

**RANCANG BANGUN SISTEM INFORMASI PENERIMAAN
BAHAN BAKU MENGGUNAKAN PHP 5.4.7 DAN MYSQL
5.5.27 PADA PT GALIH AYOM PARAMESTI**

TUGAS AKHIR

Diajukan Untuk Memenuhi Sebagian Syarat Penyelesaian Program Sarjana
Terapan Pada Program Studi Sistem Informasi Industri Otomotif
Politeknik STMI Jakarta

OLEH

FELLY HEDRIYANI LAILI

1314077



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

**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA**

TANDA PERSETUJUAN DOSEN PEMBIMBING

JUDUL TUGAS AKHIR :

**RANCANG BANGUN SISTEM INFORMASI PENERIMAAN
BAHAN BAKU MENGGUNAKAN PHP 5.4.7 DAN MYSQL
5.5.27 PADA PT GALIH AYOM PARAMESTI**

DISUSUN OLEH :

NAMA : FELLY HEDRIYANI LAILI

NIM : 1314077

PROGRAM STUDI : SISTEM INFORMASI INDUSTRI OTOMOTIF

Telah diperiksa dan disetujui untuk diajukan dan dipertahankan dalam Ujian Tugas Akhir Program Sarjana Terapan pada Program Studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

Jakarta, 21 Agustus 2018

Dosen Pembimbing

(Ahmad Juniar, S.Kom, MT)
NIP. 197906052006041002

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Felly Hedriyani Laili

NIM :1314077

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

“RANCANG BANGUN SISTEM INFORMASI PENERIMAAN BAHAN BAKU MENGGUNAKAN PHP 5.4.7 DAN MYSQL 5.5.27 PADA PT GALIH AYOM PARAMESTI”

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

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

Jakarta, 21 Agustus 2018

Felly Hedriyani Laili

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Felly Hedriyani Laili

NIM : 1314077

Judul Tugas Akhir : Rancang Bangun Sistem Informasi Penerimaan Bahan Baku Menggunakan PHP 5.4.7 dan MySQL 5.5.27 Pada PT Galih Ayom Paramesti

Pembimbing : Ahmad Juniar, S.Kom, MT

Tanggal	Keterangan	Paraf
16 Mei 2018	BAB I Pendahuluan	
18 Mei 2018	BAB II Landasan Teori	
30 Mei 2018	Bimbingan BAB III	
8 Juni 2018	Bimbingan BAB IV Pengumpulan & Pengolahan	
9 Juni 2018	Data	
7 Agustus 2018	Revisi BAB IV	
9 Agustus 2018	Bimbingan BAB V	
13 Agustus 2018	Revisi BAB V	
15 Agustus 2018	Bimbingan BAB VI, Revisi BAB V	
16 Agustus 2018	Bimbingan BAB VI, Daftar Pustaka dan Lampiran	

Mengetahui,

Ketua Program Studi
Pembimbing
Sistem Informasi Industri Otomotif

Dosen

Drs. Jacob Saragih, M.M.
S.Kom, MT
NIP : 195404281986031002
197906052006041002

Ahmad Juniar,

NIP:

ABSTRAK

Sistem informasi merupakan hal yang penting dalam sebuah perusahaan. Dengan adanya sistem informasi, perusahaan dapat menjamin kualitas informasi, informasi dapat disajikan dengan baik, cepat dan mudah dipahami. PT Galih Ayom Paramesti adalah perusahaan yang bergerak di bidang otomotif yang kegiatannya adalah memproses bahan baku besi menjadi *spare part* kendaraan roda dua. Dalam melakukan aktivitas produksi, PT Galih Ayom Paramesti terbagi ke dalam beberapa unit. Kelancaran proses di masing-masing unit ini akan menentukan keberhasilan proses produksi. Salah satu unit yang dimiliki PT Galih Ayom Paramesti adalah *Warehouse* yang bertanggung jawab terhadap penerimaan bahan baku untuk menunjang kelangsungan produksi. Tersedianya kecukupan bahan baku adalah salah satu faktor penting guna menjaga kelancaran produksi. Di beberapa bagian proses pengolahan data pada *Warehouse* masih menggunakan cara manual, misalnya dalam memberikan informasi penerimaan bahan baku yang tersedia di gudang, dan peng-*inputan* bahan baku masuk, yaitu masih menggunakan Microsoft Excel. Sistem Informasi Penerimaan Bahan Baku ini dikembangkan dengan metodologi *prototype evolusioner*, dengan *tools* pemodelan sistem *Unified Modeling Language* (UML), pemodelan data dengan kamus data, dan alat bantu perancangan dengan *Windows Navigation Diagram* (WND) dan *flowchart*. Sedangkan untuk *engine* program menggunakan bahasa pemrograman PHP dan perangkat lunak sistem manajemen basis data menggunakan MySQL. Dengan dibuatnya aplikasi sistem informasi penerimaan bahan baku, pembuatan laporan penerimaan bahan baku menjadi lebih cepat dan mudah, stok bahan baku dapat lebih mudah dilihat secara langsung dengan aplikasi yang dibuat, dan pengecekan/pemeriksaan bahan baku dapat dilakukan dengan waktu yang lebih singkat.

Kata kunci : Sistem Informasi Penerimaan Bahan Baku, *prototype evolusioner*, *Unified Modeling Language* (UML), Kamus Data, *Windows Navigation Diagram* (WND), PHP, MySQL.

KATA PENGANTAR

Alhamdulillah robbil'alamin, segala puji syukur penulis panjatkan kehadiran Allah SWT yang telah melimpahkan nikmat-Nya, terutama nikmat iman, ilmu, dan sehat sehingga dapat terselesaikannya Tugas Akhir ini dengan baik dan lancar.

Penulisan Tugas Akhir ini disusun untuk diajukan dan dipertahankan dalam Ujian Tugas Akhir Program Sarjana Terapan pada Program Studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta dengan judul **“Rancang Bangun Sistem Informasi Penerimaan Bahan Baku Menggunakan PHP 5.4.7 Dan MySQL 5.5.27 Pada PT Galih Ayom Paramesti”**.

Dalam penulisan Tugas Akhir ini, penulis banyak mendapatkan bantuan, bimbingan, petunjuk dan dukungan dari berbagai pihak. Dengan ketulusan dan kerendahan hati, perkenankanlah penulis menyampaikan ucapan terima kasih yang kepada:

- Allah SWT yang telah memberikan rahmat dan karunia-Nya.
- Kedua orang tua yang selalu memberikan dukungan, kasih sayang serta do'a untuk keberhasilan Penulis.
- Bapak Dr. Mustofa, S.T., M.T., selaku Direktur Politeknik STMI Jakarta.
- Bapak Jacob Saragih, M.M., selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
- Bapak Ahmad Juniar, S.Kom, MT. selaku Dosen Pembimbing yang telah memberikan arahan, bimbingan serta penjelasan dalam penyusunan Tugas Akhir.
- Bapak Ignatius Sumardi, Bapak Yulius Handaru Pradipto, Bapak Adi Alfianto, dan Bapak Setyo Gunawan, serta seluruh pegawai di PT Galih Ayom Paramesti yang telah membantu memberi arahan serta informasi untuk pengumpulan data.
- Seluruh karyawan di PT Galih Ayom Paramesti khususnya pada *Warehouse* atas bantuan yang telah diberikan selama berlangsungnya kegiatan Praktik Kerja Lapangan.
- Seluruh dosen dan karyawan Politeknik STMI Jakarta yang telah memberikan ilmu dan waktu guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
- Kepada Teman-tem bgtran SIIO angkatan 2014 SA01 & SA02 yang telah memotivasi dan membantu dalam penyelesaian Tugas Akhir ini.

- Kepada Erlin Puspita Dewi, Rahmeilia Khairunnisa, Ishmah Afiifatul Afnan, dan Ivana Andria Wiraputri yang saling memberikan semangat satu sama lain, saling membantu dan saling memberikan dukungan agar tugas akhir ini dapat selesai bersama-sama.
- Kepada teman-teman mahasiswa/i seluruh angkatan prodi Sistem Informasi Industri Otomotif di Politeknik STMI Jakarta.
- Serta semua pihak baik langsung maupun tidak langsung memberikan kritik, saran, dan bantuan dalam pembuatan laporan ini yang tidak bisa disebutkan satu per satu.

Semoga Allah SWT membalas kebaikan dan ketulusan semua pihak yang telah membantu menyelesaikan laporan ini dengan melimpahkan rahmat dan karunia-Nya.

Dengan segala kemampuan dan keterbatasan, penulis menyadari segala kekurangan dalam penulisan, karena itu penulis sangat mengharapkan segala kritik atau saran yang dapat membangun dari semua pihak. Dan juga berharap penulisan ini dapat berguna bagi diri pribadi maupun pihak-pihak lain yang membacanya.

Jakarta, September 2018

Penulis

DAFTAR ISI

Halaman

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

LEMBAR BIMBINGAN TUGAS AKHIR

LEMBAR PERNYATAAN KEASLIAN

ABSTRAK

KATA PENGANTAR	i
DAFTAR ISI.....	iii
DAFTAR GAMBAR	vii
DAFTAR TABEL	ix
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pokok Permasalahan	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	3
1.5 Manfaat Tugas Akhir	3
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Pengertian Sistem	6
2.1.1 Karakteristik Sistem	7
2.1.2 Klasifikasi Sistem	10
2.1.3 Metodologi Pengembangan Sistem	11
2.1.4 <i>Prototype</i> Evolusioner	12
2.1.5 Langkah Pembuatan <i>Prototype</i> Evolusioner.....	12
2.1.6 Siklus Hidup Pengembangan Sistem	13
2.2 Pengertian Data	13
2.3 Pengertian Informasi	14
2.3.1 Siklus Informasi	14
2.3.1 Kualitas Informasi	15

2.4	Pengertian Sistem Informasi	16
2.4.1	Komponen Sistem Informasi	16
2.5	Pengertian Penerimaan	17
2.6	Konsep Dasar Bahan Baku	18
2.6.1	Jenis-jenis Bahan Baku	19
2.7	Pengertian Sistem Informasi Penerimaan Bahan Baku	19
2.8	Analisis Sistem	20
2.9	<i>Flowmap</i>	21
2.10	Jenis-jenis <i>Flowchart</i>	23
2.11	<i>Unified Modeling Language</i> (UML)	23
2.11.1	<i>Use Case Diagram</i>	25
2.11.2	<i>Activity Diagram</i>	26
2.11.3	<i>Sequence Diagram</i>	28
2.11.4	<i>Class Diagram</i>	30
2.11.5	<i>Deployment Diagram</i>	31
2.12	<i>Entity Relationship Diagram</i> (ERD)	32
2.13	Conceptual Data Model (CDM)	33
2.14	Kamus Data	34
2.15	<i>Windows Navigation Diagram</i> (WND)	35
2.16	<i>Personal Home Page</i> (PHP)	37
2.17	MySQL	38
BAB III	METODOLOGI PENELITIAN.....	40
3.1	Jenis dan Sumber Data	40
3.2	Metode Pengumpulan Data	41
3.3	Instrumen Pengumpulan Data	41
3.4	Metode Pengembangan Sistem.....	42
3.5	Kerangka Penelitian.....	43
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA	47
4.1	Sejarah Perusahaan	47
4.2	Makna Logo Perusahaan.....	48

4.3	Profil Perusahaan	48
4.4	Visi dan Misi PT Galih Ayom Paramesti	49
4.5	Struktur Organisasi Perusahaan	49
4.5.1	Tugas dan Wewenang Jabatan di PT GAP	51
4.6	Bahan Baku	54
4.7	Dokumen Penerimaan Bahan Baku	57
4.8	Proses Penerimaan Bahan Baku	59
BAB V ANALISIS DAN PEMBAHASAN		61
5.1	Analisis Kebutuhan <i>Functional</i> Sistem	61
5.2	<i>Flowmap</i> Sistem Usulan	62
5.3	Pemodelan Sistem Informasi Penerimaan Bahan Baku.....	64
5.3.1	<i>Use Case Diagram</i>	64
5.3.2	<i>Activity Diagram</i>	70
5.3.3	<i>Sequence Diagram</i>	77
5.3.4	<i>Class Diagram</i>	82
5.3.5	<i>Deployment Diagram</i>	84
5.4	Pemodelan Data	84
5.4.1	<i>Entity Relationship Diagram</i> (ERD)	84
5.4.2	<i>Conceptual Data Model</i> (CDM)	86
5.5	Kamus Data	87
5.6	Analisis Desain Program	89
5.6.1	<i>Windows Navigation Diagram</i> (WND)	90
5.6.2	Rancangan <i>Interface</i> Sistem	91
5.4	Implementasi Sistem <i>Software</i> dan <i>Hardware</i>	97
BAB VI KESIMPULAN DAN SARAN		98
6.1	Kesimpulan	98
6.2	Saran	98
DAFTAR PUSTAKA		99
LAMPIRAN		

DAFTAR GAMBAR

	Halaman
Gambar II.1 Karakteristik Sistem	7
Gambar II.2 <i>Prototype</i> Evolusioner	12
Gambar II.3 Siklus Informasi.....	15
Gambar II.4 Contoh WND	36
Gambar III.1 Kerangka Penelitian	44
Gambar IV.1 Logo GAP	48
Gambar IV.2 Struktur Organisasi Keseluruhan	50
Gambar IV.3 Bahan Baku Mentah.....	54
Gambar IV.4 Bahan Baku Mentah	55
Gambar IV.5 Bahan Baku Setengah Jadi	55
Gambar IV.6 Bahan Baku Setengah Jadi	56
Gambar IV.7 Bahan Baku Penolong.....	56
Gambar IV.8 <i>Purchase Order</i>	57
Gambar IV.9 Surat Jalan	58
Gambar IV.10 <i>Flowchart</i> Yang Sedang Berjalan di <i>Warehouse</i>	60
Gambar V.1 <i>Flowmap</i> Usulan Sistem Penerimaan Bahan Baku	63
Gambar V.2 <i>Use Case Diagram</i> Usulan Sistem Penerimaan Bahan Baku ..	64
Gambar V.3 <i>Activity Diagram Login</i>	71
Gambar V.4 <i>Activity Diagram</i> Mengelola Data Master.....	72
Gambar V.5 <i>Activity Diagram</i> Pemeriksaan Bahan Baku	73
Gambar V.6 <i>Activity Diagram</i> Menginput Bahan Baku Masuk	74
Gambar V.7 <i>Activity Diagram</i> Menginput Data NG	75
Gambar V.8 <i>Activity Diagram</i> Mencetak Laporan Penerimaan	76
Gambar V.9 <i>Activity Diagram</i> Mencetak Laporan Pemeriksaan.....	76
Gambar V.10 <i>Sequence Diagram Login</i>	77
Gambar V.11 <i>Sequence Diagram</i> Mengelola Data Master	78

Gambar V.12 <i>Sequence Diagram</i> Pemeriksaan Bahan Baku	79
Gambar V.13 <i>Sequence Diagram</i> Menginput Bahan Baku Masuk	80
Gambar V.14 <i>Sequence Diagram</i> Menginput Data NG	81
Gambar V.15 <i>Sequence Diagram</i> Mencetak Laporan	82
Gambar V.16 <i>Class Diagram</i> Sistem Informasi Penerimaan Bahan Baku.....	83
Gambar V.17 <i>Deployment Diagram</i> Sistem Penerimaan BB Usulan.....	84
Gambar V.18 ERD Konseptual Sistem Informasi Penerimaan Bahan Baku..	85
Gambar V.19 CDM Sistem Informasi Penerimaan Bahan Baku.....	86
Gambar V.20 WND Sistem Informasi Penerimaan Bahan Baku	90
Gambar V.21 <i>Interface Login</i>	91
Gambar V.22 <i>Form Menu Utama</i>	91
Gambar V.23 <i>Interface Form Data Master Bahan Baku</i>	92
Gambar V.24 <i>Interface Form Data Master Supplier</i>	92
Gambar V.25 <i>Interface Form Data Master User</i>	93
Gambar V.26 <i>Interface Form</i> Pemeriksaan Bahan Baku.....	93
Gambar V.27 <i>Interface Form</i> Bahan Baku Masuk	94
Gambar V.28 <i>Interface Form</i> Laporan Penerimaan Bahan Baku.....	95
Gambar V.29 <i>Interface Form</i> Laporan Hasil Pemeriksaan Bahan Baku.....	95
Gambar V.30 <i>Interface Form</i> Data NG	96
Gambar V.31 <i>Interface Form</i> Laporan Data NG	96

DAFTAR TABEL

	Halaman
Tabel II.1 Simbol- Simbol <i>Flowmap</i>	22
Tabel II.2 Jenis Diagram Pada UML dan Fungsi-fungsinya.....	24
Tabel II.3 Simbol- Simbol <i>Use Case Diagram</i>	25
Tabel II.4 Simbol- Simbol <i>Activity Diagram</i>	27
Tabel II.5 Simbol- Simbol <i>Sequence Diagram</i>	29
Tabel II.6 Simbol- Simbol <i>Class Diagram</i>	31
Tabel II.7 Simbol- Simbol <i>Deployment Diagram</i>	32
Tabel II.8 Simbol- Simbol <i>Entity Relationship Diagram</i>	33
Tabel II.9 Simbol- Simbol <i>Conceptual Data Model</i>	34
Tabel II.10 Contoh Kamus Data Untuk Tabel Pemasok.....	35
Tabel V.1 Kebutuhan <i>Functional</i> Sistem.....	61
Tabel V.2 Definisi Aktor <i>Use Case Diagram</i> Sistem Usulan.....	65
Tabel V.3 Definisi <i>Use Case Login</i>	65

Tabel V.4	Definisi <i>Use Case</i> Menginput Data Pemeriksaan Bahan Baku ..	66
Tabel V.5	Definisi <i>Use Case</i> Mengelola Data <i>Master</i>	66
Tabel V.6	Definisi <i>Use Case</i> Menginput Bahan Baku Masuk.....	67
Tabel V.7	Definisi <i>Use Case</i> Menginput Data NG.....	68
Tabel V.8	Definisi <i>Use Case</i> Mencetak Laporan Penerimaan.....	68
Tabel V.9	Definisi <i>Use Case</i> Mencetak Laporan Hasil Pemeriksaan.....	69
Tabel V.10	Definisi <i>Use Case</i> Mencetak Label Bahan Baku	69
Tabel V.11	Definisi <i>Use Case</i> Mencetak Data NG.....	70
Tabel V.12	Tabel <i>User</i>	87
Tabel V.13	Tabel Bahan Baku	87
Tabel V.14	Tabel <i>Supplier</i>	88
Tabel V.15	Tabel Bahan Baku Masuk	88
Tabel V.16	Tabel Pemeriksaan Bahan Baku.....	89
Tabel V.17	Tabel Data NG	89

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi pada saat ini berkembang sangat pesat, hal tersebut dapat dilihat dari banyaknya perusahaan-perusahaan baik itu perusahaan dagang maupun perusahaan manufaktur yang mempergunakan peran teknologi informasi sebagai sarana untuk mempermudah dalam proses pengambilan keputusan. Teknologi informasi pada sebuah perusahaan mempunyai peranan penting, hal ini digunakan agar sebuah perusahaan dapat tetap bertahan dari semakin ketatnya persaingan antar perusahaan. Saat ini perusahaan yang mampu bertahan dalam menghadapi persaingan adalah perusahaan yang mampu memanfaatkan peran teknologi informasi.

Dengan adanya persaingan yang semakin ketat, maka dibutuhkan sikap profesional manajemen dalam mengelola perusahaan, salah satunya dengan pengelolaan penerimaan bahan baku. Pengelolaan penerimaan bahan baku merupakan aktivitas kerja yang sangat penting dalam perusahaan manufaktur. Apabila terjadi kesalahan informasi penerimaan bahan baku di gudang maka sangat berpengaruh pada kegiatan perusahaan selanjutnya.

PT Galih Ayom Paramesti adalah perusahaan manufaktur yang bergerak di bidang otomotif. Salah satu kegiatan yang ada didalam perusahaan tersebut adalah kegiatan penerimaan bahan baku, setiap bulannya PT Galih Ayom Paramesti melakukan pembelian bahan baku salah satunya besi untuk diolah menjadi *spare parts* kendaraan roda dua.

Penerimaan bahan baku sangat penting untuk PT Galih Ayom Paramesti, karena jika terjadi kesalahan dalam perhitungan transaksi bahan baku masuk dan stok bahan baku maka proses produksi akan terhambat. *Warehouse* yang bertanggung jawab terhadap penerimaan bahan baku. Bahan baku yang dipakai dalam memproduksi *spare parts* terbagi menjadi tiga jenis yaitu bahan baku

mentah, bahan baku setengah jadi, dan bahan baku penolong. Bahan baku yang digunakan adalah besi.

Dalam menjalankan kegiatannya, penerapan sistem informasi pengolahan data pada *warehouse* masih menggunakan Microsoft Excel sehingga pengolahan data menjadi lambat dan hasil laporannya masih kurang efisien. Diantaranya mengenai pembuatan laporan penerimaan bahan baku dan pemberitahuan jumlah persediaan bahan baku yang dilakukan dengan pengecekan secara langsung di lapangan. Hal-hal tersebut kurang efisien dalam segi waktu.

Untuk mengatasi masalah tersebut, diperlukan aplikasi untuk membantu perusahaan dalam mengatasi masalah tersebut. Adapun judul Tugas Akhir ini adalah “RANCANG BANGUN SISTEM INFORMASI PENERIMAAN BAHAN BAKU MENGGUNAKAN PHP 5.4.7 DAN MYSQL 5.5.27 PADA PT GALIH AYOM PARAMESTI”

1.2 Pokok Permasalahan

Permasalahan yang terjadi pada PT Galih Ayom Paramesti adalah sebagai berikut:

1. Pembuatan laporan bahan baku masuk dan stok bahan baku dilakukan dengan ditulis pada buku dan papan tulis terlebih dahulu kemudian diinput melalui Microsoft Excel, sehingga dapat berakibat berkurangnya performa pada penginputan data penerimaan bahan baku dan untuk membuat laporan penerimaan bahan baku.
2. Belum ada *database* penerimaan bahan baku, dokumen di simpan di sembarang tempat yang disebabkan kurangnya tempat penyimpanan dokumen sehingga berisiko terjadinya kehilangan data, sulit mencari data dan kerusakan data.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah sebagai berikut:

1. Membantu bagian *warehouse* dalam mengolah data bahan baku dengan menggunakan aplikasi yang sudah terintegrasi.

2. Menyediakan fasilitas penyimpanan data dokumen-dokumen bahan baku dan laporan penerimaan bahan baku dengan menggunakan *database*, mencegah terjadinya kehilangan data, kesulitan mencari data dan kerusakan data.

1.4 Batasan Masalah

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

1. Penelitian dilakukan di *warehouse* pada PT Galih Ayom Paramesti
2. Penelitian dilakukan hanya sebatas menangani masalah pengolahan data bahan baku, pendataan penerimaan bahan baku dan laporan penerimaan bahan baku.

1.5 Manfaat Tugas Akhir

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

1. Bagi penulis
 - a. Untuk memberikan pengalaman dalam menganalisis suatu sistem dan diharapkan dapat memberikan suatu solusi permasalahan.
 - b. Untuk memberikan kemampuan dalam mengaplikasikan teori tentang pengembangan sistem informasi sistem secara jelas untuk memberikan solusi terhadap masalah yang diamati.
2. Bagi institusi pendidikan
 - a. Menjalani kerja sama antara dunia pendidikan dan dunia kerja dalam rangka pengembangan ilmu.
 - b. Untuk menghasilkan lulusan yang memiliki keterampilan sesuai dengan kebutuhan dunia kerja.
3. Bagi perusahaan
 - a. Memberikan partisipasi dalam pengembangan di dunia pendidikan.
 - b. Mendapatkan saran yang diharapkan mampu menyelesaikan permasalahan yang terjadi pada perusahaan.

1.6 Sistematika Penulisan

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

BAB I PENDAHULUAN

Bab ini memuat latar belakang, rumusan permasalahan, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan laporan.

BAB II LANDASAN TEORI

Bab ini membahas tentang pengertian sistem, informasi, sistem informasi, sistem informasi penerimaan bahan baku, analisis sistem, metode pengembangan sistem, *flowchart*, *flowmap*, *Unified Modelling Language* (UML), *Hypertext Preprocessor* (PHP), dan MySQL sebagai alat bantu untuk membuat rancangan konseptual, dan teori-teori lain yang berhubungan dengan perancangan sistem.

BAB III METODOLOGI PENELITIAN

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

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini akan menguraikan tentang hasil pengamatan dalam penelitian yang telah dilakukan, seperti fungsi sistem persediaan bahan baku, alur proses sistem persediaan bahan baku, dokumen-dokumen yang terlibat dalam persediaan bahan baku dan transaksi-transaksi yang terjadi di sistem persediaan bahan baku.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari identifikasi kebutuhan sistem, analisis sistem yang meliputi diagram alir sistem yang berjalan, perancangan sistem dengan menggunakan diagram UML, perancangan basis data, perancangan tampilan layar, perancangan hierarki menu, pembuatan *flowchart* program, dan spesifikasi kebutuhan *software* dan *hardware* untuk mengimplementasi sistem yang dibangun.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan secara keseluruhan yang diperoleh dari hasil analisis dan perancangan sistem serta saran-saran untuk penerapan dan pengembangan lebih lanjut pada sistem yang bersangkutan.

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Terdapat dua kelompok pendekatan di dalam pendefinisian sistem, yaitu kelompok yang menekankan pada prosedur dan kelompok yang menekankan pada elemen atau komponennya. Pendekatan yang menekankan pada prosedur mendefinisikan sistem sebagai suatu jaringan kerja prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pendekatan sistem yang lebih menekankan pada elemen atau komponen mendefinisikan sistem sebagai kumpulan dari elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Kedua kelompok definisi ini adalah benar dan tidak bertentangan. Yang berbeda adalah cara pendekatannya (Sutabri, 2012).

Kata sistem sendiri dari bahasa Latin “*Systema*” dan bahasa Yunani “*Sustema*” adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Berikut pengertian sistem menurut beberapa ahli dari beberapa sumber referensi, mendefinisikan bahwa sistem:

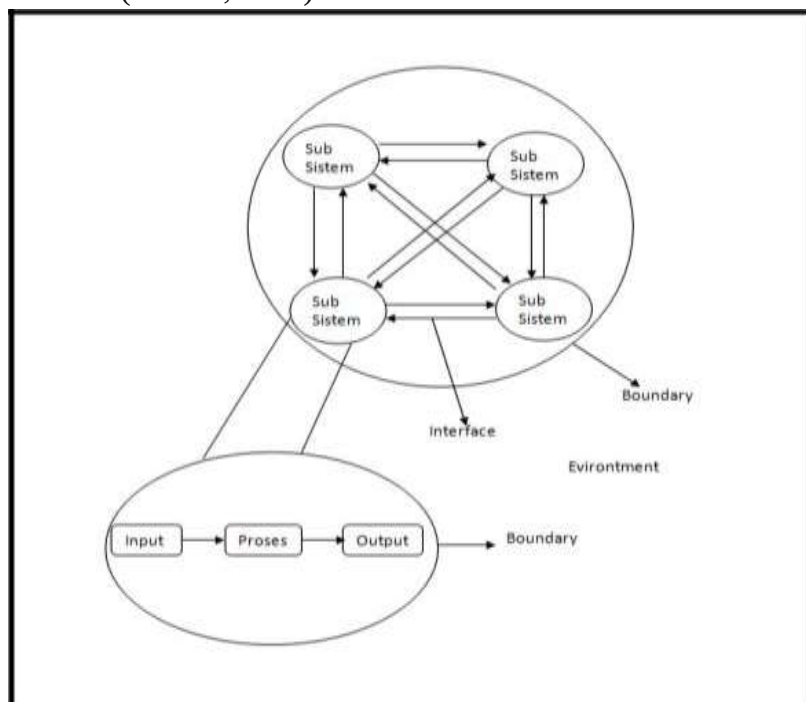
1. Sistem sebagai kumpulan atau *group* dari bagian atau komponen apapun baik fisik maupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu (Taufiq, 2013).
2. Sistem merupakan suatu kesatuan komponen atau elemen yang saling berhubungan, saling terkait dan

saling ketergantungan satu dengan yang lainnya sehingga membentuk satu kesatuan yang kompleks (Kadir, 2014).

3. Sistem sebagai sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan (Zakiyudin, 2012).

2.1.1 Karakteristik Sistem

Model umum sebuah sistem terdiri dari *input*, proses, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana mengingat sebuah sistem dapat mempunyai beberapa masukan (*input*) dan keluaran (*output*) sekaligus. Selain itu sebuah sistem juga memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud dapat dilihat seperti pada Gambar II.1 berikut (Sutabri, 2012):



Gambar II.1 Karakteristik Sistem

Sumber: Sutabri (2012)

1. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar yang disebut dengan supra sistem.

2. Batasan Sistem (*Boundary*)

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

3. Lingkungan Luar Sistem (*Environtment*)

Lingkungan luar (*environment*) dari suatu sistem adalah bentuk apapun di luar ruang lingkup atau batas sistem yang mempengaruhi operasi sistem tersebut. Lingkungan luar sistem ini dapat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus

selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan, karena jika tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem (*Interface*)

Penghubung (*interface*) merupakan media penghubung antara sistem dengan subsistem yang lainnya. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

5. Masukan Sistem (*Input*)

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer. Sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi informasi yang berguna. Keluaran ini dapat merupakan masukan bagi subsistem lainnya.

7. Pengolahan Sistem (*Procces*)

Pengolah sistem merupakan suatu bagian yang mengolah masukan (*input*) dan memprosesnya agar menjadi *output* informasi yang berguna.

8. Sasaran Sistem (*Objective*)

Suatu sistem mempunyai tujuan dan sasaran yang pasti bersifat deterministik. Jika suatu sistem tidak mempunyai sasaran maka operasi sistem tidak akan berguna. Sasaran dari sistem sangat menentukan masukan yang dibutuhkan sistem dan keluaran yang dihasilkan sistem. Suatu sistem dikatakan berhasil apabila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.2 Klasifikasi Sistem

Sistem dapat diklasifikasikan menjadi beberapa jenis, di antaranya sebagai berikut (Zakiyudin, 2012):

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak (*abstract system*) adalah sistem yang berupa gagasan atau konsep-konsep yang tidak tampak secara fisik, misalnya sistem teologia atau keagamaan, yaitu suatu sistem yang mengatur hubungan manusia dengan Tuhannya. Sedangkan sistem fisik (*physical system*) merupakan sistem yang secara fisik dapat dilihat, misalnya sistem komputer, sistem akuntansi, sistem produksi, dll.

2. Sistem Deterministik dan Sistem Probabilistik

Sistem deterministik (*deterministic system*) adalah sistem yang operasinya dapat diprediksi secara tepat. Contohnya sistem komputer, dimana pada sistem tersebut kita dapat memberikan *input* sesuai dengan tujuan *output* tertentu.

Sedangkan sistem probabilistik (*probabilistic system*) adalah sistem yang tidak dapat diprediksi atau diramal dengan pasti karena mengandung unsur probabilitas atau kemungkinan-kemungkinan.

3. Sistem Tertutup dan Sistem Terbuka

Sistem tertutup (*close system*) adalah sistem yang tidak berhubungan dan tidak dipengaruhi oleh lingkungan luarnya, dengan kata lain sistem yang tidak bertukar materi, informasi atau energi dengan lingkungannya. Sedangkan sistem terbuka (*open system*) merupakan sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan yang diketahui, yang bersifat acak maupun gangguan.

4. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah (*natural system*) merupakan sistem yang terjadi secara alamiah tanpa campur tangan manusia, seperti sistem tata surya. Sedangkan sistem buatan manusia (*human-machine system*) merupakan sistem yang dibuat oleh manusia, seperti sistem komputer, sistem telekomunikasi dan lain-lain.

5. Sistem Sederhana dan Sistem Kompleks

Sistem sederhana adalah sistem yang tidak rumit atau sistem dengan tingkat kerumitan rendah. Sedangkan sistem kompleks adalah sistem yang rumit, seperti sistem otak manusia. Sistem informasi dapat tergolong sebagai sistem yang kompleks atau sederhana tergantung pada implementasinya.

2.1.3 Metodologi Pengembangan Sistem

Metodologi adalah kesatuan metode-metode, prosedur-prosedur, konsep-konsep pekerjaan, dan aturan-aturan yang digunakan oleh suatu ilmu pengetahuan, seni atau disiplin ilmu lainnya. Sedangkan metode adalah suatu cara, teknik yang sistematis untuk mengerjakan sesuatu. Metodologi pengembangan sistem berarti metode-metode, prosedur-prosedur, konsep-konsep pekerjaan, aturan-aturan yang akan digunakan untuk mengembangkan suatu sistem informasi. Dalam pengembangan sistem informasi, perlu digunakan suatu metodologi yang dapat digunakan sebagai pedoman bagaimana dan apa yang harus dikerjakan selama pengembangan ini. Dengan mengikuti metode atau prosedur-prosedur yang diberikan oleh suatu metodologi, maka pengembangan sistem diharapkan akan dapat diselesaikan dengan berhasil. Urutan-urutan prosedur untuk pemecahan masalah ini dikenal dengan istilah algoritma (*algorithm*) (Sutabri, 2012).

2.1.4 *Prototype* Evolusioner

Prototype evolusioner (*evolutionary prototype*) terus-menerus disempurnakan sampai memiliki seluruh fungsionalitas yang dibutuhkan pengguna dari sistem yang baru. *Prototype* ini kemudian dilanjutkan produksi. Jadi satu *prototype* evolusioner akan menjadi sistem aktual.

2.1.5 Langkah Pembuatan *Prototype* Evolusioner

Menurut Darmawan dan Fauzi (2013), empat langkah dalam pembuatan suatu *prototype* evolusioner, yaitu:

1. Mengidentifikasi kebutuhan pengguna. Pengembang mewawancarai pengguna untuk mendapatkan ide mengenai apa yang diminta dari sistem.
2. Membuat satu *prototype*. Pengembang mempergunakan satu alat prototyping atau lebih untuk membuat *prototype*.
3. Menentukan apakah *prototype* dapat diterima, pengembang mendemonstrasikan *prototype* kepada para pengguna untuk mengetahui apakah telah memberikan hasil yang memuaskan, jika sudah, langkah keempat akan diambil; jika tidak, *prototype* direvisi dengan mengulang langkah satu, dua dan tiga dengan pemahaman yang lebih baik mengenai kebutuhan pengguna.
4. Menggunakan *prototype*, *prototype* menjadi sistem produksi.

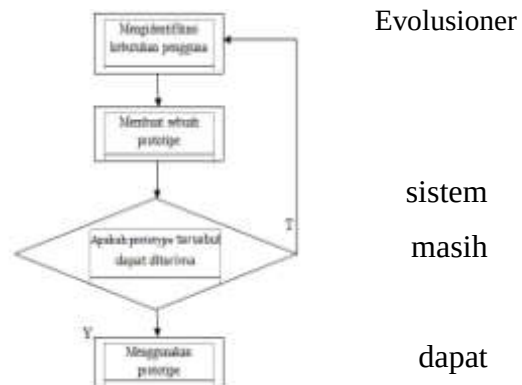
Pendekatan ini mungkin untuk dilakukan hanya ketika alat-alat *prototyping* memungkinkan *prototype* untuk memiliki seluruh unsur yang penting dari sistem yang baru.

Gambar II.2 *Prototype*

2.1.6 Siklus Hidup Pengembangan Sistem

Bila dalam operasi yang sudah dikembangkan timbul permasalahan-permasalahan yang tidak diatasi dalam tahap

pemeliharaan sistem, maka perlu dikembangkan kembali suatu



sistem untuk mengatasinya dan proses ini kembali ke proses yang pertama. Siklus ini disebut dengan siklus hidup suatu sistem.

Siklus hidup pengembangan sistem dapat didefinisikan sebagai serangkaian aktivitas yang dilaksanakan oleh profesional dan pemakai sistem informasi untuk mengembangkan dan mengimplementasikan sistem informasi. Siklus hidup pengembangan sistem informasi saat ini terbagi atas tujuh fase, yaitu (Sutabri, 2012):

1. Perencanaan Sistem
2. Analisis Sistem
3. Perancangan Sistem secara Umum/ Konseptual
4. Evaluasi dan Seleksi Sistem
5. Perancangan Sistem secara Detail
6. Pengembangan Perangkat Lunak dan Implementasi Sistem
7. Pemeliharaan/ Perawatan Sistem

2.2 Pengertian Data

Data adalah kenyataan yang menggambarkan suatu kejadian dan merupakan kesatuan nyata yang nantinya akan digunakan sebagai bahan dasar suatu informasi (Sutabri, 2012). Data adalah keterangan mengenai sesuatu hal yang sudah sering terjadi dan berupa himpunan fakta, angka, grafik, tabel, gambar, lambang, kata, huruf-huruf yang menyatakan sesuatu pemikiran, objek, serta kondisi dan situasi (Kadir, 2014).

2.3 Pengertian Informasi

Informasi adalah data yang telah diolah, sehingga lebih berguna, berarti, dan dapat dijadikan dasar untuk mengambil keputusan yang tepat

oleh orang yang menerimanya (Raharjo, 2014). Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan. Keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang. Nilai informasi sangat penting dalam konteks pengambilan keputusan.

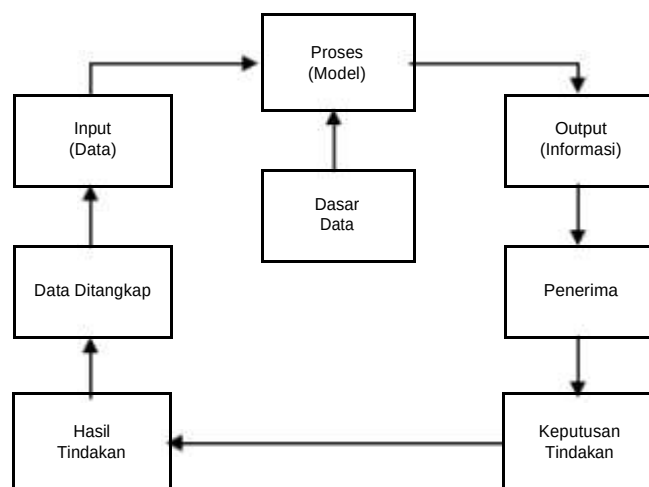
2.3.1 Siklus Informasi

Menurut Sutabri (2012), data yang diolah untuk menghasilkan informasi menggunakan suatu model proses tertentu. Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali.

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu. Di dalam dunia bisnis, kejadian-kejadian yang sering terjadi adalah transaksi perubahan dari suatu nilai yang disebut transaksi. Kesatuan nyata adalah berupa suatu obyek nyata seperti tempat, benda dan orang yang betul-betul ada dan terjadi.

Data merupakan bentuk yang masih mentah, belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu metode untuk menghasilkan informasi. Data dapat berbentuk simbol-simbol semacam huruf, angka, bentuk suara, sinyal, gambar, dan sebagainya.

Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, 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, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus informasi dapat dijelaskan pada Gambar II.1.



Gambar II.3 Siklus Informasi

(Sumber: Sutabri, 2012)

2.3.2 Kualitas Informasi

Kualitas informasi sangat dipengaruhi oleh 3 hal, yaitu (Sutabri, 2012):

1. Relevan (*Relevancy*)

Relevan berarti informasi harus memberikan manfaat bagi pemakainya. Relevansi informasi berbeda-beda untuk tiap-tiap orang satu dengan yang lainnya.

2. Akurat (*Accuracy*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak bisa menyesatkan dan harus jelas mencerminkan maksudnya. Ketidakakuratan dapat terjadi karena sumber informasi (data) mengalami gangguan atau kesengajaan sehingga merusak atau merubah data-data asli tersebut. Komponen akurat: lengkap (*completeness*), benar (*correctness*), dan aman (*security*).

c. Tepat Waktu (*timeliness*)

Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat. Informasi yang terlambat tidak mempunyai nilai yang baik, sehingga kalau digunakan sebagai dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan dalam keputusan dan tindakan. Kondisi demikian menyebabkan mahalnya suatu nilai informasi, sehingga kecepatan untuk mendapatkan, mengolah, dan mengirimkannya memerlukan teknologi-teknologi terbaru.

2.4 Pengertian Sistem Informasi

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 (Sutabri, 2012).

2.4.1 Komponen Sistem Informasi

Dalam membangun suatu sistem informasi diperlukan penggabungan elemen-elemen pendukung tersebut antara lain (Sutabri, 2012):

1. *Software*, merupakan suatu program komputer, struktur data, dan dokumen-dokumen yang saling berhubungan yang digunakan dalam metode logika dan prosedur yang dibutuhkan.
2. *Hardware*, merupakan perangkat elektronik yang memiliki kemampuan untuk melakukan proses komputerisasi.
3. *User*, adalah pengguna dan operator perangkat keras atau perangkat lunak.
4. *Data*, berupa salinan-salinan manual dan deskripsi informasi yang menggambarkan operasi sistem.

2.5 Pengertian Penerimaan

Transaksi pembelian lokal bahan baku melibatkan bagian-bagian produksi, gudang, pembelian, penerimaan barang, dan asuransi. Dokumen sumber dan dokumen pendukung yang dibuat dalam transaksi pembelian lokal bahan baku terdiri dari prosedur permintaan pembelian lain, prosedur order pembelian, prosedur penerimaan barang digudang (Mulyadi, 2015).

1. Prosedur Penerimaan Pembelian Bahan Baku

Jika persediaan bahan baku yang ada digudang sudah mencapai tingkat minimum pemesanan kembali, bagian gudang kemudian membuat surat permintaan pembelian untuk dikirim untuk dikirim ke bagian pembelian.

2. Prosedur Order Pembelian

Bagian pembelian melaksanakan pembelian atas dasar surat permintaan pembelian dari bagian gudang. Untuk pemilihan pemasok, bagian pembelian mengirimkan surat permintaan penawaran harga kepada para pemasok, yang berisi permintaan informasi harga dan syarat-syarat pembelian dari masing-masing pemasok tersebut. Setelah pemasok yang dianggap baik dipilih, bagian pembelian kemudian membuat surat order pembelian untuk dikirim kepada pemasok yang dipilih.

3. Prosedur Penerimaan Bahan Baku

Pemasok mengirim bahan baku kepada perusahaan sesuai dengan surat order pembelian yang diterimanya. Bagian penerimaan bertugas menerima barang, memcocokkan kualitas, kuantitas, jenis serta spesifikasi bahan baku yang diterima sesuai dengan surat order pembelian.

4. Prosedur Pencatatan Penerimaan Bahan Baku di Bagian Gudang.

Bagian penerimaan menyerahkan bahan baku yang diterima dari pemasok ke bagian gudang. Bagian gudang menyimpan bahan baku tersebut dan mencatat jumlah bahan baku dalam kartu gudang. Kartu gudang ini digunakan oleh bagian gudang untuk mencatat mutasi tiap-tiap barang digudang.

2.6 Konsep Dasar Bahan Baku

Bahan baku merupakan bahan yang membentuk bagian menyeluruh produk jadi. Bahan baku yang diolah dalam perusahaan manufaktur dapat diperoleh dari pembelian lokal, *import* atau dari pengolahan sendiri (Mulyadi, 2015).

Menurut Assuari (2015), bahan baku meliputi semua bahan yang dipergunakan dalam perusahaan, kecuali terhadap bahan-bahan yang secara fisik akan digabungkan dengan produk yang dihasilkan oleh perusahaan tersebut. Bahan baku diperlukan oleh pabrik untuk diolah, yang setelah melalui beberapa proses diharapkan menjadi barang jadi (*finished goods*), contohnya benang diolah menjadi kain atau kaos, kapas dipintal menjadi benang. Contoh lainnya, kertas yang merupakan bahan baku bagi perusahaan percetakan, tetapi di sisi lain kertas juga merupakan barang jadi bagi suatu pabrik yang menghasilkan kertas.

Dalam perusahaan, bahan baku terdiri dari 2 yaitu bahan baku dan bahan baku penolong. Bahan baku memiliki definisi yaitu bahan-bahan yang merupakan komponen utama yang membentuk keseluruhan dari produk jadi. Sedangkan bahan baku penolong adalah bahan yang digunakan dalam proses produksi yang nilainya kecil dan tidak dapat diidentifikasi dalam produk jadi. Contohnya produk kaos olah raga, bahan bakunya kain kaos dan bahan penolongnya adalah benang (Sujarweni, 2015).

2.6.1 Jenis-Jenis Bahan Baku

Pada umumnya terdapat dua macam jenis bahan baku, yaitu (Ristono, 2013):

1. Bahan Baku Langsung (*Direct Material*)

Bahan baku langsung atau yang biasa disebut dengan istilah *direct material* yaitu bahan yang membentuk dan merupakan bagian dari barang jadi yang biayanya dengan mudah ditelusuri dari biaya barang jadi tersebut. Jumlah bahan baku langsung bersifat variabel, artinya sangat tergantung atau dipengaruhi oleh besar kecilnya volume produksi atau perubahan output. Contoh:

- Kain adalah bahan baku pada industri garmen atau pakaian jadi.
- Tepung terigu adalah bahan baku pada pabrik roti.

2. Bahan Baku Tak Langsung (*Indirect Material*)

Bahan baku tak langsung atau *indirect material* yaitu bahan baku yang dipakai dalam proses produksi, tetapi sulit ditelusuri biayanya pada setiap barang jadi. Contoh:

- Benang adalah bahan baku tak langsung yang digunakan pada industri garmen.
- Garam dan ragi adalah bahan baku tak langsung pada pembuatan roti.

2.7 Pengertian Sistem Informasi Penerimaan Bahan Baku

Sistem informasi penerimaan bahan baku adalah sistem informasi manajemen yang menyediakan informasi untuk digunakan oleh fungsi penerimaan bahan baku. Sistem informasi penerimaan bahan baku mendukung fungsi penerimaan yang meliputi semua aktivitas yang berkaitan dengan proses saat bahan baku datang dan bahan baku masuk ke gudang (Zakiyudin, 2012). Jadi, fungsi penerimaan bahan baku berkaitan dengan manajemen proses kedatangan bahan baku dari *supplier* sampai ke gudang. Sistem informasi digunakan untuk manajemen operasi dan pemrosesan transaksi pendukung semua perusahaan yang harus merencanakan, memonitor, dan mengendalikan persediaan, pembelian, serta arus barang dan jasa. Oleh karenanya, perusahaan-perusahaan seperti perusahaan transportasi, grosir, peritel, lembaga keuangan, dan perusahaan jasa harus menggunakan sistem informasi penerimaan bahan baku untuk mengatur kedatangan bahan baku mereka agar tidak terjadi kecurangan (Wilkinson, 2012).

Tujuan dari sistem informasi penerimaan bahan baku antara lain:

1. Digunakan dalam monitoring dan mengontrol proses penerimaan bahan baku yang terjadi sehingga lebih efisien.
2. Menghasilkan efisiensi proses penerimaan, kontrol kualitas yang ketat serta menghasilkan produk yang lebih bagus.
3. Mengurangi biaya dari penggunaan berbagai inventarisasi dengan mendapatkan kontrol material yang baik.
4. Produksi yang dihasilkan tepat.

2.8 Analisis Sistem

Menurut Mcleod (2012), analisis sistem adalah penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan.

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

Dalam tahap analisis sistem, terdapat langkah-langkah dasar yang harus dilakukan oleh sistem analis, yaitu (Mcleod, 2012):

1. *Identify*, mengidentifikasi masalah.
2. *Understand*, memahami kerja sistem yang ada .
3. *Analyze*, menganalisis sistem yang ada.

4. *Report*, membuat laporan dari hasil analisis.

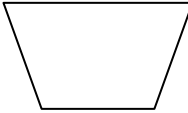
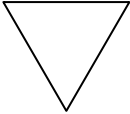
2.9 *Flowmap*

Flowmap atau bagan alir adalah bagan yang menunjukkan aliran di dalam program atau prosedur sistem secara logika. *Flowmap* ini berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *flowmap* ini harus dapat memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi. Adapun pedoman-pedoman dalam pembuatan *flowmap* adalah sebagai berikut:

1. *Flowmap* sebaiknya digambarkan dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam *flowmap* harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhir.
4. Masing-masing kegiatan didalam *flowmap* sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan didalam *flowmap* harus dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung ditempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol-simbol *flowmap* yang standar.

Adapun simbol-simbol yang sering digunakan dalam *flowmap* dapat dilihat pada Tabel II.I berikut ini:

Tabel II.1 Simbol-simbol *Flowmap*

Simbol	Nama	Fungsi
	Mulai/Akhir	Menunjukkan dimulai/akhir dari sebuah proses.
	Dokumen	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak dikertas.
	Dokumen Rangkap	Menunjukkan multi dokumen.
	Manual Process	Simbol untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer/pekerjaan manual.
	Proses	Proses pengolahan data.
	Arsip	Menunjukkan pengarsipan data.

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

Simbol	Nama	Fungsi
--------	------	--------

	Garis Aliran	Arus dari suatu proses.
	<i>Input/Output</i>	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i> .

(Sumber: Jogianto, 2005)

2.10 Jenis-Jenis *Flowchart*

Terdapat beberapa jenis bagan alir (*flowchart*), yaitu sebagai berikut (Ladjamudin, 2013):

1. Bagan alir sistem (*systems flowchart*).
2. Bagan alir dokumen (*document flowchart*).
3. Bagan alir skematik (*schematic flowchart*).
4. Bagan alir program (*program flowchart*).
5. Bagan alir proses (*process flowchart*).

2.11 *Unified Modelling Language (UML)*

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh

Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Tujuan dari UML adalah untuk menyediakan kosa kata yang umum dari istilah-istilah berbasis objek dan teknik yang cukup banyak untuk memodelkan proyek pengembangan sistem dari analisis ke desain. Menurut Dennis (2010), diagram-diagram yang ada dalam UML terbagi menjadi dua bagian utama yaitu, *Struktur diagram* dan *Behavior diagram*. *Structure diagram* biasanya digunakan untuk mepresentasikan data dan hubungan statik yang ada di dalam sebuah sistem informasi. Sedangkan *Behavior diagram* menyediakan para analis dengan sebuah gambaran hubungan yang dinamis antara instansi atau objek yang merepresentasikan sistem informasi bisnis. Versi 2.0 ini sudah diterima oleh *Object Management Group* (OMG) pada tahun 2003. Versi UML ini mendefinisikan suatu set dari 14 teknik untuk pemodelan sistem. Berikut adalah tabel dari jenis diagram yang ada dan fungsi-fungsinya:

Tabel II.2 Jenis Diagram Pada UML dan Fungsi-fungsinya

Nama Diagram	Fungsi
<i>Structure Diagrams</i>	
<i>Class Diagram</i>	Menggambarkan hubungan antara pemodelan-pemodelan <i>class</i> di dalam sistem
<i>Object Diagram</i>	Menggambarkan hubungan antara pemodelan-pemodelan objek di dalam sistem
<i>Package Diagram</i>	Mengelompokkan elemen UML lainya menjadi satu untuk membentuk level konstruksi yang lebih tinggi
<i>Deployment Diagram</i>	Menampilkan arsitektur fisik dari suatu sistem. Bisa juga digunakan untuk menunjukkan komponen <i>software</i> sedang dijadikan arsitektur fisik suatu sistem
<i>Component Diagram</i>	Menggambarkan hubungan fisik diantara komponen-komponen <i>software</i>
<i>Behavior Diagram</i>	

<i>Activity Diagram</i>	Menggambarkan proses bisnis masing-masing <i>class</i>
<i>Sequence Diagram</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam aktifitas berdasarkan urutan waktu

Tabel II.2 Jenis Diagram Pada UML dan Fungsi-fungsinya (Lanjutan)

Nama Diagram	Fungsi
<i>Behavioral State Machine Diagram</i>	Memeriksa kebiasaan dari suatu <i>class</i>
<i>Protocol State Machine Diagram</i>	Menggambarkan ketergantungan diantara perbedaan-perbedaan <i>interface</i> dari suatu <i>class</i>
<i>Use Case Diagram</i>	Menangkap kebutuhan bisnis untuk sistem dan untuk menggambarkan interaksi diantara sistem dan lingkungannya
<i>Timing Diagram</i>	Menggambarkan interaksi yang terjadi diantara suatu set dari objek-objek dan perubahan keadaan selama perjalanan waktu
<i>Interaction Overview Diagram</i>	Menggambarkan suatu ikhtisar alur dari kontrol suatu proses
<i>Communication Diagram</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam komunikasi antara satu set dari kolaborasi objek dari suatu objek

(Sumber: Dennis, 2010)

2.11.1 Use Case Diagram

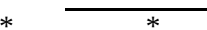
Use case adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* menggambarkan dengan cara yang sangat simpel tentang fungsi utama dari suatu

sistem dan perbedaan jenis-jenis *user* yang akan berinteraksi dengan sistem tersebut. Menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”.

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

No.	Simbol	Deskripsi
1.	<i>Use Case</i> 	– Merepresentasikan bagian utama dari sistem secara fungsional.

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

No.	Simbol	Deskripsi
2.	<i>Actor</i> 	– Seseorang atau sistem yang mendapatkan keuntungan dari sistem – Dilabelkan dengan peran – Bisa diasosiasikan dengan aktor lainnya berdasarkan spesialisasi – Diletakan diluar batas sistem
3.	<i>Boundary</i> 	– Nama dari sistem terdapat didalam atau diatas – Merepresntasikan ruang lingkup dari sistem
4.	<i>Association</i> 	– Menghubungkan suatu aktor dengan <i>use case</i> dengan apa interaksinya
5.	<i>Extend</i> 	– Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
6.	<i>Generalization</i> 	– Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
7.	<i>Include</i> 	– Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan

		fungsinnya atau sebagai syarat dijalankan <i>use case</i> ini.
--	---	--

Sumber: Alan Denis (2011)

2.11.2 Activity Diagram

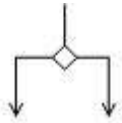
Digunakan untuk model perilaku dalam independen proses bisnis benda. Dalam banyak hal, diagram aktivitas dapat dipandang sebagai diagram aliran data yang canggih yang digunakan dalam hubungannya dengan analisis terstruktur. Namun, tidak seperti aliran data diagram, diagram aktivitas termasuk notasi yang membahas pemodelan paralel, kegiatan bersamaan dan proses. (Alan Dennis, 2011). Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan actor, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

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

Berikut adalah simbol-simbol yang ada pada Activity Diagram:

Tabel II.4 Simbol-Simbol Activity Diagram

No.	Simbol	Deskripsi
1.	<i>Initial Node</i>	Merupakan tanda awal dari sebuah aktifitas.
2.	<i>Activity</i>	Merupakan sebuah gambaran aktifitas

		yang terjadi
3.	<i>Decision Node</i> 	Pilihan untuk pengambilan keputusan

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

No.	Simbol	Deskripsi
5.	<i>Final-Activity Node</i> 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
7.	<i>Control flow</i> 	Menunjukkan urutan eksekusi
8.	<i>Object Flow</i> 	Menunjukkan aliran objek dari satu kegiatan (atau tindakan) untuk kegiatan lain (atau tindakan).

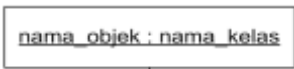
(Sumber: Alan dennis, 2011)

2.11.3 Sequence Diagram

Sequence diagram adalah salah satu dari dua jenis diagram interaksi. Mereka menggambarkan benda-benda yang berpartisipasi dalam kasus penggunaan dan pesan yang melewati antara mereka dari waktu ke

waktu untuk satu *use case*. Sebuah diagram *sequence* adalah model dinamis yang menunjukkan urutan eksplisit pesan yang lewat di antara objek dalam interaksi didefinisikan. Karena urutan diagram menekankan pemesanan berbasis waktu kegiatan yang terjadi di antara set benda, mereka sangat membantu untuk memahami spesifikasi *real-time* dan kompleks menggunakan kasus (Dennis, 2010). Berikut simbol yang ada pada *Sequence Diagram*:

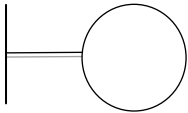
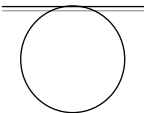
Tabel II.5 Simbol-Simbol *Sequence Diagram*

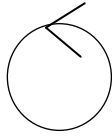
No.	Simbol	Deskripsi
1.	Aktor 	Orang atau sistem yang berasal dari manfaat dan eksternal ke sistem yang berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan
2.	Garis Hidup/<i>lifeline</i> 	Menyatakan kehidupan suatu objek.
3.	Objek 	Berpartisipasi secara berurutan dengan mengirim dan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

5.	<i>Message</i> <i>aMessage()</i> 	Pesan yang menggambarkan komunikasi yang terjadi antar objek.
6.	<i>Object Destruction</i> X	Ditempatkan pada akhir dari suatu objek <i>lifeline</i> untuk menunjukkan bahwa itu akan keluar dari eksistensi.

(Sumber: Dennis, 2010)

Tabel II.5 Simbol-Simbol *Sequence Diagram* (lanjutan)

No.	Simbol	Deskripsi
Adapun jenis-jenis objek <i>Sequence Diagram</i> sebagai berikut:		
1.	<i>Boundary</i> 	Berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form</i> cetak.
2.	<i>Entity Class</i> 	Merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.

3.	<i>Control Class</i> 	Suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
----	---	--

(Sumber: Gata, 2013)

2.11.4 Class Diagram

Sebuah diagram kelas adalah model statis yang menunjukkan kelas dan hubungan antar kelas yang tetap konstan dalam sistem dari waktu ke waktu. Sebuah *class* merupakan deskripsi dari sekumpulan objek yang memiliki properti (*attribute*), operasi (*method*), relasi (*association*), dan tingkah laku (*behavior*) yang sama. Sebuah class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda /fungsi). (Alan Dennis, 2011). Berikut simbol-simbol yang ada pada diagram kelas:

Tabel II.6 Simbol-Simbol *Class Diagram*

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

Asosiasi yang menghubungkan *class* dengan *class Multiplicity*.

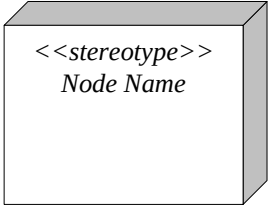
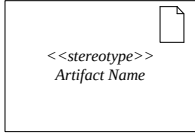
1, 1...*,
0...1

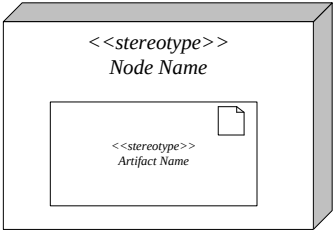
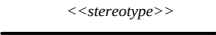
(Sumber: Alan Dennis, 2011)

2.11.5 *Deployment Diagram*

Deployment diagram digunakan untuk mewakili hubungan diantara komponen-komponen perangkat keras dalam infrastuktur fisik sebuah sistem informasi. Misalnya, ketika merancang sistem informasi yang akan terdistribusi menggunakan *wide area network*, sebuah *deployment diagram* dapat digunakan untuk menunjukkan hubungan komunikasi antara node yang berbeda dalam jaringan. *Deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen perangkat lunak dan bagaimana mereka dikerahkan dalam arsitektur atau infrastruktur dari suatu sistem informasi. Berikut adalah simbol-simbol *deployment diagram* menurut Dennis 2010:

Tabel II.7 Simbol-Simbol *Deployment Diagram*

Nama Simbol	Deskripsi	Simbol
<i>Node</i>	Sumber daya komputasi seperti komputer klien, <i>server</i> atau perangkat jaringan individu.	
<i>Artifact</i>	Spesifikasi sebuah <i>software</i> atau <i>database</i> , contoh: <i>database</i> atau tabel atau <i>layer</i> .	

<i>Node with a deployed artifact</i>	Menggambarkan sebuah artefak yang ditempatkan pada node fisik. Mendukung pemodelan distribusi perangkat lunak melalui jaringan.	
<i>Communication path</i>	Merupakan asosiasi antara dua node. Memungkinkan node untuk bertukar pesan.	

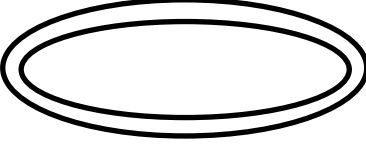
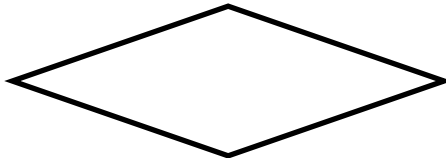
(Sumber: Dennis, 2010)

2.12 Entity Relationship Diagram (ERD)

Berdasarkan Yanto (2016), ERD adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. ERD juga merupakan gambaran yang merelasikan antara objek yang satu dengan objek yang lain dari objek dunia nyata yang sering dikenal dengan hubungan antar entitas. Sebagai contoh jika membuat ERD dari system perpustakaan maka bahan sebagai objek ERD bias berupa anggota, buku, peminjaman, pengembalian, dan lain sebagainya. Berikut ini adalah symbol-simbol dari *Entity Relationship Diagram* (ERD), dapat dilihat pada Tabel II.9.

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

No.	Simbol	Deskripsi
1.	Entitas 	Entitas merupakan data inti yang akan disimpan
2.	Atribut 	Field atau kolom data yang butuh disimpan

		dalam suatu entitas
3.	Atribut kunci primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan.
4.	Atribut multinilai / multivalve 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
5.	Relasi 	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja
6.	Asosiasi 	Penghubung antara relasi dan entitas

(Sumber: Rosa dan Shalahuddin, 2015)

2.13 Conceptual Data Model (CDM)

CDM (*Conceptual Data Model*) atau model konsep data merupakan konsep yang berkaitan dengan pandangan pemakai terhadap data yang di simpan dalam basis data. CDM sudah di buat dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel untuk keperluan implementasi ke basis data. CDM merupakan hasil penjabaran lebih lanjut dari ERD. Ada aturan-aturan yang harus diikuti dalam melakukan konversi ERD menjadi CDM. Berikut adalah simbol-simbol yang ada pada CDM :

Tabel II.9 Simbol-simbol *Conceptual Data Model*

Simbol	Deskripsi
Entitas / tabel	Entitas atau tabel yang menyimpan data

dalam basis data

nama_tabel	

Relasi

Relasi antar tabel yang terdiri atas nama relasi dan *multiplicity*

1..*

nama relasi

1..*

2.14 Kamus Data

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

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

Keuntungan penggunaan kamus data adalah (Sommerville, 2003):

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

nama ini harus dipakai secara konsisten dan tidak boleh bentrok. Kamus data dapat memeriksa keunikan nama dan memberitahu analis persyaratan sekiranya terjadi duplikasi nama.

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

Nama Tabel : Pemasok

Tipe : File master

Tabel II.10 Contoh Kamus Data Untuk Tabel Pemasok

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

(Sumber: Jogiyanto, 2005)

2.15 Windows Navigation Diagram (WND)

Menurut Dennis dkk (2015), desain struktur navigasi mendefinisikan komponen dasar antarmuka dan bagaimana mereka bekerja sama untuk menyediakan fungsionalitas kepada pengguna. *Windows Navigation Diagram* (WND) digunakan untuk menunjukkan bagaimana semua layar, bentuk, dan laporan yang digunakan oleh sistem terkait dan bagaimana pengguna berpindah dari satu ke lainnya.

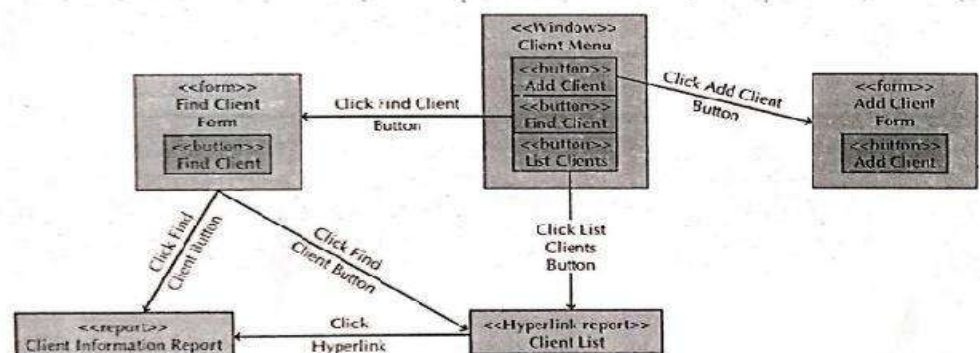
WND sangat mirip dengan *state machine*, karena keduanya memodelkan perubahan status. Mesin. Keadaannya biasanya membuat model perubahan dari suatu objek, sedangkan keadaan

model perubahan lainnya yaitu di antarmuka pengguna. Di WND setiap status antarmuka pengguna direpresentasikan sebagai kotak. Kotak biasanya sesuai dengan komponen antarmuka pengguna, seperti jendela, *form*, tombol, atau laporan.

Transisi dimodelkan sebagai panah berkepala tunggal atau berkepala dua. Panah berkepala satu menunjukkan bahwa kembali ke keadaan panggilan tidak diperlukan, sedangkan panah berkepala dua menunjukkan pengembalian yang diminta.

Item terakhir yang akan dijelaskan dalam WND adalah *stereotype*-nya. *Stereotype* dimodelkan sebagai item teks yang dilampirkan dalam *guillemet* atau kurung sudut (<<>>). *Stereotype* mewakili jenis komponen antarmuka pengguna dari kotak pada diagram.

Struktur navigasi dasar antarmuka mengikuti struktur dasar dari proses bisnis itu sendiri, sebagaimana didefinisikan dalam kasus penggunaan dan model perilaku. Analisis dimulai dengan kasus-kasus penggunaan esensial dan mengembangkan aliran mendasar dari kontrol sistem ketika bergerak dari objek ke objek. Analisis kemudian memeriksa skenario penggunaan untuk melihat seberapa baik dari WND.



Gambar II.4 Contoh WND

(sumber: Dennis et al, 2015)

2.16 *Personal Home Page (PHP)*

Hypertext Preprocessor (PHP) adalah bahasa *server-side scripting* yang digunakan untuk aplikasi web yang dinamis dan interaktif. Sebuah halaman PHP adalah sebuah halaman HTML yang memiliki *server-side scripts* yang ditempatkan dalam *server* dan diproses oleh *web server* sebelum dikirim ke *browser* pemakai (Welling dan Thomson, 2003).

Server-side scripts dijalankan ketika *browser* melakukan permintaan *file.php* dari *server*. PHP dipanggil oleh *web server*, dimana proses script perintah yang ada di suatu halaman dieksekusi mulai dari awal sampai akhir di dalam mesin PHP. Setelah *script* PHP tersebut diolah, hasilnya akan ditampilkan kepada *client* melalui *web browser* berupa tampilan HTML. Menurut Welling dan Thomson (2003), beberapa keunggulan PHP adalah:

1. *High Performance*

PHP sangat efisien. Dengan menggunakan *server* tunggal yang tidak mahal, *user* dapat melakukan banyak pekerjaan setiap harinya.

2. *Database Integration*

PHP mempunyai sambungan ke banyak sistem basis data, antara lain MySQL, PostgreSQL, Oracle, Informix, dan Sysbase Databases.

3. *Built-in-Libraries*

PHP dirancang khusus untuk web, dan mempunyai banyak *built-in-function* untuk menampilkan banyak fungsi di dalam web.

4. Harga yang murah

PHP adalah perangkat lunak gratis.

5. Mudah dalam pembelajaran dan penggunaan

Sintaks PHP berdasarkan bahasa pemrograman lainnya, terutama C dan Java.

6. *Portability*

PHP dapat digunakan di banyak sistem operasi yang berbeda.

7. Ketersediaan *Source Code*.

Kode PHP dapat langsung diakses dan dimodifikasi secara bebas.

2.17 MySQL

Menurut Anhar (2010) MySQL adalah sebuah perangkat lunak sistem manajemen basis data (DBMS) dari sekian banyak DBMS seperti Oracle, MSSQL, Postgre SQL, dan lain-lain. MySQL merupakan DBMS yang multithread, multi-user yang bersifat gratis dibawah GNU *General Public Licence* (GPL). MySQL memiliki beberapa kelebihan antara lain:

1. Portabilitas

MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

2. *Open Source*

MySQL didistribusikan secara *open source*, dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.

3. *Multi User*

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. Jenis Kolom

MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed* atau *unsigned integer*, *float*, *double*, *char*, *text*, *date*, *timestamp*, dan lain-lain.

6. Perintah dan Fungsi

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *select* dan *where* dalam perintah (*query*).

7. Keamanan

MySQL memiliki beberapa lapisan sekuritas seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.

8. Konektivitas

MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix socket (UNIX), atau Named Pipes (NT).

9. Lokalisasi

MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk di dalamnya.

10. Antar Muka

MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

11. Klien dan Peralatan

MySQL dilengkapi dengan berbagai peralatan (*tools*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.

BAB III

METODOLOGI PENELITIAN

Metode penelitian adalah cara yang digunakan oleh peneliti dalam pengumpulan data penelitiannya. Berdasarkan pengertian tersebut dapat dikatakan bahwa metode penelitian adalah cara yang dipergunakan untuk mengumpulkan data yang diperlukan dalam penelitian. (Moleong, 2012).

Metode penelitian adalah cara-cara ilmiah untuk mendapatkan data yang valid, dengan tujuan dapat ditemukan, dikembangkan dan dibuktikan, suatu pengetahuan tertentu sehingga pada gilirannya dapat digunakan untuk memahami, memecahkan, dan mengantisipasi masalah. (Sugiyono, 2013).

Metode penelitian adalah suatu cabang ilmu pengetahuan yang membicarakan atau mempersoalkan cara-cara melaksanakan penelitian (yaitu meliputi kegiatan-kegiatan mencari, mencatat, merumuskan, menganalisis sampai menyusun laporannya) berdasarkan fakta-fakta atau gejala-gejala secara ilmiah. (Suharsaputra, 2014)

Berikut ini akan dijelaskan tentang metodologi penelitian yang akan digunakan pada penelitian ini.

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 terhadap sistem yang sedang berjalan dan wawancara dengan pegawai *warehouse* sebagai sumber informasinya. Dalam penelitian ini data tersebut berupa aliran data sistem informasi penerimaan bahan baku, prosedur penerimaan bahan baku pada PT Galih Ayom Paramesti.

2. Data Sekunder

Data sekunder merupakan sumber data penelitian yang diperoleh secara tidak langsung melalui media perantara. Dalam penelitian ini data tersebut diperoleh dari jurnal penelitian tentang penerimaan bahan baku, buku yang berisi tentang sumber-sumber yang menjelaskan tentang penerimaan bahan baku, dan situs internet tentang penerimaan bahan baku.

3.2 Metode Pengumpulan Data

Langkah pertama yang dilakukan sebagai dasar dari kerangka pemikiran dalam penelitian ini adalah dengan melakukan studi pendahuluan terhadap objek penelitian. Secara sederhana, pengumpulan data diartikan sebagai proses atau kegiatan yang dilakukan peneliti untuk mengungkap atau menjangkau berbagai fenomena, informasi atau kondisi lokasi penelitian sesuai dengan lingkup penelitian. Pengumpulan data sangat penting dalam proses penelitian, karena tanpa data lapangan, proses analisis data dan kesimpulan hasil penelitian, tidak dapat dilaksanakan.

3.3 Instrumen Pengumpulan Data

Metode yang digunakan dalam pengumpulan data, diantaranya:

1. Studi lapangan

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

- a. Pengamatan, tahap ini dilakukan secara langsung di bagian *warehouse* PT Galih Ayom Paramesti dengan mengamati sistem informasi penerimaan bahan baku pada gudang, mulai dari bahan baku datang, disimpan sampai proses bisnis persediaan bahan baku. Melalui teknik ini, data yang dibutuhkan diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.
- b. Wawancara, yaitu mencari data yang dibutuhkan secara langsung dengan memberikan pertanyaan spontan tentang proses bisnis

penerimaan bahan baku yang diperlukan pada penyusunan tugas akhir ini. Wawancara ini dilakukan kepada pegawai di bagian *warehouse* yaitu Bapak Sani dan Bapak Jerry

2. Studi Pustaka

Tahap ini merupakan tahap awal pengumpulan data dari sumber-sumber seperti buku yang berhubungan analisis dan desain sistem informasi, mengelola kebutuhan dan penerimaan bahan baku, PHP dan MySQL, sistem informasi manajemen, pengembangan perangkat lunak, dan juga mencari data tambahan yang diperlukan seperti konsep pemrograman PHP, dan beberapa contoh *coding* program melalui media *internet*.

3.4 Metode Pengembangan Sistem

Metode pengembangan sistem adalah menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah berjalan (Arikunto, 2006). Pada penelitian ini untuk mengatasi masalah yang ada pada sistem, diputuskan untuk membuat pengembangan sistem. Dalam pengembangan sistem ini digunakan metode *Evolutionary Prototype*. Alasan menggunakan metode *Prototype* ini pada PT Galih Ayom Paramesti karena metode *Prototype* lebih menghemat waktu dalam pengembangan sistem, penerapan menjadi lebih mudah karena karyawan mengetahui apa yang diharapkannya, dan lebih efisien dibanding dengan metode *Waterfall* dan metode RAD (*Rapid Application Development*) yang memakan waktu lama dalam pengembangan sistem dan kurang efisien.

Tahapan-tahapan pada model *Prototype* adalah sebagai berikut:

1. Identifikasi kebutuhan *user*

Melakukan diskusi dengan karyawan pengolahan data persediaan bahan baku di *warehouse* tentang kondisi sistem saat ini dan kebutuhan sistem yang diinginkan.

2. Membuat *Prototype*

Membuat *Prototype* dari sistem yang telah dijelaskan oleh karyawan pengolahan data persediaan bahan baku di *warehouse*.

3. Menyesuaikan dan evaluasi *Prototype* dengan keinginan *user*

Menanyakan kepada karyawan pengolahan data persediaan bahan baku di *warehouse* tentang *Prototype* yang sudah dibuat, apakah sesuai atau tidak dengan kebutuhan sistem.

4. Menggunakan *Prototype*

Sistem mulai dikembangkan dengan *Prototype* yang sudah dibuat oleh karyawan pengolahan data persediaan bahan baku di *warehouse*.

3.5 Kerangka Penelitian

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

1. Pendahuluan

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

2. Identifikasi Masalah

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

3. Tujuan Penelitian

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

4. Batasan Masalah

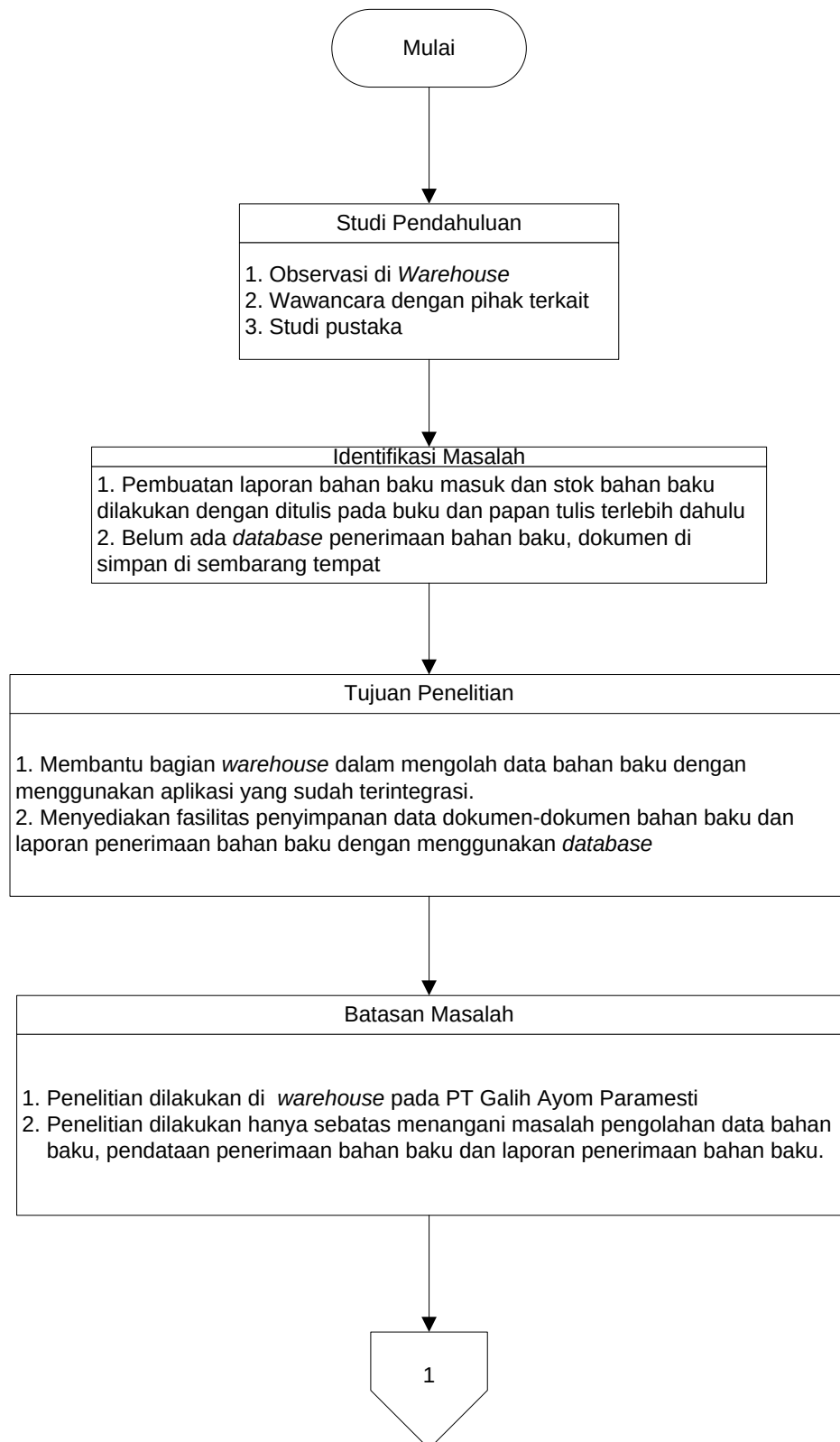
Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya.

5. Penerapan Metode *Prototype Evolusioner*

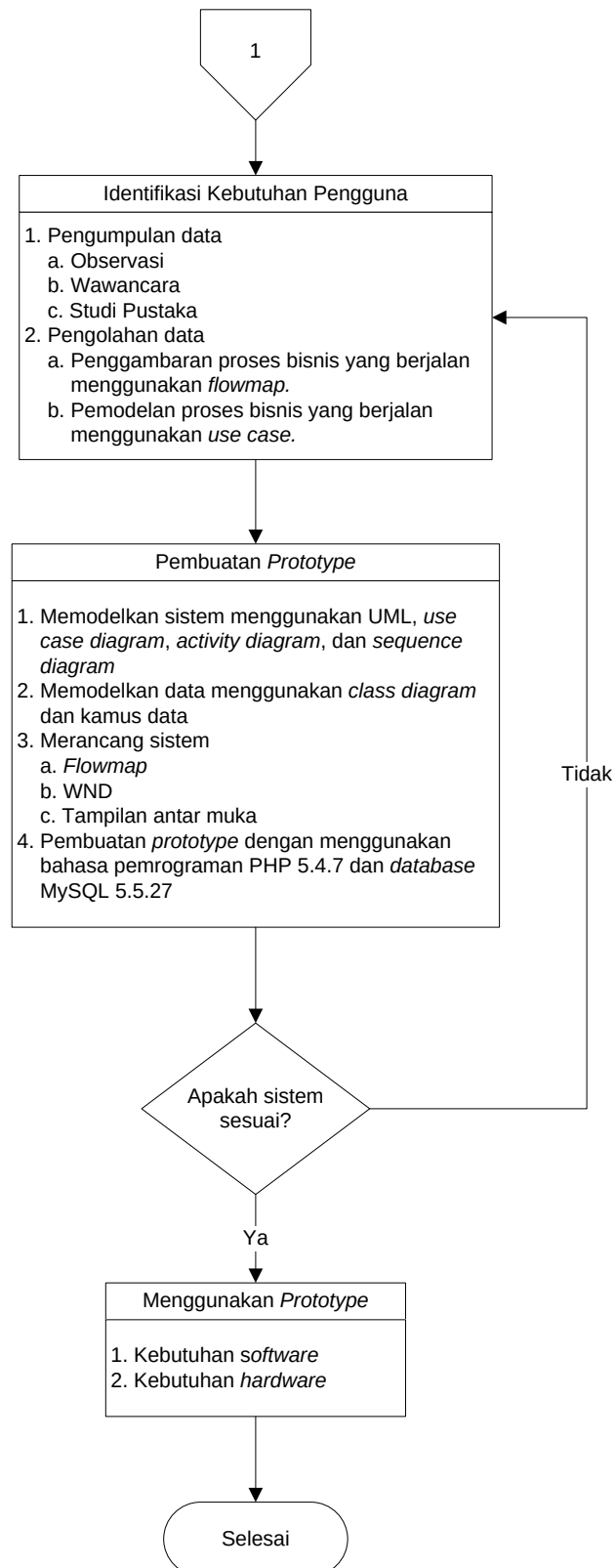
- a. Mengidentifikasi kebutuhan pelanggan
Menganalisis kebutuhan *user* terhadap program yang akan dibuat.
Dengan melakukan metode pengumpulan data dan pengolahan data.
- b. Membuat prototipe
Membuat sebuah prototipe sesuai dengan analisis yang dilakukan.
Pada tahap ini akan dirancang aplikasi tentang sistem penyimpanan bahan baku. Perancangan sistem nantinya akan memuat tentang:
 - i. Model sistem menggunakan *Unified Modeling Language* (UML).
 - ii. Perancangan *database* dengan menggunakan MYSQL versi 5.5.27
- c. Menggunakan prototipe
Memutuskan untuk menggunakan prototipe yang telah divalidasi menggunakan metode *prototype evolutioner*. Apabila prototipe tidak diterima maka akan kembali ke tahap identifikasi.

.

Kerangka penelitian dibuat dalam bentuk *flowchart* yang menggambarkan tahap-tahap kegiatan mulai dari awal hingga akhir dapat dilihat pada Gambar III.1.



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



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

PENGUMPULAN DAN PENGOLAHAN DATA

Sejarah Perusahaan

PT Galih Ayom Paramesti (GAP) didirikan pada tahun 1992 dalam rangka untuk memenuhi permintaan industri otomotif. Setelah bertahun-tahun, produk GAP tumbuh dalam berbagai produk, terutama pada produk berbasis logam. GAP selalu mengembangkan produknya sepanjang pelanggan membutuhkan. Pengembangan dilakukan melalui penelitian yang berkelanjutan dalam sistem yang telah diterapkan oleh GAP. Melalui manajemen produk yang ada secara spesifik, GAP selalu meningkatkan efektivitas dan efisiensi dari proses produk.

Dalam jangka panjang, GAP ditargetkan untuk menjadi pemasok terbaik di sektor otomotif, bidang manufaktur. Sementara itu, kesadaran akan konsep “Ramah Lingkungan” muncul. Untuk alasan itu, GAP berusaha untuk menjadi perusahaan yang ramah lingkungan. Dengan itu, semua kegiatan yang akan dilakukan dalam rangka untuk mencapai visi perusahaan. Visi GAP dengan jelas menyatakan bahwa “kami ingin sukses bersama-sama dengan mitra bisnis dan ramah lingkungan”.

Karena kepuasan pelanggan menjadi tujuan, GAP selalu memperbaiki dan mengembangkan layanan dan standar manajemen secara bersamaan. PT Galih Ayom Paramesti akan selalu memprioritaskan layanan pelanggan. GAP yakin bahwa GAP akan tumbuh seiring kepuasan pelanggan dapat dipenuhi oleh pelayanan yang baik. GAP ingin orang-orang dimana saja untuk dilayani oleh produknya.

4.2 Makna Logo Perusahaan



Gambar IV.1 Logo GAP

Sumber : PT Galih Ayom Paramesti

Logo merupakan sebuah simbol, atau lambang yang menggambarkan ciri dari barang, lembaga, perusahaan, instansi, ataupun website. Begitu pula dengan PT Galih Ayom Paramesti memiliki logo yang bermakna. Lingkaran dan tulisan GAP sendiri menggambarkan arti dari GAP itu yaitu PT Galih Ayom Paramesti yang artinya Galih adalah mencari dan melatih potensi diri. Ayom yaitu mengayomi dengan segenap hati, dan yang terakhir yaitu Paramesti yaitu dengan menjunjung nama Tuhan.

4.3 Profil Perusahaan

Profil singkat dari PT Galih Ayom Paramesti adalah sebagai berikut:

Nama	: PT Galih Ayom Paramesti
Alamat	: Jl. Inspeksi Tarum Barat, Pekopen, Lambang Jaya Tambun, Bekasi, Jawa Barat, Indonesia
Telepon	: (+62) 2188374577, 98169016, 70604371
Fax	: (+62) 2188374576
Tahun Terdaftar	: 2010
Karyawan	: ±56 orang
Luas Pabrik	
- Luas keseluruhan	: 2.500 m ²

4.4 Visi dan Misi PT Galih Ayom Paramesti

PT Galih Ayom Paramesti sebagai perusahaan yang bergerak dibidang komponen otomotif mempunyai visi, misi, dan tujuan sebagai berikut:

Visi :

Manajemen PT Galih Ayom Paramesti mempunyai cita-cita mengelola usahanya menjadi salah satu perusahaan Industri Komponen otomotif dan komponen lain (*machining*) yang terbaik dan dipercaya di lingkungan Nasional dengan menggunakan kemampuan sumber daya manusia yang kompeten.

Misi :

PT Galih Ayom Paramesti akan menjadi sebuah perusahaan yang menghasilkan produk dan jasa yang berkualitas tinggi dan aman, untuk memperoleh laba yang memadai sehingga perusahaan berkembang dan memberi manfaat kepada pemegang saham, karyawan serta pihak lain dengan melakukan perbaikan terus menerus.

4.5 Struktur Organisasi Perusahaan

Untuk menjalankan usahanya, setiap perusahaan memerlukan suatu struktur organisasi yang baik. Struktur organisasi dapat diartikan sebagai susunan dan hubungan antar bagian dan posisi dalam suatu perusahaan. Suatu struktur organisasi menggambarkan pembagian kerja, pelimpahan wewenang, kesatuan perintah dan tanggung jawab yang jelas.

Struktur organisasi yang tersusun dengan baik akan memudahkan koordinasi, integrasi, serta meningkatkan efektivitas dan efisiensi suatu perusahaan di dalam mencapai tujuannya.

Berikut adalah struktur organisasi PT Galih Ayom Paramesti secara umum pada gambar IV.2:

tata-tertib, keadilan dan kesempatan bagi semua untuk berkontribusi secara tepat, menentukan urutan agenda, mengarahkan diskusi ke arah konsensus, dan menjelaskan tindakan dan kebijakan.

- c. Mengambil keputusan sebagaimana didelegasikan oleh BOD atau pada situasi tertentu yang dianggap perlu.

2. *Dept. Head PPIC:*

- a. Membuat planning produksi harian, bulanan serta tahunan.
- b. Menerima, menyimpan, merawat dan mengirim barang sesuai dengan kualitas yang telah ditentukan.
- c. Mengkoordinasikan kegiatan operasional dengan produksi.
- d. Mengontrol pengiriman dari pemasok dan pelanggan.
- e. Menentukan loading mesin dan perencanaan kerja.
- f. Menentukan waktu *delivery* sesuai dengan permintaan pelanggan.

3. *Dept. Head Produksi:*

- a. Memimpin, memastikan serta mengontrol jajaran bawahan dalam pelaksanaan kegiatan produksi agar terlaksana sesuai jadwal produksi
- b. Membuat pemetaan keterampilan operator produksi dan di evaluasi secara periodik 6 bulan.
- c. Mengembangkan kemampuan dan keterampilan operator dengan cara memberikan pelatihan secara berkala serta membuat jadwal dan modul pelatihan.
- d. Membantu mengatasi kesulitan dan memecahkan persoalan-persoalan produksi meliputi mesin, operator dan produk.
- e. Tanggap darurat bila terjadi sesuatu hal yang luar biasa.
- f. Memberikan ijin sesuai prosedur kepada karyawan (operator) bila meninggalkan tugas/ijin pulang kerja atau keluar kantor dengan alasan-alasan tertentu.

4. *Dept. Head Building and Maintenance :*

- a. Membantu pelaksanaan produksi dalam hal pemeliharaan mesin dan peralatannya dengan memastikan kondisi mesin siap pakai.
- b. Melakukan pemeriksaan mesin setiap hari.
- c. Melakukan pemeriksaan mesin secara berkala sesuai jadwal yang telah disepakati dengan departemen terkait.

- d. Mendata setiap mesin yang ada dan menginformasikan ke kepala produksi.
- e. Membuat permintaan *spare part* ke kepala produksi dan mencari pemasok.
- f. Menginformasikan setiap terjadi masalah mesin, alat/tool ke kepala produksi untuk diambil keputusan.

5. *Dept. Head Finance and Administration :*

- a. Melaksanakan kegiatan pembayaran, penagihan dan laporan keuangan.
- b. Membuat administrasi keuangan dengan pemerintah dan instansi lain yang terkait.
- c. Membuat sistem penilaian karya dari kebijakan sampai IK nya.
- d. Menyiapkan atau merencanakan pelatihan sesuai dengan kebutuhan.
- e. Membuat laporan ke top manajemen tentang keefektifan dan kesesuaian kerja karyawan.
- f. Memelihara catatan yang sesuai mengenai pendidikan, pelatihan, keterampilan dan pengalaman.

6. *Dept. Head Marketing :*

- a. Menerima, menyimpan serta merawat permintaan penawaran harga dari pelanggan sesuai waktu yang telah ditetapkan.
- b. Mendata permintaan atau *inquiry* harga yang masuk dari pelanggan
- c. Membuat penawaran harga ke pelanggan.
- d. Menginformasikan dan memberikan data kepada departemen *engineering* untuk pembuatan estimasi harga.
- e. Membuat laporan penawaran yang sudah terkirim dan disetujui pelanggan kepada top manajemen dan departemen terkait untuk di tindak lanjuti.

7. *Dept. Head Quality Control :*

- a. Memeriksa setiap kedatangan barang dari pemasok (*Material, Machining, Heatreatment planning*).
- b. Membuat laporan *reject in house*, reject pemasok dan *claim* pelanggan
- c. Membuat *checksheet, part inspection standard, inspection record*.
- d. Merevisi *checksheet, part inspection standard, inspection record* jika ada perubahan dari pelanggan.
- e. Membuat QOCP/ *control plan* dan FMEA bersama *engineering* dan produksi.
- f. Membuat alat ukur.
- g. Membuat jadwal kalibrasi eksternal.

8. *Dept. HRD:*

- a. Melakukan proses seleksi dan rekrutmen terhadap karyawan lain.

- b. Mengadakan training pada karyawan baru.
 - c. Menyiapkan kebutuhan training kepada karyawan.
 - d. Melakukan pembayaran upah tenaga kerja.
 - e. Mengatur sistem absensi karyawan.
 - f. Memberikan semangat setiap hari kepada karyawan.
9. Dept. *Engineering* Produksi:
- a. Proses *engineering* dan perhitungan nilai/ harga barang.
 - b. Membuat *master plan new project* dan persiapan *new project* hingga *finish good*.
 - c. Membuat proses atau perencanaan alat penunjang produksi.
 - d. Membuat program perbaikan proses produksi.
 - e. Mengevaluasi usulan perbaikan proses dalam proses produksi.
 - f. Mengkoordinasi dengan bagian terkait bila terjadi penyimpangan dalam proses produksi.
 - g. Mengawasi proses produksi apakah sesuai planning dalam hal teknik.
 - h. Memastikan dokumen yang berhubungan dengan kerja *engineering* yang berada di lingkungan kerja di *update*.
 - i. Mengevaluasi dalam hal standar *time/cost* apakah sesuai estimasi.

4.6 Bahan Baku

PT Galih Ayom Paramesti untuk menghasilkan produk komponen otomotif mempunyai tiga macam bahan baku yaitu bahan baku mentah, bahan baku setengah jadi dan bahan baku penolong, berikut ini merupakan beberapa jenis bahan baku:

1. Bahan baku mentah



Gambar IV.3 Bahan Baku Mentah

Sumber : PT Galih Ayom Paramesti (2017)



Gambar IV.4 Bahan Baku Mentah

Sumber : PT Galih Ayom Paramesti (2017)

2. Bahan baku setengah jadi



Gambar IV.5 Bahan Baku Setengah Jadi

Sumber : PT Galih Ayom Paramesti (2017)



Gambar IV.6 Bahan Baku Setengah Jadi

Sumber : PT Galih Ayom Paramesti (2017)

3. Bahan baku penolong



Gambar IV.7 Bahan Baku Penolong

Sumber : PT Galih Ayom Paramesti (2017)

4.7 Dokumen Penerimaan Bahan Baku

Proses penerimaan bahan baku merupakan kegiatan untuk mengendalikan bahan baku masuk dan keluar serta menyediakan bahan baku sesuai dengan kebutuhan proses produksi. Kegiatan ini sangat penting dan berpengaruh terhadap kelancaran proses produksi. Dokumen yang terlibat dalam penerimaan bahan baku adalah sebagai berikut:

1. Purchase Order (PO)

Pada dokumen *purchase order* terdapat *description*, *quantity/unit*, *unit price*, *total price*. Dokumen *purchase order* ini dibuat sesuai dengan list kebutuhan material berdasarkan list penawaran harga yang diberikan *supplier*

PT. GALIH AYOM PARAMESTI
 Jl. Inspeksi Taruni Barat, Pekopen Lembang Jaya,
 Tambora, Bekasi 17510 Indonesia
 Phone : (021) 8837 4577
 Fax : (021) 8837 4578

PURCHASE ORDER

Kepada PT. INTERMEDIO RAYA Kawasan Industri Ranta Jl. Meris V No. 15, Tangerang Fax : (021) 5919404 / 896-9239 Telp : (021) 5918943 U/p : Bp. KARTONO		Dokumen No. : CTM-PCN-PUOR-005-00 No Purchase Order : GAP142/PO/01/2016 Tanggal : 29 SEPTEMBER 2016 Revisi : 0		
No.	Description	Jumlah/Setoran Qty/Unit	Harga Satuan/ Unit Price	Jumlah Harga/ Total Price
1	DOWEL Pin (K04)-12x48) (Ø 10.1 ± 0.15 X Ø 8.2 ± 0.2 X 14 ± 0.2) M4TL SWICH-10R Sales Forecast : JULI 2016 Qty : 180.000 Pcs JULI 2016 Qty : 190.000 Pcs Note : 1. Kurs USD Rp 13.500,- 2. Harap diartikan checklist dan terlihat pada saat pengiriman	180.000 PCS	255.000	45.900.000,00
CATATAN : Pengiriman : <i>29 September</i> Pembayaran : 1 BULAN SETELAH INVOICE DITERIMA Keterangan : HD PO HARAP DICATAT DISURAT JALAN			Sub Total 1 : 45.900.000,00 Discount 10 % : - Sub Total 2 : 45.900.000,00 PPN 10 % : 4.990.000,00 Total : 50.890.000,00	
JAWAB : 01.506.027.3-004.000 ALAMAT : Komplek Kecamatan I No. 32 Kel. Jatinegara Cakung, Jakarta Timur - 13130			PEMESAN : PERSETUJUAN PENJUAL IGNATIUS SUMARDI	

CNC Turned Parts, Jig & Fixture Manufacturing

Surat Jalan ke-Fax dan ke Perusa di Jakarta 29/9/16

Gambar IV.8 Purchase Order
 Sumber : PT Galih Ayom Paramesti (2016)

2. Surat Jalan

Pada surat jalan terdapat nomor surat jalan, tanggal, deskripsi barang, keterangan dan *pieces*. Surat jalan ini dibuat setelah *supplier* menerima PO dari *purchasing*, *supplier* akan menyiapkan bahan baku yang dipesan dan membuat surat jalan. Setelah itu bahan baku dikirim oleh *supplier*.

TRUST STEEL INDO
Carbon & Alloy Steel Specialist

PT. Trust Steel Indo
Jalan Raya Imam Bonjol No. 41
RT 009 / RW 013, Sukadana
Cikarang Barat, Bekasi
Tel : +62 21 8852 8680
Fax : +62 21 8852 8638

Delivery Order

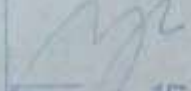
Delivery To:	TSPH 3002
Delivery Date:	20 Aug 2018
PO No:	3A7110POM110018

Buyer To:	Ship To:
PT. GALIH AYOM PARAMESTI Jl. Insipok Tarum Barat - Pekanbaru Lembang Jaya - Tambora - Delas Jakarta 11110 CN - 8804877	PT. GALIH AYOM PARAMESTI Jl. Insipok Tarum Barat - Pekanbaru Lembang Jaya - Tambora - Delas Jakarta 11110 CN Jakarta Indonesia 021 -

Item Description:	Length (mm):	Qty:
Steel Round Bar S235JR - 25mm	8800mm	17

Prepared By:  **TRUST STEEL INDO**

Shipped By: 

Received By: 

Gambar IV.9 Surat Jalan
Sumber : PT Galih Ayom Paramesti (2017)

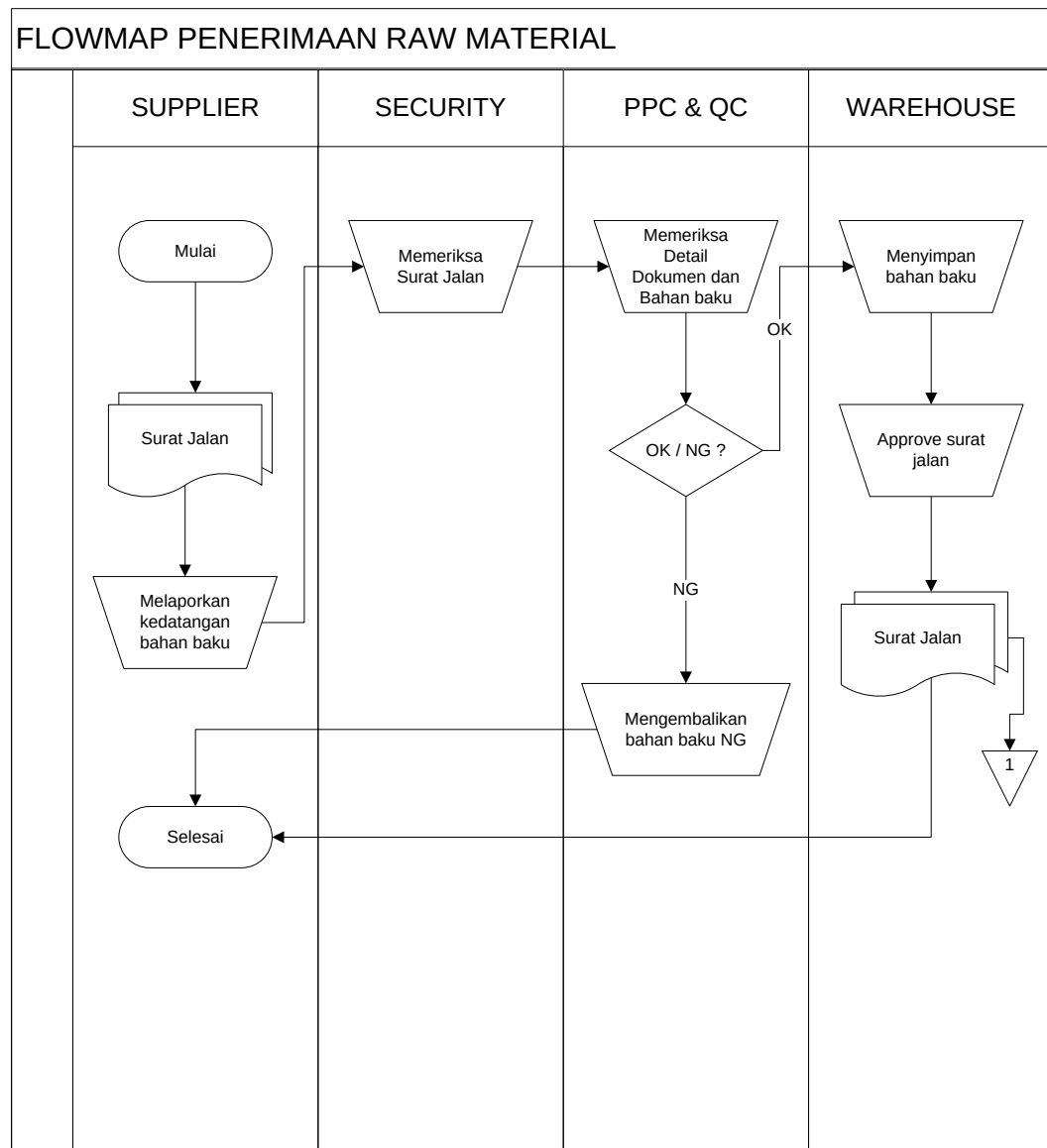
4.8 Proses Penerimaan Bahan Baku

Pada PT Galih Ayom Paramesti saat penerimaan bahan baku sebelumnya dilakukan proses pengadaan bahan baku dengan menggunakan kategori *Make to Order*, proses-proses tersebut diuraikan sebagai berikut ini:

1. *Supplier* melaporkan kedatangan *raw material* kepada *security*.
2. *Security* melakukan pemeriksaan surat jalan.

3. Surat jalan dari *supplier* diserahkan kepada petugas administrasi *warehouse*.
4. PPC melakukan pencatatan dan pencocokan barang yang datang, sedangkan QC memeriksa kualitas dari *raw material* yang datang.
5. Dokumen hasil pemeriksaan kedatangan *raw material* menjadi acuan untuk menentukan *raw material* itu NG atau tidak.
6. Jika kualitas dan kuantitas *raw material* yang diterima tidak sesuai (NG) dengan dokumen pemeriksaan maka dilakukan konfirmasi ketidak sesuaian dan penanganan *raw material* NG.
7. Jika kualitas dan kuantitas *raw material* OK maka akan disimpan di *warehouse/gudang*.
8. Surat jalan ditandatangani oleh petugas administrasi *warehouse* jika sudah sesuai.

Berikut merupakan *flowmap* proses penerimaan *raw material* di *warehouse*:



Gambar IV.10 Flowchart Yang Sedang Berjalan di Warehouse
(Sumber: Pengolahan Data, 2018)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan *Functional* Sistem

Berdasarkan hasil pengolahan data dan analisis sistem, maka dibutuhkan perancangan aplikasi sistem informasi penerimaan bahan baku pada *warehouse* pada PT Galih Ayom Paramesti. Berikut adalah daftar kebutuhan *Functional* sistem untuk aplikasi sistem informasi penerimaan bahan baku pada *warehouse*.

Tabel V.1 Kebutuhan *Functional* Sistem

No.	Masalah	Kebutuhan User	Kebutuhan <i>Functional</i> Sistem	Uraian
1.	Pengisian <i>form</i> penerimaan bahan baku dan laporan bulanan dilakukan dengan <i>input</i> melalui <i>Microsoft Excel</i> dimana admin seringkali salah ketik karena kesulitan dalam melihat data	a. Dapat mengisi <i>form</i> penerimaan bahan baku dengan mudah dan cepat	a. Terdapat form penerimaan bahan baku yang simple untuk mencegah kesalahan dalam menginput jumlah bahan baku yang masuk	a. Proses yang dapat dilakukan pada penerimaan bahan baku yaitu: Dapat menambah dan mencari data penerimaan bahan baku

	pada <i>sheet</i> dan dokumen yang berbeda.	b. Pembuatan Laporan penerimaan bahan baku lebih cepat tanpa input ulang data	b. Menyediakan menu laporan bulanan	b. Pembuatan laporan penerimaan bahan baku secara otomatis dan terdapat menu untuk mencetak laporan
--	---	---	-------------------------------------	---

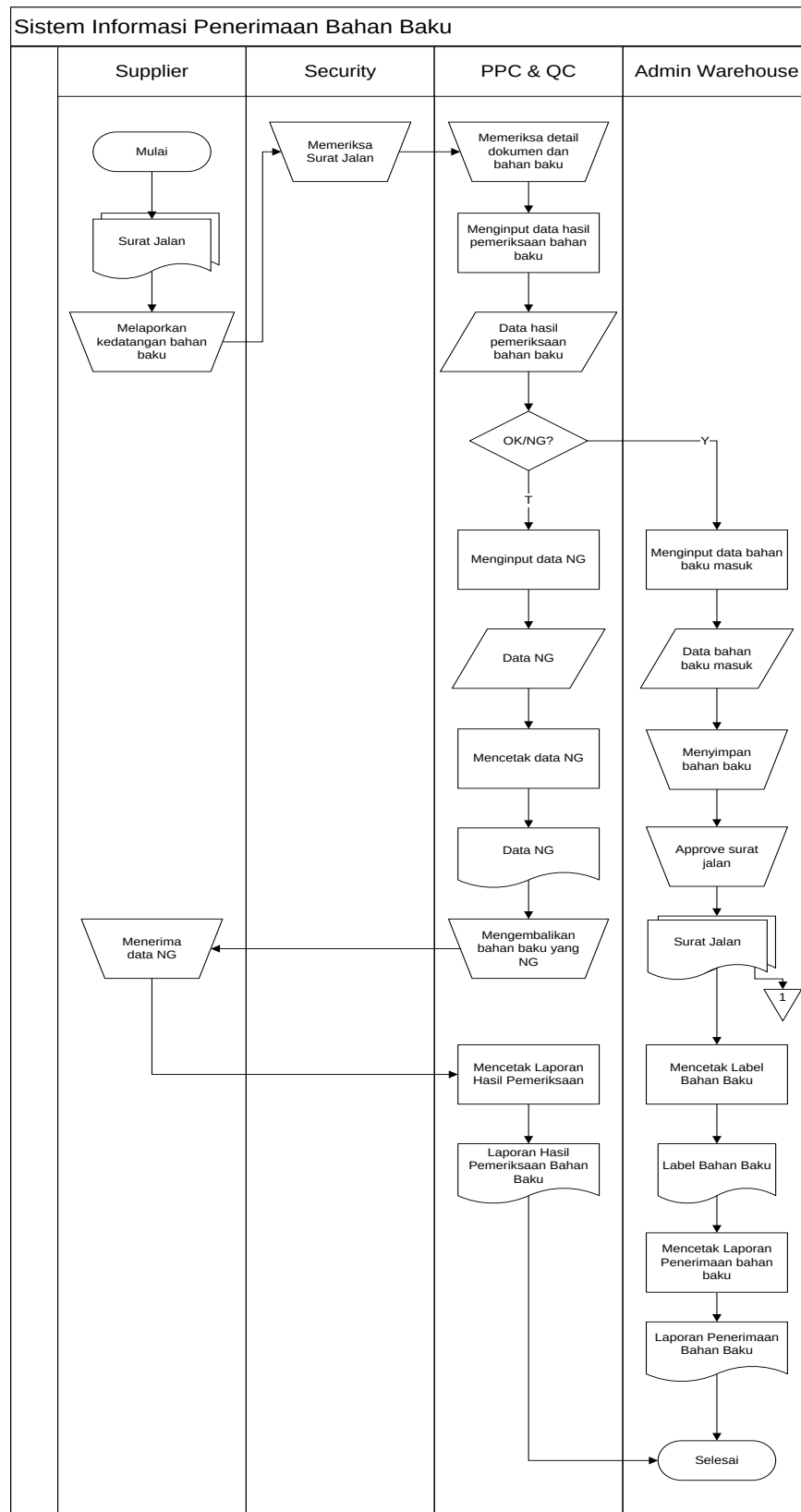
Tabel V.1 Kebutuhan *Functional* Sistem (Lanjutan)

No.	Masalah	Kebutuhan User	Kebutuhan <i>Functional</i> Sistem	Uraian
2.	Admin menyimpan dokumen di sembarang tempat yang disebabkan kurangnya tempat penyimpanan dokumen sehingga berisiko terjadinya kehilangan data, sulit mencari data dan kerusakan data.	Pengelolaan data yang mudah dan cepat	Pengelolaan Data Master	Data master diantaranya adalah data <i>user</i> , data bahan baku, data <i>supplier</i> . Proses yang dapat dilakukan pada pengelolaan data tersebut adalah: - Dapat menambah, mengubah, mencari dan menghapus data tersebut.

(Sumber: Hasil analisis, 2018)

5.2 Flowmap Sistem Usulan

Proses yang diusulkan bertujuan untuk mengembangkan prosedur dan proses suatu data agar tujuan dari suatu organisasi dapat tercapai. Perancangan dibuat untuk meminimalkan kekurangan dari sistem lama yang kurang efektif dan efisien ke sistem baru yang terintegrasi dengan basis data sehingga lebih mudah dan cepat dalam pengerjaan proses penerimaan dan pengecekan bahan baku di PT Galih Ayom Paramesti *Flowmap* Program Usulan Sistem Penerimaan Bahan Baku dapat dilihat pada Gambar V.1.



Gambar V.1 *Flowmap* Usulan Sistem Informasi Penerimaan Bahan Baku
(Sumber: Hasil analisis, 2018)

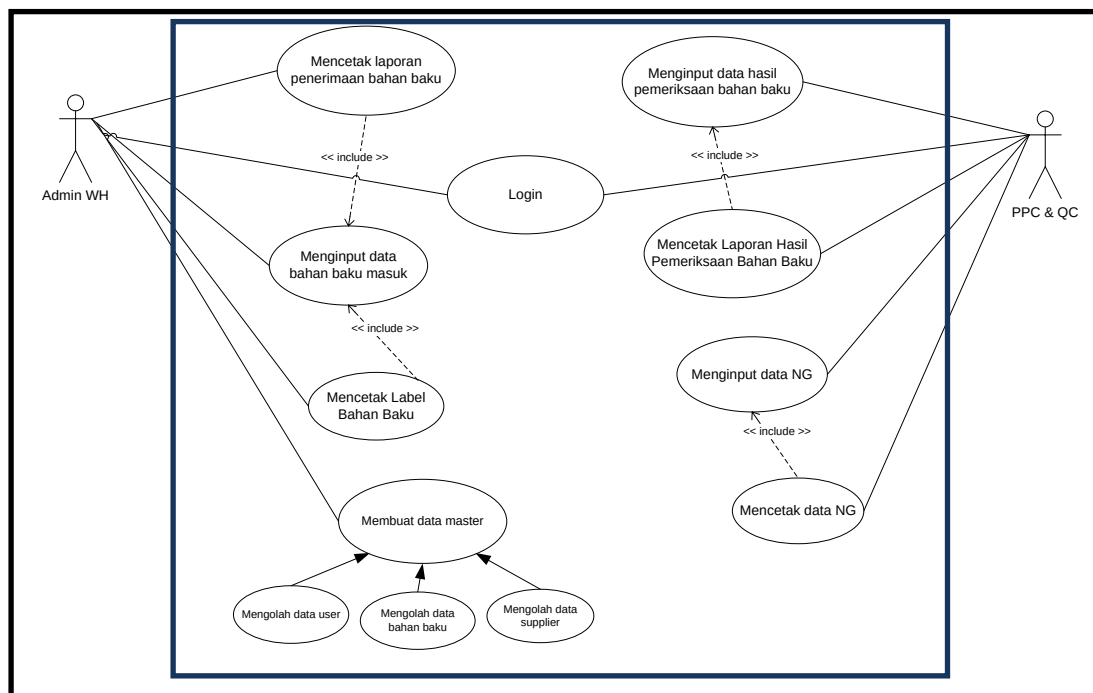
5.3 Pemodelan Sistem Informasi

Penerimaan Bahan Baku

Pemodelan sistem informasi penerimaan bahan baku yang diusulkan menggunakan pemodelan sistem UML (*Unified Modeling Language*) yang meliputi pembuatan *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, *Deployment Diagram* dan juga pembuatan Kamus Data. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi yang akan dibangun sebagai berikut:

5.3.1 Use case diagram

Use case diagram sistem informasi pada penerimaan bahan baku dapat dilihat pada Gambar V.2 di bawah ini:



Gambar V.2 *Use Case Diagram* Usulan Sistem Informasi Penerimaan Bahan Baku

(Sumber: Hasil Analisis, 2018)

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* sistem informasi penerimaan bahan baku usulan:

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

No.	Aktor	Definisi
1.	Admin Warehouse	Admin Warehouse adalah aktor yang bertugas mengelola data <i>master</i> , menginput data bahan baku masuk, mencetak laporan penerimaan bahan baku, mencetak label bahan baku.
2.	PPC & QC	PPC & QC adalah aktor yang bertugas menginput hasil pemeriksaan bahan baku, menginput data NG, dan mencetak laporan hasil pemeriksaan dan data NG

(Sumber: Hasil Analisis, 2018)

2. Skenario *Use Case*

Skenario *use case* sistem informasi penerimaan bahan baku usulan dapat dilihat pada poin berikut:

a. *Use case login*

Berikut skenario *use case login* yang terdapat pada tabel V.3:

Tabel V.3 Definisi *Use Case Login*

Nama <i>Use Case</i>	<i>Login</i>
<i>Primary Actor</i>	Admin Warehouse dan PPC & QC
<i>Use Case Description</i>	<i>Use case</i> ini menggambarkan <i>login</i> sesuai dengan hak akses dari <i>user</i> . Dalam hal ini hak aksesnya sesuai jabatan.
<i>Relationship</i>	<i>Association</i> : Admin Warehouse, dan PPC & QC
<i>Normal Flow Event</i>	1. <i>User</i> membuka aplikasi. 2. <i>User</i> memasukkan <i>user id</i> dan <i>password</i>. 3. Sistem mengecek ke <i>database</i>, apakah <i>user id</i> dan <i>password</i> benar. 4. Jika <i>user id</i> dan <i>password</i> benar, maka muncul tampilan menu utama. Jika <i>user id</i> dan <i>password</i> salah maka muncul pesan “<i>User does not match</i>”

	<i>password” pada window login.</i>
--	-------------------------------------

(Sumber: Hasil Analisis, 2018)

b. *Use case* Menginput Data Hasil Pemeriksaan Bahan Baku

Berikut skenario *Use case* menginput data hasil pemeriksaan bahan baku yang terdapat pada tabel V.4:

Tabel V.4 Definisi *Use case* menginput data hasil pemeriksaan bahan baku

<i>Nama Use Case</i>	<i>Use case</i> menginput data hasil pemeriksaan bahan baku
<i>Primary Actor</i>	PPC & QC
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses untuk menginput data hasil pemeriksaan bahan baku yang sesuai pesanan atau tidak.
<i>Relationship</i>	<i>Association:</i> PPC & QC <i>Include:</i> Login
<i>Nama Use Case</i>	<i>Use case</i> memeriksa bahan baku
<i>Normal Flow Event</i>	1. PPC & QC memilih menu pengelolaan. 2. PPC & QC memilih sub menu hasil pemeriksaan bahan baku. 3. Sistem menampilkan data sub menu hasil pemeriksaan bahan baku yang dipilih. 4. PPC & QC memasukkan data hasil pemeriksaan bahan baku lalu mengklik tombol cek. 5. Sistem menampilkan data hasil pemeriksaan bahan baku.

(Sumber: Hasil Analisis, 2018)

c. *Use case* mengelola data *master*

Berikut skenario *Use case* mengelola data *master*:

Tabel V.5 Definisi *Use case* mengelola data *master*

<i>Nama Use Case</i>	<i>Use case</i> mengelola data <i>master</i>
<i>Primary Actor</i>	Admin Warehouse
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses untuk melakukan pengolahan data <i>master</i> diantaranya adalah data <i>user</i> , data bahan baku, dan data <i>supplier</i> .
<i>Relationship</i>	<i>Association:</i> Admin Warehouse <i>Include:</i> Login

Tabel V.5 Definisi *Use case* memeriksa bahan baku (lanjutan)

<i>Normal Flow Event</i>	1. Admin Warehouse memilih menu data master. 2. Admin Warehouse memilih sub menu data master. 3. Sistem menampilkan data sub menu data master yang dipilih.
--------------------------	---

	4. Admin Warehouse dapat melakukan proses tambah, ubah, cari dan hapus data <i>user</i> , data bahan baku, dan data <i>supplier</i> . 5. Sistem akan memperbarui <i>database</i> . 6. Sistem akan menampilkan data.
--	---

(Sumber: Hasil Analisis, 2018)

d. *Use case* menginput bahan baku masuk

Berikut skenario *Use case* menginput bahan baku masuk yang terdapat pada tabel V.6:

Tabel V.6 Definisi *Use case* menginput bahan baku masuk

<i>Nama Use Case</i>	<i>Use case</i> menginput bahan baku masuk
<i>Primary Actor</i>	Admin Warehouse
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses menginput data penerimaan bahan baku masuk
<i>Relationship</i>	Association: Admin Warehouse Include: Login
<i>Normal Flow Event</i>	1. Admin Warehouse memilih menu pengelolaan. 2. Admin Warehouse memilih sub menu bahan baku masuk. 3. Sistem menampilkan <i>form</i> bahan baku masuk. 4. Admin Warehouse dapat melakukan proses tambah, ubah, cari dan hapus data penerimaan bahan baku. 5. Sistem akan memperbarui <i>database</i> . 6. Sistem akan menampilkan data.

(Sumber: Hasil Analisis, 2018)

e. *Use case* Menginput Data NG

Berikut skenario *Use case* menginput data NG yang terdapat pada tabel V.7:

Tabel V.7 Definisi *Use case* menginput data NG

<i>Nama Use Case</i>	<i>Use case</i> menginput data NG
<i>Primary Actor</i>	PPC & QC
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses menginput data-data bahan baku yang rusak atau cacat
<i>Relationship</i>	Association: PPC & QC Include: Login
<i>Normal Flow Event</i>	1. PPC & QC memilih menu pengelolaan. 2. PPC & QC memilih sub menu input data NG. 3. Sistem menampilkan <i>form</i> data NG. 4. PPC & QC mengisi <i>form</i> data NG. 5. PPC & QC dapat melakukan proses tambah, ubah, cari, hapus dan cetak data NG.

	6. Sistem akan menampilkan data NG.
--	-------------------------------------

(Sumber: Hasil Analisis, 2018)

f. *Use case* Mencetak Laporan Penerimaan Bahan Baku

Berikut skenario *Use case* mencetak laporan penerimaan bahan baku yang terdapat pada tabel V.8:

Tabel V.8 Definisi *Use case* mencetak laporan penerimaan bahan baku

<i>Nama Use Case</i>	<i>Use case</i> mencetak laporan
<i>Primary Actor</i>	Admin Warehouse
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses mencetak laporan penerimaan bahan baku
<i>Relationship</i>	<i>Association:</i> Admin Warehouse <i>Include:</i> Login, menginput bahan baku masuk, meninput data hasil pemeriksaan bahan baku, menginput data NG
<i>Normal Flow Event</i>	1. Admin Warehouse memilih menu laporan. 2. Admin Warehouse memasukkan tanggal laporan yang ingin di cetak. 3. Sistem menampilkan data laporan yang ingin dicetak.

(Sumber: Hasil Analisis, 2018)

g. *Use case* Mencetak Laporan Hasil Pemeriksaan Bahan Baku

Berikut skenario *Use case* mencetak laporan hasil pemeriksaan bahan baku yang terdapat pada tabel V.9:

Tabel V.9 Definisi *Use case* mencetak laporan hasil pemeriksaan bahan baku

<i>Nama Use Case</i>	<i>Use case</i> mencetak laporan
<i>Primary Actor</i>	PPC & QC
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses mencetak laporan hasil pemeriksaan bahan baku
<i>Relationship</i>	<i>Association:</i> PPC & QC <i>Include:</i> Login, menginput data hasil pemeriksaan bahan baku
<i>Normal Flow Event</i>	1. PPC & QC memilih menu laporan. 2. PPC & QC memasukkan tanggal laporan yang ingin di cetak. 3. Sistem menampilkan data laporan yang ingin dicetak.

(Sumber: Hasil Analisis, 2018)

h. *Use case* Mencetak Label Bahan Baku

Berikut skenario *Use case* mencetak label bahan baku yang terdapat pada tabel V.10:

Tabel V.10 Definisi *Use case* mencetak label bahan baku

<i>Nama Use Case</i>	<i>Use case</i> mencetak laporan
<i>Primary Actor</i>	Admin Warehouse
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses mencetak label bahan baku
<i>Relationship</i>	<i>Association:</i> Admin Warehouse <i>Include:</i> Login, menginput bahan baku masuk
<i>Normal Flow Event</i>	1. Admin Warehouse memilih menu pengelolaan. 2. Admin Warehouse memilih sub menu bahan baku masuk. 3. Admin Warehouse memilih data yang ingin dicetak. 4. Sistem menampilkan data yang ingin dicetak.

(Sumber: Hasil Analisis, 2018)

i. *Use case* Mencetak Data NG

Berikut skenario *Use case* mencetak data NG yang terdapat pada tabel V.11:

Tabel V.11 Definisi *Use case* mencetak laporan hasil pemeriksaan bahan baku

<i>Nama Use Case</i>	<i>Use case</i> mencetak data NG
<i>Primary Actor</i>	PPC & QC
<i>Use Case Description</i>	<i>Use Case</i> ini adalah proses mencetak data bahan baku yang NG
<i>Relationship</i>	<i>Association:</i> PPC & QC <i>Include:</i> Login, menginput data NG
<i>Normal Flow Event</i>	1. PPC & QC memilih menu pengelolaan. 2. PPC & QC memilih sub menu data NG. 3. PPC & QC memilih data yang ingin dicetak. 4. Sistem menampilkan data yang ingin dicetak

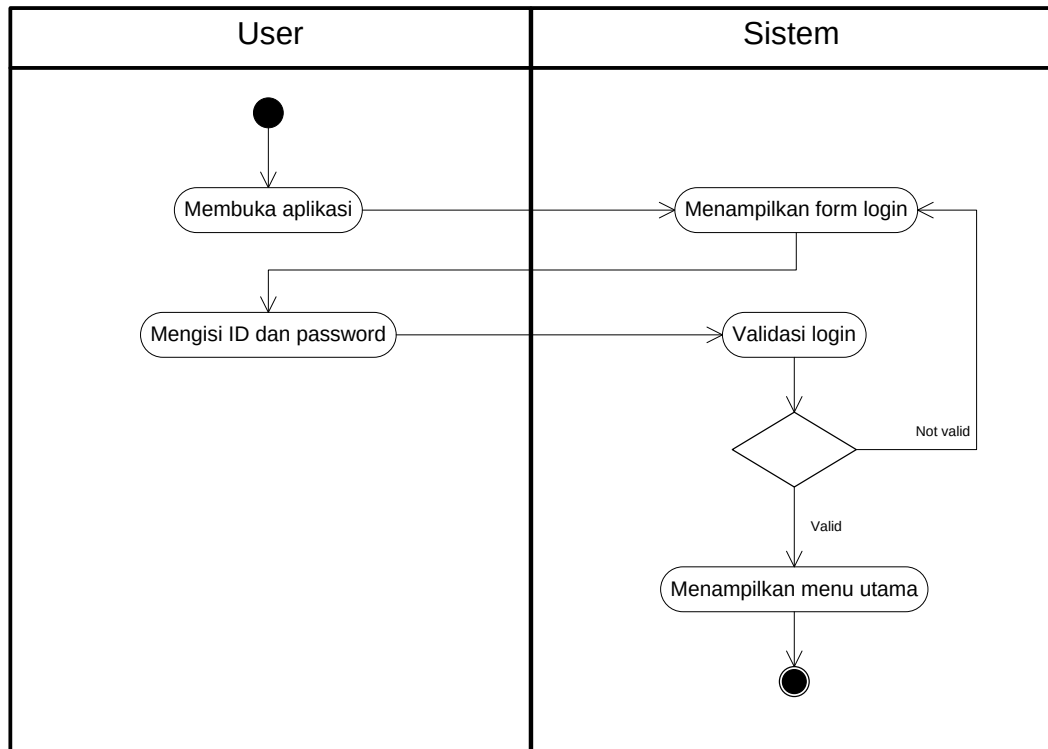
(Sumber: Hasil Analisis, 2018)

5.3.2 Activity Diagram

Activity diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar memahami sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem informasi penerimaan bahan baku sebagai berikut:

1. Activity Diagram Login

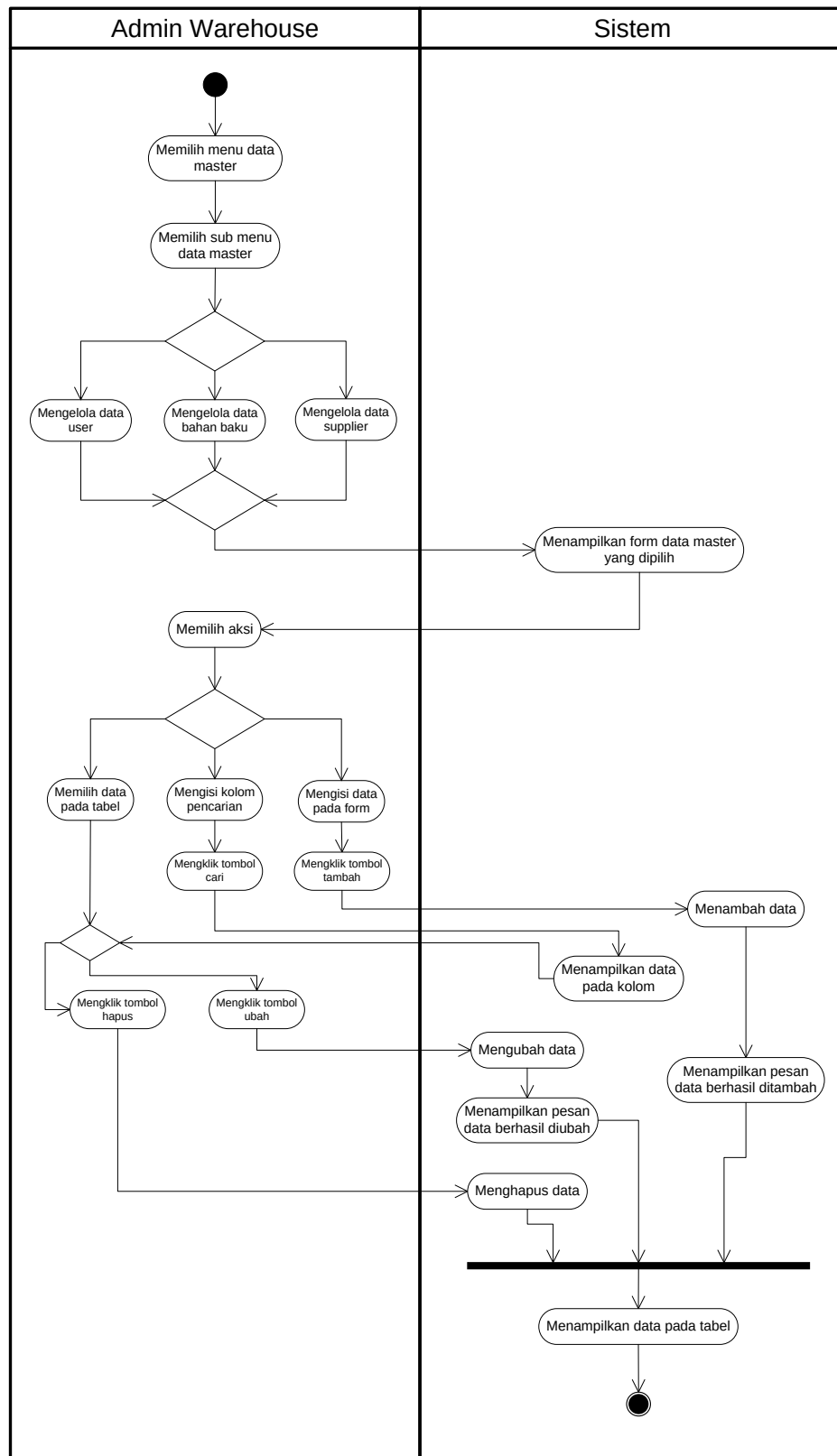
Activity diagram login ini menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu Admin Warehouse, Purchasing, dan PPC & QC untuk dapat masuk ke dalam sistem informasi penerimaan bahan baku. *Activity diagram* dapat dilihat pada Gambar V.3 berikut:



Gambar V.3 Activity Diagram Login
(Sumber: Hasil Analisis, 2018)

2. Activity Diagram Mengelola Data Master

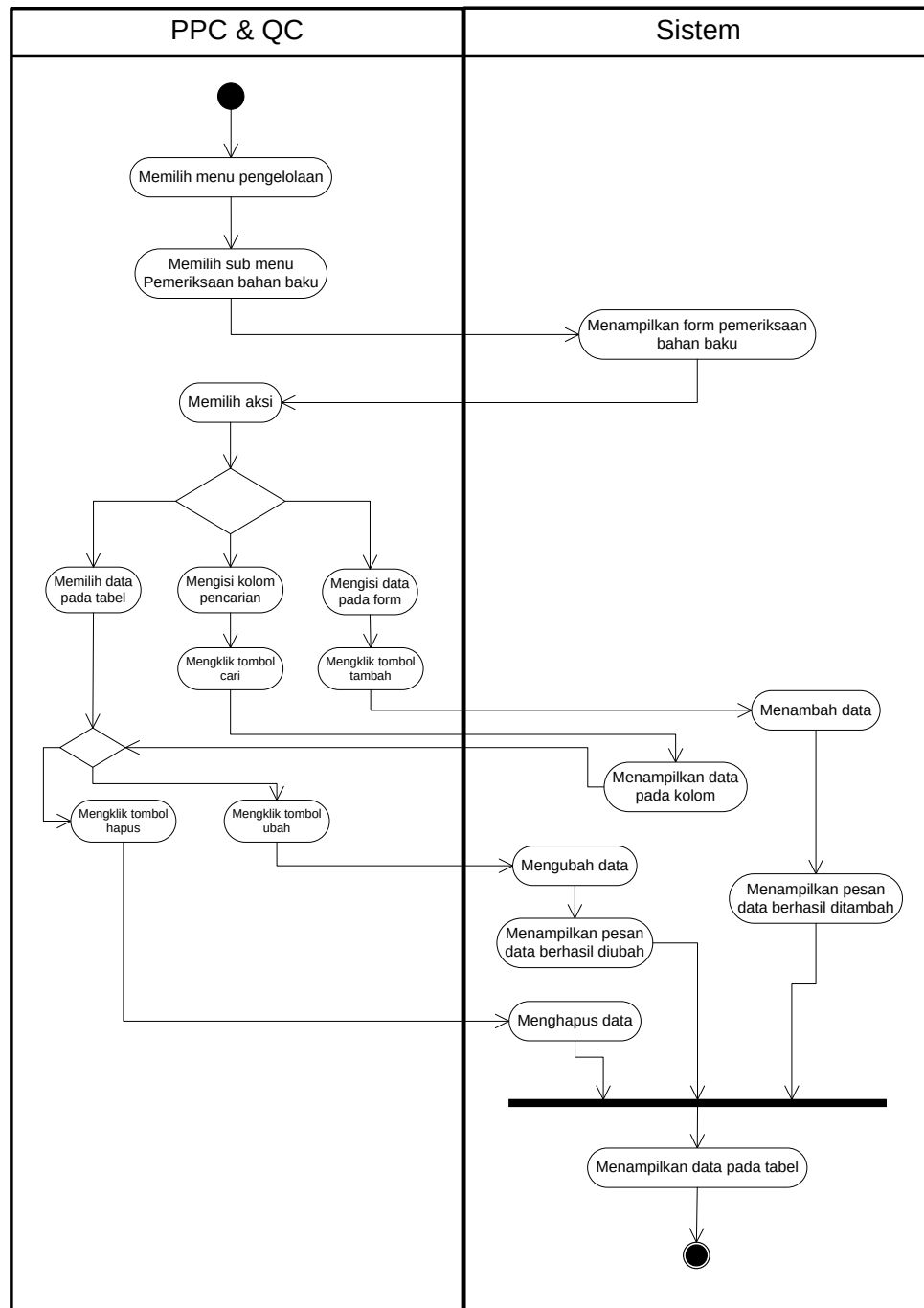
Activity diagram mengelola data master ini menjelaskan alur aktivitas yang dilakukan oleh Admin Warehouse agar data yang ada dalam *database* selalu diperbarui. *Activity diagram* mengelola data master dapat dilihat pada Gambar V.4 sebagai berikut:



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

3. Activity Diagram Pemeriksaan Bahan Baku

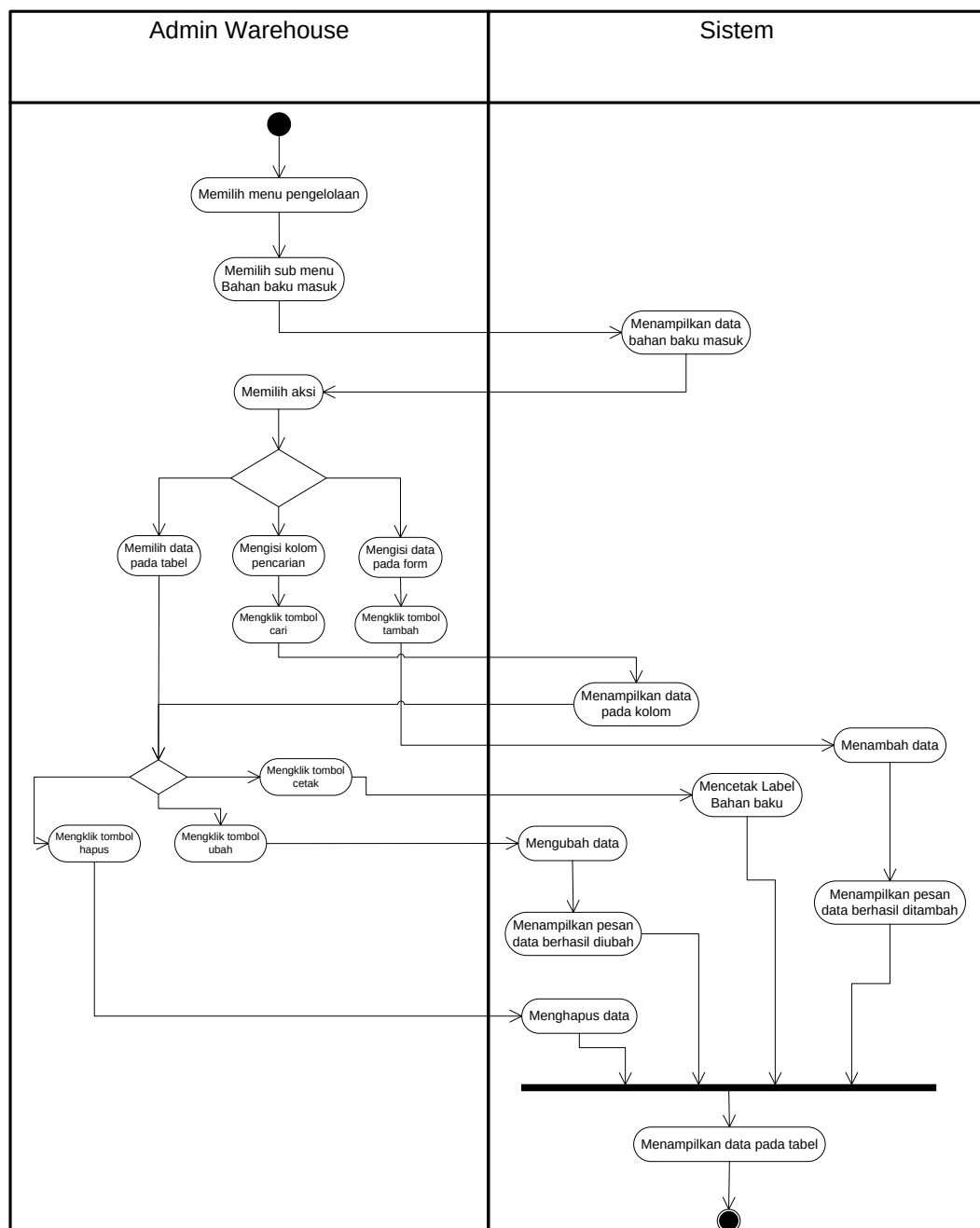
Activity diagram pemeriksaan bahan baku berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada pemeriksaan bahan baku. Activity diagram bahan baku dapat dilihat pada Gambar V.5 sebagai berikut:



Gambar V.5 Activity Diagram Pemeriksaan bahan baku
(Sumber: Hasil Analisis, 2018)

4. Activity Diagram Menginput Bahan Baku Masuk

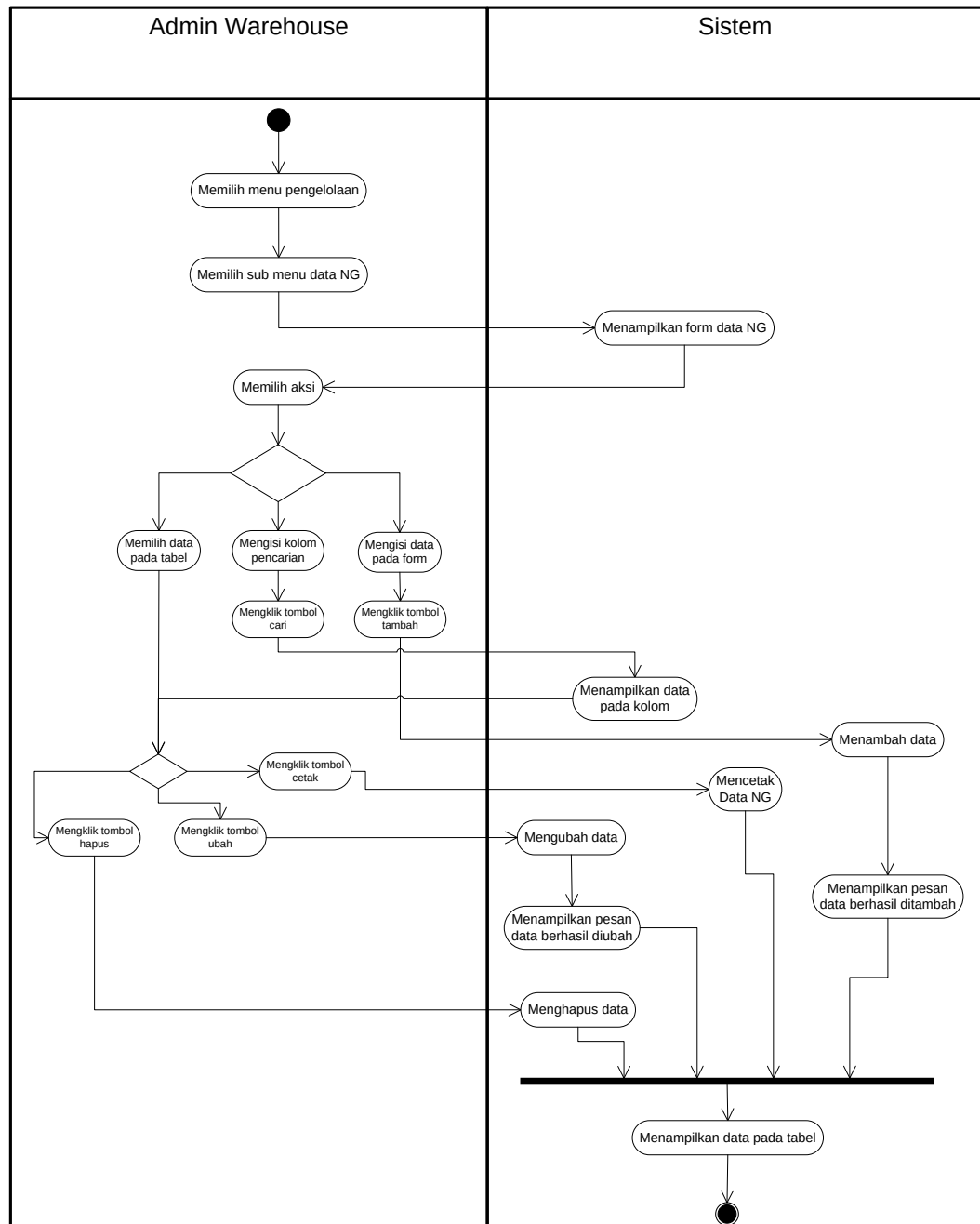
Activity diagram menginput bahan baku masuk berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada penginputan bahan baku masuk. Activity diagram membuat menginput bahan baku masuk dapat dilihat pada Gambar V.6 sebagai berikut:



Gambar V.6 Activity Diagram Menginput Bahan Baku Masuk
(Sumber: Hasil Analisis, 2018)

5. Activity Diagram Menginput data NG

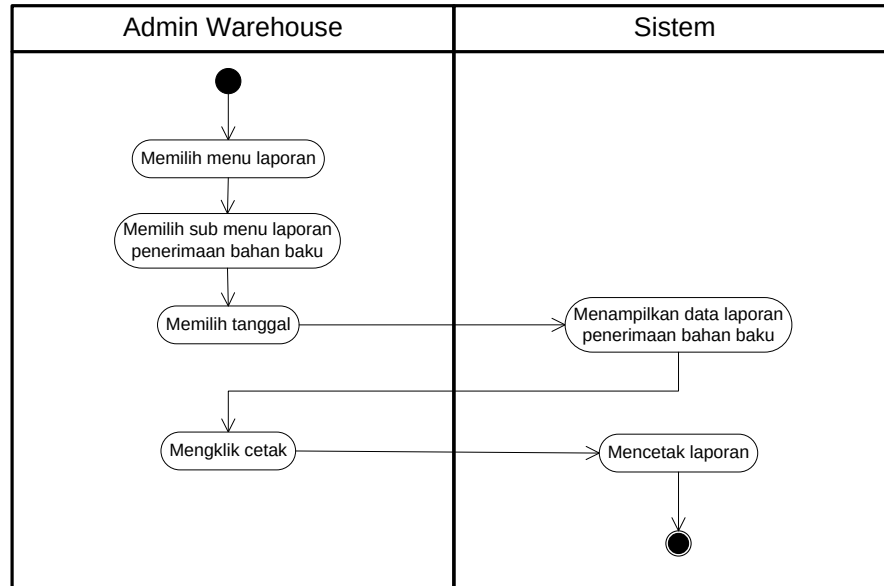
Activity diagram menginput data NG berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada penginputan data NG. Activity diagram menginput data NG dapat dilihat pada Gambar V.7 sebagai berikut:



Gambar V.7 Activity Diagram Menginput data NG
(Sumber: Hasil Analisis, 2018)

6. Activity Diagram Mencetak Laporan Penerimaan Bahan Baku

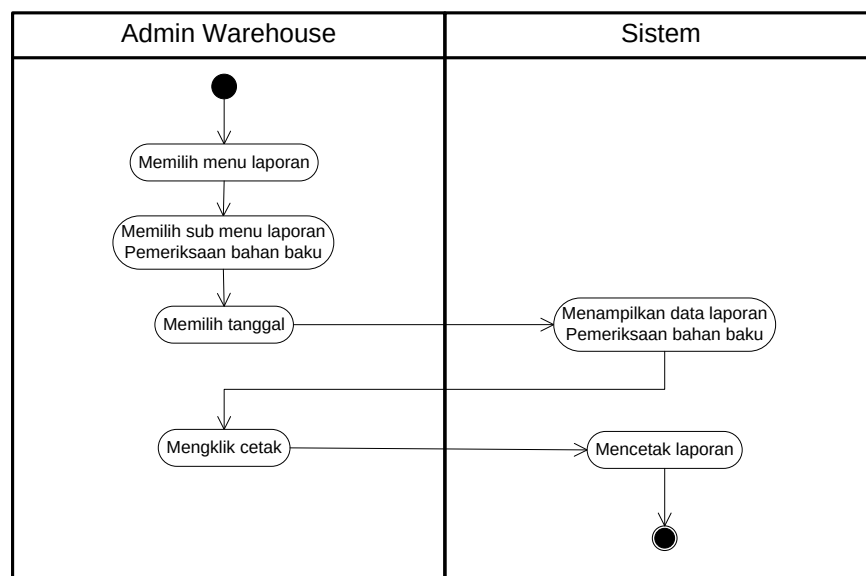
Activity diagram mencetak laporan penerimaan bahan baku dapat dilihat pada Gambar V.8 sebagai berikut:



Gambar V.8 Activity Diagram Mencetak Laporan Penerimaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

7. Activity Diagram Mencetak Laporan Pemeriksaan Bahan Baku

Activity diagram mencetak laporan hasil pemeriksaan bahan baku dapat dilihat pada Gambar V.9 sebagai berikut:



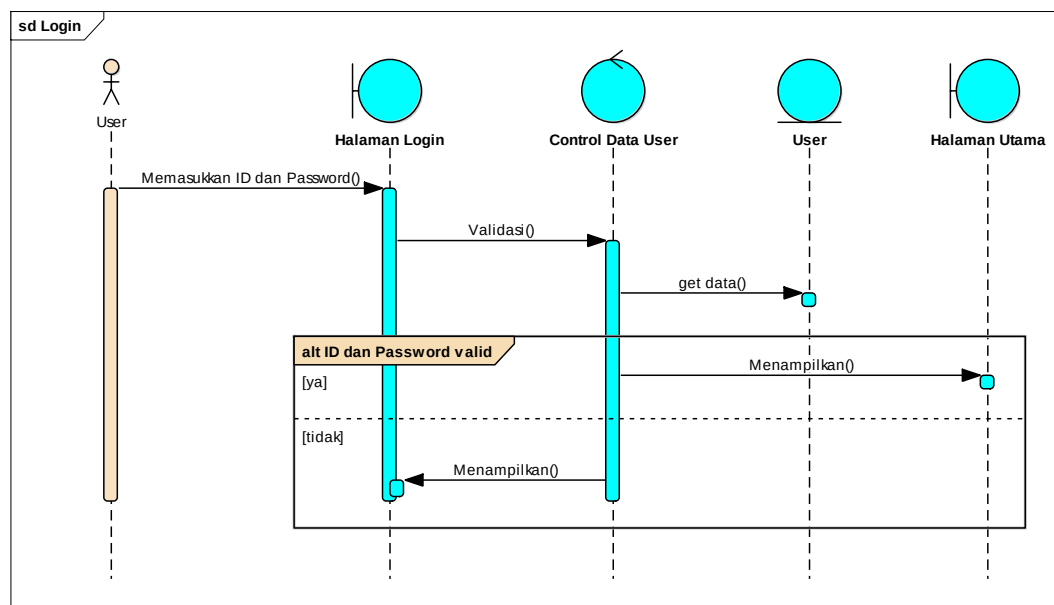
Gambar V.9 Activity Diagram Mencetak Laporan Pemeriksaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

5.3.3 Sequence Diagram

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

1. Sequence Diagram Login

Sequence diagram login menjelaskan sebuah *sequence diagram* dalam proses *login*. Proses ini dilakukan oleh *user* yang mempunyai hak akses pada sistem. Adapun *sequence diagram* dari *use case login* dapat dilihat pada Gambar V.10.

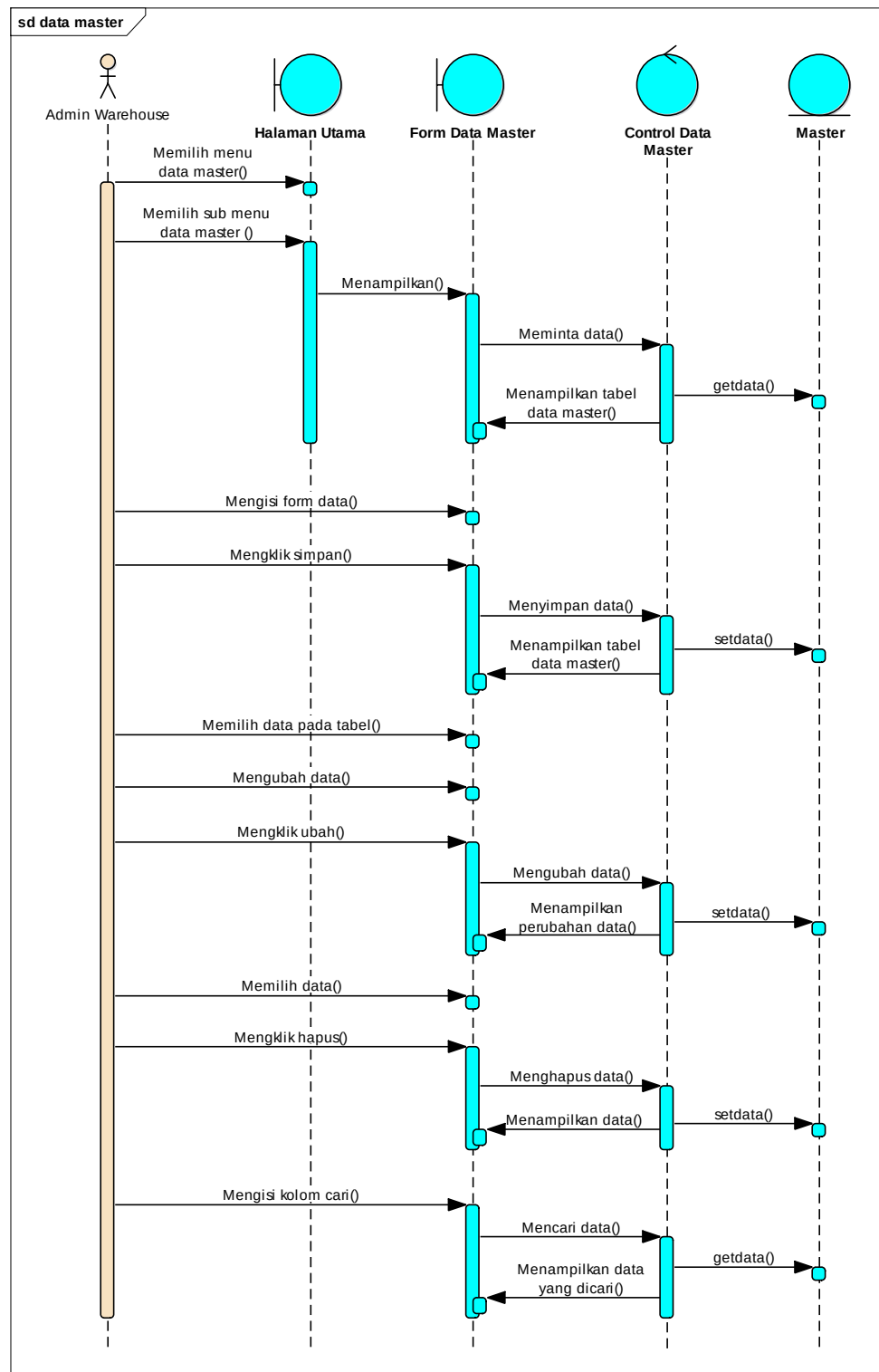


Gambar V.10 Sequence Diagram Login
(Sumber: Hasil Analisis, 2018)

2. Sequence Diagram Mengelola Data Master

Sequence diagram mengelola data *master* ini menjelaskan interaksi yang dilakukan Admin Warehouse dengan sistem dalam mengelola data master yaitu data *User*, data *Bahan Baku*, dan data *Supplier* agar data yang ada dalam

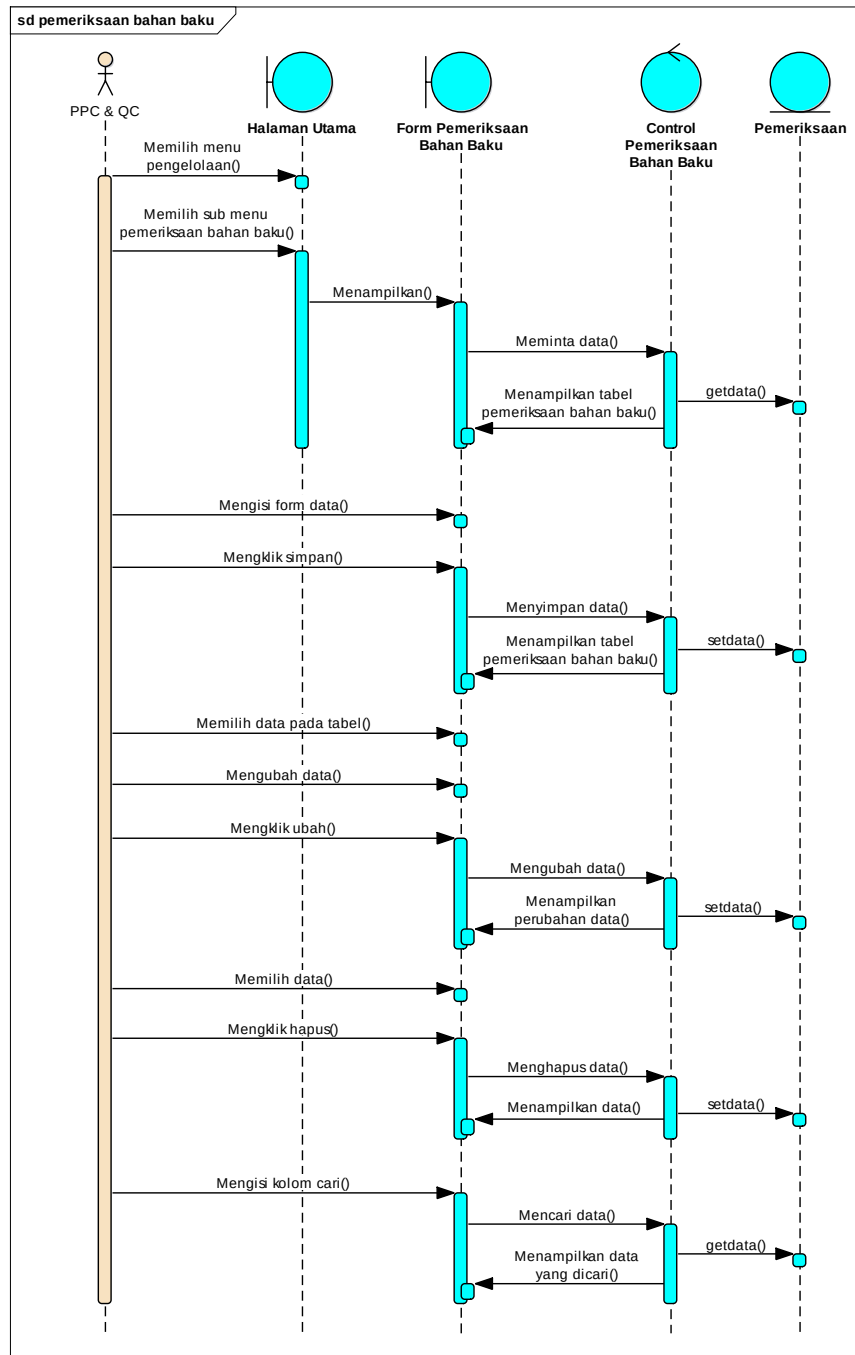
database selalu diperbarui. *Sequence diagram* proses mengelola data *master* dapat dilihat pada Gambar V.11.



Gambar V.11 *Sequence Diagram* Mengelola Data Master
(Sumber: Hasil Analisis, 2018)

3. Sequence Diagram Pemeriksaan Bahan Baku

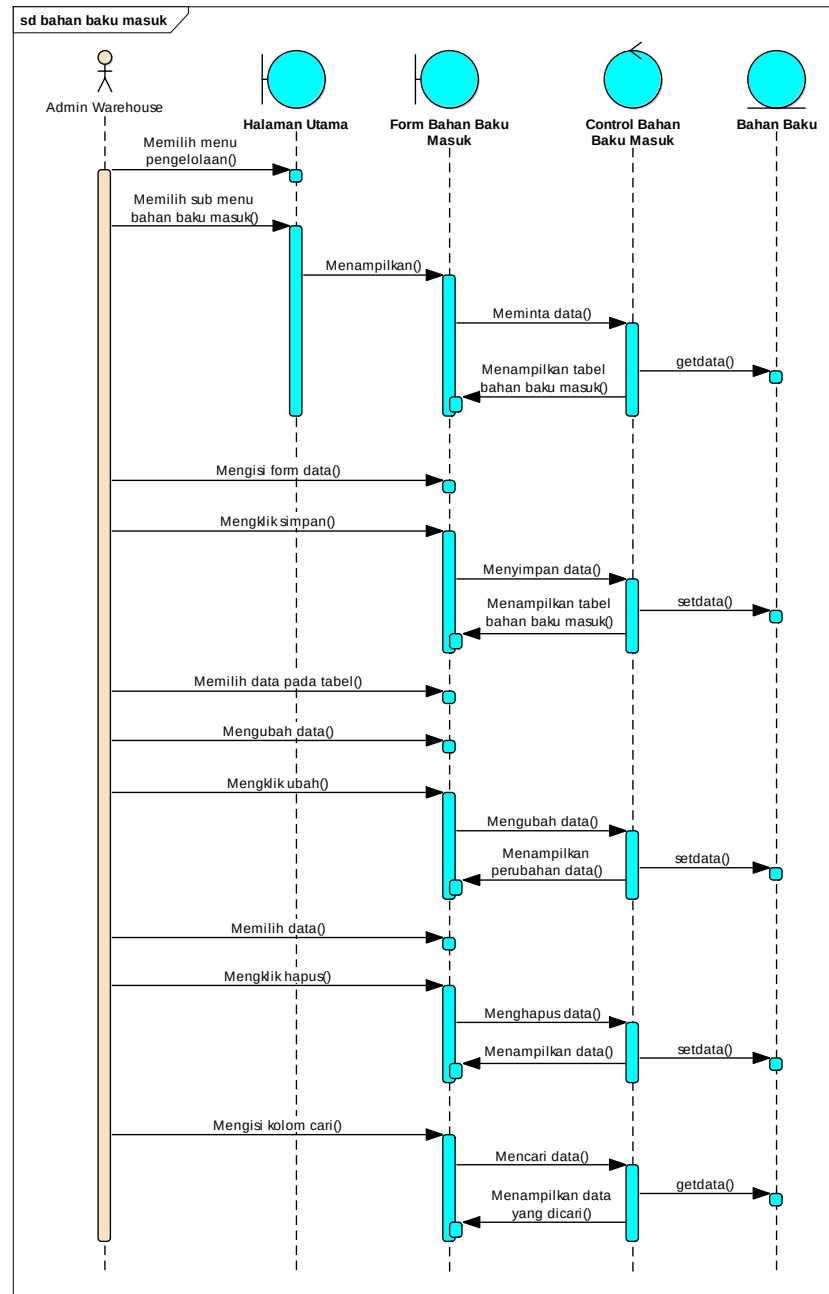
Sequence diagram pemeriksaan bahan baku ini menjelaskan sebuah *sequence diagram* dalam proses memeriksa stok dan kondisi bahan baku yang masuk oleh PPC & QC. *Sequence diagram* proses pemeriksaan bahan baku dapat dilihat pada Gambar V.12.



Gambar V.12 *Sequence Diagram* Pemeriksaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

4. Sequence Diagram Menginput Bahan Baku Masuk

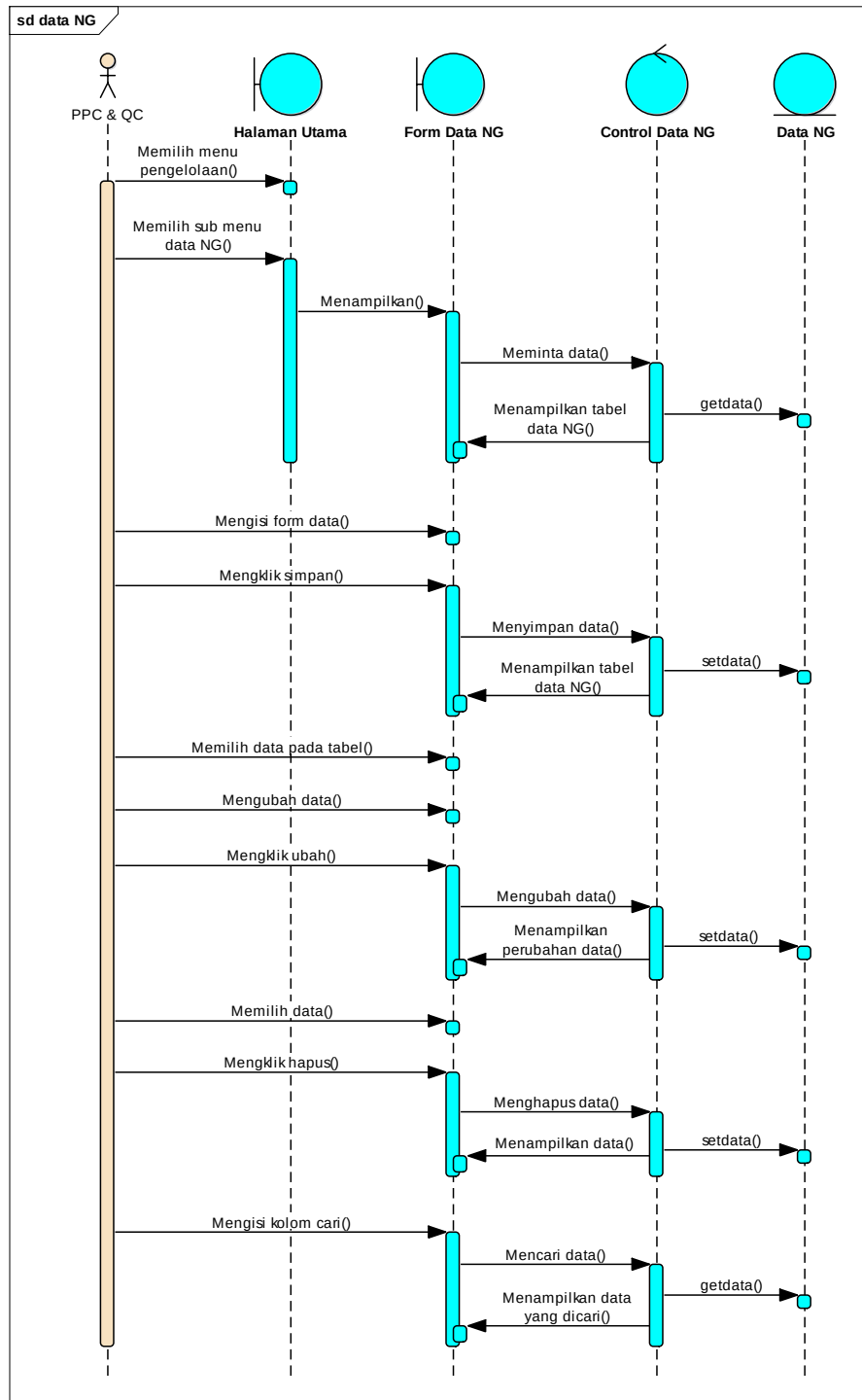
Sequence diagram membuat menginput bahan baku masuk ini menjelaskan sebuah *sequence diagram* dalam proses menginput bahan baku masuk oleh *Part Control*. *Sequence diagram* proses menginput bahan baku masuk dapat dilihat pada Gambar V.13.



Gambar V.13 Sequence Diagram Menginput Bahan Baku Masuk
(Sumber: Hasil Analisis, 2018)

5. Sequence Diagram Menginput Data NG

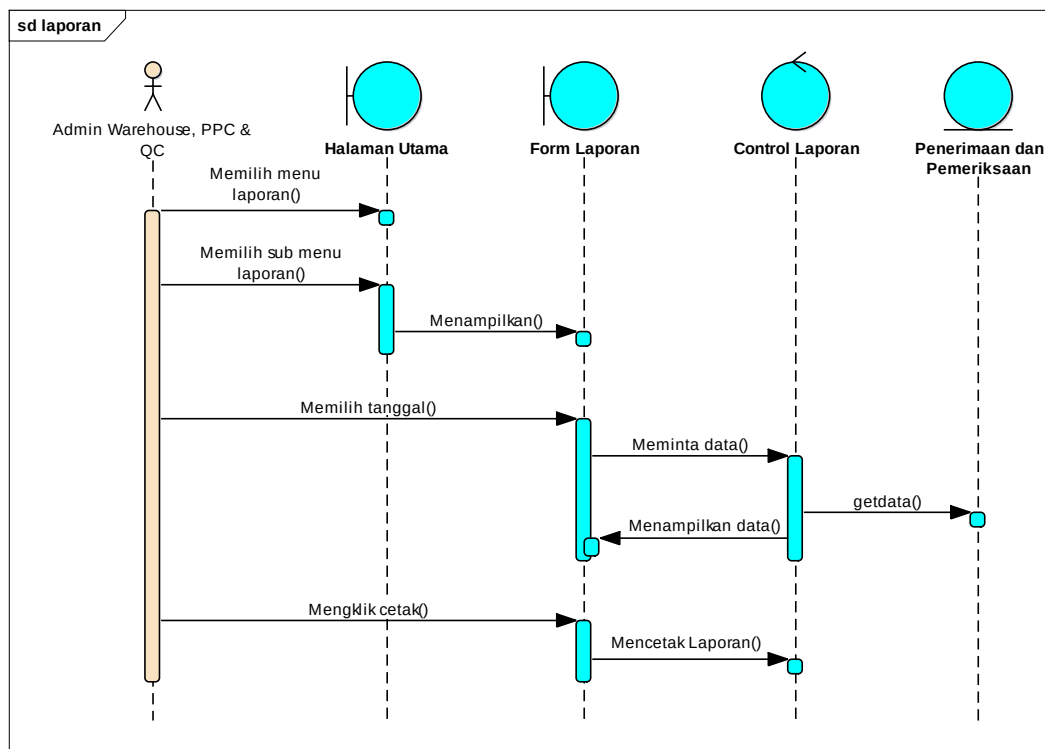
Sequence diagram menginput data NG ini menjelaskan sebuah sequence diagram dalam proses menginput data NG oleh PPC & QC.



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

6. Sequence Diagram Mencetak Laporan

Sequence diagram mencetak laporan menjelaskan sebuah *sequence diagram* dalam proses mencetak laporan pemeriksaan dan penerimaan oleh Admin Warehouse dan PPC & QC. *Sequence diagram* proses mencetak laporan dapat dilihat pada Gambar V.15.

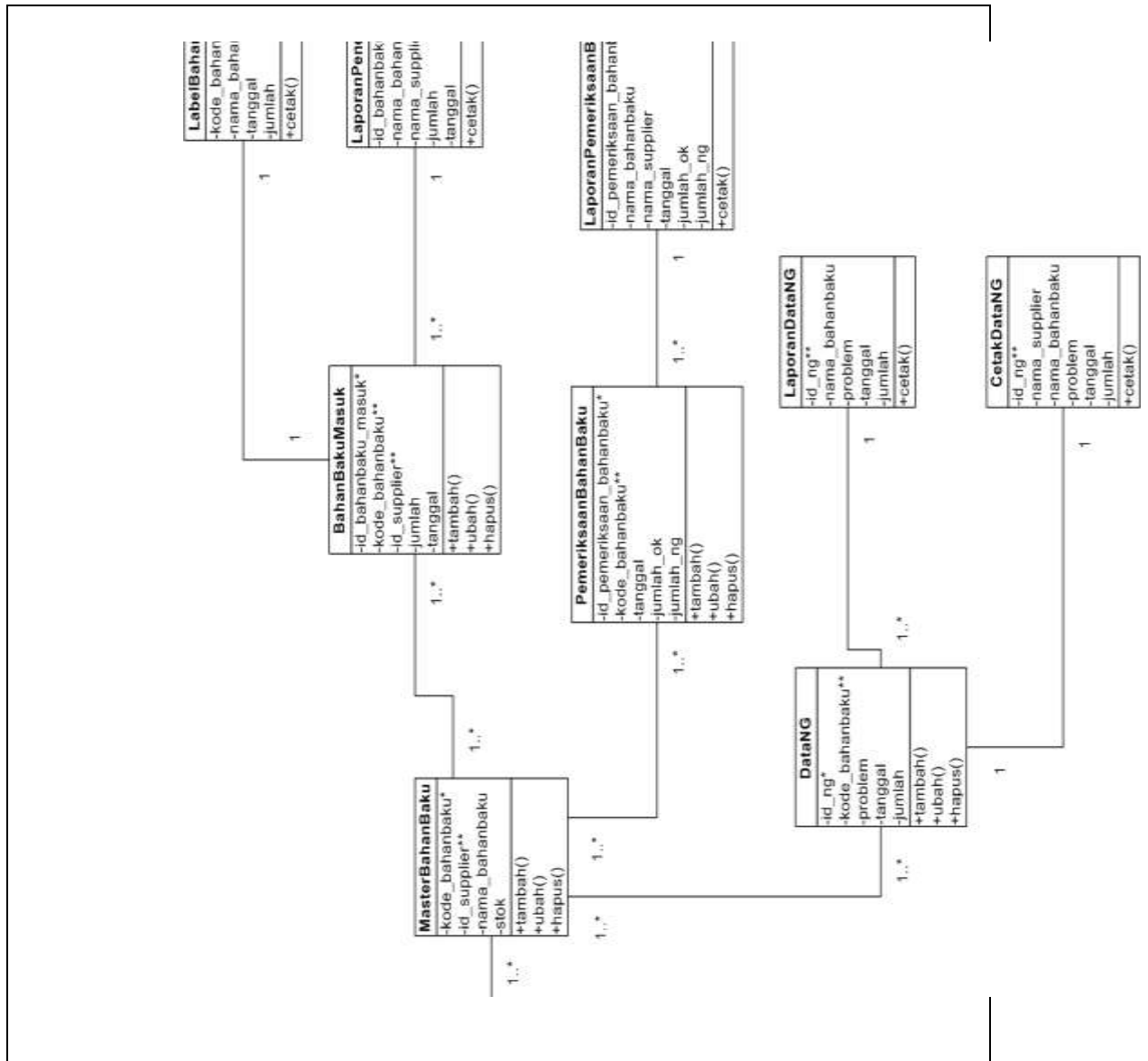


Gambar V.15 *Sequence Diagram* Mencetak Laporan
(Sumber: Hasil Analisis, 2018)

5.3.4 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.

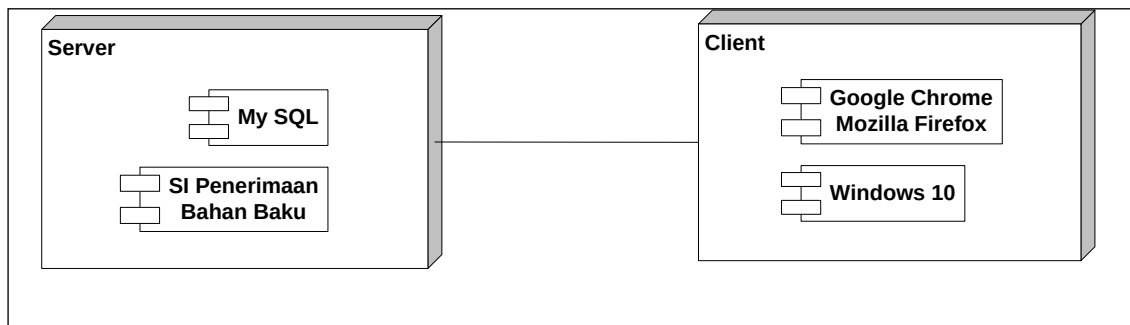
Selama proses analisis, *class diagram* memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama tahap desain, *class diagram* berperan dalam merangkap struktur dari semua kelas yang membentuk arsitektur yang dibuat. *Class diagram* sistem informasi persediaan bahan baku yang diusulkan dapat dilihat pada Gambar V.16.



Gambar V.16 *Class Diagram* Sistem Informasi Penerimaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

5.3.5 Deployment Diagram

Deployment diagram digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur fisik sebuah sistem. *Deployment diagram* sistem informasi penerimaan bahan baku yang diusulkan dapat dilihat pada Gambar V.17.



Gambar V.17 *Deployment Diagram* Sistem Informasi Penerimaan Bahan Baku Usulan

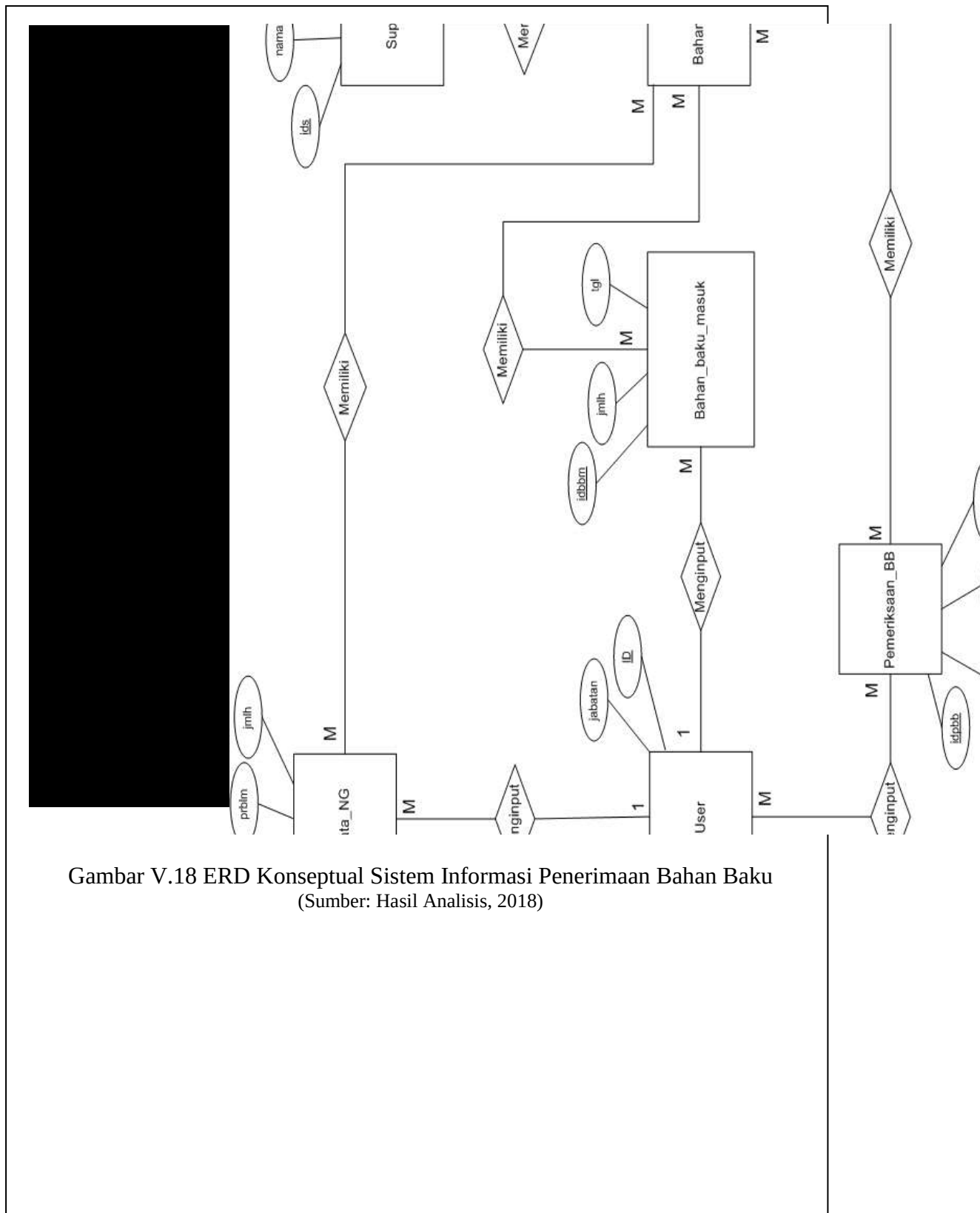
(Sumber: Hasil Analisis, 2018)

5.4 Pemodelan Data

Pemodelan data merupakan metode yang digunakan untuk menentukan dan menganalisis persyaratan data yang diperlukan untuk mendukung proses bisnis suatu organisasi. Pemodelan data ada 2. Yaitu:

5.4.1. Entity Relationship Diagram (ERD)

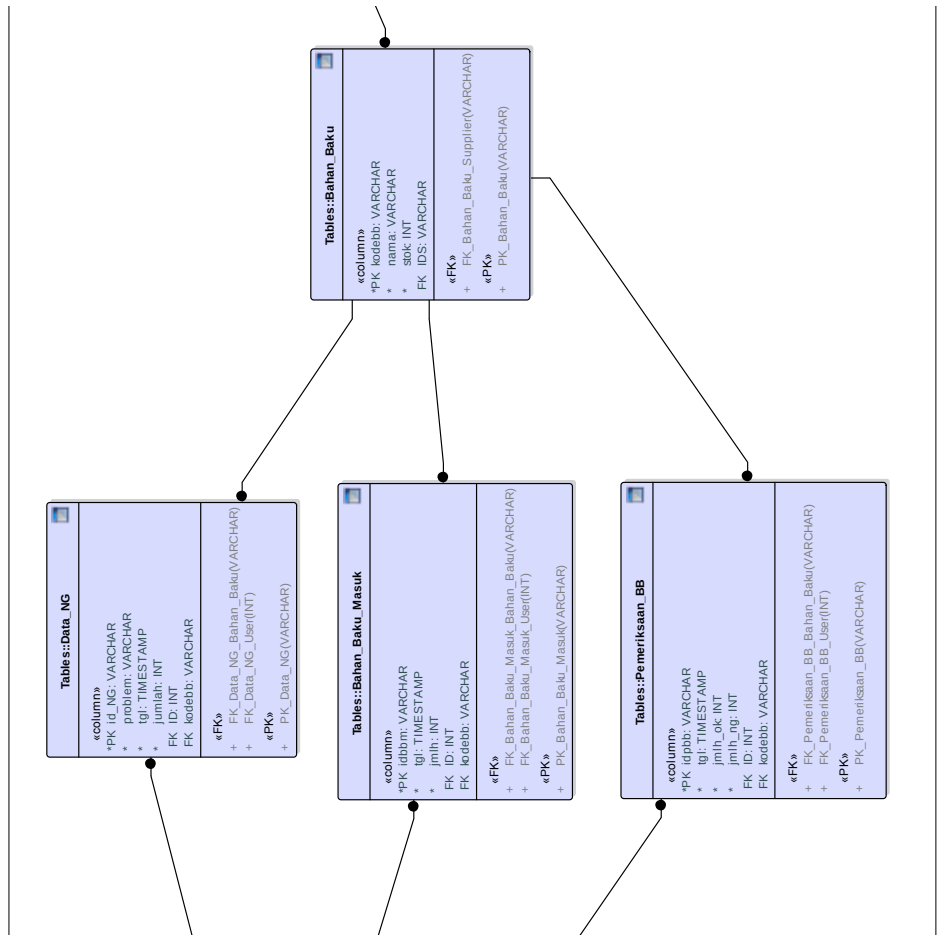
ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarannya digunakan beberapa notasi dan simbol. ERD sistem informasi penerimaan bahan baku yang diusulkan dapat dilihat pada Gambar V.18.



Gambar V.18 ERD Konseptual Sistem Informasi Penerimaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

5.4.2. Conceptual Data Model (CDM)

CDM (*Conceptual Data Model*) merupakan penjabaran lebih lanjut dari ERD. Pada gambar V.19 adalah CDM Sistem Informasi Penerimaan Bahan Baku.



Gambar V.19 CDM Sistem Informasi Penerimaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

5.5 Kamus Data

Kamus data membantu dalam pendefinisian data agar pendefinisian data tersebut dapat dilakukan dengan lengkap dan terstruktur. Pembentukan kamus data dilaksanakan dalam tahap analisis dan perancangan suatu sistem. Berikut adalah kamus data yang menjelaskan isi entitas dari sistem:

1. Spesifikasi Tabel Pengguna

Nama Tabel : *User*

Fungsi : Untuk menyimpan data *user*

Tipe : File data master

Tabel V.12 Tabel User

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	ID	ID_Pengguna	Int	10	Primary Key
2.	nama	Nama_Pengguna	varchar	20	Not Null
3.	pass	Password	Int	7	Not Null
4.	jabatan	Jabatan	varchar	10	Not Null

(Sumber: Hasil Analisis, 2018)

2. Spesifikasi Tabel Bahan Baku

Nama Tabel : Bahan_Baku

Fungsi : Untuk menyimpan data bahan baku

Tipe : File data master

Tabel V.13 Tabel Bahan Baku

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	kodebb	Kodebahanbaku	varchar	10	Primary Key
2.	ids	id_supplier	varchar	10	Foreign Key
3.	nama	namabahanbaku	varchar	20	Not Null
4.	stok	jumlah	Int	3	Not Null

(Sumber: Hasil Analisis, 2018)

3. Spesifikasi Tabel Supplier

Nama Tabel : Supplier

Fungsi : Untuk menyimpan data supplier

Tipe : File data master

Tabel V.14 Tabel Supplier

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	ids	ID_Supplier	varchar	10	Primary Key
2.	nama	namasupplier	varchar	20	Not Null
3.	alamat	alamatupplier	varchar	20	Not Null

4.	cp	contactperson	<i>varchar</i>	20	<i>Not Null</i>
5.	telp	Telepon	<i>Int</i>	12	<i>Not Null</i>
6.	ket	keterangan	<i>varchar</i>	20	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

4. Spesifikasi Tabel Bahan Baku Masuk

Nama Tabel : Bahan_baku_masuk

Fungsi : Untuk menyimpan data bahan baku masuk

Tipe : File data pengelolaan

Tabel V.15 Tabel Bahan Baku Masuk

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	idbbm	id_bahanbakumasuk	<i>varchar</i>	10	<i>Primary Key</i>
2.	kodebb	Kodebahanbaku	<i>varchar</i>	10	<i>Foreign Key</i>
3.	ids	id_supplier	<i>varchar</i>	10	<i>Foreign Key</i>
4.	tgl	Tanggal_masuk	<i>Timestamp</i>		<i>Not Null</i>
5.	jmlh	Jumlah	<i>Int</i>	10	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

5. Spesifikasi Tabel Pemeriksaan Bahan Baku

Nama Tabel : Pemeriksaan_BB

Fungsi : Untuk menyimpan data hasil pemeriksaan bahan baku

Tipe : File data pengelolaan

Tabel V.16 Tabel Pemeriksaan Bahan Baku

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	idpbb	id_pemeriksaan	<i>varchar</i>	10	<i>Primary Key</i>
2.	kodebb	Kodebahanbaku	<i>varchar</i>	10	<i>Foreign Key</i>
3.	tgl	Tgl_pemeriksaan	<i>Timestamp</i>		<i>Not Null</i>

4.	jmlh_ok	Jumlah_OK	<i>Int</i>	3	<i>Not Null</i>
5.	jmlh_ng	Jumlah_NG	<i>Int</i>	3	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

6. Spesifikasi Tabel Data NG

Nama Tabel : Data_NG

Fungsi : Untuk menyimpan data NG

Tipe : File data pengelolaan

Tabel V.17 Tabel Data NG

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	id_ng	ID_NG	<i>varchar</i>	10	<i>Primary Key</i>
2.	kodebb	Kodebahanbaku	<i>varchar</i>	10	<i>Foreign Key</i>
3.	prblm	Problem	<i>varchar</i>	20	<i>Not Null</i>
4.	tgl	Tanggal	<i>Timestam p</i>		<i>Not Null</i>
5.	jmlh	Jumlah	<i>Int</i>	3	<i>Not Null</i>

(Sumber: Hasil Analisis, 2018)

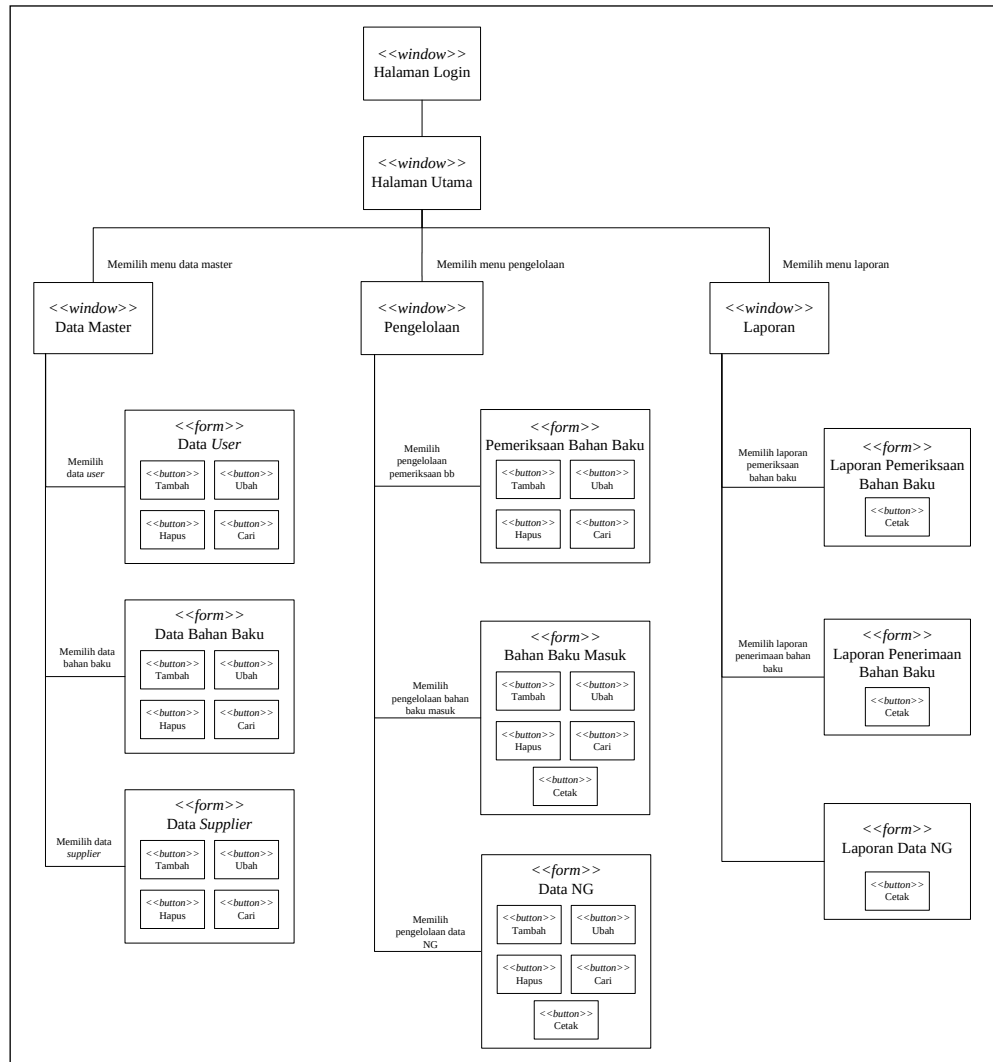
5.6 Analisis Desain Program

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

5.6.1 Windows Navigation Diagram (WND)

Windows Navigation Diagram (WND) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hirarkis. Diagram ini memuat semua modul judul yang ada dalam sistem beserta nama dan nomornya. Diagram ini menunjukkan *window-window* tersebut. Sebuah window dapat digambarkan sebagai sebuah *state*. *State* ini memiliki nama dan berisi gambar *miniature* window. Transisi antar *state*

dipicu oleh ditekannya sebuah tombol yang menghubungkan dua *window*. Perancangan WND aplikasi usulan sistem informasi penerimaan bahan baku dapat dilihat pada Gambar V.20



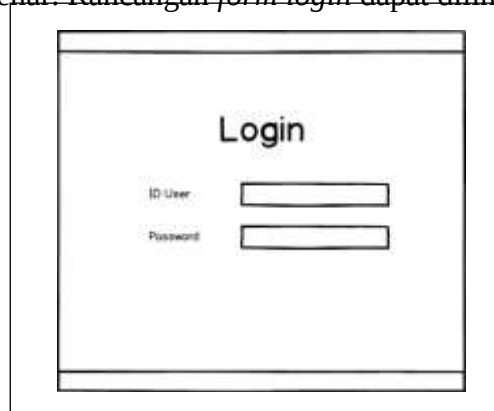
Gambar V.20 WND Sistem Informasi Penerimaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

5.6.2 Perancangan *Interface* Sistem

Rancangan *interface* dari program penerimaan dan pengiriman komponen ini adalah sebagai berikut:

1. *Form Login*

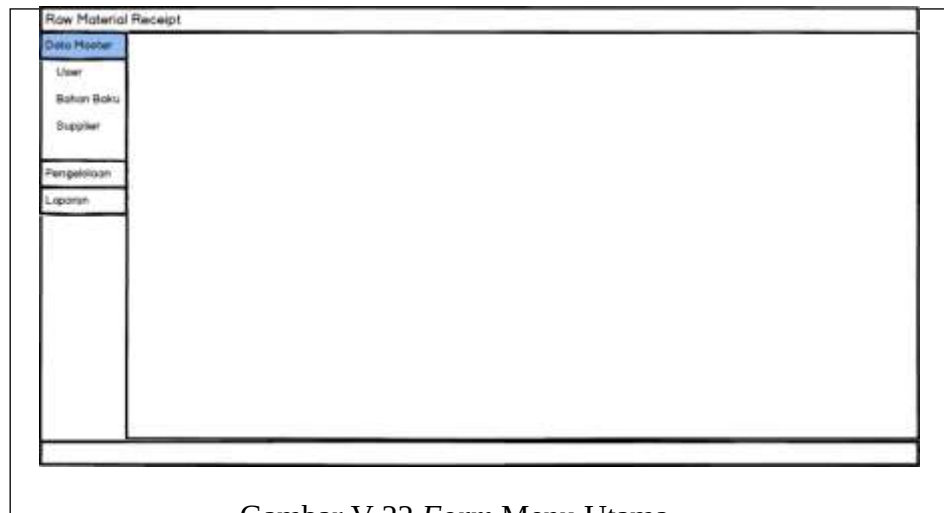
Form login adalah *form* yang digunakan untuk masuk ke dalam program aplikasi. Untuk masuk ke dalam aplikasi, pengguna harus memasukkan ID dan *password* yang benar. Rancangan *form login* dapat dilihat pada Gambar V.21.

A rectangular window titled "Login" with a thin border. Inside, the word "Login" is centered at the top. Below it, there are two horizontal input fields. The first field is preceded by the text "ID User" and the second field is preceded by the text "Password".

Gambar V.21 *Interface Login*
(Sumber: Hasil Analisis, 2018)

2. *Form Menu Utama*

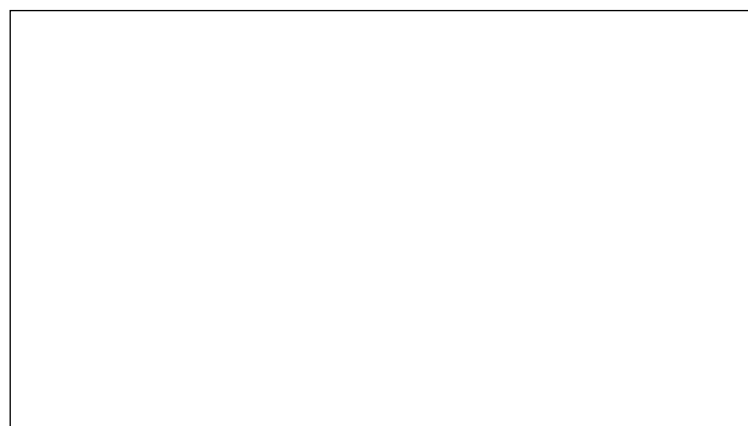
Form menu utama adalah tampilan awal pada aplikasi yang berisi empat menu. Menu tersebut adalah Data Master, Pengelolaan, dan Laporan.

A rectangular window titled "Raw Material Receipt". On the left side, there is a vertical sidebar menu. The first item is "Data Master" and is highlighted with a blue background. Below it are "User", "Bahan Baku", "Supplier", "Pengelolaan", and "Laporan". The main area of the window is a large empty rectangle.

Gambar V.22 *Form Menu Utama*
(Sumber: Hasil Analisis, 2018)

3. *Form Data Bahan Baku*

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

A large, empty rectangular box with a thin black border, intended for the design of the data management form.

Gambar V.23 *Interface Form Data Master Bahan Baku*
(Sumber: Hasil Analisis, 2018)

4. *Form Data Supplier*

Form data supplier adalah *form* yang digunakan untuk mengelola data *supplier*. Rancangan *interface* dari *form data supplier* dapat dilihat pada Gambar V.24 berikut:

Gambar V.24 *Interface Form Data Master Supplier*
(Sumber: Hasil Analisis, 2018)

5. *Form Data User*

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

Gambar V.25 *Interface Form Data Master User*
(Sumber: Hasil Analisis, 2018)

6. *Form Data Pemeriksaan Bahan Baku*

Form data pemeriksaan bahan baku adalah *form* yang digunakan untuk pemeriksaan bahan baku yang tersedia. Rancangan *interface* dari *form* pemeriksaan bahan baku dapat dilihat pada Gambar V.26 berikut:

Gambar V.26 *Interface Form Pemeriksaan Bahan Baku*
(Sumber: Hasil Analisis, 2018)

7. *Form Bahan Baku Masuk*

Form bahan baku masuk adalah *form* yang digunakan untuk mengelola data bahan baku yang masuk di bagian *Warehouse*. Pada *interface form* bahan baku masuk terdapat empat tombol, yaitu cari, tambah, ubah dan hapus. Tombol cari digunakan untuk mencari data bahan baku masuk yang diinginkan. Tombol simpan atau ubah digunakan untuk menyimpan data bahan baku

masuk ke dalam database. Sedangkan tombol hapus digunakan untuk menghapus data bahan baku masuk yang dipilih. Data bahan baku masuk, dikelola oleh Admin Warehouse. Rancangan *interface* dari *form* bahan baku masuk dapat dilihat pada Gambar V.27 berikut:

The screenshot shows a web application interface titled 'Raw Material Receipt'. On the left is a sidebar menu with options: 'Data Master', 'Pergerakan', 'SD', 'Penerimaan', 'Bahan Baku' (highlighted), and 'Laporan'. The main content area is titled 'Form Bahan Baku Masuk'. It contains several input fields: 'ID Bahan Baku Masuk' (text), 'Kode Bahan Baku' (dropdown), 'ID Supplier' (dropdown), 'Tanggal' (calendar icon), and 'Jumlah' (text). To the right of these fields is a placeholder for a 'Logo'. Below the fields are three buttons: 'Ubah', 'Hapus', and 'Simpan'. There is also a search bar with a magnifying glass icon and the text 'Cari'. At the bottom is a table with the following columns: 'ID Bahan Baku Masuk', 'Kode Bahan Baku', 'ID Supplier', 'Tanggal', and 'Jumlah'. The table has three empty rows for data entry.

Gambar V.27 *Interface Form Bahan Baku Masuk*
(Sumber: Hasil Analisis, 2018)

8. *Form* Laporan Penerimaan Bahan Baku

Form Laporan penerimaan bahan baku adalah *form* yang digunakan untuk mencetak laporan penerimaan bahan baku pada bagian Warehouse dengan memilih tanggal laporan yang ingin dicetak. *Form* ini dikelola oleh Admin Warehouse. Rancangan *interface* dari *form* laporan penerimaan bahan baku dapat dilihat pada Gambar V.28 berikut:

Raw Material Receipt

Data Master

Pengelolaan

Laporan

Laporan Hasil

Laporan

Laporan Penerimaan Bahan Baku

Tanggal

Logo

Tanggal	Kode Bahan Baku	Nama Bahan Baku	ID Supplier	Nama Supplier	Jumlah

Cetak

Gambar V.28 *Interface Form* Laporan Penerimaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

9. *Form* Laporan Hasil Pemeriksaan Bahan Baku

Form Laporan hasil pemeriksaan bahan baku adalah form yang digunakan untuk mencetak laporan hasil pemeriksaan bahan baku dengan memilih tanggal laporan yang ingin dicetak. Form ini dikelola oleh Admin Warehouse. Rancangan *interface* dari *form* laporan hasil pemeriksaan bahan baku dapat dilihat pada Gambar V.29 berikut:

Raw Material Receipt

Data Master

Pengelolaan

Laporan

Laporan Hasil

Laporan

Laporan Hasil Pemeriksaan Bahan Baku

Tanggal

Logo

Tanggal	ID Pemeriksaan	Kode Bahan Baku	Nama Bahan Baku	Jumlah OK	Jumlah NG

Cetak

Gambar V.29 *Interface Form* Laporan Hasil Pemeriksaan Bahan Baku
(Sumber: Hasil Analisis, 2018)

10. Form Data NG

Form data NG adalah *form* yang digunakan untuk menginput data bahan baku yang NG. Rancangan *interface* dari *form* data NG dapat dilihat pada Gambar V.30 berikut:

ID Data NG	Kode Bahan Baku	Problem	Tanggal	Jumlah

Gambar V.30 *Interface Form Data NG*
(Sumber: Hasil Analisis, 2018)

11. Form Laporan Data NG

Form Laporan data NG adalah form yang digunakan untuk mencetak laporan data bahan baku yang NG dengan memilih tanggal laporan yang ingin dicetak. Form ini dikelola oleh Admin Warehouse. Rancangan *interface* dari *form* laporan data NG dapat dilihat pada Gambar V.31 berikut:

Tanggal	ID Data NG	Kode Bahan Baku	Problem	Jumlah

Gambar V.31 *Interface Form Laporan Data NG*
(Sumber: Hasil Analisis, 2018)

5.7 Implementasi Sistem *Software* dan *Hardware*

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

1. Analisis kebutuhan *software*

Software yang dibutuhkan antara lain sebagai berikut:

- a. Laragon 3.4.0 untuk pengelolaan *database*.
- b. Browser Mozilla Firefox atau Google Chrome untuk menjalankan Laragon.
- c. Sistem Operasi minimal Windows XP.

2. Analisis kebutuhan *hardware*

- a. *Processor* : minimal Pentium IV (1,8 GHz).
- b. RAM : minimal 512 MB.
- c. *Harddisk* : minimal 64 GB.
- d. Sistem Operasi : Microsoft Windows XP atau di atasnya.
- e. *Printer* untuk pencetakan dokumen.

Kebutuhan *non functional requirements* adalah secara umum berisi batasan-batasan pada pelayanan atau fungsi yang disediakan oleh sistem. Kebutuhan *non functional requirements* sistem informasi penerimaan dan pengiriman komponen usulan terdapat 2 yaitu:

1. Untuk dapat masuk kedalam sistem ini, Pengguna harus melakukan *login* terlebih dahulu menggunakan ID dan *Password*.
2. Sistem hanya membahas penerimaan bahan baku di bagian *warehouse*, tidak membahas pembelian bahan baku.

BAB VI

KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan hasil rancang bangun sistem yang dilakukan mengenai sistem informasi penerimaan bahan baku di PT Galih Ayom Paramesti dapat disimpulkan beberapa hal yaitu:

1. Dengan dibuatnya sistem informasi penerimaan bahan baku, pembuatan laporan penerimaan bahan baku menjadi lebih cepat dan mudah.
2. Dengan mengimplementasikan sistem informasi penerimaan bahan baku, maka data tersimpan di *database*, sehingga tidak terjadi kehilangan data, kesulitan mencari data, dan kerusakan data.

Saran

Saran yang dapat diberikan untuk pengembangan sistem informasi penerimaan bahan baku ini selanjutnya adalah sebagai berikut:

1. Memperbaiki pembuatan *output* dan *interface* aplikasi sistem informasi penerimaan bahan baku yang lebih baik untuk kenyamanan pengguna.
2. Melakukan pelatihan terlebih dahulu kepada pengguna aplikasi sistem informasi penerimaan bahan baku untuk mencegah kesalahan pengguna.
3. Melakukan perawatan terhadap sistem informasi penerimaan bahan baku tersebut agar jangka waktu penggunaan menjadi lebih lama.

DAFTAR PUSTAKA

- Assauri, Sofjan. 2015. *Manajemen Pemasaran*. Jakarta: Raja Grafindo.
- Assauri, Sofjan. 2015. *Manajemen Operating Produksi*. Jakarta: Raja Grafindo.
- Connolly, Thomas M., Carolyn, E. Begg. 2005. *Database System : A practical approach to design, implamentation, and management*, fourth edition. USA: Pearson Education Limited.
- Dennis, Alan., Barbara Haley., dan David Tegarden. 2015. *System Analysis and Design, Fifth Edition*. America: Willey.inc.
- Kadir, Abdul. 2014. *Pengenalan Sistem Infromasi* Yogyakarta: ANDI.
- Kementrian Riset dan Teknologi. 2013. *Modul Data 1spasial*. Bandung
- M, H.A. *Rusdiana, Irfan, Moch.* 2014. *Sistem Informasi Manajemen*. Bandung: Pustaka Setia. Jaya, I. D.
- McLeod, R. J., Scholl, G. 2012. *Sistem Informasi Manajemen Edisi 11*. Jakarta: Salemba Empat.
- Mulyadi. 2015. *Pengertian Sistem Informasi Persediaan*. Jakarta: Salemba Empat.
- Moleong, Lexy J. 2012. *Metodologi Penelitian Kualitatif*, Bandung: Remaja Rosdakarya.
- Ristono, Agus. 2013. *Manajemen Persediaan*. Yogyakarta: Graha ilmu
- Sjahrial, Dermawan. 2012. *Pengantar Manajemen Keuangan*. Jakarta: Mitra Wacana Media.
- Sofyan, Diana. 2013. *Perencanaan Dan Pengendalian Produksi*. Yogyakarta: Graha Ilmu.
- Sugiyono. 2013. *Metode Penelitian Kuantitatif, Kualitatif dan R&D*. Bandung: Alfabeta.

Suharsaputra, U.2012. *Metode Penelitian Kuantitatif, Kualitatif dan Tindakan*. Bandung: Refika Aditama

Sulindawati, Muhamad Fathoni. 2014. “*Pengantar Analisis Perancangan Sistem.*” Jurnal Saintikom Volume 9, No.2.

Sujarweni, V. Wiratna. 2015. *Metodologi Penelitian*. Yogyakarta: Pustaka Baru Press.

Sutabri, Tata. 2012. *Sistem Informasi Manajemen*.Yogyakarta: ANDI

Taufiq, Rohmat. 2013. *Sistem Informasi Manajemen*.Yogyakarta: Graha Ilmu.

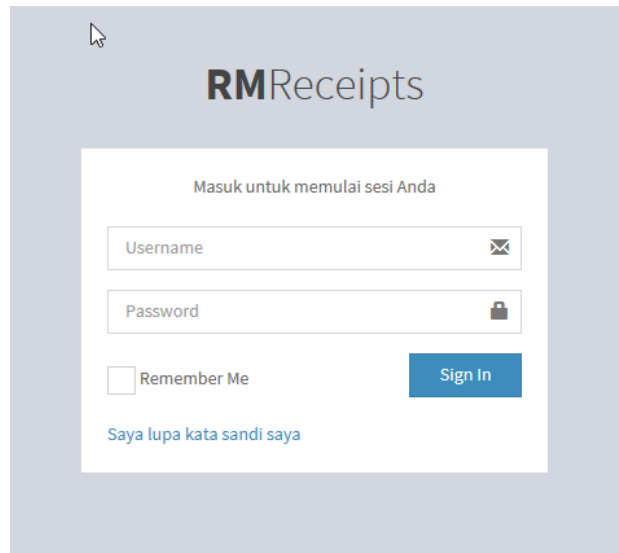
Wilkinson, Joseph W. 2012. *Sistem Akunting dan Informasi, Edisi keempat jilid satu*. Jakarta: Binarupa Aksara.

Zakiyudin, Ais. 2012. *Sistem Informasi Manajemen*. Jakarta: Mitra Wacana Media.

LAMPIRAN A

TAMPILAN PROGRAM

1. Halaman Login



The screenshot shows the login page for 'RMReceipts'. The page has a light gray background. At the top center, the text 'RMReceipts' is displayed in a large, dark font. Below this, a white rectangular box contains the login form. Inside the box, the text 'Masuk untuk memulai sesi Anda' is centered. There are two input fields: 'Username' with an eye icon on the right, and 'Password' with a lock icon on the right. Below the password field is a checkbox labeled 'Remember Me'. To the right of the checkbox is a blue button labeled 'Sign In'. At the bottom of the white box, there is a link that says 'Saya lupa kata sandi saya'.

2. Halaman Utama



3. Data Master

RMReceipts

Log Out

Dashboard

Master Data

- User
- Stasiun Balok
- Supplier

Pengelolaan

Laporan

Profile

Change Password

Users

Add User

Full Name

Username

Email

Password

Password

Password (Re-type)

Confirmation Password

Role

☒ Admin Warehouse

☐ Purchasing

☐ PPC & QC

Submit

Cancel

RMReceipts

Log Out

Dashboard

Master Data

- User
- Stasiun Balok
- Supplier

Pengelolaan

Laporan

Profile

Change Password

Users

List of User

Add User

Showing 1 to 1 of 1 entries

No	Role	Full Name	Username	Email	Created Date	Created By	Updated By	Action
1	Admin Warehouse	Administrator	admin	admin@mmwcojtgap.com	2018-08-01 00:00:00	Administrator	Administrator	Edit

Showing 1 to 1 of 1 entries

Previous1Next

RM/Receipts

Master Data > Bahan Baku

Dashboard

Master Data

- User
- Bahan Baku
- Supplier

Pengelolaan

Laporan

Logout

Profile

Change Password

Bahan Baku

Add Bahan Baku

Supplier

Pilih Supplier

Kode

Nama

Stok

Simpan

Cancel

RM/Receipts

Master Data > Bahan Baku

Dashboard

Master Data

- User
- Bahan Baku
- Supplier

Pengelolaan

Laporan

Logout

Profile

Change Password

Bahan Baku

List of Bahan Baku

Add Bahan Baku

Showing 1 to 1 of 1 entries

Search

No	Supplier	Kode	Nama	Stok	Created Date	Updated Date	Action
1	PT ABC	0001	Besi	78	Sun, 19 Aug 2018 12:27 PM	Sun, 19 Aug 2018 12:36 PM	EditDelete

Showing 1 to 1 of 1 entries

Previous1Next

RM/Receipts

Log Out

Dashboard

Master Data

User

System Role

Supplier

Pengelolaan

Laporan

Profile

Change Password

Supplier

Add Supplier

Nama Supplier

Alamat

Contact Person

No Telp

Keterangan

SubmitCancel

RM/Receipts

Log Out

Dashboard

Master Data

User

System Role

Supplier

Pengelolaan

Laporan

Profile

Change Password

Supplier

List of Supplier

Show 10 entries

Search

Add Supplier

No	Nama Supplier	Alamat	Contact Person	No Telp	Keterangan	Created Date	Updated Date	Action
1	PT ABC	Jakarta	Bp. Herman	062291406124		Sun, 19 Aug 2018 12:26 PM	Sun, 19 Aug 2018 12:26 PM	EditDelete

Showing 1 to 1 of 1 entries

Previous1Next

4. Data Pengelolaan

Log Out

Dashboard

Master Data

Pengelolaan

- Purchase Order
- Penerimaan
- Bahan Baku Masuk

Laporan

Profil

Change Password

Purchase Order

Add Purchase Order

Kode Bahan Baku

Pilih Kode Bahan Baku

Tanggal

Jumlah

0

Harga Satuan

0

Jumlah Harga

0

Submit

Cancel

Log Out

Dashboard

Master Data

Pengelolaan

- Purchase Order
- Penerimaan
- Bahan Baku Masuk

Laporan

Profil

Change Password

Purchase Order

List of Purchase Order

Show 10 entries

Search

No	Kode Bahan Baku	Tanggal	Jumlah	Harga Satuan	Jumlah Harga	Created At	Action
No data available in table							

Showing 0 to 0 of 0 entries

Previous

Next

RMReceipts

Log Out

Dashboard

Master Data

Pengelolaan

- Purchase Order
- Pemeriksaan
- Bahan Baku Masuk

Laporan

ACCOUNT SETTINGS

- Profile
- Change Password

Pemeriksaan

Add Pemeriksaan

Purchase Order

PMH Supplier

Tanggal

dd—yyyy

Stok

Submit

Cancel

RMReceipts

Log Out

Dashboard

Master Data

Pengelolaan

- Purchase Order
- Pemeriksaan
- Bahan Baku Masuk

Laporan

ACCOUNT SETTINGS

- Profile
- Change Password

Pemeriksaan

List of Pemeriksaan

10 entries

Search

No	ID Pemeriksaan	PO ID	Tanggal	Jumlah OK	Jumlah NG	Action
No data available in table						

Showing 0 to 0 of 0 entries

Previous

Next

RMReceipts

Log Out

Dashboard

Master Data

Pengelolaan

- Purchase Order
- Pembelian
- Bahan Baku Masuk

Laporan

Account Settings

- Profile
- Change Password

Bahan Baku Masuk

Add Bahan Baku Masuk

Supplier

Pilih Supplier

Kode Bahan Baku

Pilih Kode Bahan Baku

Tanggal

dd/mm/yyyy

Jumlah

Submit

Cancel

RMReceipts

Log Out

Dashboard

Master Data

Pengelolaan

- Purchase Order
- Pembelian
- Bahan Baku Masuk

Laporan

Account Settings

- Profile
- Change Password

Bahan Baku Masuk

List of Bahan Baku Masuk

Add Bahan Baku Masuk

Showing 10 entries

Search

#	ID Bahan Baku Masuk	ID Supplier	Kode Bahan Baku	Tanggal	Jumlah	Created Date	Action
No data available in table							

Showing 0 to 0 of 0 entries

PreviousNext

5. Laporan

RMReceipts

Menu Utama

Dashboard

Master Data

Pengelolaan

Laporan

Laporan

Laporan Hasil

Account Settings

Profile

Change Password

Laporan

Laporan Hasil Pemeriksaan Bahan Baku

Tanggal

dd—yyyy

Cari

Reset

Hasil

Cetak

Tanggal	ID Pemeriksaan	Kode Bahan Baku	Nama Bahan Baku	Jumlah OK	Jumlah NG
Data tidak ditemukan					

RMReceipts

Menu Utama

Dashboard

Master Data

Pengelolaan

Laporan

Laporan

Laporan Hasil

Account Settings

Profile

Change Password

Laporan

Laporan Bahan Baku Masuk

Tanggal

dd—yyyy

Cari

Reset

Hasil

Cetak

Tanggal	Kode Bahan Baku	Nama Bahan Baku	ID Supplier	Nama Supplier	Jumlah
Data tidak ditemukan					

KODE PROGRAM

6. Master

```
<!DOCTYPE html>

<html>

<head>

  <meta charset="utf-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <title>

    @yield('title_prefix', config('adminlte.title_prefix', ''))

    @yield('title', config('adminlte.title', 'AdminLTE 2'))

    @yield('title_postfix', config('adminlte.title_postfix', ''))

  </title>

  <!-- Tell the browser to be responsive to screen width -->

  <meta content="width=device-width, initial-scale=1, maximum-scale=1,
user-scalable=no" name="viewport">

  <!-- Bootstrap 3.3.7 -->

  <link                                rel="stylesheet"                                href="{{
asset('vendor/adminlte/vendor/bootstrap/dist/css/bootstrap.min.css')}}">

  <!-- Font Awesome -->

  <link  rel="stylesheet"  href="{{  asset('vendor/adminlte/vendor/font-
awesome/css/font-awesome.min.css')}}">

  <!-- Icons -->

  <link                                rel="stylesheet"                                href="{{
asset('vendor/adminlte/vendor/Icons/css/ions.min.css')}}">

  @if(config('adminlte.plugins.select2'))

    <!-- Select2 -->
```

```
        <link                                rel="stylesheet"
href="//cdnjs.cloudflare.com/ajax/libs/select2/4.0.3/css/select2.css">
    @endif
```

```
    <!-- Theme style -->
    <link                                rel="stylesheet"                                href="{{
asset('vendor/adminlte/dist/css/AdminLTE.min.css') }}">
```

```
    @if(config('adminlte.plugins.datatables'))
        <!-- DataTables with bootstrap 3 style -->
        <link                rel="stylesheet"                href="//cdn.datatables.net/v/bs/dt-
1.10.18/datatables.min.css">
    @endif
```

```
    @yield('adminlte_css')
```

```
    <!--[if lt IE 9]>
    <script
src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
    <script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->
```

```
    <!-- Google Font -->
    <link                                rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,60
0,700,300italic,400italic,600italic">
</head>
<body class="hold-transition @yield('body_class')">
```

```
@yield('body')
```

```
<script src="{{ asset('vendor/adminlte/vendor/jquery/dist/jquery.min.js')
}}"></script>
```

```
<script src="{{ asset('vendor/adminlte/vendor/jquery/dist/jquery.slimscroll.min.js')
}}"></script>
```

```
<script src="{{ asset('vendor/adminlte/vendor/bootstrap/dist/js/bootstrap.min.js')
}}"></script>
```

```
@if(config('adminlte.plugins.select2'))
```

```
<!-- Select2 -->
```

```
<script
src="//cdnjs.cloudflare.com/ajax/libs/select2/4.0.3/js/select2.min.js"></script>
>
```

```
@endif
```

```
@if(config('adminlte.plugins.datatables'))
```

```
<!-- DataTables with bootstrap 3 renderer -->
```

```
<script src="//cdn.datatables.net/v/bs/dt-
1.10.18/datatables.min.js"></script>
```

```
@endif
```

```
@if(config('adminlte.plugins.chartjs'))
```

```
<!-- ChartJS -->
```

```
<script
src="//cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.0/Chart.bundle.min.js"></sc
ript>
```

```
@endif
```

```
@yield('adminlte_js')
```

```
</body>
```

```
</html>
```

7. Login

```
@extends('adminlte::master')
```

```
@section('adminlte_css')
```

```
    <link                                rel="stylesheet"                                href="{{
asset('vendor/adminlte/plugins/iCheck/square/blue.css') }}">
```

```
    <link rel="stylesheet" href="{{ asset('vendor/adminlte/css/auth.css') }}">
```

```
    @yield('css')
```

```
@stop
```

```
@section('body_class', 'login-page')
```

```
@section('body')
```

```
    <div class="login-box">
```

```
        <div class="login-logo">
```

```
            <a href="{{ url(config('adminlte.dashboard_url', 'home')) }}">{!!
config('adminlte.logo', '<b>Admin</b>LTE') !!}</a>
```

```
        </div>
```

```
        <!-- /.login-logo -->
```

```
        <div class="login-box-body">
```

```
            <p                                                                class="login-box-msg">{{
trans('adminlte::adminlte.login_message') }}</p>
```

```
            <form action="{{ url(config('adminlte.login_url', 'login')) }}"
method="post">
```

```
                {!! csrf_field() !!}
```

```

        <div class="form-group has-feedback {{ $errors->has('username') ?
'has-error' : " }}">
            <input type="text" name="username" class="form-control"
value="{{ old('username') }}"
                placeholder="{{ trans('adminlte::adminlte.username') }}">
            <span class="glyphicon glyphicon-envelope form-control-
feedback"></span>
            @if ($errors->has('username'))
                <span class="help-block">
                    <strong>{{ $errors->first('username') }}</strong>
                </span>
            @endif
        </div>

```

```

        <div class="form-group has-feedback {{ $errors->has('password') ?
'has-error' : " }}">
            <input type="password" name="password" class="form-control"
                placeholder="{{ trans('adminlte::adminlte.password') }}">
            <span class="glyphicon glyphicon-lock form-control-
feedback"></span>
            @if ($errors->has('password'))
                <span class="help-block">
                    <strong>{{ $errors->first('password') }}</strong>
                </span>
            @endif
        </div>
        <div class="row">
            <div class="col-xs-8">
                <div class="checkbox icheck">
                    <label>

```

```

        <input type="checkbox" name="remember"> {{
trans('adminlte::adminlte.remember_me') }}
    </label>
</div>
</div>
<!-- /.col -->
<div class="col-xs-4">
    <button type="submit"
        class="btn btn-primary btn-block btn-flat">{{
trans('adminlte::adminlte.sign_in') }}</button>
</div>
<!-- /.col -->
</div>
</form>
<div class="auth-links">
    <a href="{{ url(config('adminlte.password_reset_url',
'password/reset')) }}"
        class="text-center"
    >{{ trans('adminlte::adminlte.i_forgot_my_password') }}</a>
    <br>
    @if (config('adminlte.register_url', 'register'))
        <a href="{{ url(config('adminlte.register_url', 'register')) }}"
            class="text-center"
        >{{ trans('adminlte::adminlte.register_a_new_membership')
    }}</a>
    @endif
</div>
</div>
<!-- /.login-box-body -->
</div><!-- /.login-box -->
@stop

```

```

@section('adminlte_js')
    <script src="{{ asset('vendor/adminlte/plugins/iCheck/ichack.min.js')
    }}"></script>
    <script>
        $(function () {
            $('input').iCheck({
                checkboxClass: 'icheckbox_square-blue',
                radioClass: 'iradio_square-blue',
                increaseArea: '20%' // optional
            });
        });
    </script>
    @yield('js')
@stop

```

8. Page

```

@extends('adminlte::master')

@section('adminlte_css')
    <link rel="stylesheet"
        href="{{ asset('vendor/adminlte/dist/css/skins/skin-'.
        config('adminlte.skin', 'blue') . '.min.css')) }}">
    @if(config('adminlte.plugins.bootstrap-datepicker'))
        <link rel="stylesheet" href="{{
        asset('vendor/adminlte/vendor/bootstrap-datepicker/css/bootstrap-
        datepicker3.min.css') }}">
    @endif
    @stack('css')
    @yield('css')
@stop

```

```

@section('body_class', 'skin-' . config('adminlte.skin', 'blue') . ' sidebar-
mini ' . (config('adminlte.layout') ? [
    'boxed' => 'layout-boxed',
    'fixed' => 'fixed',
    'top-nav' => 'layout-top-nav'
][config('adminlte.layout')] : '')) . (config('adminlte.collapse_sidebar') ? '
sidebar-collapse ' : ''))

```

```

@section('body')
    <div class="wrapper">

        <!-- Main Header -->
        <header class="main-header">
            @if(config('adminlte.layout') == 'top-nav')
                <nav class="navbar navbar-static-top">
                    <div class="container">
                        <div class="navbar-header">
                            <a href="{{ url(config('adminlte.dashboard_url', 'home'))
}}" class="navbar-brand">
                                {!! config('adminlte.logo', '<b>Admin</b>LTE') !!}
                            </a>
                            <button type="button" class="navbar-toggle collapsed"
data-toggle="collapse" data-target="#navbar-collapse">
                                <i class="fa fa-bars"></i>
                            </button>
                        </div>

                        <!-- Collect the nav links, forms, and other content for
toggling -->
                        @auth

```

```

        <div class="collapse navbar-collapse pull-left" id="navbar-
collapse">
            <ul class="nav navbar-nav">
                @each('adminlte::partials.menu-item-top-nav',
$adminlte->menu(), 'item')
            </ul>
        </div>
        @endauth
        <!-- /.navbar-collapse -->
    @else
        <!-- Logo -->
        <a href="{{ url(config('adminlte.dashboard_url', 'home')) }}"
class="logo">
            <!-- mini logo for sidebar mini 50x50 pixels -->
            <span class="logo-mini">{!! config('adminlte.logo_mini',
'<b>A</b>LT') !!}</span>
            <!-- logo for regular state and mobile devices -->
            <span class="logo-lg">
                 {!! config('adminlte.logo', '<b>Admin</b>LTE') !!}
            </span>
        </a>

        <!-- Header Navbar -->
        <nav class="navbar navbar-static-top" role="navigation">
            <!-- Sidebar toggle button-->
            <a href="#" class="sidebar-toggle" data-toggle="push-menu"
role="button">
                <span class="sr-only">{{
trans('adminlte::adminlte.toggle_navigation') }}</span>
            </a>

```

```

@endif
<!-- Navbar Right Menu -->
<div class="navbar-custom-menu">

    <ul class="nav navbar-nav">
        <li>
            @if(config('adminlte.logout_method') == 'GET' ||
!config('adminlte.logout_method')                &&
version_compare(Illuminate\Foundation\Application::VERSION, '5.3.0',
'<'))
                <a href="{ url(config('adminlte.logout_url',
'auth/logout')) }}">
                    <i class="fa fa-fw fa-power-off"></i> {{
trans('adminlte::adminlte.log_out') }}
                </a>
            @else
                <a href="#"
                    onclick="event.preventDefault();
document.getElementById('logout-form').submit();"
                >
                    <i class="fa fa-fw fa-power-off"></i> {{
trans('adminlte::adminlte.log_out') }}
                </a>
                <form id="logout-form" action="{ {
url(config('adminlte.logout_url', 'auth/logout')) }}" method="POST"
style="display: none;">
                    @if(config('adminlte.logout_method'))
                        {{
method_field(config('adminlte.logout_method')) }}
                    @endif

```

```

        {{ csrf_field() }}
    </form>

    @endif
</li>
</ul>
</div>

@if(config('adminlte.layout') == 'top-nav')
</div>

@endif

</nav>
</header>

@if(config('adminlte.layout') != 'top-nav')
<!-- Left side column. contains the logo and sidebar -->
<aside class="main-sidebar">

    <!-- sidebar: style can be found in sidebar.less -->
    <section class="sidebar">
        <!-- Sidebar Menu -->
        @auth
        <ul class="sidebar-menu" data-widget="tree">
            @each('adminlte::partials.menu-item', $adminlte->menu(),
'item')
        </ul>
        @endauth
        <!-- /.sidebar-menu -->
    </section>
    <!-- /.sidebar -->
</aside>

@endif

```

```

<!-- Content Wrapper. Contains page content -->
<div class="content-wrapper">
    @if(config('adminlte.layout') == 'top-nav')
    <div class="container">
        @endif

    <!-- Content Header (Page header) -->
    <section class="content-header">
        <h1>
            @yield('content_header')
        </h1>
    </section>

    <!-- Main content -->
    <section class="content">
        @include('layouts.components.alert')
        @yield('content')
    </section>
    <!-- /.content -->
    @if(config('adminlte.layout') == 'top-nav')
    </div>
    <!-- /.container -->
    @endif
</div>
<!-- /.content-wrapper -->

</div>
<!-- ./wrapper -->
@stop

@section('adminlte_js')

```

```

        <script      src="{{      asset('vendor/adminlte/dist/js/adminlte.min.js')
    }}"></script>

    <script>
        $('div.alert').not('.alert-important').delay(3000).fadeOut(350);
    </script>

    @if(config('adminlte.plugins.bootstrap-datepicker'))
        <script      src="{{      asset('vendor/adminlte/vendor/bootstrap-
datepicker/js/bootstrap-datepicker.min.js') }}"></script>

        <script      src="{{      asset('vendor/adminlte/vendor/bootstrap-
datepicker/locales/bootstrap-datepicker.id.min.js') }}"
            charset="UTF-8"></script>

    @endif

    @stack('js')
    @yield('js')
@stop

```

9. Home

```

@extends('adminlte::page')

@section('content_header')
    <h1>Dashboard</h1>
@stop

@section('content')
    @box
        @slot('body')
            <p>Selamat datang kembali, <b>{{ auth()->user()->full_name
        }}</b></p>
        @endslot
    @endbox
@stop

```

10. Box

```
<div class="box box-default color-palette-box">
  @isset($title)
    <div class="box-header with-border">
      <h3 class="box-title"><i class="fa fa-tag"></i> {{ $title }}</h3>
      @isset($tools)
        <div class="box-tools pull-right">
          {{ $tools }}
        </div>
      @endisset
    </div>
  @endisset
  <div class="box-body">
    {!! $body !!}
  </div>
  <!-- /.box-body -->
</div>
```