl Tipar Inspeksi Cakung Drain, Cakung Barat  
Jakarta 13910  
Telp 021 468 35020  
Fax 021 46835025

Dies Manufacturing Division  
Kawasan Industri Pulogadung, Jl Pulo Ayang Raya  
Blok F No2  
Jakarta Timur  
Telp 021 4602574-6  
Fax 021 4608904

Pabrik Perakitan Pegangsaan  
Jl Raya Pegangsaan 2 km 2.2 Kelapa Gading

Jakarta 14250  
Telp 021 46822510

Pabrik Perakitan Karawang  
Kawasan Industri Indotaisei, Kota Bukit Indah, Karawang  
Jawa Barat  
Telp 0264 8379090

Astra Honda training Center  
Jl Agung Timur IX Blok 01 Kav 25-26, Sunter II  
Jakarta 14350  
Telp 021 65308080  
Fax 021 6510460

Tanggal Pendirian : 11 Juni 1971 (d/h Federal Motor)  
Agen Tunggal Pemegang Merek (ATPM),

Aktivitas : Manufaktur, Perakitan dan Distributor Sepeda Motor  
HONDA

Total Distribusi  
(2014) : 5.051.100 Unit

Kapasitas Produksi : 5.300.000 unit/tahun

Status Kepemilikan : 50% PT. Astra International Tbk  
50% Honda Motor Co., Ltd

Referensi Standar : JIS (*Japan Industrial Standard*)  
SII (Standar Industri Indonesia)  
SNI (Standar Nasional Indonesia)  
HES (*Honda Engineering Standard*)  
ISO 9001

ISO 14001  
 ISO 17025  
 OHSAS 18001

Jam Kerja

Kantor : 07.30 - 16.30 WIB  
 Pabrik Shift I : 07.00 - 16.00 WIB  
 Pabrik Shift II : 16.00 - 24.00 WIB  
 Pabrik Shift III : 24.00 - 07.00 WIB  
 Jumlah Karyawan : 22.563 orang (Desember 2014)  
 Website : <http://www.astra-honda.com>

#### 4.3 Visi dan Misi Perusahaan

1. Visi

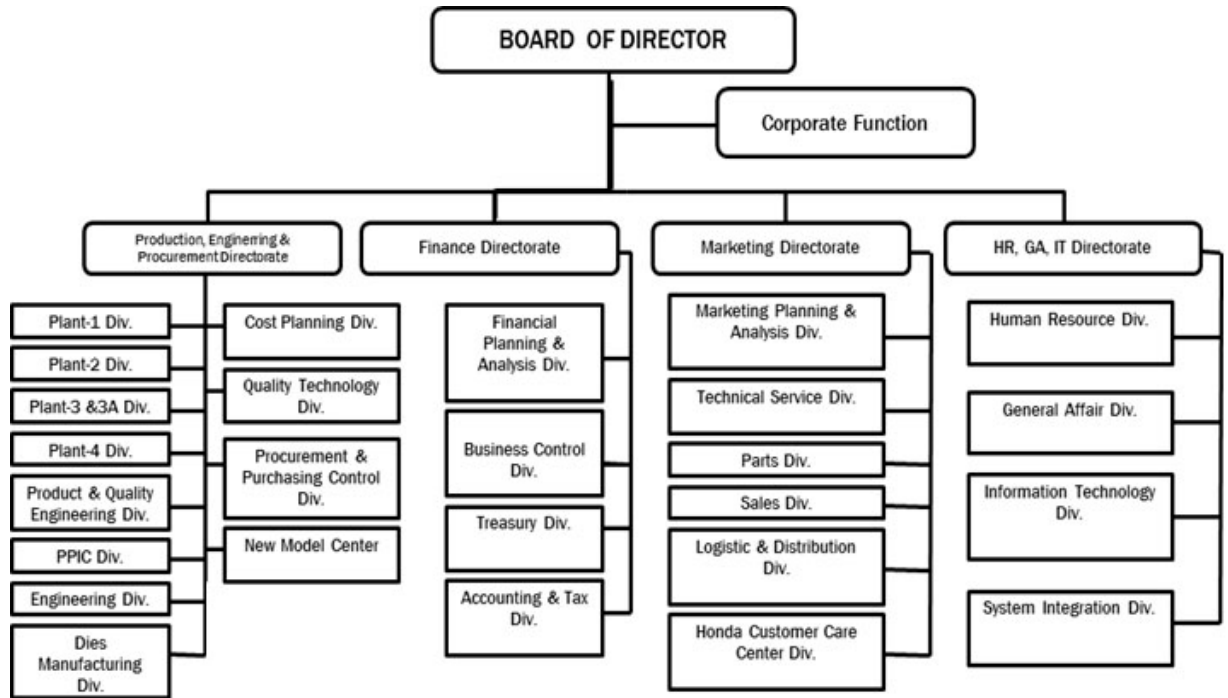
Menjadi pemimpin pasar sepeda motor di Indonesia dengan cara merealisasikan mimpi dan menciptakan kegembiraan para pelanggan serta berkontribusi bagi masyarakat Indonesia.

2. Misi

Menciptakan solusi mobilitas bagi masyarakat Indonesia dengan produk dan layanan terbaik.

#### 4.4 Struktur Organisasi Perusahaan

Demi Kelancaran pelaksanaan dan mempermudah dalam koordinasi maka PT. Astra Honda Motor membentuk struktur organisasi yang menjelaskan tugas dan tanggung jawab sesuai dengan fungsi, tanggung jawab dan wewenangnya. Struktur beberapa orang Direktur antara lain *Finance Directorate*, *Production Directorate*, *Marketing Directorate*, dan *HR, GA, IT Directorate*. Adapun bentuk struktur organisasi PT Astra Honda Motor sampai *Business Control Division* digambarkan dalam bagan pada gambar IV.1.



Gambar IV.1 Struktur Organisasi PT Astra Honda Motor  
Sumber: PT Astra Honda Motor, 2015

Di bawah ini akan dijelaskan mengenai tugas dan tanggung jawab tiap bagian dari struktur organisasi PT Astra Honda Motor.

#### 1. *President Director*

*President Director* mempunyai tugas untuk memimpin dan mengkoordinasikan seluruh fungsi dari *corporate* serta *directorate* seperti *Corporate Internal Audit*, *Corporate Risk Management*, *Corporate Secretary & Legal*, *Corporate Communication*, *Corporate Strategic Planning*, *Production Directorate*, *Finance Directorate*, *Marketing Directorate*, *HR, GA, IT Directorate*, serta melakukan koordinasi Integritas, Sinkronisasi dengan Instansi terkait.

2. *Productions Directorate*

*Productions Directorate* terdiri dari beberapa divisi yaitu *Plant 1, Plant 2, Plant 3&3A, Plant 4, Product & Quality Engineering, PPIC, Engineering, Dies Manufacturing, Cost Planning, Procurement Engineering, Quality Technology, Procurement & Purchasing Control & New Mode Center*.

Dimana tugas dari divisi–divisi tersebut adalah :

- a. Mengatur kegiatan kegiatan yang diperlukan bagi terselenggaranya proses produksi.
- b. Melakukan pengawasan terhadap pelaksanaan produksi.
- c. Melakukan perencanaan terhadap kapasitas produk, *layout* pabrik serta bahan baku produksi.

3. *Marketing Directorate*

*Marketing Directorate* terdiri dari beberapa divisi yaitu *Marketing Planning & Analysis, Technical Service, Parts, Sales, Logistic & Disribution & Honda Customer Care Center*. Dimana tugas dari divisi–divisi tersebut adalah :

- a. Mengkoordinasikan tim *marketing* untuk mempresentasikan profil perusahaan dan produk perusahaan kepada pelanggan.
- b. Melakukan pengawasan terhadap *staff marketing* atas target penjualan.
- c. Bertanggung jawab menyerahkan laporan berkala kepada Direktur Utama.

4. *HR, IT, GA Directorate*

*HR, IT, GA Directorate* terdiri dari beberapa divisi yaitu *Human Resourcess, General Affair, Information Technology & Sistem Intergration*. Dimana tugas dari divisi–divisi tersebut adalah :

- a. Melaksanakan sistem kepegawaian berdasarkan ketentuan dari peraturan perusahaan dan perundangan-undangan yang berlaku.

- b. Melakukan pengawasan terhadap tingkat kualitas kerja karyawan dengan berkoordinasi secara berkala kepada manajer atau koordinasi divisi.
- c. Melakukan pengawasan terhadap tingkat kedisiplinan karyawan dengan mendata setiap pelanggaran yang ditemukan.
- d. Mengadakan seleksi penerimaan karyawan baru apabila adanya permohonan dari manajer atau koordinator divisi setelah mendapatkan persetujuan dari manajer operasional.
- e. Melakukan tindakan khusus kepada karyawan yang melakukan pelanggaran terhadap tata tertib perusahaan atas laporan dari HR Administrasi.
- f. Membuat kebijakan terkait kepegawaian setelah berkoordinasi dengan Direktur Operasional. Melakukan pengendalian dan operasional *software* atau perangkat lunak.
- g. Menciptakan dan menerapkan sistem informasi terkait penerimaan data pengolahan data dan pengeluaran *output* data.
- h. Menangani permasalahan-permasalahan terkait perangkat lunak dan sistem yang telah terkomputerisasi.
- i. Melakukan koordinasi untuk perawatan gedung beserta perlengkapan inventaris kantor.

5. *Finance Directorate*

- a. Bertanggung jawab untuk mengatur rencana keuangan perusahaan sesuai kebijakan.
- b. Membiayai beberapa fungsi Produksi, *Marketing*, Operasi dan *finance* itu sendiri.
- c. Mengelola *Accounting* dan *Tax*.
- d. Menyelenggarakan kontrol pada bisnis yang berjalan.

6. *Accounting & Tax Division*

- a. Melakukan penghitungan dan pendataan perihal Akuntansi Manajemen Keuangan dan Sistem Informasi Keuangan.
- b. Menyelenggarakan proses keuangan yang akuntabel.
- c. Menyusun Sistem Informasi Akuntansi dan Keuangan yang *up to date*.
- d. Memenuhi semua kewajiban dan pertanggung jawaban keuangan perusahaan kepada pihak yang berwenang.

7. *Financial Planning & Analysis Division*

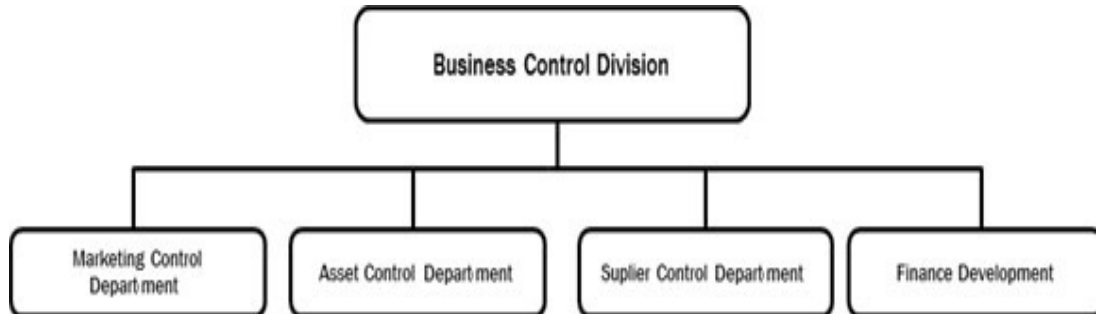
- a. Menyusun anggaran perusahaan sesuai dengan kebijakan yang ditetapkan.
- b. Menyajikan laporan manajemen yang bermakna bagi Direktur *Finance*.

8. *Treasury Division*

- a. Melakukan peramalan rencana pendanaan baik melalui penggunaan hutang dan ekuitas.
- b. Menyelenggarakan tata kelola modal perusahaan.
- c. Mempertahankan hubungan baik dengan komunitas investasi untuk tujuan penggalangan dana.
- d. Menciptakan strategi manajemen resiko yang menerapkan taktik hedging untuk melindungi perusahaan dari segala macam resiko.

9. *Business Control Division*

- a. Melaksanakan pengawasan terhadap *Supplier AHM*.
- b. Melaksanakan pengawasan terhadap *Asset AHM*.
- c. Melaksanakan pengawasan terhadap program penjualan yang dilakukan oleh *Sales Division*



Gambar IV.2 Struktur Organisasi Divisi *Business Control*  
 Sumber: PT Astra Honda Motor, 2015

10. *Financial Development Department*

- a. Membentuk menganalisa dan menginterpretasikan data statistik maupun informasi keuangan.
- b. Bertanggung jawab untuk mengevaluasi kinerja *staff* maupun manajer.
- c. Berpartisipasi dalam menyusun anggaran dan peramalan keuangan, institusi dan pengawasan terhadap perencanaan, pelaksanaan prosedur analisa dan pelaporan selisih.
- d. Bertanggung jawab terhadap perencanaan perpajakan sejalan dengan peraturan Ditjen Pajak terkait dengan peraturan pemerintah.

11. *Marketing Control Department*

- a. Mengidentifikasi kesalahan dan penyimpangan yang jika tidak terdeteksi memberikan dampak materil pada kewajaran penyajian kesesuaian pada program penjualan.
- b. Menganalisa berjalannya proses klaim program penjualan.
- c. Melakukan verifikasi berkas klaim yang diajukan oleh Main Dealer.
- d. Mengukur tingkat efisiensi, efektifitas dan produktifitas dari program penjualan.

12. *Supplier Control Department*

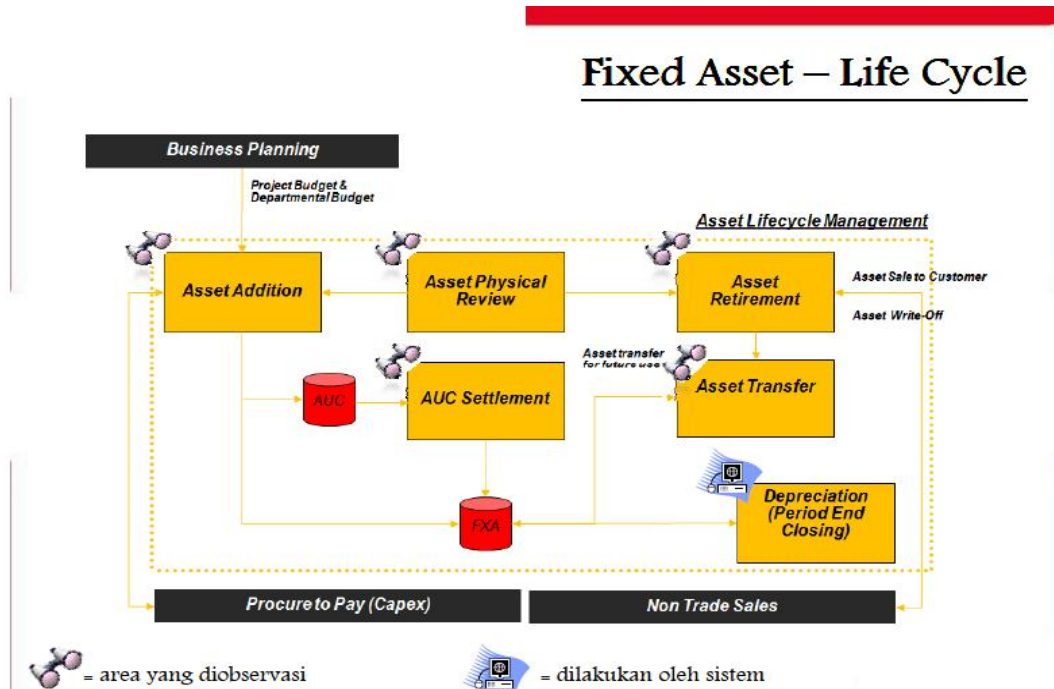
- a. Mengkordinasikan *supplier* pendukung dari AHM dalam penyuplaian barang pendukung.
- b. Melakukan kontrol harga dari setiap *supplier* yang mendukung produksi.

13. *Asset Control Department*

- a. Memonitoring *asset* yang digunakan tiap pabrik.
- b. Mengukur tingkat efisiensi, efektifitas dan produktifitas dari penggunaan *asset*.
- c. Melakukan perhitungan dan pendataan perihal *asset*.
- d. Melakukan pengawasan *life cycle* dari setiap *asset* yang dimiliki oleh AHM.

#### **4.5 Life Cycle Fixed Assets**

Berikut merupakan bentuk gambaran *Life Cycle* dari suatu *Fixed Assets* dari awal proses perencanaan pengadaan hingga aset tersebut tidak dapat digunakan.



Gambar IV.3 *Fixed Assets – Life Cycle*  
(Sumber: PT. Astra Honda Motor, 2015)

Dalam *Life Cycle Fixed Assets* yang ada di PT Astra Honda Motor dimulai dari penyediaan budget yang dianalisis dan dikelola oleh *Budget Department* sebelum masuk ke dalam proses *Asset Addition*. Untuk proses *Asset Addition* sendiri di dalamnya terdapat tiga tahap, yaitu *Investment Plan*, *Ordering* dan *Construction/Installation*. Apabila suatu aset yang sudah masuk dalam tahap *Construction/Installation* maka aset sudah dapat disebut sebagai AUC (*Assets Under Construction*) yang akan dimasukkan ke data master dalam SAP.

Selanjutnya AUC yang sudah memenuhi kriteria akan masuk proses *Settlement* menjadi *Fixed Assets*. Saat suatu aset dinyatakan sebagai *Fixed Assets* maka proses *Depreciation* akan berjalan otomatis oleh sistem. Proses *Retirement* akan dilakukan apabila umur manfaatnya suatu aset telah habis sesuai dengan perhitungan sistem dari proses *Depreciation*.

Aset yang sudah diproses *Retirement* akan masuk ke dalam proses *Asset Transfer* untuk dipindahkan ke tempat khusus yang selanjutnya akan dilelang ataupun didonasikan. *Fixed Assets* yang belum habis umur manfaatnya juga dapat masuk ke dalam proses *Asset Transfer* akibat dari kebutuhan perusahaan. Semisal dalam proses casting di plant 1 memiliki mesin *automixing* yang berlebih sedangkan di plant 2 terdapat mesin *automixing* yang sedang rusak maka mesin yang ada di plant 1 akan masuk ke dalam proses *Asset Transfer* untuk dipindahkan ke plant 2.

*Asset Physical Review* merupakan proses yang terkait dengan pengecekan keadaan fisik dari suatu aset baik itu AUC, *Fixed Assets* maupun DOFAR (*Declaration of Fixed Assets Retirement*).

#### **4.6 Prosedur Kontrol AUC**

Prosedur kontrol dalam *AUC (Asset Under Construction)* terbagi menjadi 3 prosedur, yaitu prosedur *order*, observasi AUC dan *settlement*. Ketiga hal tersebut tersebut tidak lepas dari kontrol yang dilakukan oleh *Asset Control Department*.

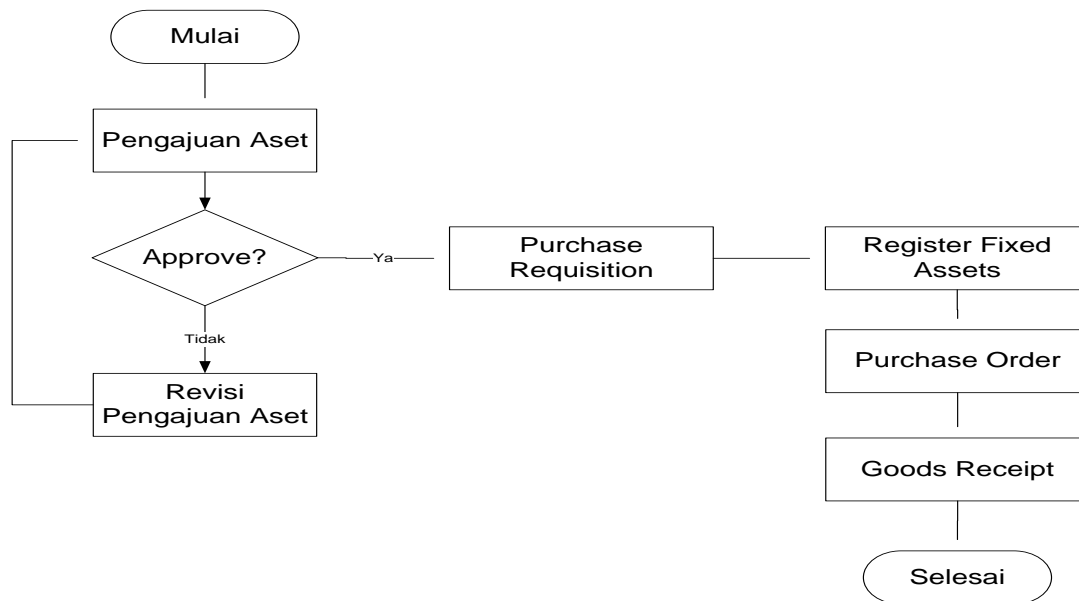
##### **4.6.1 Prosedur Order**

Pemesanan suatu aset tergantung pada kebutuhan perusahaan baik yang digunakan untuk kegiatan operasional maupun non-operasional. Semua pembelian aset tersebut harus menggunakan nama perusahaan sebagai pemilik dan mengikuti prosedur kebijakan yang telah disetujui oleh direksi. Berikut merupakan salah satu contoh list data pengajuan aset.

Asset description	Kategori	Plant Description	Cap.date	Acquis.val.	Asset Controller
AUTOMIXING DIE LUBE TS	Replacement	Plant Karawang, Gd C Lt 1 HPDC Line A	11/06/2014	1,823,553,390	Wahyu Tri Eka
PNEUMATIC DECK LEVELLER H100	Replacement	Plant Karawang, Gd B Lt 1 P. Plastik Combat 1	11/06/2014	1,823,553,390	Hendri Supomo
SPECTRO FLEX 6100	Replacement	Plant Karawang, Gd B Lt 1 P. Steel	25/02/2014	5,980,000,000	Sutarjo
CHAIN HOIST 250KG PAINT CED	Replacement	Plant Sunter, Sunter Tengah Plastic Injection Lantai 2	20/03/2013	153,354,747,142	Riska Indah
CHAIN HOIST 500KG PAINT ABS	Replacement	Part Centre, Tipar Cakung PQE Lantai 1	18/03/2014	12,288,971,126	Purnomo Andika
PERKINS-STAMFORD PRESS	Capacity Up	Plant Karawang, Temp Dawuan	29/10/2013	240,000,000	Jaya Kurniawan
WORKING DECK & LIFTER LPDC	Capacity Up	Plant Karawang, Gd C Lt 1 LPDC Line B	10/10/2013	539,000,000	Joko Wijajaya

Gambar IV.4 List Data Pengajuan Aset  
(Sumber: PT. Astra Honda Motor, 2015)

Berikut ini merupakan gambaran mengenai alur proses order di PT Astra Honda Motor.



Gambar IV.5 Flowchart Prosedur Order  
(Sumber: Pengolahan Data, 2015)



## 2. *Purchase Requisition*

Proses pembuatan daftar permintaan aset yang dipesan untuk masuk ke proses pembelian.

## 3. *Register Fixed Assets*

Proses input data aset yang akan dipesan ke dalam sistem SAP yang dilakukan sebagai penyimpanan data master. Data Aset tersebut meliputi:

- a. *Description Asset*
- b. *Cost Center*
- c. *Responsible Cost Center*
- d. *Plant*
- e. *Location*
- f. *Asset Controller*

## 4. *Purchase Order*

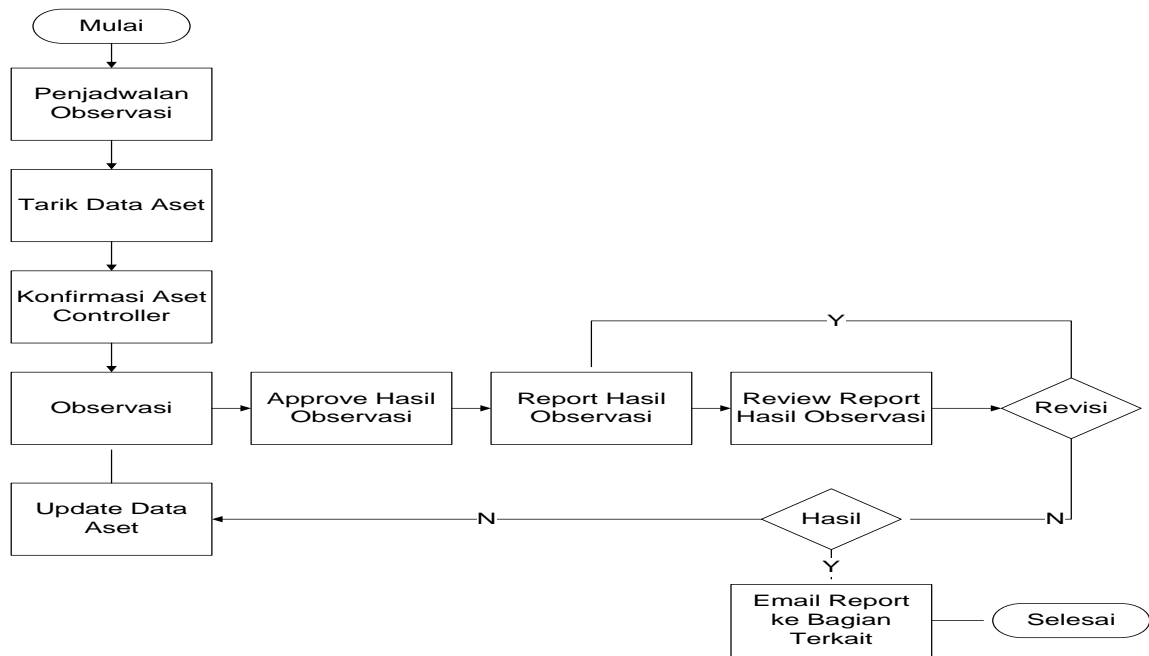
Proses pembuatan *Purchase Order* oleh bagian *Procurement* yang akan ditujukan pada vendor terkait untuk melakukan pembelian aset.

## 5. *Goods Receipt*

Proses penerimaan suatu aset yang telah dibeli melalui vendor dengan berkas berita acara serah terima yang menjadi tanggung jawab dari pihak *Procurement* PT Astra Honda Motor.

### **4.6.2 Prosedur Observasi AUC**

Berikut Prosedur dalam pelaksanaan observasi yang dilakukan oleh *Asset Control* terhadap AUC:



Gambar IV.7 *Flowchart* Prosedur Observasi

(Sumber: Pengolahan Data, 2015)

1. Melakukan penjadwalan observasi untuk masing-masing *plant*.
2. Menarik data aset yang akan diobservasi dari SAP. Data observasi AUC yang tercatat pada Master Data Asset dan dibuat memiliki dasar dan kriteria, diantaranya adalah:
  - a. Berdasarkan *Top 5 Major Project (New Plant, New Model and Capacity Up)*.
  - b. Berdasarkan *Top 5 Others Project (Maintenance and Replacement)*.
  - c. Top 5 berdasarkan nominal dari *project* terkait.
3. Melakukan konfirmasi via email maupun telepon ke *Asset Controller* terkait minimal satu minggu sebelum observasi dilaksanakan. Email berisikan jadwal beserta pelaksana lapangan dengan melampirkan data observasi AUC.
4. Mencetak dokumen observasi AUC untuk dibawa pada saat pelaksanaan

5. Melakukan observasi sesuai dengan jadwal yang telah ditetapkan bersama dengan *Asset Controller* terkait, hal-hal yang perlu dilakukan pengecekan pada saat observasi meliputi.
  - a. Kesesuaian data SAP dengan aktual:
    - Lokasi.
    - *Responsible Cost Center (Asset Controller)*.
    - Deskripsi.
  - b. Kesesuaian antara *project* awal dengan aktual.
  - c. Kegunaan (Proses Kerja).
  - d. *Progress* pengerjaan.
  - e. Kesiapan proses *settlement*.
  - f. Jika AUC berada di *supplier*, lakukan pengecekan melalui surat jalan dan dokumen pendukung berdasarkan AUC terkait.
6. Memberikan keterangan pada dokumen observasi AUC jika terjadi ketidaksesuaian antara data dengan kondisi aktual.
7. Setelah observasi dilakukan, *Observer* dan *Asset Controller* menandatangani dokumen observasi AUC yang menyatakan kesepakatan mengenai hasil observasi yang telah dilakukan.

Asset No.	Asset description	Capex	Year	Acquired	GR	Asset Category	Asset Location Code	Asset Location Description	Plant Code	Plant Description	Model	Keterangan	Settlement
20000007176	AUC for 10200000295-0-NEW PLANT 4	366,316,163,975	2013	05-03-13	100% EG	1700A	Temp Dawuan	1700	1700	Plant Karawang	New Model		Slip Settle
20000007251	AUC for 10200000400-0-NEW PARTS WAREHOUSE OFFICE	153,354,747,142	2013	20-03-13	100% GA	1700A	Temp Dawuan	1700	1700	Plant Karawang	New Model		Slip Settle
20000007897	AUC for 10500005528-0-ROAD LOAD SIMULATOR SYSTEM	12,288,971,126	2014	18-03-14	100% PQ	24F1	Tipe Cakung PCE Lantai 1	24F1	1700	Plant Karawang	New Model		Slip Settle
20000008039	AUC for 10500005705-0-CO2 SYSTEM	240,000,000	2013	29-10-13	100% QT	1700A	Temp Dawuan	1700	1700	Plant Karawang	New Model	Y Backup	belum siap
20000008129	AUC for 10500005729-0-WORKING DECK & LIFTER UPDC	539,000,000	2013	10-10-13	100% EG	4C1AW1318	Karawang Gd C U 1 PPCC Line B	4C1AW1318	1700	Plant Karawang	New Model	Y Backup	belum siap
20000008337	AUC for 10500005734-0-FIRE AIR FILLER MC	170,000,000	2014	04-09-14	100% EG	21B1	Peganganan Utra Assembling C & D U 1	21B1	1700	Plant Karawang	New Model		Cup Settle
20000008390	AUC for 10500005752-0-RETICOL SYSTEM	386,222,200	2014	04-02-14	100% EG	4C1YK2328	Karawang Gd C U 1 PPCC Line A	4C1YK2328	1700	Plant Karawang	New Model		Slip Settle
20000008435	AUC for 10500005775-0-CHAIN HOIST 250KG PAINT CED	35,100,000	2015	26-02-15	100% EG	4B1UL5660	Karawang Gd B U 1 P. Steel	4B1UL5660	1700	Plant Karawang	Capacity Up		Slip Settle
20000008436	AUC for 10500005776-0-CHAIN HOIST 1 T PAINTING CED	40,500,000	2015	26-02-15	100% EG	4B1UL5660	Karawang Gd B U 1 P. Steel	4B1UL5660	1700	Plant Karawang	Capacity Up		Slip Settle
20000008437	AUC for 10500005777-0-CHAIN HOIST 500KG PAINT ABS	40,500,000	2015	26-02-15	100% EG	4B1UC5955	Karawang Gd B U 1 P. Plastic Combat 1	4B1UC5955	1700	Plant Karawang	Capacity Up		Slip Settle
20000008438	AUC for 10500005778-0-CHAIN HOIST 500KG PAINT ABS	40,500,000	2015	26-02-15	100% EG	4B1UC5955	Karawang Gd B U 1 P. Plastic Combat 2	4B1UC5955	1700	Plant Karawang	Capacity Up		Slip Settle
20000008570	AUC for 10200000295-1-UMB BANGUNAN PLANT 4	2,788,566,600	2013	30-09-13	100% EG	1700A	Temp Dawuan	1700	1700	Plant Karawang	Capacity Up		Slip Settle
20000008621	AUC for 10800003614-0-SPECTRO FLEX 6100	66,885,000	2013	04-12-13	100% EG	14A1	AHTC Fasum Lantai 1	14A1	1700	Plant Karawang	Capacity Up	Ck 91/p	belum siap
20000008655	AUC for 10500005875-0-AUTOMATING DIE LUBE	72,500,000	2014	20-06-14	100% EG	4C1YK2328	Karawang Gd C U 1 PPCC Line A	4C1YK2328	1700	Plant Karawang	Capacity Up		Slip Settle
20000008656	AUC for 10500005875-0-AUTOMATING DIE LUBE	72,500,000	2014	20-06-14	100% EG	4C1YK2328	Karawang Gd C U 1 PPCC Line A	4C1YK2328	1700	Plant Karawang	Capacity Up		Slip Settle
20000008657	AUC for 10500005880-0-AUTOMATING DIE LUBE	72,500,000	2014	20-06-14	100% EG	4C1YK2328	Karawang Gd C U 1 PPCC Line A	4C1YK2328	1700	Plant Karawang	New Model		Slip Settle
20000008672	AUC for 10500005885-0-PNEUMATIC DECK LEVELER	92,379,470	2014	07-03-14	100% EG	14A1	AHTC Fasum Lantai 1	14A1	1700	Plant Karawang	New Model		Slip Settle
20000008673	AUC for 10500005886-0-PNEUMATIC DECK LEVELER	92,379,470	2014	07-03-14	100% EG	14A1	AHTC Fasum Lantai 1	14A1	1700	Plant Karawang	New Model		Slip Settle
20000008674	AUC for 10500005887-0-PNEUMATIC DECK LEVELER	92,379,470	2014	07-03-14	100% EG	14A1	AHTC Fasum Lantai 1	14A1	1700	Plant Karawang	New Model		Slip Settle
20000008675	AUC for 10500005888-0-PNEUMATIC DECK LEVELER	92,379,470	2014	07-03-14	100% EG	14A1	AHTC Fasum Lantai 1	14A1	1700	Plant Karawang	New Model		Slip Settle
20000008676	AUC for 10500005889-0-PNEUMATIC DECK LEVELER	92,379,470	2014	07-03-14	100% EG	14A1	AHTC Fasum Lantai 1	14A1	1700	Plant Karawang	New Model		Slip Settle
20000008677	AUC for 10500005890-0-PNEUMATIC DECK LEVELER	92,379,470	2014	07-03-14	100% EG	14A1	AHTC Fasum Lantai 1	14A1	1700	Plant Karawang	New Model		Slip Settle

Gambar IV.8 Dokumen Approval Observasi  
(Sumber: PT. Astra Honda Motor, 2015)

8. Membuat laporan hasil observasi AUC.
9. Laporan hasil observasi AUC ditinjau oleh *Asset Control Department Head* dan *Business Control Division Head*.
10. Mengirimkan laporan hasil observasi AUC yang telah ditinjau oleh *Asset Control Department Head* dan *Business Control Division Head* ke bagian terkait:
  - a. *Financial Planning Analysis Division* (untuk melihat kesesuaian *project* dengan aktual).
  - b. *Accounting & Tax Division* (untuk memperbaiki ketidaksesuaian antara data di sistem dan aktual).
  - c. *Asset Controller* (untuk menindaklanjuti temuan hasil observasi).
11. Melakukan monitoring atas temuan observasi untuk memastikan bahwa hasil observasi telah ditindaklanjuti oleh bagian terkait.

#### **4.6.3 Settlement AUC**

Settlement *AUC (Assets Under Construction)* merupakan pengakuan aset tetap yang berasal dari *AUC* pada saat selesai dan siap untuk dipergunakan sesuai dengan tujuan yang dimaksud. Berikut syarat suatu aset dapat dilakukan *settlement*.

1. Aset sudah mencapai *good receipt* hingga 100 %
2. Aset sudah digunakan sesuai fungsinya.

Berikut contoh data hasil observasi sebelum dilakukan *settlement*:

Observasi AUC (April 2015)												
No	Asset	Asset description	Cap.date	Acquis.val.	GR	Asset Control	Kode Lokasi	Deskripsi Lokasi	Foto	Project	Keterangan	Settle
1	20000000893	AUC for 105000006259-0-Bench Hardness tester	18/09/2014	100,000,000	100%	DME000	1700A	Temp Davuan		New Model		Siap Settle
9	20000001187	AUC for 107000003314-0-Panaboard	18/03/2015	12,398,400	100%	EC4000	1700A	Temp Davuan		New Model		Siap Settle
10	200000008428	AUC for 104000000573-0-OVERHEAD CRANE 7.5/5 T LPDC	09/10/2013	1,413,950,000	100%	FPR000	1700A	Temp Davuan		New Model	Lokasi P5	Siap Settle
11	200000008576	AUC for 102000001239-1-IMB BANGUNAN PLANT 4	30/09/2013	2,788,956,600	100%	FPR000	1700A	Temp Davuan		New Model	Dokumen IME ini berada di Legal	
12	20000000870	AUC for 104000000638-0-DOCKING LEVELLER	08/12/2014	125,000,000	100%	FPR000	1700A	Temp Davuan		New Model	Responsible dari aset ini dimiliki oleh Wahu	Siap Settle
13	200000009371	AUC for 104000000610-0-Frak	08/12/2014	975,000,000	100%	FPR000	1700A	Temo Davuan		New Model	Responsible dari aset ini dimiliki	Siao Settle

Gambar IV.9 Data Hasil Observasi  
(Sumber: PT. Astra Honda Motor, 2015)

#### 4.7 Klasifikasi AUC

PT Astra Honda Motor mengelompokan AUC menjadi 6 bagian, yaitu:

##### 1. Machinery

Mesin, peralatan yang terangkai menjadi satu yang berhubungan. Jika bagian-bagian yang membentuk rangkaian tersebut dapat berdiri sendiri maka secara pencatatan harus dicatat sebagai asset terpisah.



Gambar IV.10 Turning 2 - Turret L MC  
(Sumber: PT. Astra Honda Motor, 2015)

## 2. *Factory Equipment*

Semua peralatan yang mendukung kegiatan produksi. Jika peralatan tersebut terdiri dari beberapa bagian yang dapat berdiri sendiri maka secara pencatatan harus dicatat sebagai aset terpisah.



Gambar IV.11 Roller Annealing  
(Sumber: PT. Astra Honda Motor, 2015)

## 3. *Mould*

Semua peralatan cetak yang mendukung produksi dan digunakan untuk memproduksi barang cetakan dari material plastik.



Gambar IV.12 Injection Machine 300A  
(Sumber: PT. Astra Honda Motor, 2015)

#### 4. *Dies Press*

Semua peralatan cetak yang mendukung produksi dan digunakan untuk memproduksi barang cetakan dari material *steel* atau *coil*.



Gambar IV.13 Stamping Press MC 80T  
(Sumber: PT. Astra Honda Motor, 2015)

#### 5. *Jig*

Semua peralatan berupa Jig atau alat bantu yang terkait dengan proses *welding* dan *machining* yang digunakan untuk mendukung operasional.



Gambar IV.14 Jig Leak Tester Cr  
(Sumber: PT. Astra Honda Motor, 2015)

#### 4.8 Fungsi – Fungsi Terkait

Berikut adalah bagian-bagian yang terkait dalam proses control AUC :

##### 1. *Asset Controller*

Pihak pemilik aset yang bertanggung jawab atas *settlement* aset, pemeliharaan aset, pemindahan aset, optimalisasi penggunaan aset dan *retirement* aset. *Asset Controller* terbagi menjadi 8 bagian yang masing-masing tersebar di semua *plant*.

##### a. PEA (Proses *Engineering A*)

*Asset Controller* yang khusus menangani aset untuk proses *casting*, *polimer* dan *casting wheel*.

##### b. PEB (Proses *Engineering B*)

*Asset Controller* yang khusus menangani aset untuk proses *machining*.

##### c. PEC (Proses *Engineering C*)

*Asset Controller* yang khusus menangani aset untuk proses *welding* dan *press*.

##### d. PED (Proses *Engineering D*)

*Asset Controller* yang khusus menangani aset untuk proses *painting steel and palstic*.

##### e. PEE (Proses *Engineering E*)

*Asset Controller* yang khusus menangani aset untuk proses *assembling, engine unit* dan *assy Wheel*.

f. *FPR (Facility Provider)*

*Asset Controller* yang khusus menangani aset berupa pengembangan bangunan.

g. *QTM (Quality Technology Manufacture)*

*Asset Controller* yang khusus menangani aset yang memiliki fungsi untuk *jig* dan kebutuhan laboratorium teknologi mesin.

h. *GA (General Affair)*

*Asset Controller* yang khusus menangani aset berupa *office equipment* dan *safety equipment*.

2. *Asset User*

Pihak pengguna aset yang bertanggung jawab atas keberadaan, kondisi dan kelengkapan aset yang digunakan.

3. *Asset Accountant*

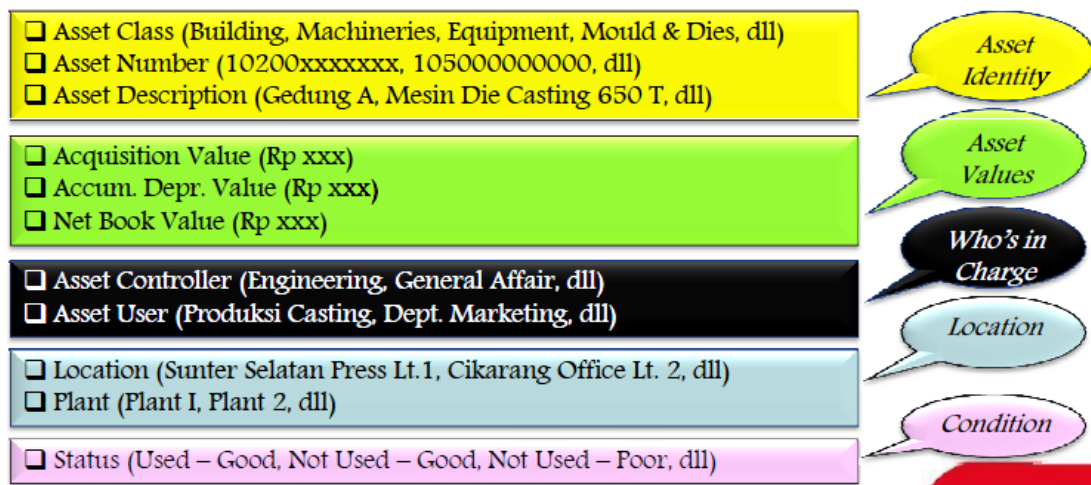
Pihak yang bertanggung jawab atas keakuratan pencatatan aset, pendespresiasian aset dan pemeliharaan master data aset.

4. *Asset Control*

Pihak yang bertanggung jawab untuk membuat, mensosialisasikan dan mengevaluasi kebijakan pengelolaan aset serta berkoordinasi dengan fungsi terkait lainnya untuk memastikan bahwa kebijakan telah berjalan.

#### **4.9 Fungsi SAP**

Perangkat Lunak SAP dalam proses kontrol AUC sendiri berperan dalam pembuatan master data yang diproses berdasarkan pengajuan aset yang sudah mendapat approval. Berikut gambaran database untuk master data AUC.



Gambar IV.15 Master Data  
(Sumber: PT. Astra Honda Motor, 2015)

#### 4.10 Flow Map Sistem yang Sedang Berjalan

Bagan alir *flow map* menunjukkan arus dari pekerjaan secara keseluruhan dari sistem. Bagan alir ini digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. *Flowmap* ini menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem. Adapun proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Proses dimulai dari *Asset Controller* yang bertugas melakukan pengajuan suatu aset yang dibutuhkan untuk menunjang kegiatan operasional perusahaan.
2. Aset yang diajukan tersebut akan dilakukan *approve* oleh *Department Head Asset Controller*. Pengajuan *approve* menggunakan berkas *hardcopy*. Aset akan dinilai oleh *Department Head Asset Controller* apakah benar-benar diperlukan atau tidak.
3. Apabila aset yang diajukan disetujui maka *Asset Controller* melakukan *Register Fixed Assets* ke SAP. *Asset Controller* harus membuat *Purchase Requisition* untuk diteruskan pada bagian *procurement* yang akan membuat *purchase order*. Jika aset tidak mendapatkan approval maka

*Asset Controller* harus merevisi pengajuan sesuai dengan saran dari *Department Head*.

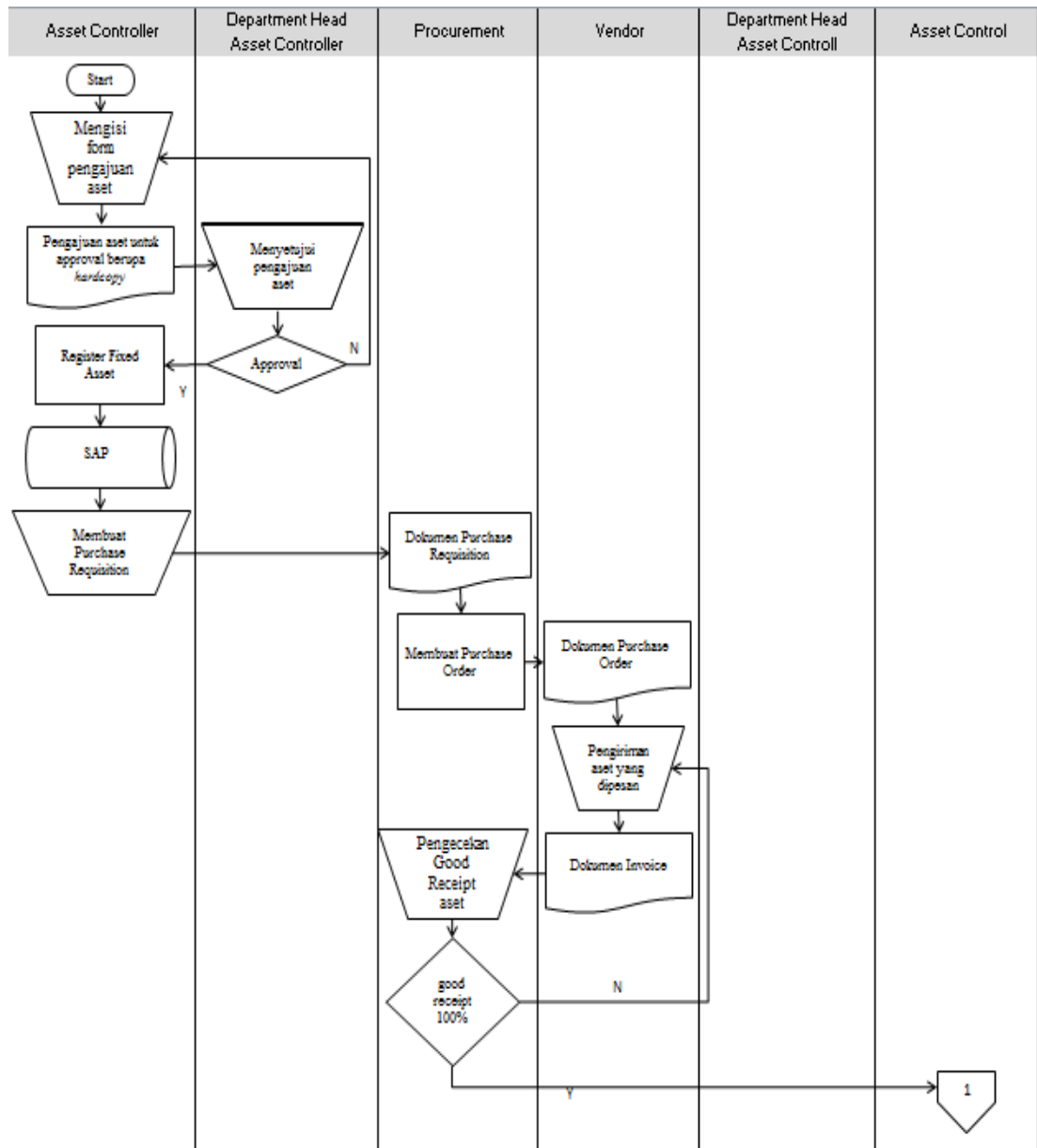
4. *Asset Control* akan mengontrol dan memonitor pengadaan aset tersebut dengan melakukan observasi untuk memastikan aset yang dibeli sesuai dengan pengajuan awal dan *good receipt* sudah mencapai 100%.
5. Hasil observasi akan dilaporkan kepada *Department Head Asset Control* berupa data excel mengenai bagaimana kondisi aset tersebut dilapangan apakah ada perbedaan atau tidak dengan pengajuan awal dan berapa jumlah aset yang siap untuk dilakukan proses *settlement*.

Observasi AUC (April 2015)

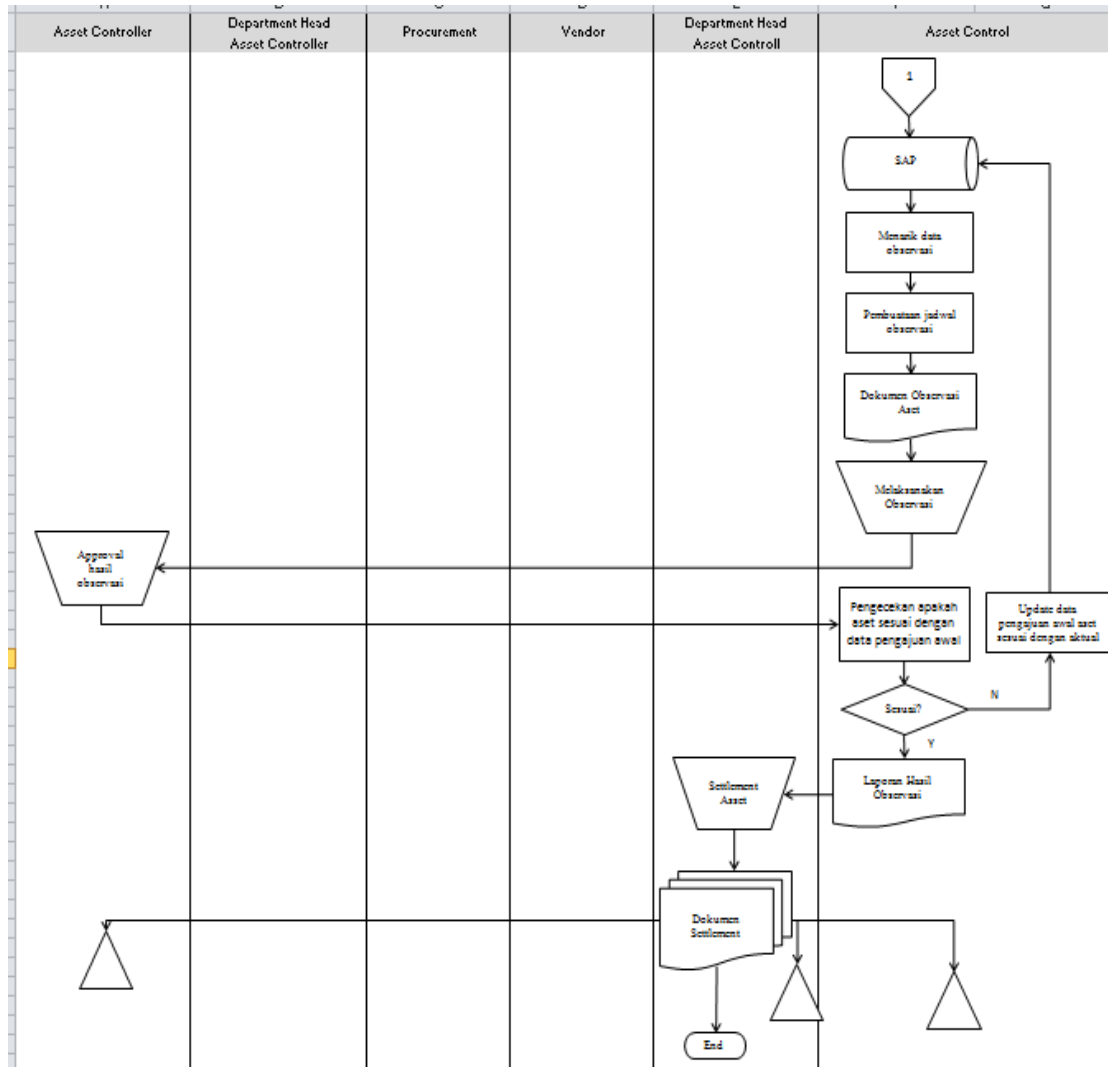
No	Asset	Asset description	Cap. di	Acquis. val	GR	Asset Control	Kode Lokasi	Deskripsi Lokasi	Foto	Project	Keterangan	Settle
1	200000003933	AUC for 105000006253-D-Bench Hardness tester	18-03-14	100.000.000	100%	DME000	1700A	Temp Davuan		New Model		Siap Settle
9	200000001877	AUC for 107000003314-D-Paraboard	18-03-15	12.398.400	100%	ED0000	1700A	Temp Davuan		New Model		Siap Settle
10	200000000420	AUC for 104000000573-D-OVERHEAD CRANE 7.5 T L POC	00-10-12	1.413.350.000	100%	FFR000	1700A	Temp Davuan		New Model	Lokasi PS	Siap Settle
12	200000003107	AUC for 104000000698-D-DOCKING LEVELLER	08-12-14	125.000.000	100%	FFR000	1700A	Temp Davuan		New Model	Responsible dari aset ini dimiliki oleh Wahjo	Siap Settle
13	200000009921	AUC for 104000000810-D-Tak	08-10-14	975.000.000	100%	FFR000	1700A	Temp Davuan		New Model	Responsible dari aset ini dimiliki oleh Ruan Sula	Siap Settle

Gambar IV.16 Laporan Hasil Observasi  
(Sumber: PT. Astra Honda Motor, 2015)

6. Setelah *Department Head Asset Control* menerima laporan hasil observasi, kemudian *Department Head Asset Control* melakukan *settlement* terhadap aset yang sudah siap dipakai. Informasi *settlement* aset dikirimkan ke *Asset Controller* dan *Asset Control* menggunakan email internal perusahaan.



Gambar IV.17 Flow Map Analisis Sistem yang Berjalan  
 Sumber: Hasil Pengamatan (2015)



Gambar IV.18 *Flow Map* Analisis Sistem yang Berjalan (Lanjutan)  
 Sumber: Hasil Pengamatan (2015)

#### 4.11 Use Case Diagram Sistem Saat Ini

*Use case diagram* menggambarkan siapa saja aktor yang melakukan prosedur dalam sistem serta fungsi-fungsi (proses) yang terlibat dalam transformasi pada sistem tersebut. *Use case diagram* sistem kontrol AUC saat ini pada PT Astra Honda Motor dapat dilihat pada Gambar IV.12.



evaluasi dan diterapkan perbaikan proses AUC berdasarkan dari instruksi *Departement Head*. Oleh karena itu perlu di buat *Procurement Control System* dimana dapat mempermudah dan mempercepat proses control AUC

#### 4.13 Penerapan Metode SAW (*Simple Addictive Weighting*)

Berdasarkan hasil diskusi dengan bagian *Asset Control Department*, aset yang harus di dahulukan dalam proses obaservasi dipertimbangkan berdasarkan tanggal penggunaan, tipe *project* dan harga dari aset tersebut. Kriteria tanggal penggunaan dibobotkan dengan nilai 0.4, tipe *project* dibobotkan dengan nilai 0.4 dan harga asset dibobotkan dengan nilai 0.2. Berikut contoh kasus untuk perhitungan metode SAW dalam prioritas aset:

Kriteria Tanggal Penggunaan:

No.	Capture Date	Nilai
1.	> 30 Hari	5
2.	≤ 30 Hari dan > 23 Hari	10
3.	≤ 23 Hari dan > 16 Hari	15
4.	≤ 16 Hari dan > 9 Hari	18
5.	≤ 9 Hari dan > 3 Hari	22
6.	≤ 3 Hari	30

Kriteria Project:

No.	Project	Nilai
1.	Replacement	35
2.	Maintainance	25
3.	Capacity Up	20
4.	New Plant	10
5.	New Model	10

Kriteria Harga Aset :

Nilai yang diambil sesuai dengan harga aktual aset tersebut.

Aset	Capture Date	Project	Harga
A	30 Desember 2017	New Model	10.000
B	01 Desember 2017	Replacement	5.000

Aset	Convert Nilai (Capture Date)	Convert Nilai (Project)	Harga
A	5	10	10.000
B	22	35	5.000

Normalisasi:

$$\begin{aligned}
 R11 &= 5/22 = 0.22727 & R12 &= 10/35 = 0.28571 & R13 &= 10000/10000 = 1 \\
 R21 &= 22/22 = 1 & R22 &= 35/35 = 1 & R23 &= 5000/10000 = 0.5
 \end{aligned}$$

Penyelesaian SAW:

$$\begin{aligned}
 A &= (0.22727 \times 0.4) + (0.28571 \times 0.4) + (1 \times 0.2) = 0.40519 \\
 B &= (1 \times 0.4) + (1 \times 0.4) + (0.5 \times 0.2) = 0.90000
 \end{aligned}$$

Maka aset yang diprioritaskan adalah aset B.

## BAB V

### ANALISIS DAN PEMBAHASAN

#### 5.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk mendapatkan informasi kebutuhan sistem, dengan mengetahui kebutuhan sistem diharapkan dapat diusulkan perbaikan terhadap sistem yang diamati. Berdasarkan hasil pengolahan data, maka diidentifikasi dibutuhkannya suatu aplikasi untuk menunjang proses pengadaan asset hingga settlement di PT Astra Honda Motor. Dalam merancang aplikasi ini, tahap pengidentifikasian kebutuhan sistem merupakan tahapan yang perlu dilakukan agar aplikasi yang dibuat dapat menjawab permasalahan yang ada.

Berikut adalah kebutuhan sistem yang diidentifikasi untuk *Procurement Control System (PCS)* :

Tabel V.1 Kebutuhan Sistem

<b>Kebutuhan Sistem</b>	
<i>Project Name</i>	<i>Procurement Control System</i>
<i>Business Need</i>	Terwujudnya sebuah sistem yang dapat mempermudah proses pengadaan hingga <i>settlement</i> aset.
<i>Business Requirement</i>	Memberikan sistem yang dapat membantu dalam melakukan kontrol terhadap proses pengadaan hingga settlement aset.
<i>Business Value</i>	<ol style="list-style-type: none"> <li>1. Mempercepat dalam proses persetujuan aset.</li> <li>2. Melakukan integrasi sistem pengadaan dengan proses observasi hingga <i>settlement</i> aset.</li> <li>3. Mempermudah <i>Asset Control</i> dalam melakukan observasi di lapangan.</li> <li>4. Mempermudah dalam monitor <i>settlement</i> aset.</li> </ol>
<i>Special Issues or Constrains</i>	Pembuatan sistem dilakukan selama enam bulan.

(Sumber: Analisis Data, 2017)

## 5.2 Analisis Kebutuhan Rinci Sistem

Analisis kebutuhan rinci sistem untuk *Procurement Control System* pada *Asset Control Department* di PT Astra Honda Motor dapat dilihat pada table V.2 berikut ini :

Tabel V.2 Kebutuhan Rinci Sistem

No	Kebutuhan Rinci Usulan	Uraian
1	Pengelolaan Data Master	Proses yang dapat dilakukan oleh <i>Asset Control</i> pada pengelolaan data master diantaranya : a. <i>Asset Control</i> dapat menambah data master apabila terdapat data baru. b. <i>Asset Control</i> dapat melakukan pencarian data dengan memasukkan kata kunci pada fasilitas cari yang ada pada sistem usulan. c. <i>Asset Control</i> dapat mengubah dan menghapus data master
2	Pengajuan Aset	Proses yang dapat dilakukan oleh <i>Asset Controller</i> untuk mengajukan pembelian suatu aset.
3	<i>Approval</i> Pengajuan Aset	Proses yang dapat dilakukan oleh <i>Asset Controller Dept. Head</i> untuk memberikan <i>approval</i> pada pengajuan aset.
4	Membuat <i>Purchase Order</i>	Proses yang dapat dilakukan oleh <i>Procurement</i> untuk membuat <i>purchase order</i> berdasarkan aset yang telah disetujui.
5	<i>Update Good Receipt</i>	Proses yang dapat dilakukan oleh <i>Procurement</i> untuk <i>update</i> data <i>good receipt</i> berdasarkan invoice yang diterima dari vendor.
6	Pelaksanaan Observasi	Proses yang dapat dilakukan oleh <i>Asset Control</i> untuk <i>update</i> data observasi aset di lapangan.

No	Kebutuhan Rinci Usulan	Uraian
7	<i>Approval</i> Hasil Observasi	Proses yang dapat dilakukan oleh <i>Asset Controller</i> untuk memberikan <i>approval</i> hasil observasi aset di lapangan.
8	Monitor Hasil <i>Approval</i> Observasi	Proses yang dapat dilakukan oleh <i>Asset Control</i> untuk <i>monitor</i> data hasil <i>approval</i> observasi apakah aset siap <i>settlement</i> atau tidak.
9	<i>Approval Settlement</i>	Proses yang dapat dilakukan oleh <i>Asset Control Dept. Head</i> untuk memberikan <i>approval</i> pada <i>settlement</i> aset.

(Sumber: Analisis Data, 2017)

### 5.3 Analisis Sistem Usulan

Tahap ini adalah tahap kedua pada metodologi *prototipe evolusioner* yaitu membuat *prototipe* untuk model sistem setelah sebelumnya melakukan tahap mengidentifikasi kebutuhan. Analisis proses sistem informasi menggunakan *tools* pemodelan sistem UML (*unified modeling language*) yang meliputi pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, *deployment diagram*, *class diagram* dan juga pembuatan kamus data. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun. Selanjutnya akan dilakukan perancangan *procurement control system* berdasarkan hasil analisis.

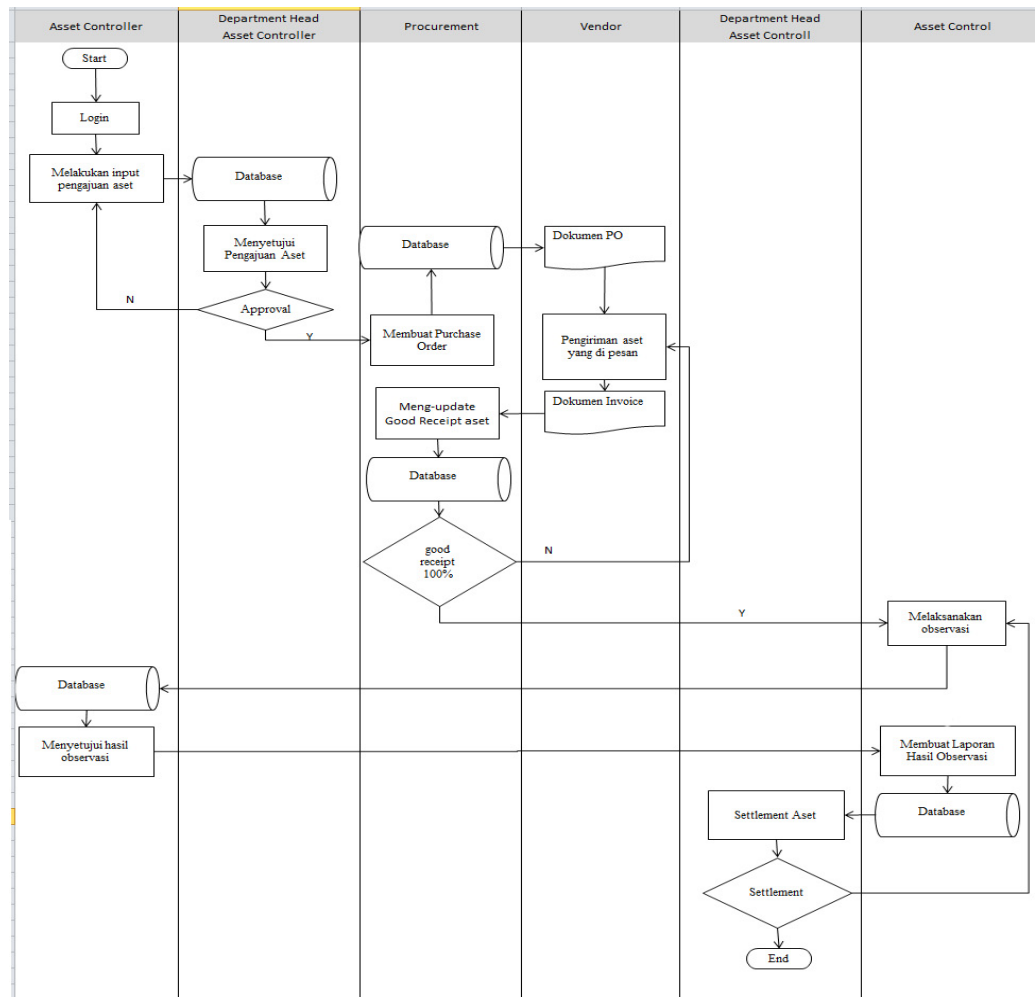
#### 5.3.1 Flowmap Sistem yang Diusulkan

Bagan alir *flow map* menunjukkan arus dari pekerjaan secara keseluruhan dari sistem. Bagan alir ini digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. *Flow map* ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem. Adapun proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Proses dimulai dari *Asset Controller* yang bertugas melakukan pengajuan suatu aset yang dibutuhkan untuk menunjang kegiatan

operasional perusahaan dengan cara input data aset yang diajukan ke *Procurement Control System (PCS)*.

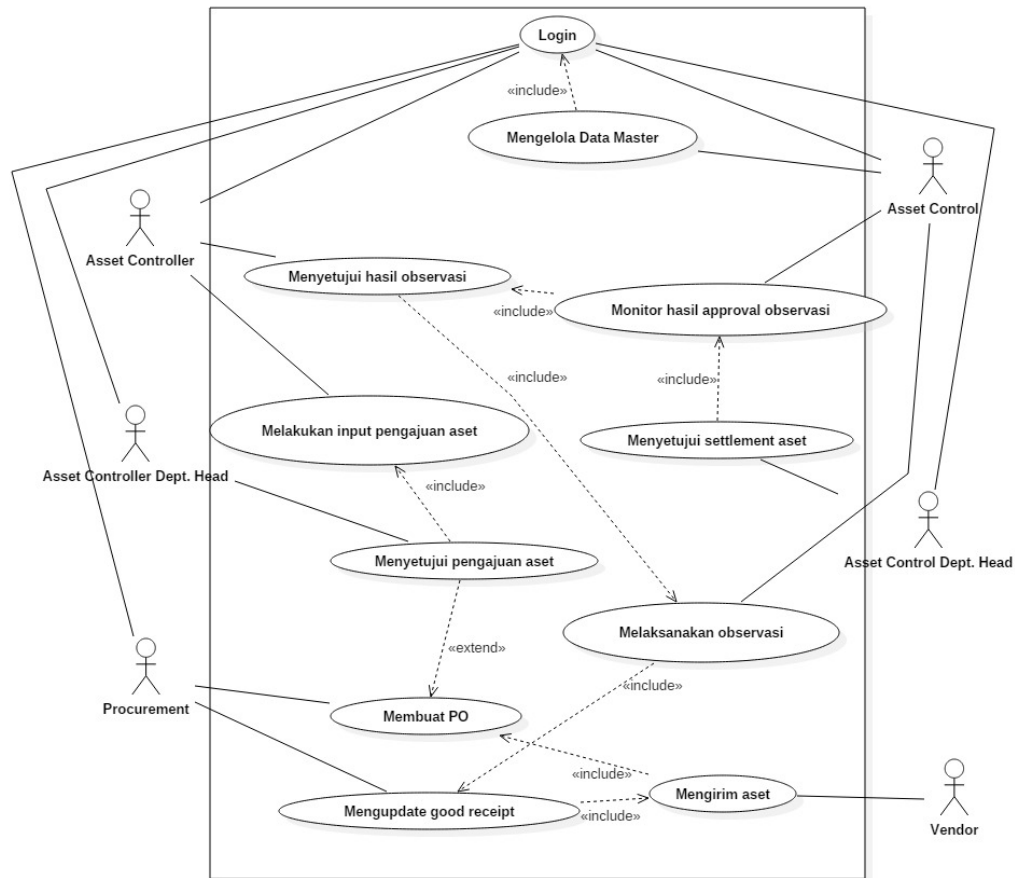
2. Aset yang di ajukan tersebut akan dilakukan approval oleh *Department Head Asset Controller* berupa pengajuan approval menggunakan sistem sehingga tanpa ada berkas *hardcopy*, aset akan dinilai oleh *Department Head Asset Controller* apakah benar-benar diperlukan atau tidak..
3. Apabila aset yang diajukan disetujui dapat dimonitor langsung melalui sistem sehingga bagian *procurement* dapat langsung membuat *purchase order* untuk aset yang telah disetujui tersebut.
4. *Asset Control* akan mengontrol dan memonitor pengadaan aset tersebut dengan melakukan observasi untuk memastikan aset yang dibeli sesuai dengan pengajuan awal dan *good receipt* sudah mencapai 100%. Apabila terdapat ketidaksesuaian data aset maka dapat langsung *update* data sesuai dengan aktual melalui sistem.
5. Hasil obervasi akan dilaporkan kepada *Department Head Asset Control* melalui sistem mengenai bagaimana kondisi aset tersebut dilapangan apakah ada perbedaan atau tidak dengan pengajuan awal dan berapa jumlah aset yang siap untuk dilakukan proses *settlement*.
6. *Department Head Asset Control* dapat langsung melihat laporan hasil observasi melalui sistem, kemudian *Department Head Asset Control* memberikan approval terhadap aset yang siap *settlement*.



Gambar V.1 Flowmap Sistem yang Diusulkan  
Sumber: Hasil Analisis Data (2017)

### 5.3.2 Use Case Diagram

*Use case diagram* berfungsi untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Dengan kata lain *use case diagram* juga digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah *use case diagram Procurement Control System (PCS)* yang diusulkan dapat dilihat pada Gambar V.1.



Gambar V.2 Use Case Diagram Procurement Control System  
Sumber: Hasil Analisis (2017)

Di dalam *Use Case Diagram*, sebuah aktor tidak hanya untuk mewakili orang tetapi juga untuk mewakili proses atau sistem lain yang berinteraksi dengan sistem. Oleh karena itu, tahapan selanjutnya yang perlu dilakukan yaitu mendefinisikan aktor-aktor apa saja yang terkait dalam sistem dan menjelaskan peran atau siapa aktor tersebut. Pendefinisian aktor pada *Use Case Diagram* Gambar V.2 dapat dilihat pada Tabel V.3.

Tabel V.3 Definisi Aktor *Use Case Diagram* yang Diusulkan

No	Aktor	Deskripsi
1.	<i>Asset Control (Admin)</i>	Orang yang bertugas untuk update data hasil observasi, untuk selanjutnya hasil tersebut dapat di monitor dan approval oleh <i>Asset Control Dept. Head</i> dengan menggunakan <i>PCS</i> . Hak akses yang dimiliki adalah pengelolaan data master dan <i>update</i> observasi.
2.	<i>Asset Controller (User)</i>	Orang yang bertugas input data pengajuan aset, untuk selanjutnya data tersebut akan melalui approval oleh <i>Asset Controller Dept. Head</i> dengan menggunakan <i>PCS</i> . Hak akses yang dimiliki adalah penginputan pengajuan data aset, approval hasil observasi dan melihat <i>settlement</i> aset pada sistem.
3.	<i>Asset Control Dept. Head</i>	Orang yang melakukan approval data hasil observasi untuk proses <i>settlement</i> dengan menggunakan <i>PCS</i> . Hak akses yang dimiliki adalah <i>approval</i> data hasil observasi pada sistem.
4.	<i>Asset Controller Dept. Head</i>	Orang yang melakukan approval data pengajuan aset, untuk selanjutnya proses pembuatan PO oleh <i>Procurement</i> dengan menggunakan <i>PCS</i> . Hak akses yang dimiliki adalah approval data pengajuan aset.
5.	<i>Procurement</i>	Orang yang melakukan pembuatan PO dan update data <i>good receipt</i> dengan menggunakan <i>PCS</i> . Hak akses yang dimiliki adalah pembuatan PO, <i>update</i> data <i>good receipt</i> dan melihat data aset.
6.	<i>Vendor</i>	Orang yang bertugas mengirimkan aset sesuai dengan PO.

Sumber: Hasil Analisis (2017)

Setelah mendefinisikan aktor yang terkait dalam *use case Diagram*, tahap selanjutnya yaitu mendefinisikan *use case* itu sendiri. *Use case* ini merupakan suatu fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor yang dinyatakan dengan menggunakan kata kerja di awal frase nama *use case*. Pendefinisian *use case* pada *use case diagram* Gambar V.2 dapat dilihat pada Tabel V.4.

Tabel V.4 *Use Case Description Login*

<b>Nama Use Case</b>	<b>Login</b>
Deskripsi Use Case	Use case ini menggambarkan proses awal untuk masuk ke dalam aplikasi dengan mengisi <i>username</i> dan <i>password</i> .
Aktor	<i>Asset Controller, Asset Control, Asset Controller Dept. Head, Asset Control Dept. Head</i> dan <i>Procurement</i>
Normal Flow Events:	<ol style="list-style-type: none"> <li>1. <i>User</i> membuka <i>form login</i></li> <li>2. <i>User</i> mengisi <i>username</i> dan <i>password</i>.</li> <li>3. Jika <i>username</i> dan <i>password</i> salah maka akan muncul pesan kesalahan (tidak valid).</li> <li>4. Apabila <i>username</i> dan <i>password</i> benar maka <i>user</i> akan masuk ke dalam <i>Procurement Control System</i>.</li> </ol>

Sumber: Hasil Analisis Data (2017)

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

<b>Nama Use Case</b>	<b>Mengelola Data Master</b>
Deskripsi Use Case	Use case ini menggambarkan proses generalisasi pengelolaan data master.
Aktor	<i>Asset Control.</i>
Normal Flow Events:	<ol style="list-style-type: none"> <li>1. <i>User login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu master data.</li> <li>3. Sistem menampilkan menu data master.</li> <li>4. <i>User</i> melakukan proses tambah, edit, hapus data master.</li> </ol>

Tabel V.6 *Use Case Description Melakukan Input Pengajuan Aset*

<b>Nama Use Case</b>	<b>Melakukan Input Pengajuan Aset</b>
Deskripsi Use Case	Use case ini menggambarkan proses yang meliputi menginput data aset yang akan diajukan.
Aktor	<i>Asset Controller.</i>

<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu order dan submenu submission of asset.</li> <li>3. Sistem menampilkan menu data pengajuan aset.</li> <li>4. <i>User</i> memilih add data.</li> <li>5. Sistem menampilkan <i>form</i> pengajuan aset.</li> <li>6. <i>User</i> mengisi <i>form</i> pengajuan aset dan pilih save untuk menyimpan dan back untuk kembali ke menu data pengajuan aset.</li> </ol>
----------------------------	--

Tabel V.7 *Use Case Description* Menyetujui Pengajuan Aset

<b>Nama Use Case</b>	<b>Menyetujui Pengajuan Aset</b>
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses yang meliputi <i>approve</i> data aset yang akan diajukan.
Aktor	<i>Asset Controller Dept. Head.</i>
<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu order dan submenu submission of asset.</li> <li>3. Sistem menampilkan menu data pengajuan aset.</li> <li>4. <i>User</i> memilih wait approve.</li> <li>5. Sistem menampilkan data detail pengajuan aset.</li> <li>6. <i>User</i> memilih approve untuk menyetujui aset, reject untuk menolak aset dan back untuk kembali ke menu data pengajuan aset.</li> </ol>

Tabel V.8 *Use Case Description* Membuat PO

<b>Nama Use Case</b>	<b>Membuat PO</b>
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses yang meliputi pembuatan data PO dan cetak PO.
Aktor	<i>Procurement.</i>

<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu order dan submenu purchase order.</li> <li>3. Sistem menampilkan menu data purchase order.</li> <li>4. <i>User</i> memilih add data.</li> <li>5. Sistem menampilkan data detail pengajuan aset.</li> <li>6. <i>User</i> memilih approve untuk menyetujui aset, reject untuk menolak aset dan back untuk kembali ke menu data pengajuan aset.</li> </ol>
----------------------------	--

Tabel V.9 *Use Case Description* Mengirim Aset

<b>Nama Use Case</b>	<b>Mengirim Aset</b>
Deskripsi Use Case	<i>Use case</i> ini menggambarkan pengiriman aset dari vendor sesuai dengan PO.
Aktor	Vendor.
<i>Normal Flow Events:</i>	1. Vendor melakukan pengiriman aset sesuai dengan PO.

Tabel V.10 *Use Case Description* Update Data Good Receipt

<b>Nama Use Case</b>	<b>Update Data Good Receipt</b>
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses <i>update good receipt</i> aset.
Aktor	<i>Procurement</i> .
<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu order dan submenu good receipt.</li> <li>3. Sistem menampilkan menu data good receipt.</li> <li>4. <i>User</i> memilih update.</li> <li>5. Sistem menampilkan data detail good receipt.</li> <li>6. <i>User update</i> good receipt dan pilih save untuk menyimpan dan back untuk kembali ke menu data good receipt.</li> </ol>

Tabel V.11 *Use Case Description* Melaksanakan Observasi

<b>Nama Use Case</b>	<b>Melaksanakan Observasi.</b>
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses yang meliputi <i>update</i> data aset yang nantinya akan menjadi data hasil observasi.
Aktor	<i>Asset Control.</i>
<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu observation.</li> <li>3. Sistem menampilkan menu data observasi.</li> <li>4. <i>User</i> memilih observe.</li> <li>5. Sistem menampilkan data detail observasi.</li> <li>6. <i>User update</i> data observasi dan pilih update untuk memperbarui data observasi dan back untuk kembali ke menu data observasi.</li> </ol>

Tabel V.12 *Use Case Description* Menyetujui Hasil Observasi

<b>Nama Use Case</b>	<b>Menyetujui Hasil Observasi</b>
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses yang meliputi pemberian approval data hasil observasi dilapangan.
Aktor	<i>Asset Controller.</i>
<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu observation.</li> <li>3. Sistem menampilkan menu data observasi.</li> <li>4. <i>User</i> memilih wait approve.</li> <li>5. Sistem menampilkan data detail observasi.</li> <li>6. <i>User</i> memilih approve untuk menyetujui aset, reject untuk menolak aset dan back untuk kembali ke menu data pengajuan aset.</li> </ol>

Tabel V.13 *Use Case Description* Monitor Approval Hasil Observasi

<b>Nama Use Case</b>	<b>Monitor Approval Hasil Observasi</b>
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses yang meliputi <i>monitor</i> data approval hasil observasi.

Aktor	<i>Asset Control.</i>
<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu settlement.</li> <li>3. Sistem menampilkan menu data aset siap settle.</li> <li>4. <i>User</i> memilih re-observe.</li> <li>5. Sistem menampilkan data re-observe</li> <li>6. <i>User</i> memilih observe.</li> <li>7. <i>User update</i> data re-observe dan pilih update untuk memperbarui data re-observe dan back untuk kembali ke menu data observasi.</li> </ol>

Tabel V.14 Use Case Description Menyetujui Settlement

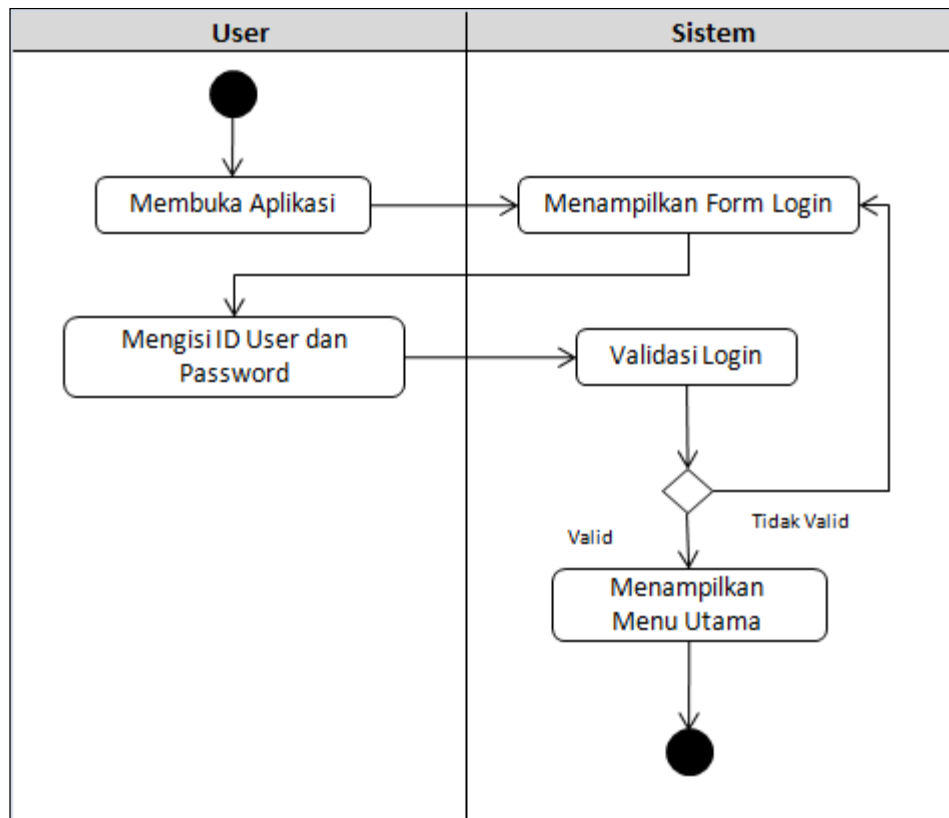
<b>Nama <i>Use Case</i></b>	<b>Menyetujui <i>Settlement</i></b>
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses yang meliputi pemberian approval <i>settlement</i> aset.
Aktor	<i>Asset Control Dept. Head.</i>
<i>Normal Flow Events:</i>	<ol style="list-style-type: none"> <li>1. <i>User Login</i> dan masuk menu utama.</li> <li>2. <i>User</i> memilih menu settlement.</li> <li>3. Sistem menampilkan menu data aset siap settle.</li> <li>4. <i>User</i> memilih wait approve.</li> <li>5. Sistem menampilkan data detail aset siap settle.</li> <li>6. <i>User</i> memilih approve untuk menyetujui settlement, rej-observe untuk mengulang observasi dan back untuk kembali ke menu data aset siap settle.</li> </ol>

### 5.3.3 Activity Diagram

Penggunaan *activity diagram* bertujuan untuk menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, mulai dari titik awal, melalui kondisi sampai titik akhir. Berikut adalah *activity diagram procurement control system* yang diusulkan:

### 1. Activity Diagram Login

Diagram pada Gambar V.3. di bawah ini merupakan *activity diagram* login yang dilakukan oleh user. Pada diagram tersebut dijelaskan bagaimana alur aktivitas login yang harus dilakukan user.

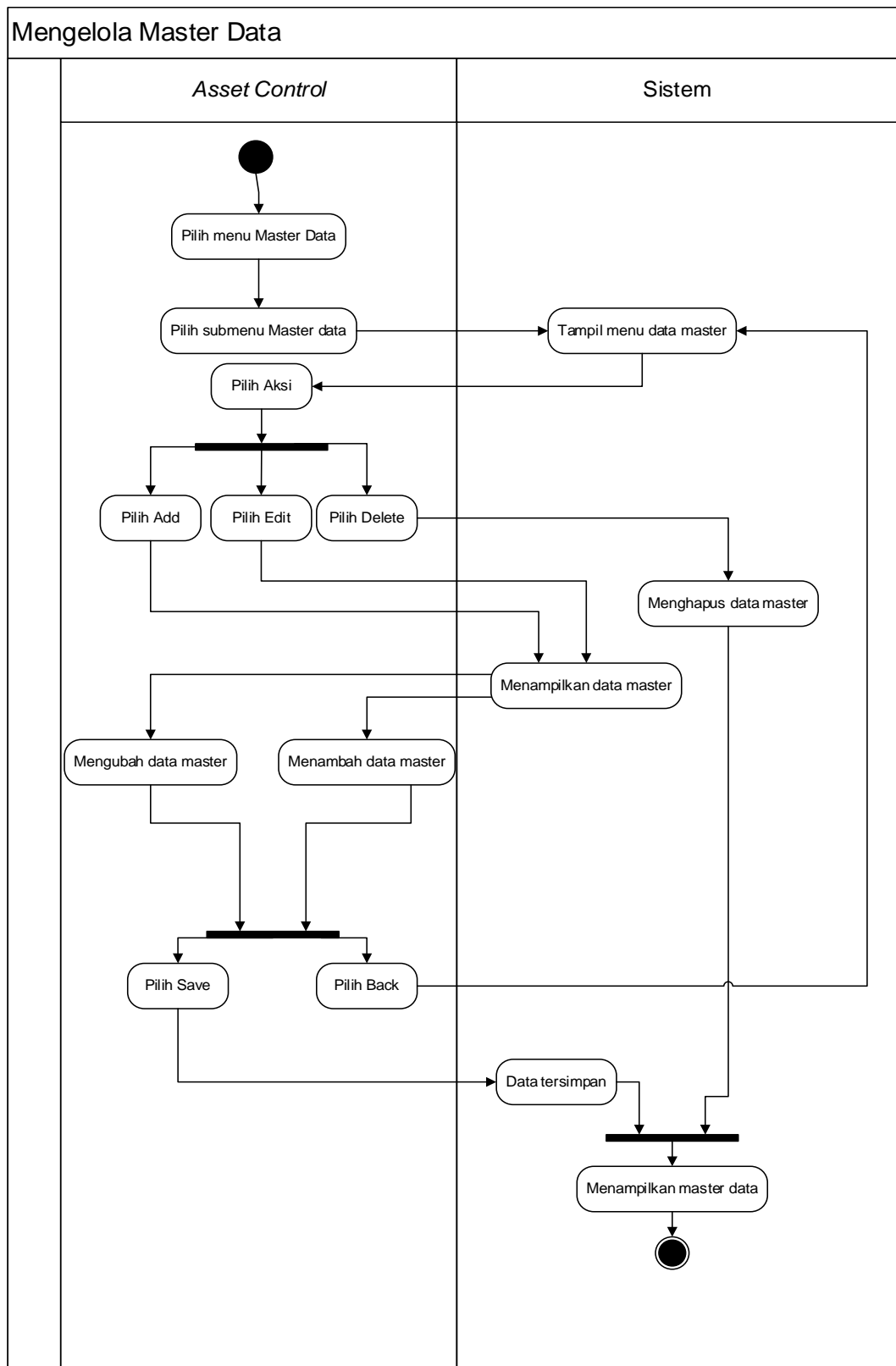


Gambar V.3. Activity Diagram Login

Sumber: Hasil Analisis (2017)

### 2. Activity Diagram Mengelola Master Data

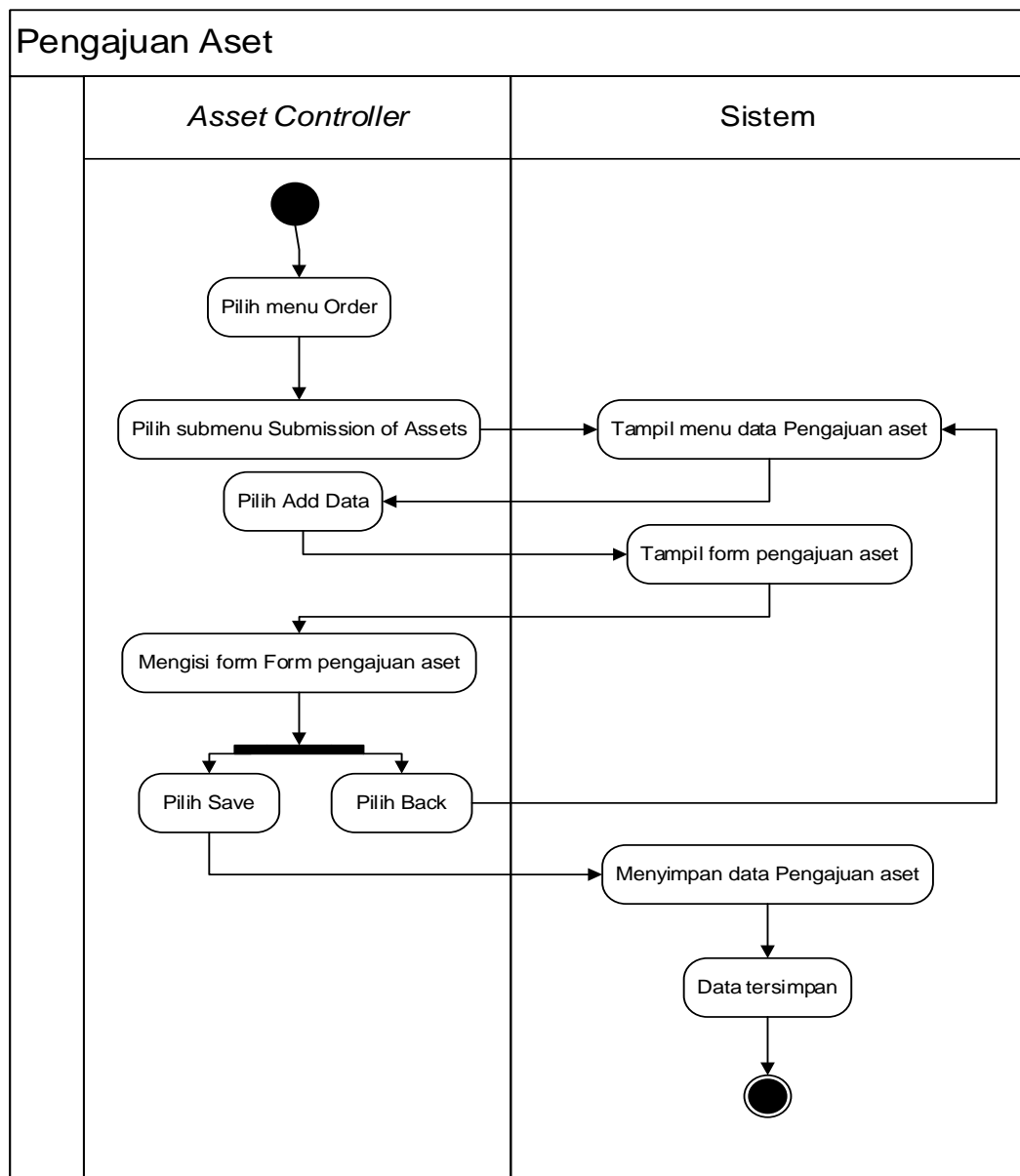
Diagram pada Gambar V.4. di bawah ini merupakan *activity diagram* pengelolaan master data. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan untuk mengelola master data.



Gambar V.4. Activity Diagram Mengelola Master Data  
Sumber: Hasil Analisis (2017)

### 3. Activity Diagram Pengajuan Aset

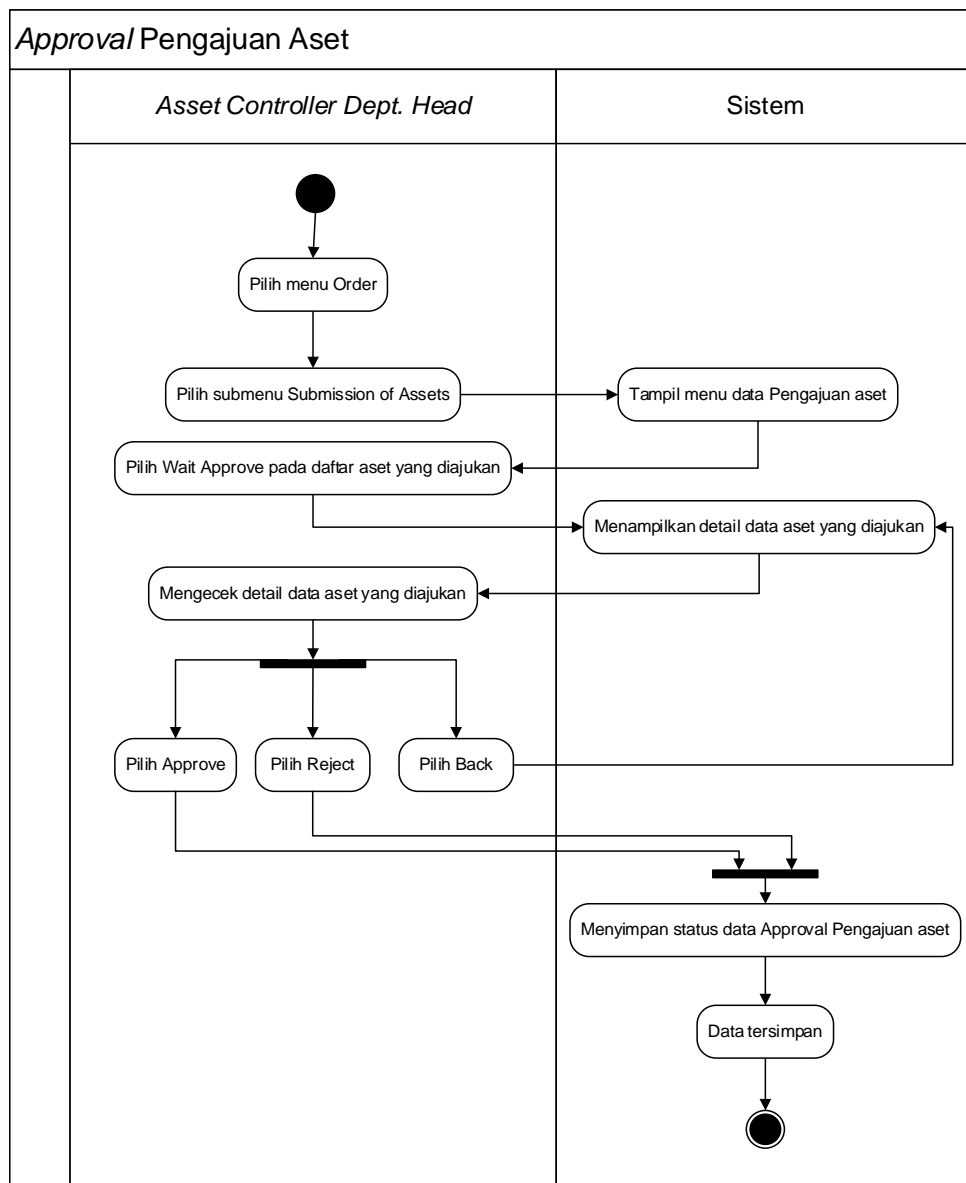
Diagram pada Gambar V.5. di bawah ini merupakan *activity diagram* pengajuan aset yang dilakukan oleh *Asset Controller*. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam pengajuan aset. Serta penjelasan terhadap fungsi-fungsi yang dapat dilakukan pada proses pengajuan aset.



Gambar V.5. Activity Diagram Pengajuan Aset  
Sumber: Hasil Analisis (2017)

#### 4. Activity Diagram Approval Pengajuan Aset

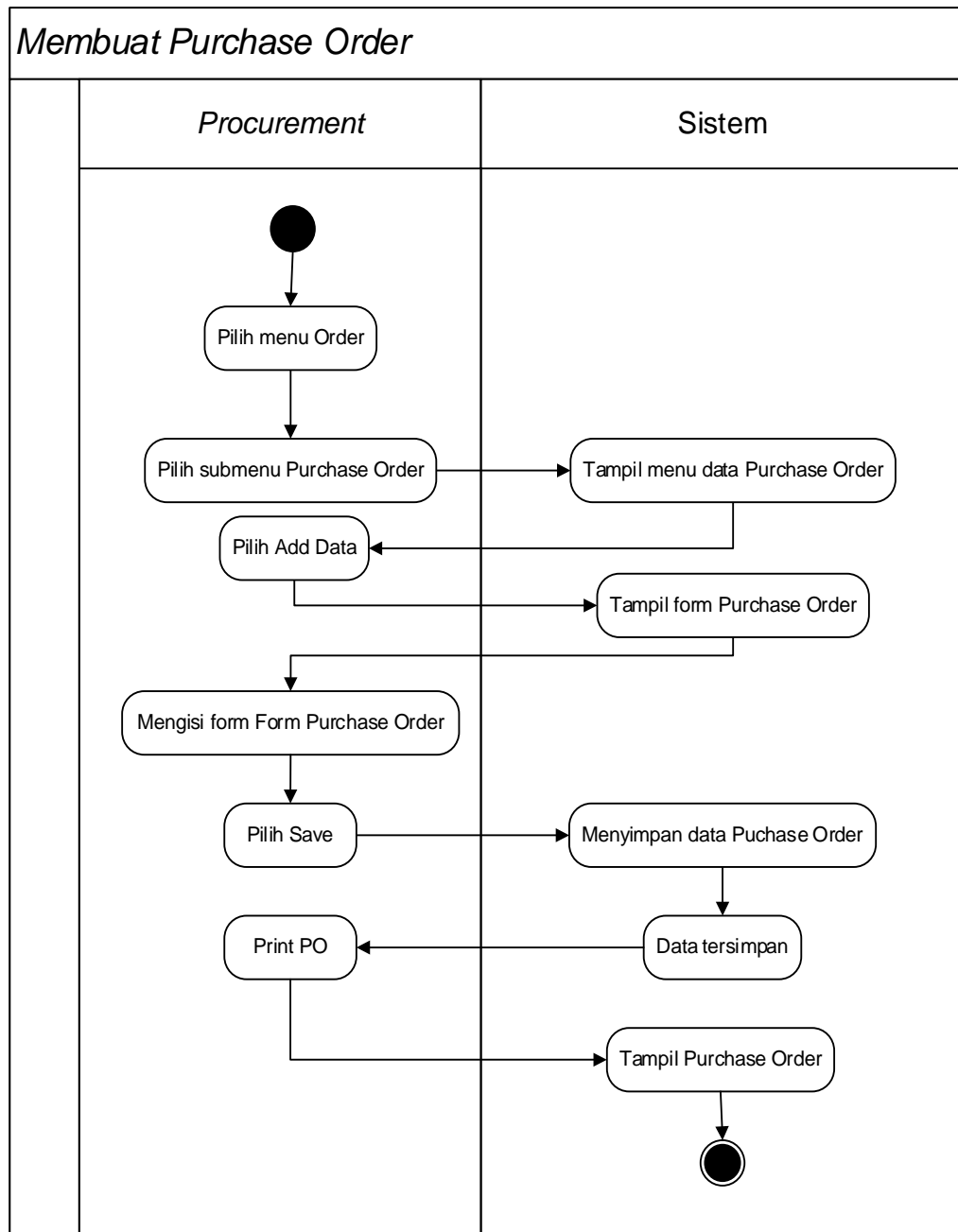
Diagram pada Gambar V.6. di bawah ini merupakan *activity diagram* pemberian approval pengajuan aset yang bisa dilakukan oleh *Asset Controller Dept. Head*. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan untuk monitor dan memberikan approval pengajuan aset.



Gambar V.6. Activity Diagram Approval Pengajuan Aset  
Sumber: Hasil Analisis (2017)

### 5. Activity Diagram Membuat Purchase Order

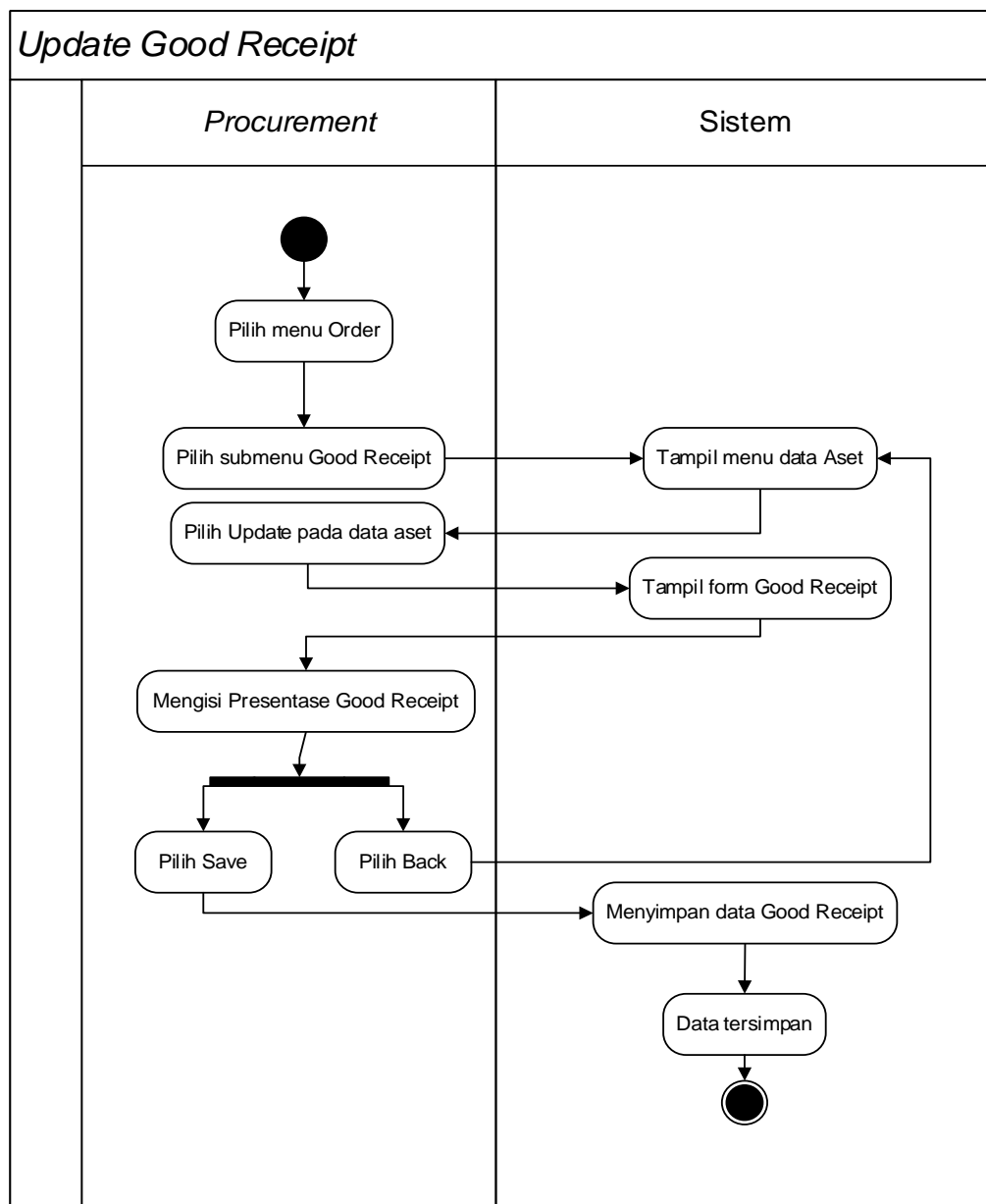
Diagram pada Gambar V.7. di bawah ini merupakan *activity diagram* membuat *purchase order* yang dilakukan oleh *procurement*.



Gambar V.7. Activity Diagram Membuat Purchase Order  
Sumber: Hasil Analisis (2017)

### 6. Activity Diagram Update Good Receipt

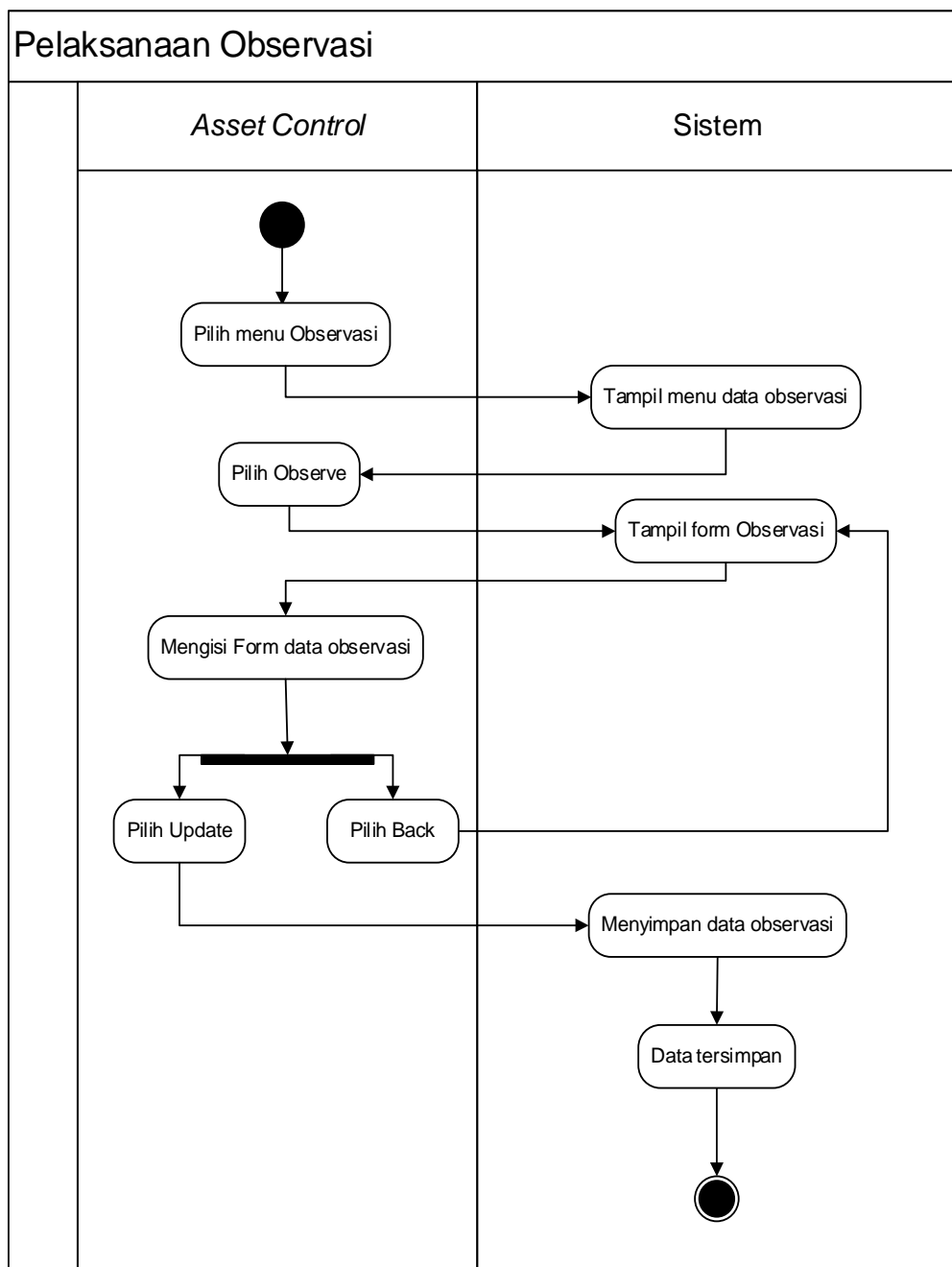
Diagram pada Gambar V.8. di bawah ini merupakan *activity diagram update good receipt* yang dilakukan oleh *Procurement*. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam *update good receipt*.



Gambar V.8. Activity Diagram Update Good Receipt  
Sumber: Hasil Analisis (2017)

### 7. Activity Diagram Pelaksanaan Observasi

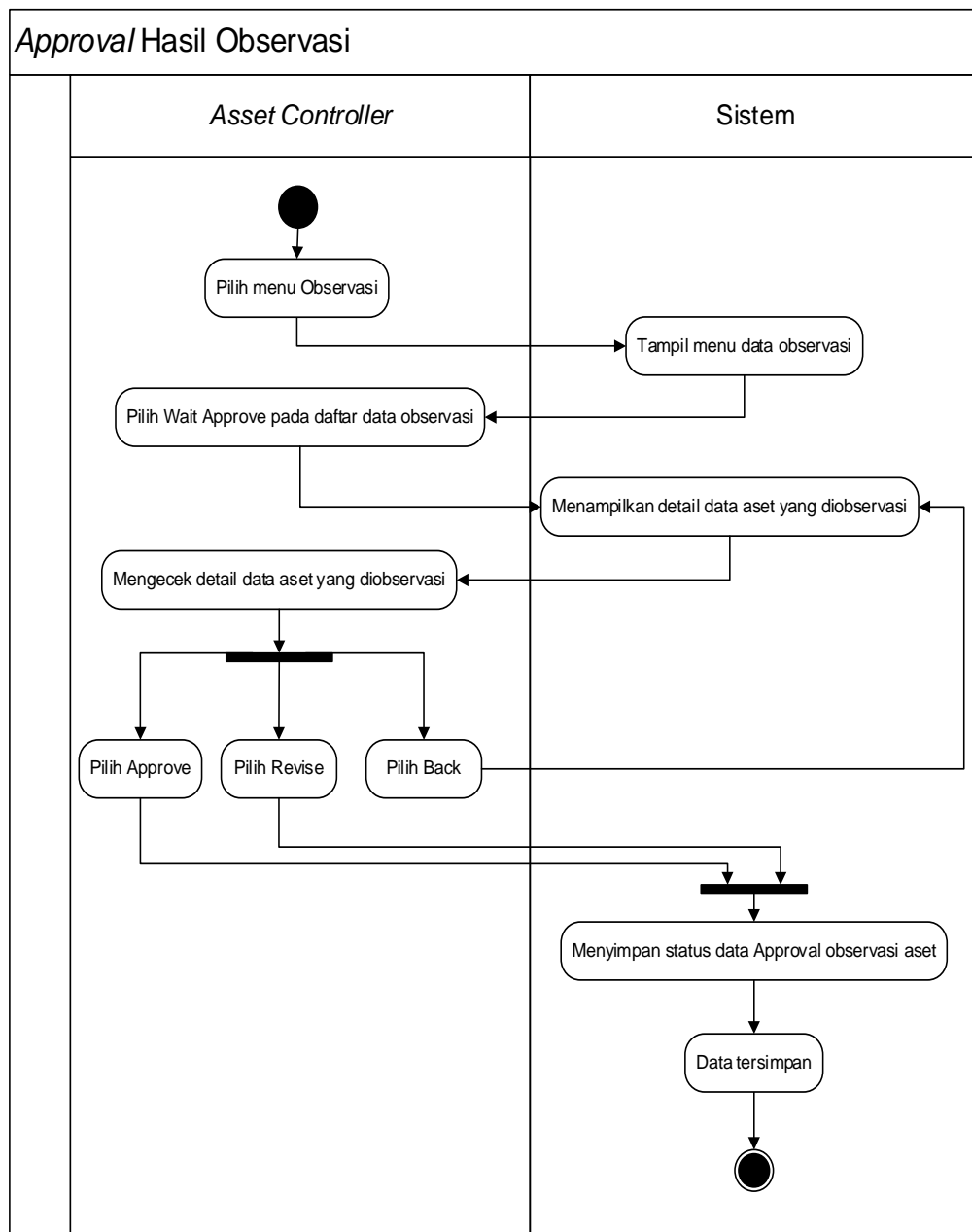
Diagram pada Gambar V.9. di bawah ini merupakan *activity diagram update* data observasi yang dilakukan oleh *Asset Control*. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam *update* data observasi.



Gambar V.9. Activity Diagram Pelaksanaan Observasi  
Sumber: Hasil Analisis (2017)

### 8. Activity Diagram Approval Hasil Observasi

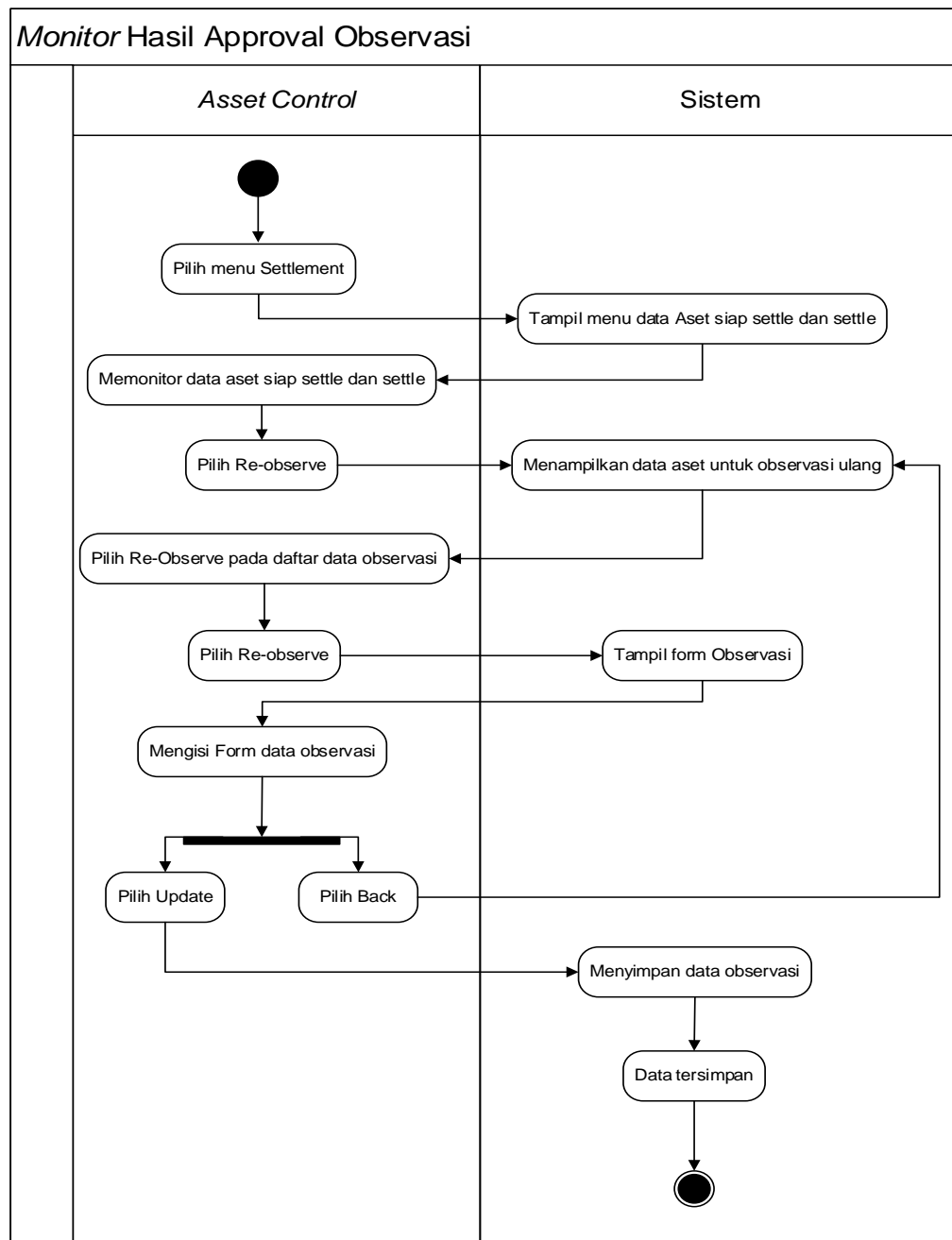
Diagram pada Gambar V.10. di bawah ini merupakan *activity diagram* dalam memberikan approval terhadap hasil observasi yang dilakukan oleh *Asset Controller*. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam pemberian approval hasil observasi.



Gambar V.10. Activity Diagram Approval Hasil Observasi  
Sumber: Hasil Analisis (2017)

### 9. Activity Diagram Monitor Hasil Approval Observasi

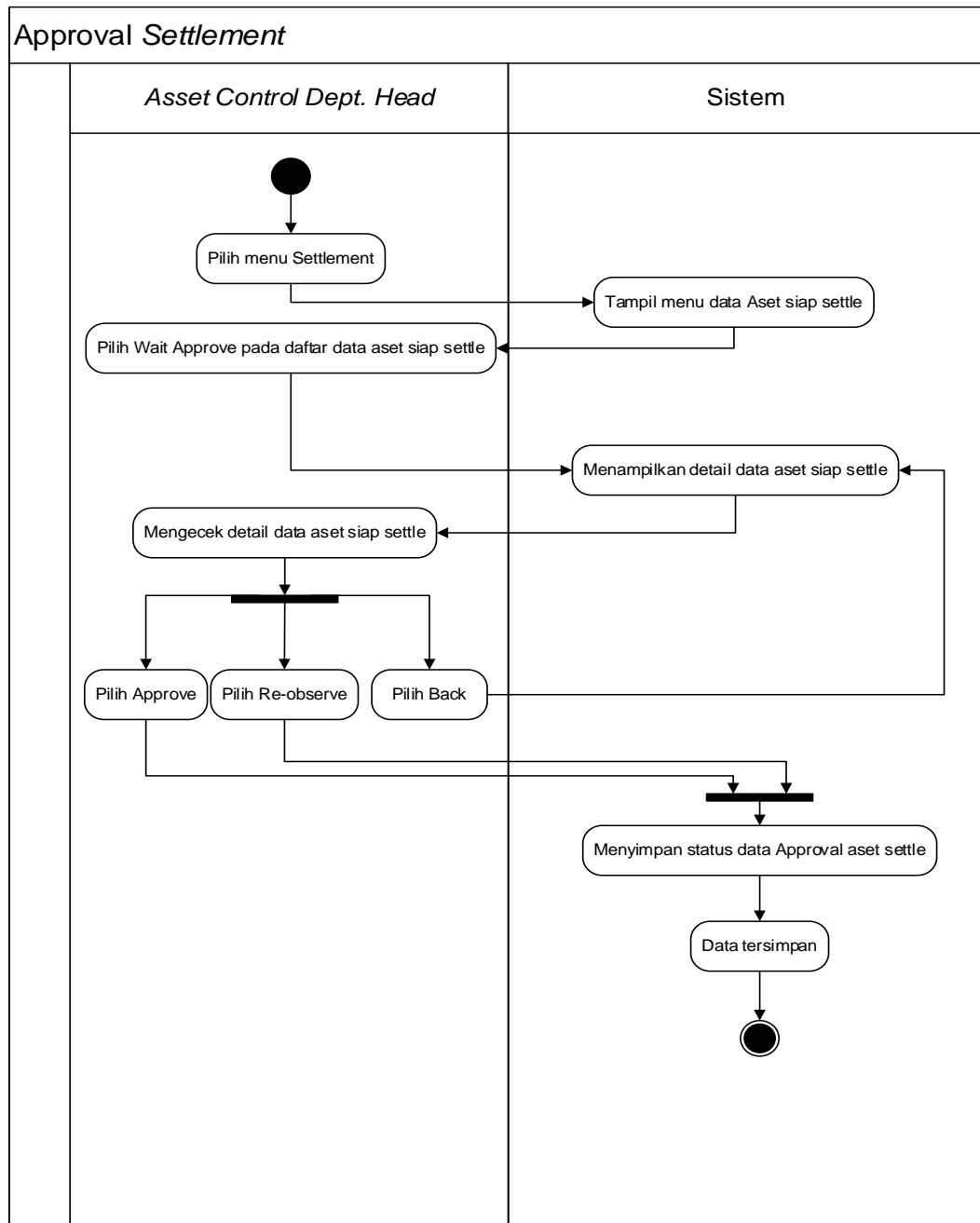
Diagram pada Gambar V.11. di bawah ini merupakan *activity diagram monitor* data hasil approval observasi yang dilakukan oleh *Asset Control*. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam *monitor* data hasil approval observasi.



Gambar V.11. Activity Diagram Monitor Hasil Approval Observasi  
Sumber: Hasil Analisis (2017)

### 10. Activity Diagram Approval Settlement

Diagram pada Gambar V.12. di bawah ini merupakan *activity diagram* dalam memberikan approval *settlement* aset yang dilakukan oleh *Asset Control Dept. Head*. Pada diagram tersebut dijelaskan bagaimana alur yang harus dilakukan dalam pemberian approval *settlement* aset.



Gambar V.12. Activity Diagram Approval Settlement  
Sumber: Hasil Analisis (2017)

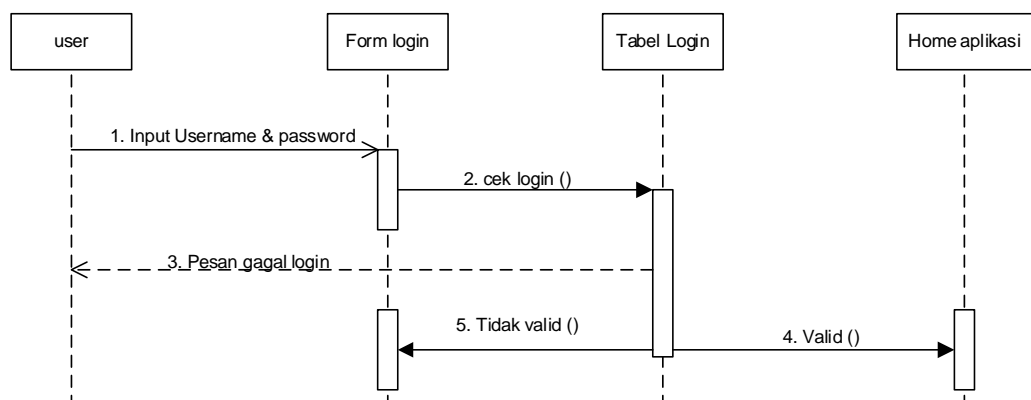
### 5.3.4 Sequence Diagram

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Diagram ini menjelaskan urutan proses yang dilakukan aktor untuk melakukan suatu tugas.

#### 1. Sequence Diagram Login

Pada Gambar V.13. menjelaskan tentang *sequence diagram login user*. Dalam proses *login* ini terdapat beberapa macam hak akses yang telah diberikan sesuai dengan fungsi masing-masing, antara lain:

- a. *Asset Control* : *User* tersebut adalah yang memiliki hak akses penuh kedalam sistem.
- b. *Asset Controller* : *User* tersebut adalah yang melakukan pengajuan aset.
- c. *Asset Control Dept. Head* : *User* tersebut adalah yang melakukan monitor dan approval aset siap *settle*.
- d. *Asset Controller Dept. Head* : *User* tersebut adalah yang melakukan monitor dan approval pengajuan aset.
- e. *Procurement* : *User* tersebut adalah yang melakukan pembuatan PO dan *update good receipt*.

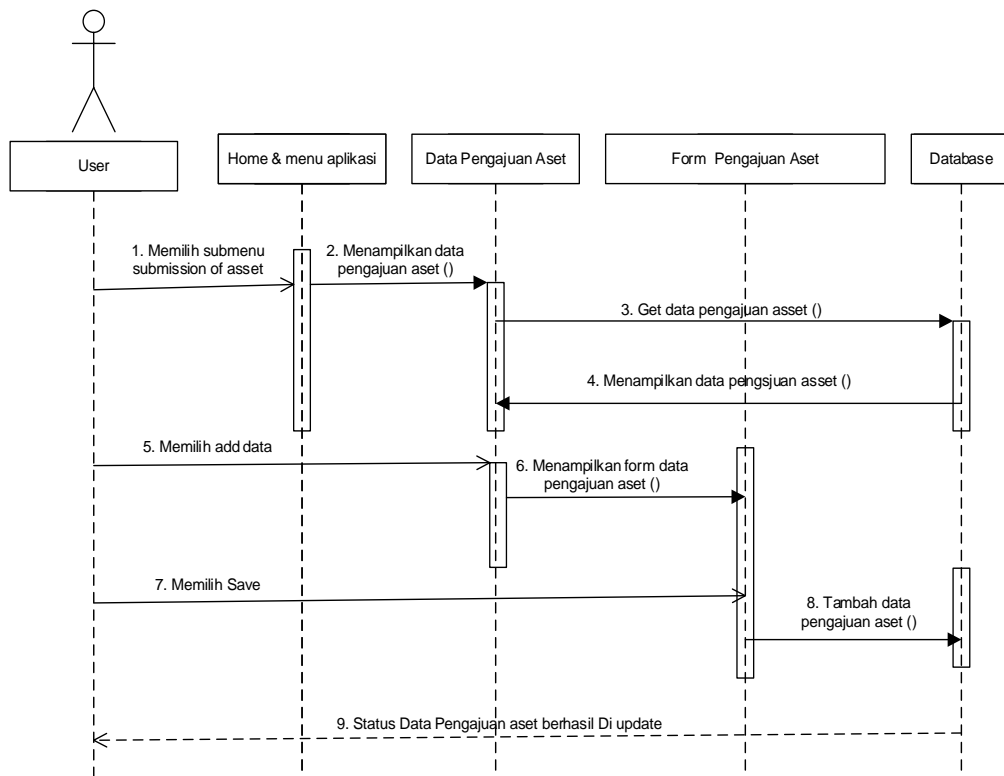


Gambar V.13. *Sequence Diagram Login*

Sumber: Analisis Data (2017)

## 2. *Sequence Diagram* Pengajuan Aset

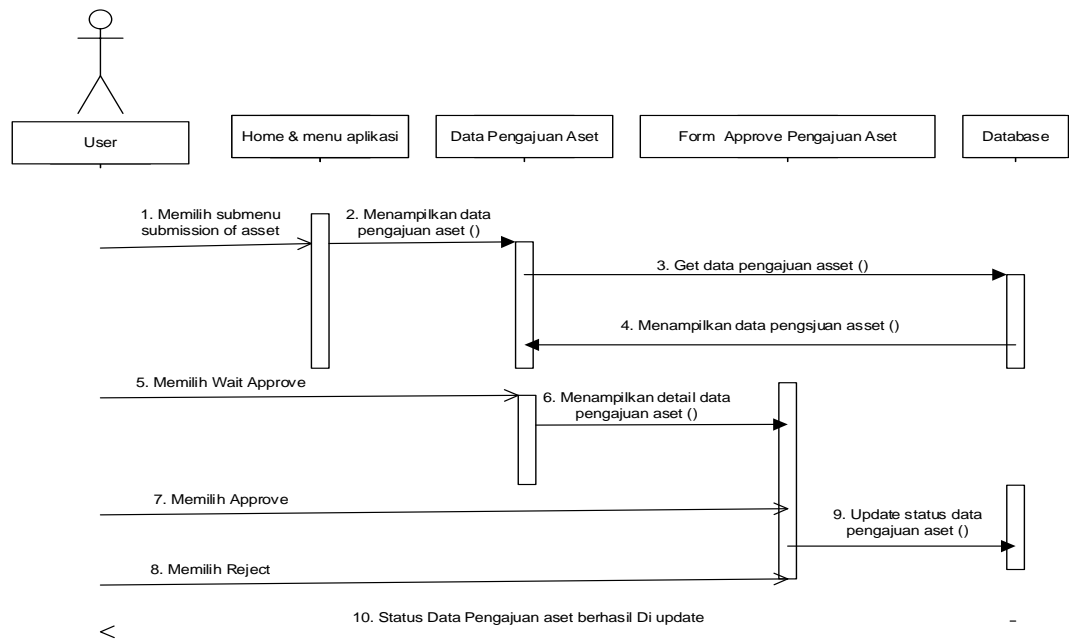
*Sequence diagram* pengajuan aset menjelaskan sebuah *sequence diagram* untuk membuat pengajuan aset. Proses ini dilakukan oleh *Asset Controller* yang sudah melakukan proses *login* sebelumnya. Adapun *sequence diagram* pengajuan aset dapat dilihat pada Gambar V.14. di bawah ini.



Gambar V.14. *Sequence Diagram* Pengajuan Aset  
Sumber: Analisis Data (2017)

## 3. *Sequence Diagram* Approval Pengajuan Aset

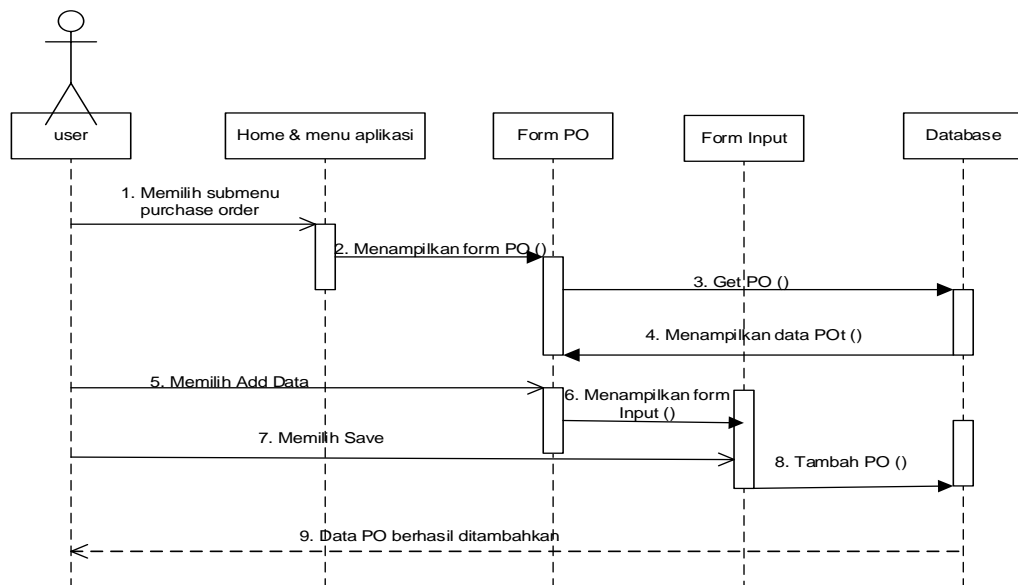
*Sequence diagram* approval pengajuan aset merupakan sebuah *sequence diagram* untuk memberikan approval terhadap pengajuan aset. Adapun *sequence diagram* approval pengajuan aset dapat dilihat pada Gambar V.15. di bawah ini.



Gambar V.15. *Sequence Diagram Approval Pengajuan Aset*  
 Sumber: Analisis Data (2017)

#### 4. *Sequence Diagram Membuat Purchase Order*

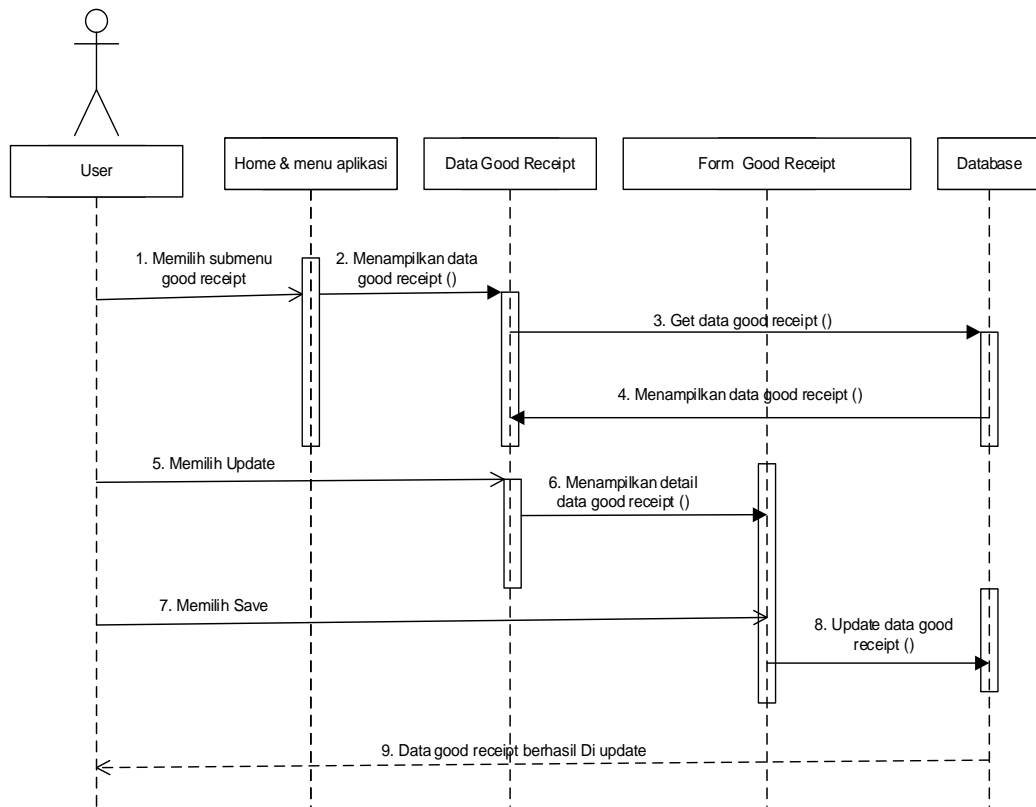
*Sequence diagram* membuat *purchase order* merupakan *sequence diagram* untuk mencetak *purchase order*. Adapun *sequence diagram* membuat *purchase order* dapat dilihat pada gambar V.16. dibawah ini.



Gambar V.16. *Sequence Diagram Membuat Purchase Order*  
 Sumber: Analisis Data (2017)

### 5. *Sequence Diagram Update Good Receipt*

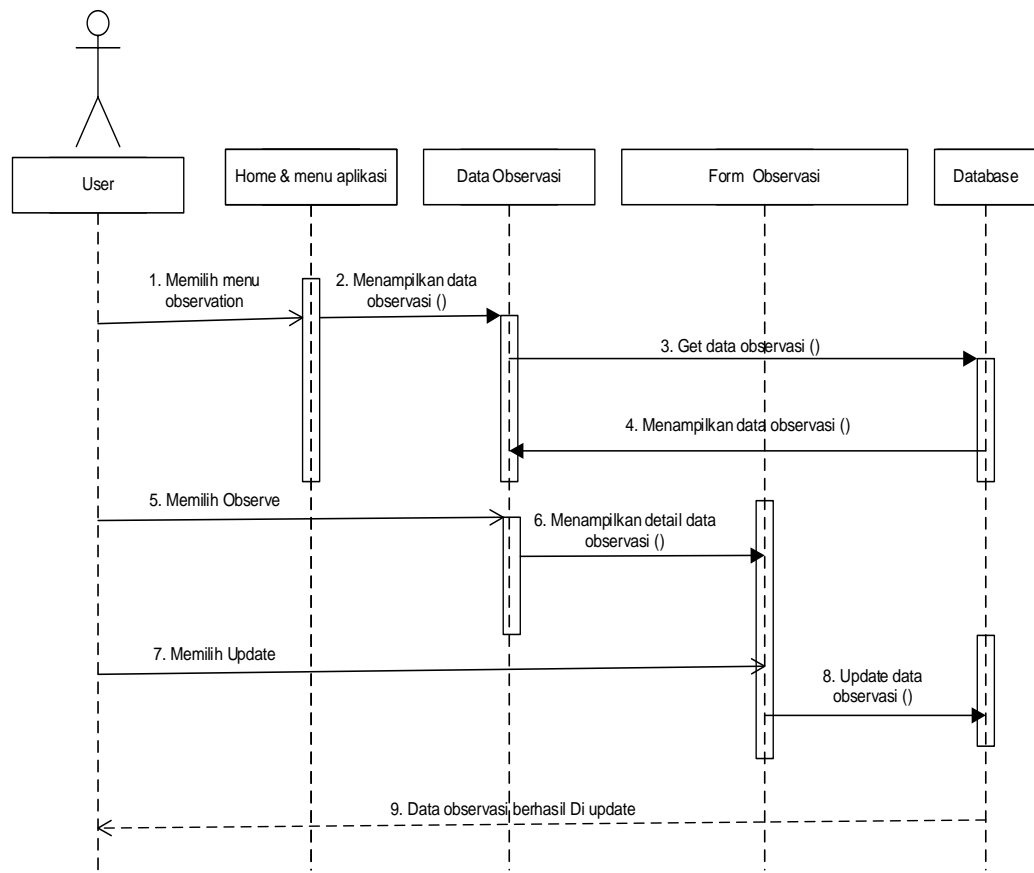
*Sequence diagram update good receipt* merupakan *sequence diagram* untuk melakukan *update* data *good receipt* sesuai dengan *invoice* dari vendor. Adapun *sequence diagram update good receipt* dapat dilihat pada gambar V.17. dibawah ini.



Gambar V.17. *Sequence Diagram Update Good Receipt*  
Sumber: Analisis Data (2017)

### 6. *Sequence Diagram Update Data Observasi*

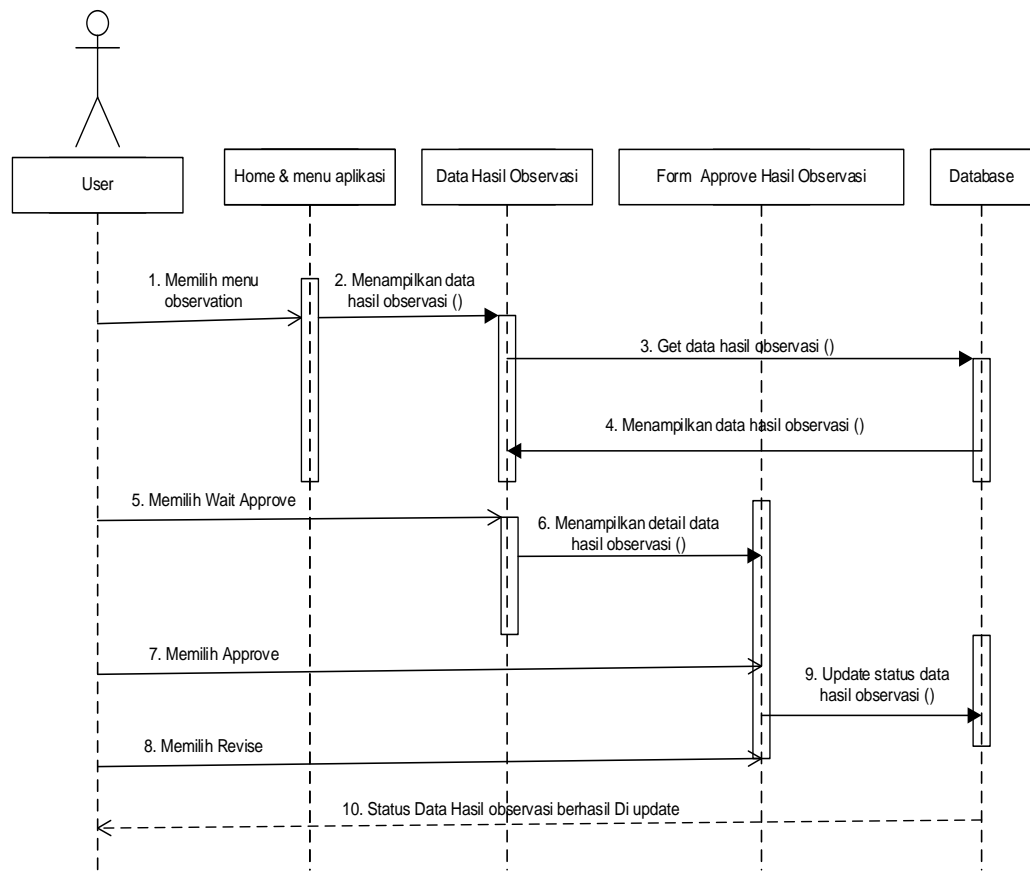
*Sequence diagram update data observasi* merupakan *sequence diagram* untuk melakukan *update* data aset saat melakukan observasi di lapangan. Adapun *sequence diagram update data observasi* dapat dilihat pada Gambar V.18. dibawah ini.



Gambar V.18. *Sequence Diagram Update Data Observasi*  
 Sumber: Analisis Data (2017)

#### 7. *Sequence Diagram Approval Hasil Observasi*

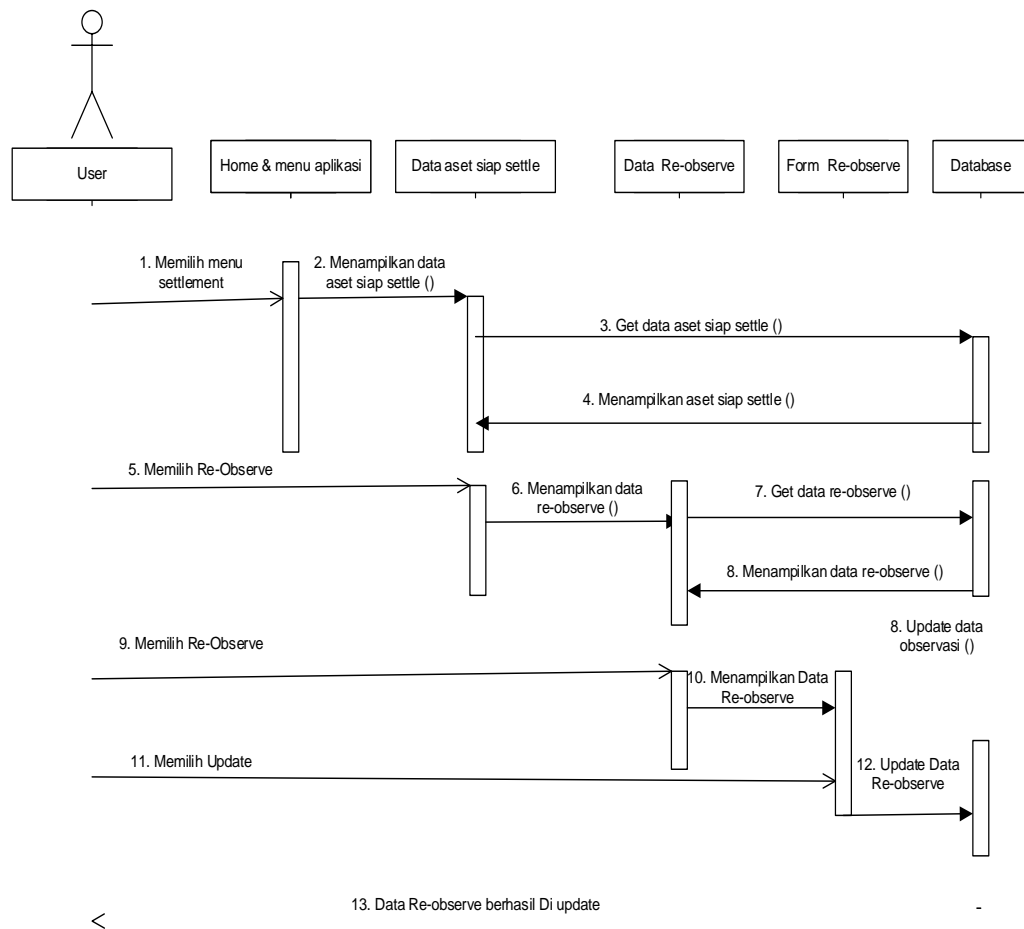
*Sequence diagram approval hasil observasi* merupakan *sequence diagram* melakukan approval terhadap hasil observasi. Adapun *sequence diagram approval hasil observasi* dapat dilihat pada Gambar V.19. dibawah ini.



Gambar V.19. *Sequence Diagram* Approval Hasil Observasi  
 Sumber: Analisis Data (2017)

#### 8. *Sequence Diagram* Monitor Hasil Approval Observasi

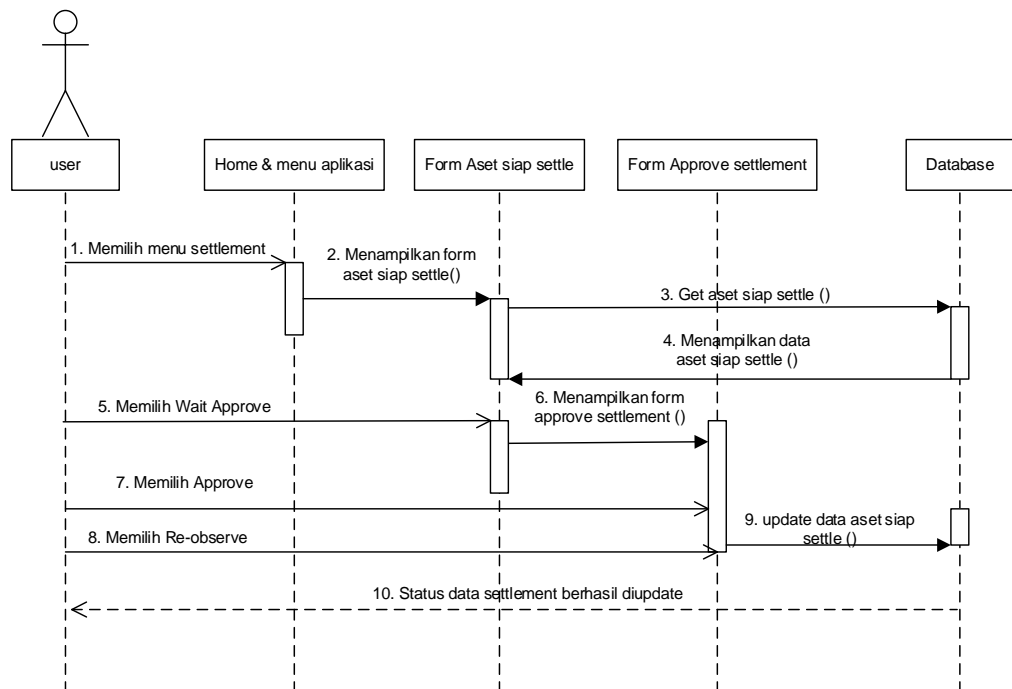
*Sequence diagram update* hasil observasi merupakan *sequence diagram* untuk melakukan *monitor* terhadap hasil approval observasi. Adapun *sequence diagram monitor* terhadap hasil approval observasi dapat dilihat pada Gambar V.20. dibawah ini.



Gambar V.20. *Sequence Diagram* Update Hasil Observasi  
 Sumber: Analisis Data (2017)

### 9. *Sequence Diagram Approval Settlement*

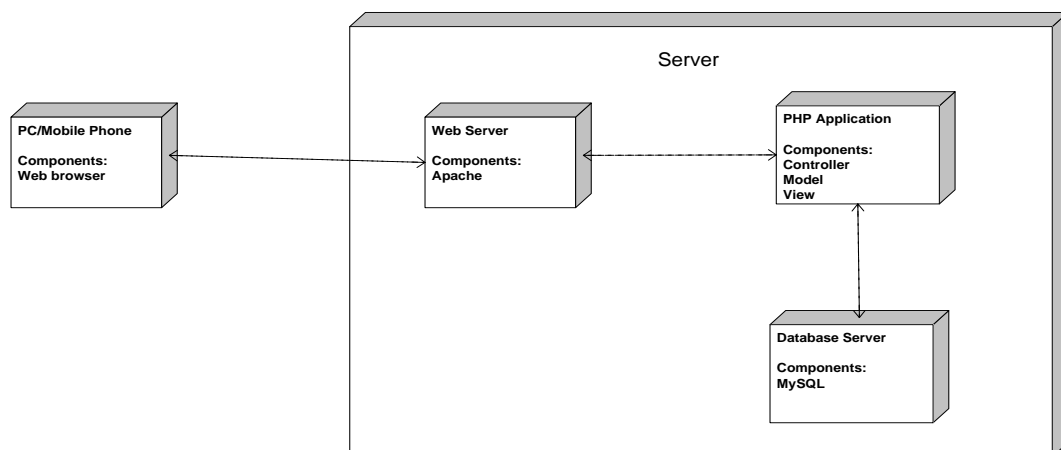
*Sequence diagram approval settlement* merupakan *sequence diagram* proses *approve settlement* terhadap aset yang siap untuk digunakan. Adapun *sequence diagram approval settlement* dapat dilihat pada Gambar V.21. dibawah ini.



Gambar V.21. *Sequence Diagram Approval Settlement*  
 Sumber: Analisis Data (2017)

**5.3.5 Deployment Diagram**

*Deployment diagram* digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur fisik sebuah sistem. *Deployment diagram procurement control system* yang diusulkan dapat dilihat pada Gambar V.22. dibawah ini.

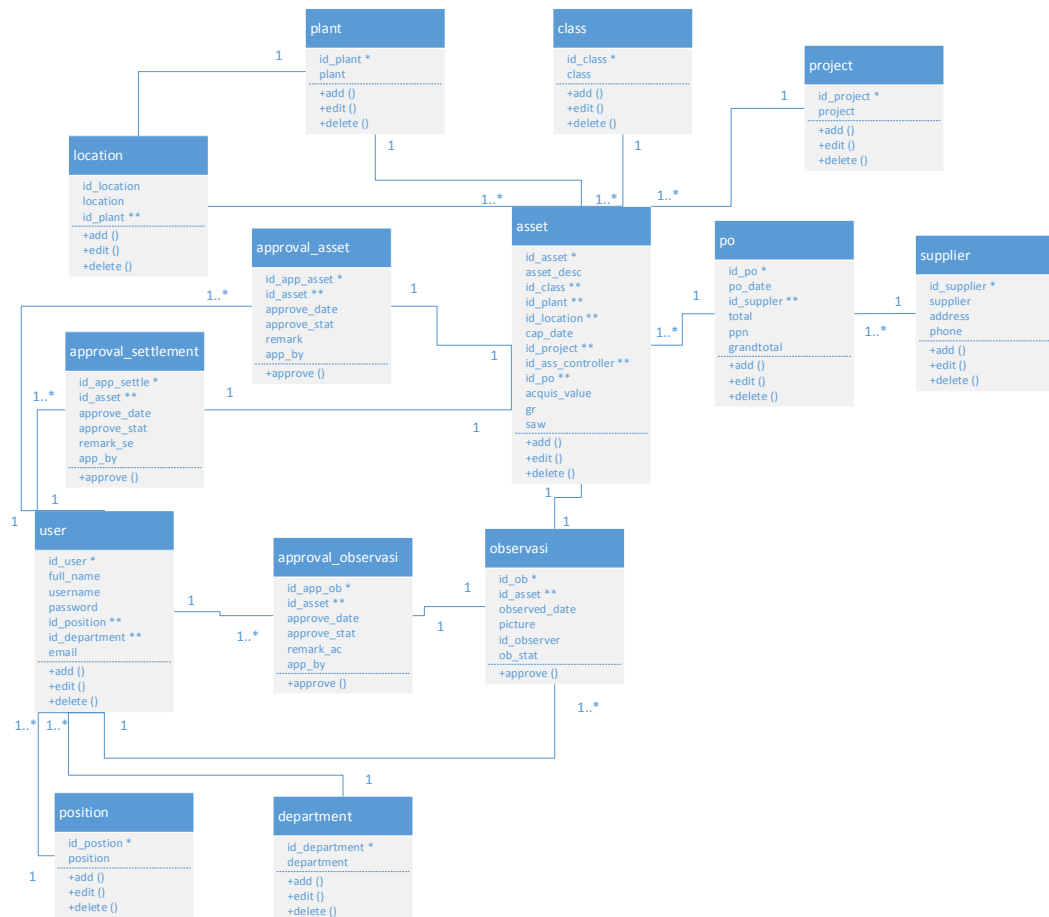


Gambar V.22. *Deployment Diagram Procurement Control System*  
 Sumber: Analisis Data (2017)

### 5.3.6 Class Diagram

*Class diagram* membantu dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem.

Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam merangkap struktur dari semua kelas yang membentuk arsitektur yang dibuat. *Class diagram procurement control system* yang diusulkan dapat dilihat pada Gambar V.23. dibawah ini.



Gambar V.23. *Class Diagram Procurement Control System*  
Sumber: Analisis Data (2017)

## 5.4 Perancangan Basis Data

Dalam perancangan basis data pendefinisian struktur data dari file-file *database* akan didokumentasikan dalam kamus data. Kamus data ini diperlukan untuk membuat file secara fisik. Kamus data dapat berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan di dalam system secara persis sehingga pemakai dan penganalisis system mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses.

### 1. Spesifikasi Tabel

Nama tabel : *asset*  
 Fungsi : Untuk menyimpan data aset  
 Tipe : File master

Tabel V.15 Tabel *asset*

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id asset	id_asset	Char	12	<i>Primary Key</i>
2	Deskripsi asset	asset_desc	Varchar	100	
3	Id kelas	id_class	Char	5	<i>Foreign Key</i>
4	Id plant	id_plant	Char	4	<i>Foreign Key</i>
5	Id lokasi	id_location	Char	10	<i>Foreign Key</i>
6	Tanggal penggunaan	cap_date	Date		
7	Id Project	id_project	Char	3	<i>Foreign Key</i>
8	Asset Controller	id_ass_controller	Char	5	<i>Foreign Key</i>
9	Id Supplier	id_supplier	Char	5	<i>Foreign Key</i>
10	Id PO	id_po	Char	10	<i>Foreign Key</i>
11	Harga	acq_val	Double		
12	Good Receipt	Gr	Tinyint	3	
13	Simple Addictive Weighting	Saw	decimal	3,3	

Sumber: Analisis Data (2017)

## 2. Spesifikasi Tabel

Nama tabel : class  
 Fungsi : Untuk menyimpan data kelas aset  
 Tipe : File master

Tabel V.16 Tabel class

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id kelas	id_Class	Char	5	<i>Primary Key</i>
2	Deskripsi Kelas	Class	Varchar	100	

Sumber: Analisis Data (2016)

## 3. Spesifikasi Tabel

Nama tabel : plant  
 Fungsi : Untuk menyimpan data plant  
 Tipe : File master

Tabel V.17 Tabel plant

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id Plant	id_plant	char	4	<i>Primary Key</i>
2	Deskripsi Plant	Plant	Varchar	50	

Sumber: Analisis Data (2017)

## 4. Spesifikasi Tabel

Nama tabel : location  
 Fungsi : Untuk menyimpan data lokasi  
 Tipe : File master

Tabel V.18 Tabel location

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id lokasi	id_location	Char	10	<i>Primary Key</i>
2	Deskripsi lokasi	Location	Varchar	100	
3	Id Plant	id_plant	char	4	<i>Foreign Key</i>

Sumber: Analisis Data (2017)

## 5. Spesifikasi Tabel

Nama tabel : position

Fungsi : Untuk menyimpan data *position*

Tipe : File master

Tabel V.19 Tabel asset\_controller

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id Posisi	id_position	Char	4	<i>Primary Key</i>
2	Deskripsi Posisi	Position	Varchar	100	

Sumber: Analisis Data (2017)

## 6. Spesifikasi Tabel

Nama tabel : department

Fungsi : Untuk menyimpan data *department*

Tipe : File master

Tabel V.20 Tabel department

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id Department	id_department	Char	2	<i>Primary Key</i>
2	Deskripsi Department	Department	Varchar	100	

Sumber: Analisis Data (2017)

## 7. Spesifikasi Tabel

Nama tabel : supplier

Fungsi : Untuk menyimpan data *supplier*

Tipe : File master

Tabel V.21 Tabel supplier

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id Supplier	Id_supplier	Char	5	<i>Primary Key</i>
2	Nama Supplier	Supplier	Varchar	30	
3	Alamat Supplier	Address	Varchar	25	
4	Nomor telepon	Phone	Char	15	

Sumber: Analisis Data (2017)

## 8. Spesifikasi Tabel

Nama tabel : project  
 Fungsi : Untuk menyimpan data tipe proyek  
 Tipe : File master

Tabel V.22 Tabel project

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id project	id_project	Char	3	<i>Primary Key</i>
2	Tipe project	Project	Varchar	20	

Sumber: Analisis Data (2016)

## 9. Spesifikasi Tabel

Nama tabel : user  
 Fungsi : Untuk menyimpan data *user*  
 Tipe : File master

Tabel V.23 Tabel user

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id user	id_user	Char	5	<i>Primary Key</i>
2	Nama lengkap	full_name	Varchar	50	
3	Nama user	Username	Varchar	20	
4	Password	Password	Varchar	100	
5	Id posisi	id_position	Char	4	<i>Foreign Key</i>
6	Id department	id_department	Char	4	<i>Foreign Key</i>
7	Email	Email	Varchar	50	

Sumber: Analisis Data (2017)

## 10. Spesifikasi Tabel

Nama tabel : observasi  
 Fungsi : Untuk menyimpan data observasi  
 Tipe : File transaksi

Tabel V.24 Tabel observasi

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id observasi	id_ob	Char	10	<i>Primary Key</i>
2	Id asset	id_asset	Char	12	<i>Foreign Key</i>
3	Gambar asset	Picture	Varchar	250	
4	Tanggal observasi	observed_date	Date		
5	Id observer	Id_observer	Char	2	<i>Foreign Key</i>
6	Keterangan	remark_ob	Varchar	250	

Sumber: Analisis Data (2017)

### 11. Spesifikasi Tabel

Nama tabel : approval\_asset

Fungsi : Untuk menyimpan data approval asset

Tipe : File Transaksi

Tabel V.25 Tabel approval\_asset

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id approval asset	id_app_asset	Char	12	<i>Primary Key</i>
2	Id asset	id_asset	Char	12	<i>Foreign Key</i>
3	Tanggal approve	approve_date	Date		
4	Status approve	approve_stat	Char	1	
5	Keterangan	Remark	Varchar	250	
6	Id approver	App_by	Char	5	

Sumber: Analisis Data (2017)

### 12. Spesifikasi Tabel

Nama tabel : approval\_observasi

Fungsi : Untuk menyimpan data approval observasi

Tipe : File Transaksi

Tabel V.26 Tabel approval\_observasi

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id approval observasi	id_app_ob	Char	12	<i>Primary Key</i>
2	Id asset	id_asset	Char	12	<i>Foreign Key</i>
3	Tanggal approve	approve_date	Date		
4	Status approve	approve_stat	Char	1	
5	Keterangan	remark_ac	Varchar	250	
6	Id approver	App_by	Char	5	

Sumber: Analisis Data (2017)

### 13. Spesifikasi Tabel

Nama tabel : approval\_settlement

Fungsi : Untuk menyimpan data approval settlement

Tipe : File Transaksi

Tabel V.27 approval\_settlement

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id approval settle	id_app_settle	Char	12	<i>Primary Key</i>
2	Id asset	id_asset	Char	12	<i>Foreign Key</i>
3	Tanggal approve	approve_date	Date		
4	Status approve	approve_stat	Char	1	
5	Keterangan	remark_se	Varchar	250	
6	Id approver	App_by	Char	5	

Sumber: Analisis Data (2017)

### 14. Spesifikasi Tabel

Nama tabel : po

Fungsi : Untuk menyimpan data *purchase order*

Tipe : File Transaksi

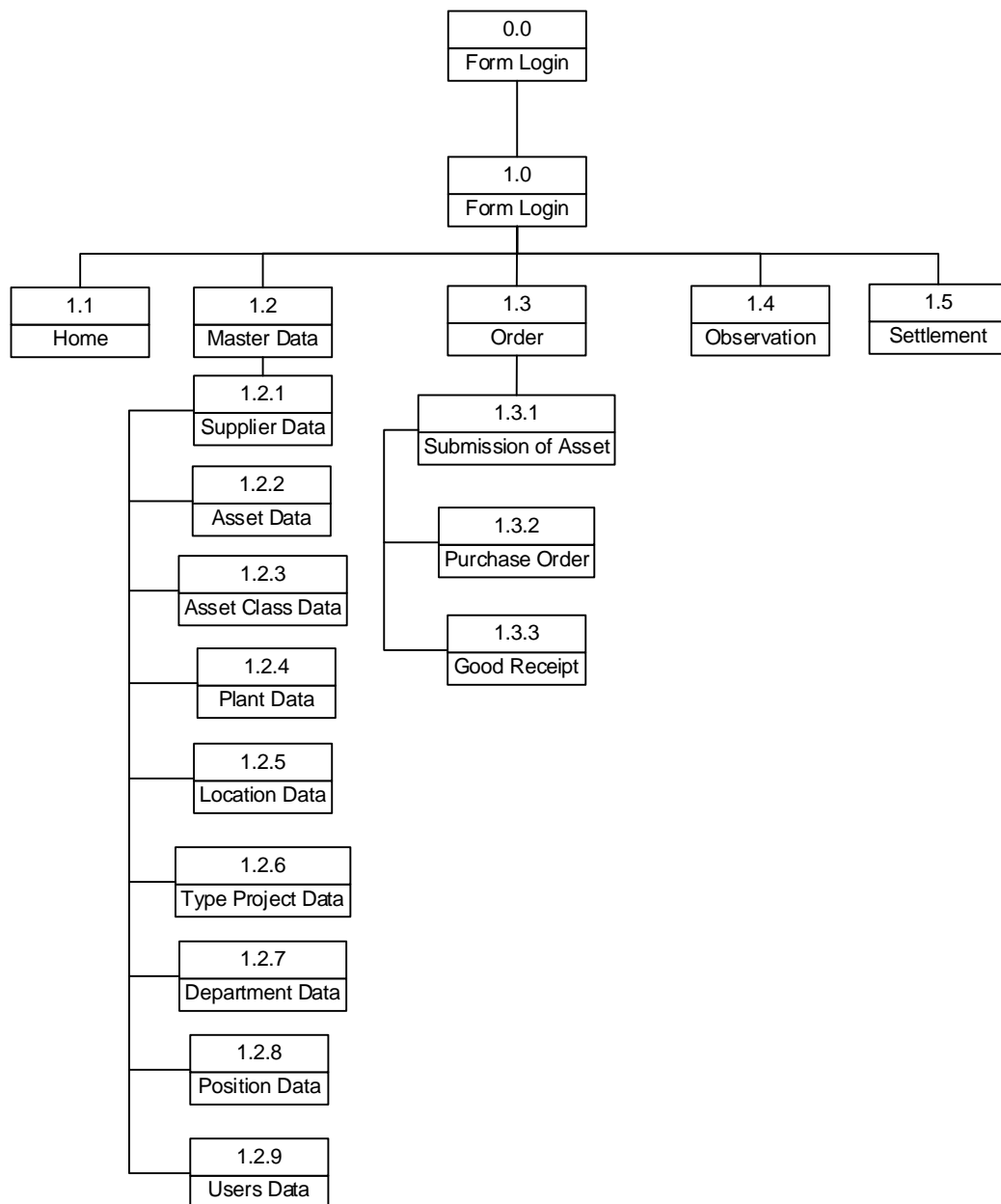
Tabel V.28 Tabel po

No	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id purchase order	id_po	Char	10	<i>Primary Key</i>
2	Tanggal po	po_date	Date		
3	Id vendor	id_supplier	Char	5	
4	Total po	total	Double	1	
5	Ppn	ppn	Double	250	
6	Grandtotal po	grandtotal	Double	5	

Sumber: Analisis Data (2017)

### 5.5 *Hierarchy plus Input-Process-Output (HIPO) PCS*

*Hierarchy plus Input-Process-Output (HIPO)* digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hierarkis. HIPO yang digunakan untuk mendokumentasikan struktur dari program absensi ini yaitu menggunakan diagram HIPO tipe daftar isi visual (DIV). Diagram ini memuat semua modul yang ada dalam sistem beserta nama dan nomornya.



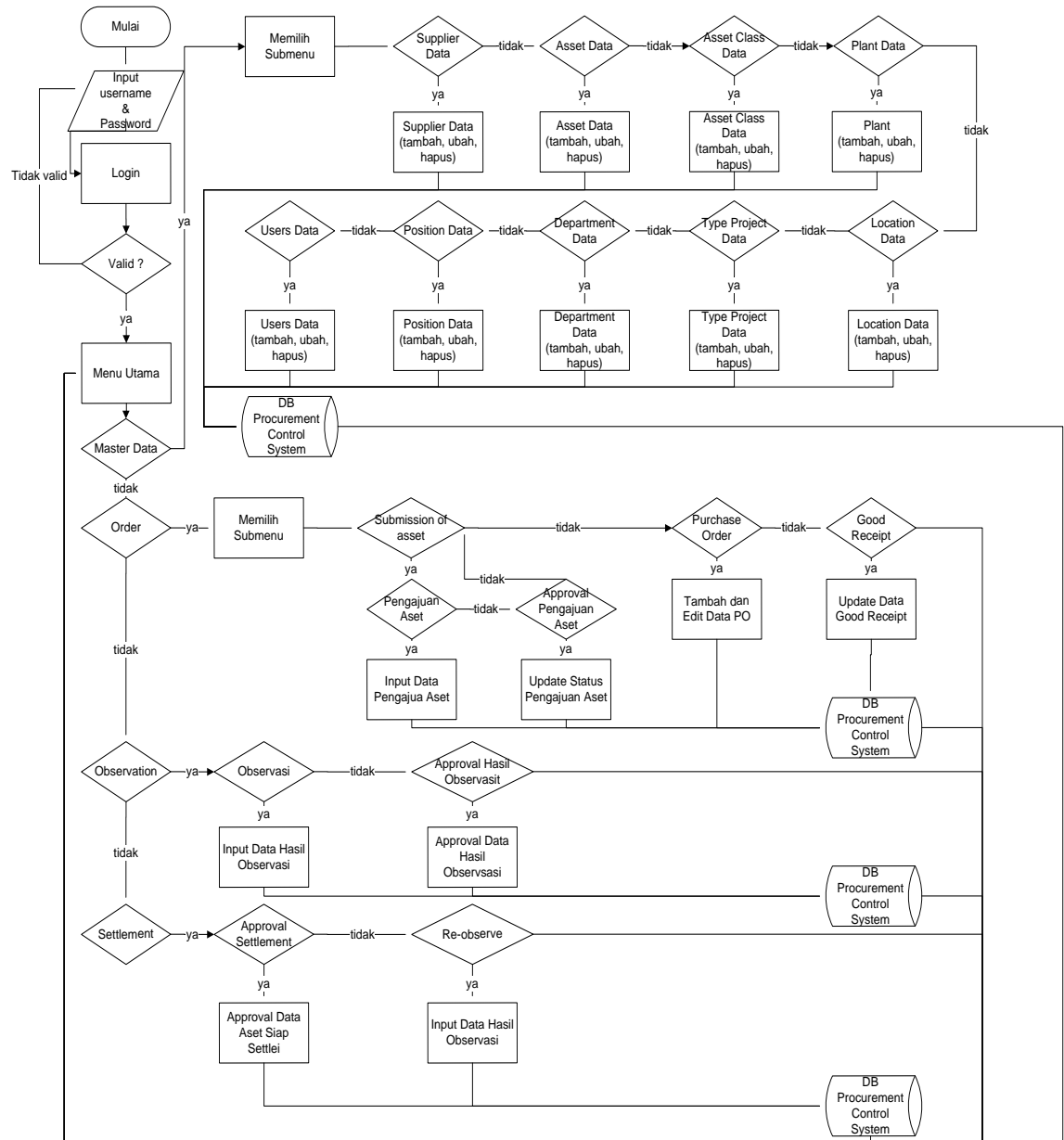
Gambar V.22 Struktur Menu Program

(Sumber: Analisis Data, 2017)

## 5.6 Flowchart Program

*Flowchart* yang digunakan untuk mendokumentasikan *Procurement Control System* 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.24. menunjukkan *program logic flowchart Procurement Control System* :



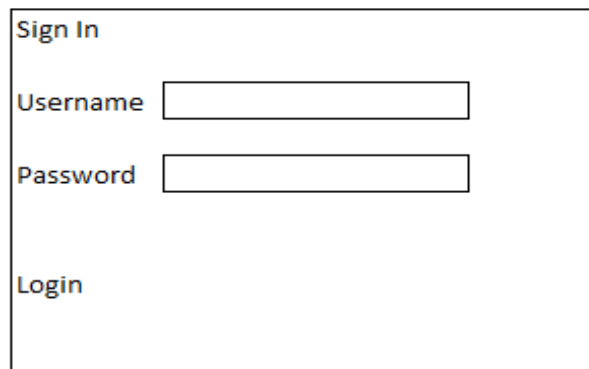
Gambar V.24. *Program Logic Flowchart Procurement Control System*  
Sumber: Hasil Analisis Data (2017)

## 5.7 Perancangan *Interface Procurement Control System*

Rancangan *interface* dari program *Procurement Control System* ini adalah sebagai berikut:

### 1. Rancangan *Form Login*

Rancangan *form login* adalah *form* yang digunakan untuk masuk ke dalam aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan username dan password yang benar. Rancangan *form login* dapat dilihat pada Gambar V.25.



The image shows a login form titled "Sign In". It contains two input fields: "Username" and "Password". Below the input fields is a "Login" button.

Gambar V.25. Rancangan *Form Login*  
(Sumber: Analisis Data, 2017)

### 2. Rancangan Daftar Menu

Rancangan Daftar menu adalah tampilan menu yang masing-masing memiliki fungsi berbeda. Pada daftar menu berisi lima menu. Lima menu tersebut adalah *Home*, *Master Data*, *Order*, *Observation* dan *Settlement*. Rancangan *form* menu dapat dilihat pada Gambar V.26.

Procurement Control System				
Home	Master Data	Order	Observation	Settlement
	-Asset Data	-Submissin of Asset		
	-Supplier Data	-Purchase Order		
	-Plant Data	-Good Receipt		
	-Location Asset Data			
	-Asset Controller Data			
	-Observer Data			
	-Access Right Data			

Gambar V.26. Rancangan *Form Menu*  
(Sumber: Analisis Data, 2017)

### 3. Rancangan Tampilan Untuk Menampilkan Data *Supplier*

Rancangan tampilan untuk menampilkan data *supplier* pada Gambar V.27. adalah tampilan yang digunakan untuk menampilkan *supplier* yang berada didalam database. Berikut penjelasan mengenai penggunaan pada *form* ini:

- a. Tombol Edit : Untuk mengubah data *supplier*.
- b. Tombol Delete : Untuk menghapus data *supplier*.

The screenshot shows a web application interface for a Procurement Control System. At the top, there is a navigation menu with options: Home, Master Data, Order, Observation, and Settlement. Below the menu is a search bar with the label 'Search' and an input field. A button labeled 'Add Data' is positioned to the left of the main data table. The table itself has the following structure:

No	Id Supplier	Supplier Name	Address	Phone	Action
					Edit Delete
					Edit Delete
					Edit Delete

Gambar V.27. Rancangan Tampilan Untuk Menampilkan Data *Supplier*  
(Sumber: Analisis Data, 2017)

### 4. Rancangan *Form* Tambah Data *Supplier*

Rancangan *form* tambah data *supplier* pada Gambar V.28. adalah *form* yang digunakan untuk menambah data *supplier* kedalam database. Berikut penjelasan mengenai penggunaan pada *form* ini:

- c. Tombol Save : Untuk menyimpan data *supplier*.
- d. Tombol Back : Untuk membatalkan proses penambahan data *supplier*

Procurement Control System				
Home	Master Data	Order	Observation	Settlement
Edit Data Supplier				
ID Supplier	<input type="text"/>			
Supplier Name	<input type="text"/>			
Address	<input type="text"/>			
Phone	<input type="text"/>			
		<input type="button" value="Save"/> <input type="button" value="Back"/>		

Gambar V.28. Rancangan *Form* Tambah Data *Supplier*  
(Sumber: Analisis Data, 2017)

#### 5. Rancangan *Form* Edit Data *Supplier*

Rancangan *form* edit data *supplier* pada Gambar V.29. adalah *form* yang digunakan untuk mengedit data *supplier* yang sudah disimpan sebelumnya. Berikut adalah penjelasan mengenai penggunaan *form* edit data *supplier* adalah sebagai berikut:

- a. Tombol Back : Untuk kembali ke list data *supplier*.
- b. Tombol Save : Untuk menyimpan data yang sudah diedit.

Procurement Control System				
Home	Master Data	Order	Observation	Settlement
Edit Data Supplier				
ID Supplier	<input type="text"/>			
Supplier Name	<input type="text"/>			
Address	<input type="text"/>			
Phone	<input type="text"/>			
		<input type="button" value="Save"/> <input type="button" value="Back"/>		

Gambar V.29. Rancangan *Form* Edit Data *Supplier*  
(Sumber: Analisis Data, 2017)

## 6. Rancangan Tampilan Untuk Menampilkan Data Observasi

Rancangan tampilan untuk menampilkan data observasi pada Gambar V.30. adalah tampilan yang digunakan untuk menampilkan observasi yang berada didalam database. Berikut penjelasan mengenai penggunaan pada *form* ini:

- a. Tombol Observe : Untuk *update* data observasi.

Procurement Control System						
Home	Master Data	Order	Observation	Settlement		
Filter	<input type="text"/>					
						Search <input type="text"/>
No	Id Asset	Asset_Desc	Plant	Location	Asset Controller	Action
						Observe
						Observe
						Observe

Gambar V.30. Rancangan Tampilan Untuk Menampilkan Data Observasi  
(Sumber: Analisis Data, 2017)

## 5.7 Implementasi Sistem

Untuk dapat menggunakan aplikasi, tentunya diperlukan suatu spesifikasi perangkat lunak ataupun perangkat keras yang mendukung agar aplikasi dapat berjalan dengan baik.

### 5.7.1 Kebutuhan Perangkat Lunak

Untuk menggunakan *Procurement Control System* ini diperlukan suatu minimum kebutuhan perangkat lunak sebagai berikut:

- a. XAMPP 1.7.1 untuk pengelolaan *database*.
- b. *Browser* Mozilla Firefox atau Google Chrome untuk menjalankan fitur phpMyadmin XAMPP.
- c. Sistem Operasi minimal Windows XP.

### 5.7.2 Kebutuhan Perangkat Keras

Untuk mendukung kebutuhan implementasi *Procurement Control System* pada PT Astra Honda Motor diperlukan dukungan sistem komputer yang memadai dalam segi *hardware*, diantaranya sebagai berikut:

- a. *Processor* : minimal Pentium IV (2,6 GHz).
- b. RAM : minimal 1 GB.
- c. *Harddisk* : minimal 64 GB.
- d. Sistem Operasi : Microsoft Windows XP atau di atasnya.

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **6.1 Kesimpulan**

Berdasarkan pembuatan aplikasi yang dilakukan mengenai sistem informasi *Procurement Control System* di PT Astra Honda Motor, dapat disimpulkan beberapa hal yaitu:

1. Proses pengajuan *Fixed Asset* menjadi lebih cepat karena *approval* dilakukan secara sistem dan dapat termonitor secara *real time*.
2. Proses pengajuan *Fixed Asset* menjadi terintegrasi dengan proses observasi hingga *settlement*. Integrasi sistem ini menjadi lebih mempermudah dalam melakukan kontrol pengadaan aset.
3. Proses penarikan data aset dengan *good receipt* 100% per-awal yang memperlambat proses *settlement* tidak diperlukan lagi karena data *good receipt* dapat termonitor secara *real time*.
4. Observer dapat langsung melakukan *update* data melalui sistem saat melakukan observasi di lapangan dan prioritas aset terhitung otomatis dengan metode SAW (*Simple Addictive Weighting*).
5. Proses *settlement* dapat termonitor secara *real time* tanpa harus menunggu email internal mengenai *settlement* aset.

#### **6.2 Saran**

Adapun saran yang diharapkan agar *Procurement Control System* ini dapat berjalan dengan baik yaitu:

1. Untuk penerapan sistem baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi kepada bagian-bagian yang terkait yang akan menggunakan sistem baru ini.
2. Sebaiknya dilakukan pemeliharaan aplikasi secara berkala, sehingga aplikasi dapat berjalan dengan baik.

3. Sebaiknya tersedia sumber daya manusia yang berkompeten dalam bidang IT untuk dapat memperbaiki jika terjadi kesalahan atau *error* pada aplikasi.

## DAFTAR PUSTAKA

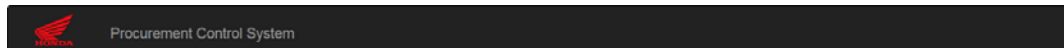
- Al-Bahra, Ladjamudin. 2006. *Rekayasa Perangkat Lunak*. Yogyakarta: Graha Ilmu
- Davis, Gordon B. 1995. *Sistem Informasi Manajemen*. Jakarta: PT. Pustaka Binaman Pressindo
- Dey, Ian. 1995. *Grounding Grounded Theory – Guidelines For Qualitative Inquiry*. San Diego: Academic Press.
- Dennis, A., Wixom, B.H., Tegarden, D. 2005. *System Analysis And Design With UML Version 2.0*. USA: John Wiley & Sons, Inc.
- Indrajit. 2001. *Analisis dan Perancangan Sistem Berorientasi Objek*. Bandung: Informatika.
- Jimmy, Chr., Gaol, L. 2008. *Sistem Informasi Manajemen*. Jakarta: Grasindo.
- Jogiyanto. 2005. *Analisis & Desain*. Yogyakarta: Andi.
- Jogiyanto. 2005. *Sistem Teknologi Informasi*. Yogyakarta: Andi.
- Kadir, Abdul. 2009. *Dasar Perancangan Dan Implementasi Database Relasional*. Yogyakarta: Andi.
- Kasiram, M. 2006. *Metodologi Penelitian Kualitatif-Kuantitatif*. Malang: Press.
- Laudon, Kenneth C. dan Jane P. Laudon. 2007. *Sistem Informasi Manajemen*. Edisi Kesepuluh. Jakarta: Salemba Empat.
- McLeod., Raymond., S, George. 2004. *Sistem Informasi Manajemen*. Jakarta: PT. Indeks.
- Ramakrishnan, Raghu dan Johannes Gehrke. (2003). *Database Management System*. 3rd Edition. New York: McGraw-Hill.
- Rosa A.S. dan M. Shalahuddin. 2013. *Rekayasa Perangkat Lunak*. Bandung: Informatika.
- Saputra, Agus. 2011. *Trik Kolaborasi Codeigniter & JQuery*. Yogyakarta: Lokomedia

- Sidharta, Lani. 1995. *Pengantar Sistem Informasi Bisnis*. Jakarta: Elex Media Komputindo.
- Silberschatz dkk. 2002. *Data Base Concept. Fourth Edition*. New York: Mc.Graw-Hill Higher Education.
- Simarmata, Janner. 2010. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi Offset.
- Solichin, Achmad. 2010. *MySQL 5 Dari Pemula Hingga Mahir*. Jakarta: Universitas Budi Luhur.
- Sommerville, I., 2003. *Rekayasa Perangkat Lunak*. Edisi 6, diterjemahkan oleh: Hanum, Y. Jakarta: Erlangga.
- Munawar. 2005. *Pemodelan Visual dengan UML*. Yogyakarta: Graha Ilmu.
- Murdick, dkk. 1991. *Sistem Informasi Untuk Manajemen Modern*. Jakarta: Erlangga.
- Nursalim. *Tipe Data MySQL*. 21 Desember 2015. [naura-lab.blogspot.co.id/2014/12/mengenal-tipe-data-di-mysql.html](http://naura-lab.blogspot.co.id/2014/12/mengenal-tipe-data-di-mysql.html).
- Peranginangin, Kasiman. 2006. *Aplikasi Web dengan PHP dan MySQL*, Yogyakarta: Andi.
- Rogayah. *UML dan Diagram Activity*. 30 Juli 2016. [rogayah.staff.gunadarma.ac.id/Downloads/files/38682/UML+Diagram+Activity.pdf/](http://rogayah.staff.gunadarma.ac.id/Downloads/files/38682/UML+Diagram+Activity.pdf/).
- Whitten, Jeffrey L., Lonnie D. Bentley. 2007. *Metode Desain dan Analisis Sistem. Edisi Ketujuh*. Yogyakarta: Andi Offset.
- ..... *Peranan Sistem Informasi Manajemen*. 23 Agustus 2015. <http://library.upnvj.ac.id/pdf/3tipdf/208513005/bab2>.
- ..... *Sejarah AHM*. 23 Agustus 2015. <http://www.astra-honda.com/index.php/sejarah-astra-hondamotor/>.
- ..... *Sistem Informasi Manajemen*. 11 Agustus 2015. <http://isjd.pdii.lipi.go.id/admin/jurnal/122087180.pdf>.
- ..... *Simple Additive Weighting*. 28 Desember 2016. [www.etunas.co.id/blog/2015/08/13/pengertian-metode-simple-additive-weighting-saw/](http://www.etunas.co.id/blog/2015/08/13/pengertian-metode-simple-additive-weighting-saw/).
- ..... *Teori Aktiva Tetap*. 26 Juni 2016. [repository.uin-suska.ac.id/4789/3/BAB%20II](http://repository.uin-suska.ac.id/4789/3/BAB%20II).

## LAMPIRAN

### TAMPILAN PROGRAM

#### 1. Login

A white rectangular form with a light blue header that says 'Sign In'. Below the header, there are two input fields. The first is labeled 'Username' and has a person icon to its left. The second is labeled 'Password' and has a lock icon to its left. Below these fields is a green button with the word 'Login' in white text.

#### Kode Program :

```
<html>
  <head>
    <title>Procurement Control System</title>
    <link href="<?php echo base_url('assets/css/bootstrap.min.css');?>"
rel="stylesheet">
    <link href="<?php echo base_url('assets/css/mine.css');?>"
rel="stylesheet">
    <script src="<?php echo
base_url('assets/js/jquery.min.js');?>"></script>
    <script src="<?php echo base_url('assets/js/bootstrap.js');?>"></script>
  </head>
  <body>

    <div class="navbar navbar-inverse">
    <div class="container.fluid">
```

```

<div class="navbar-header">
    <div class="row" style="margin:5px;">
        <div class="col-md-10">
            
        </div>
    </div>
</div>
<div>
    <? echo $query;?>
    <div class="container">
        <div style="margin-top:50px;" class="mainbox col-md-4 col-md-offset-
4">
            <div class="panel panel-info" >
                <div class="panel-heading">
                    <div class="panel-title">Sign In</div>
                    <div style="float:right; font-size: 80%; position: relative; top:-
10px"></div>
                </div>
                <div style="padding-top:30px" class="panel-body" >
                    <form class="form-horizontal" role="form" action="<?php
echo site_url('login_c/do_login');?>" method="post">
                        <div
style="margin-bottom: 25px" class="input-group">

```

```
                <span                class="input-group-addon"><i
class="glyphicon glyphicon-user"></i></span>
                <input    type="text"    class="form-control"
name="username" value="" placeholder="Username" required>
        </div>
```

```
        <div style="margin-bottom: 25px" class="input-group">
                <span                class="input-group-addon"><i
class="glyphicon glyphicon-lock"></i></span>
                <input    type="password"    class="form-control"
name="password" placeholder="Password" required>
        </div>
        <?php echo $this->session->flashdata('message');?>
```

```
        <div style="margin-top: 10px" class="form-group">
                <!-- Button -->
                <div class="col-sm-12 controls">
                        <button    class="btn btn-success"    type="submit"
style="">Login </button>
                </div>
        </div>
```

```
        </form>
    </div>
</div>
```

```
</div>
```

```

        <div class="navbar navbar-inverse navbar-fixed-bottom"
style="line-height:10px">
            <div>
                <p class="navbar-text text-center" style="font-
size:12">#169; Copyright by Riyan Alfian, Sekolah Tinggi Manajemen
Industri</a></p>
            </div>
        </div>
    </body>
</html>

```

## 2. Master Data Supplier

The screenshot shows the Procurement Control System (PCS) interface. The header includes the PCS logo and navigation tabs: Home, Master Data (selected), Order, Observation, and Settlement. A user profile for 'admin' is visible in the top right. Below the navigation is a '+ Add Data' button and a search bar for 'Search Supplier (ID/Name)'. The main content area displays a table with the following data:

No.	ID Supplier	Supplier Name	Address	Phone	Action
1	S0001	PT Metindo Erasakti	Jl Jababek 02	467233	
2	S0002	PT Nusahadi Daya Cipta	Jl Narogong 12	432033	
3	S0003	PT Era Sakti	Jl Indah Pamulang	453200	

The footer of the interface contains the copyright notice: © Copyright by Riyan Alfian, Sekolah Tinggi Manajemen Industri.

### Kode Program :

```

<div class="container">
<div class="nav navbar-nav navbar-right">

```

```

    <form class="navbar-form navbar-left" role="search" action="<?php echo
site_url('supplier_c/cari');?>" method="post">
    <div class="form-group">
        <label>Search Supplier (ID/Name)</label>
        <input type="text" class="form-control" placeholder="Search"
name="cari" required>
    </div>
    <button type="submit" class="btn btn-default"><i class="glyphicon
glyphicon-search"></i> Search</button>
    </form>
</div>
<a href="<?php echo site_url('supplier_c/tambah');?>" class="btn btn-
primary"><i class="glyphicon glyphicon-plus"></i> Add Data</a>
<hr>
<?php echo $message;?>
<div class="dataTable_wrapper">
<Table class="table table-striped table-bordered table-hover">
    <thead>
        <tr class="bg-success">
            <th>No.</th>
            <th>ID Supplier</th>
            <th>Supplier Name</th>
            <th>Address</th>
            <th>Phone</th>
            <th colspan=2>Action</th>
        </tr>
    </thead>
    <?php $no=0; foreach($supplier as $row ): $no++;?>
    <tr class="odd gradeX">
        <td><?php echo $no;?></td>
            <td><?php echo $row->id_supplier;?></td>

```

```

        <td><?php echo $row->supplier;?></td>
        <td><?php echo $row->address;?></td>
        <td><?php echo $row->phone;?></td>
        <td><a href="<?php echo site_url('supplier_c/edit/'. $row-
>id_supplier);?>"<i class="glyphicon glyphicon-edit"></i></a></td>
        <td><a href="#" class="hapus" kode="<?php echo $row-
>id_supplier;?>"><i class="glyphicon glyphicon-trash"></i></a></td>
    </tr>
<?php endforeach;?>
</Table>
</div>
<div style="margin-bottom:60px">
<?php echo $pagination;?>
</div>
<script>
$(function(){
    $(".hapus").click(function(){
        var kode=$(this).attr("kode");

        $("#idhapus").val(kode);
        $("#myModal").modal("show");
    });

    $("#konfirmasi").click(function(){
        var kode=$("#idhapus").val();

        $.ajax({
            url:"<?php echo site_url('supplier_c/hapus');?>",
            type:"POST",
            data:"kode="+kode,
            cache:false,

```

```

        success:function(html){
            location.href="<?php
            site_url('supplier_c/index/delete_success');?>";
        }
    });
});
});
</script>

```

### 3. Observation

The screenshot shows the Procurement Control System (PCS) interface. At the top, there is a navigation bar with the PCS logo and menu items: Home, Master Data, Order, Observation, and Settlement. A user profile for 'admin' is visible in the top right. Below the navigation bar, there is a search bar with the text 'Search ID / Asset Description' and a search button. A 'Plant' dropdown menu and a 'Filter' button are also present. The main content area displays a table with the following data:

No.	ID Asset	Asset Description	Plant	Location	Asset Controller	Urgency Level	Action
1	201111124564	asli	Plant II - Pegangaan	Pegangaan Utara Off & Part Prep Lt 3	Admin	0.867	Wait Approval
2	201111124561	Indah	Plant III - Cikarang	Cikarang Barat Injection Lantai 1	Admin	0.818	Observe
3	201111124554	Susi	Plant III - Cikarang	Cikarang Barat Injection Lantai 1	Admin	0.800	Observe
4	201111124558	Rinda	Plant I - Sunter	Sunter Tengah Plastic Injection Lantai 2	Admin	0.800	Observe
5	201111124557	Garing	Plant I - Sunter	Sunter Tengah Plastic Injection Lantai 2	Admin	0.687	Wait Approval

At the bottom of the interface, there is a copyright notice: © Copyright by Rryan Alfian, Sekolah Tinggi Manajemen Industri.

### Kode Program :

```

<div class="container">
<div class="nav navbar-nav navbar-right">
    <form class="navbar-form navbar-left" role="search" action="<?php echo
    site_url('supplier_c/cari');?>" method="post">
        <div class="form-group">
            <label>Search Supplier (ID/Name)</label>
            <input type="text" class="form-control" placeholder="Search"
            name="cari" required>
        </div>

```

```

        <button type="submit" class="btn btn-default"><i class="glyphicon
glyphicon-search"></i> Search</button>
    </form>
</div>
<a href="<?php echo site_url('supplier_c/tambah');?>" class="btn btn-
primary"><i class="glyphicon glyphicon-plus"></i> Add Data</a>
<hr>
<?php echo $message;?>
<div class="dataTable_wrapper">
<Table class="table table-striped table-bordered table-hover">
    <thead>
        <tr class="bg-success">
            <th>No.</th>
            <th>ID Supplier</th>
            <th>Supplier Name</th>
            <th>Address</th>
            <th>Phone</th>
            <th colspan=2>Action</th>
        </tr>
    </thead>
    <?php $no=0; foreach($supplier as $row ): $no++;?>
    <tr class="odd gradeX">
        <td><?php echo $no;?></td>
            <td><?php echo $row->id_supplier;?></td>
            <td><?php echo $row->supplier;?></td>
            <td><?php echo $row->address;?></td>
            <td><?php echo $row->phone;?></td>
            <td><a href="<?php echo site_url('supplier_c/edit/'.$row-
>id_supplier);?>"<i class="glyphicon glyphicon-edit"></i></a></td>
            <td><a href="#" class="hapus" kode="<?php echo $row-
>id_supplier;?>"><i class="glyphicon glyphicon-trash"></i></a></td>

```

```

        </tr>
        <?php endforeach;?>
</Table>
</div>
<div style=" margin-bottom:60px">
<?php echo $pagination;?>
</div>
<script>
    $(function(){
        $(".hapus").click(function(){
            var kode=$(this).attr("kode");

            $("#idhapus").val(kode);
            $("#myModal").modal("show");
        });

        $("#konfirmasi").click(function(){
            var kode=$("#idhapus").val();

            $.ajax({
                url:"<?php echo site_url('supplier_c/hapus');?>",
                type:"POST",
                data:"kode="+kode,
                cache:false,
                success:function(html){
                    location.href="<?php
site_url('supplier_c/index/delete_success');?>";
                }
            });
        });
    });
</script>

```