

**RANCANG BANGUN SISTEM INFORMASI PEMBELIAN
BAHAN BAKU MENGGUNAKAN PHP 5.6.12 DAN MYSQL
5.6.26 PADA PT BANEEX INDONESIA**

TUGAS AKHIR

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

DISUSUN OLEH:
YEYEN MARYANI
1314035



POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA
2018

LEMBAR PERSETUJUAN DOSEN PEMBIMBING
POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

TANDA PERSETUJUAN DOSEN PEMBIMBING

JUDUL TUGAS AKHIR:

**RANCANG BANGUN SISTEM INFORMASI PEMBELIAN BAHAN BAKU MENGGUNAKAN
PHP 5.6.12 DAN MYSQL 5.6.26 PADA PT BANEEX INDONESIA**

DISUSUN OLEH :

NAMA : Yeyen Maryani
NIM : 1314035
PROGRAM STUDI : Sistem Informasi Industri Otomotif

Telah diperiksa dan disetujui untuk memenuhi sebagian syarat penyelesaian Program Sarjana Terapan Program Studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

Jakarta, 13 Agustus 2018

Dosen Pembimbing

Ahmad Juniar, S.Kom, MT

NIP : 197906052006041002

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Yeyen Maryani
NIM : 1314035
Judul Tugas Akhir : Rancang Bangun Sistem Informasi Pembelian Bahan Baku Menggunakan PHP 5.6.12 Dan MySQL 5.6.26 Pada PT Baneex Indonesia
Pembimbing : Ahmad Juniar, S.Kom, MT

No	Tanggal	Keterangan	Paraf
1	11 Mei 2018	Bab 1 . Pendahuluan. Perbaiki hal 3	
2	16 Mei 2018	Revisi Bab 1	
3	17 Mei 2018	Bab II Landasan Teori. Perbaiki hal 6, 14, 16, 17, 18, 19, 22.	
4	18 Mei 2018	Lanjutan Bab II Landasan Teori . Hal 29, 30, 31, 32.	
5	30 Mei 2018	Revisi Bab II	
6	31 Mei 2018	Bab III Metodologi Penelitian	
7	8 Juni 2018	Bab 1V Pengumpulan & Pengolahan Data. Perbaiki Hal 46, 54, 68. 82.	
8	30 Juli 2018	Bab V Analisis Pembahasan. Perbaiki hal 80,82	
9	1 Agustus 2018	Bab V Analisis & Pembahasan. Perbaiki Hal 96, Activity Diagram.	
10	4 Agustus 2018	Bab V Analisis & Pembahasan Sequence Diagram, Class Diagram, ERD, CMD, WND, Kamus Data, Desain Interface.	
11	7 Agustus 2018	Kesimpulan, Saran & Daftar Pustaka	

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif

Dosen Pembimbing

Drs. Jacob Saragih, M.M.
NIP : 195404281986031002

Ahmad Juniar, S.Kom, MT
NIP : 197906052006041002

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Yeyen Maryani

NIM :1314035

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 PEMBELIAN BAHAN BAKU MENGGUNAKAN PHP 5.6.12 DAN MYSQL 5.6.26 PADA PT BANEEX INDONESIA”

- **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, 13 Agustus 2018

Yeyen Maryani

KATA PENGANTAR

Puji syukur dipanjatkan kepada Allah SWT, atas rahmat dan karunia-Nya sehingga penulis dapat menyusun dan menyelesaikan Tugas Akhir dengan judul “**Rancang Bangun Sistem Informasi Pembelian Bahan Baku Menggunakan PHP 5.6.12 dan MYSQL 5.6.26 Pada PT Baneex Indonesia**”.

Penyusunan Tugas akhir ini bertujuan untuk memenuhi sebagian syarat penyelesaian Program Diploma Empat (D-IV) pada program studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

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

1. Allah SWT atas rahmat dan karunia-Nya.
2. Keluarga tercinta yang selalu mendoakan dan selalu memberikan dukungan baik moril maupun materil selama menyusun Laporan Tugas Akhir.
3. Bapak Dr. Mustofa, S.T., M.T., selaku Direktur Politeknik STMI Jakarta.
4. Bapak Jacob Saragih, M.M., selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
5. Bapak Ahmad Juniar, S.Kom, MT selaku dosen pembimbing yang telah memberikan arahan dan masukan dalam penyusunan Laporan Tugas Akhir.
6. Ibu Risa selaku pembimbing yang telah membantu mengarahkan dan membimbing selama Kerja Lapangan.
7. Ibu Sekar, Bapak Ahmad, Ibu Etha, Bapak Iwan dan seluruh pegawai di PT Baneex Indonesia yang telah membantu memberi arahan serta informasi untuk pengumpulan data.
8. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.

9. Teman-teman mahasiswa/i Politeknik STMI Jakarta Program Studi Sistem Informasi terutama kelas SA01 2014 yang selalu memberi motivasi dan semangat dalam menyusun Laporan Tugas Akhir ini khususnya kepada pak fadhla fanini dan bang riza agustin para calon orang sukses.
10. Rusdiansyah, Roni sudrajat, Muhammad Hardiansyah, Dimas Pangestu, Rizkiana Kunastiti.
11. Saudari Nur afifa *also known as* pipih atau eceu , calon sarjana di Politeknik STMI Jakarta juga, terimakasih karena tidak pernah memberikan semangat kepada penulis melainkan telah menjadi teman dari zaman suka naik bis sekolah muter-muter sampai pusing hingga sekarang.
12. Indri Fitriana Alfia Nisa, Indria Aprilyana, dan Yulia sahadah yang selalu setia menjadi teman *karaoke* dan telah menjadi teman seperjuangan hingga saat ini.

Semoga Allah SWT membalas kebaikan dan ketulusan semua pihak yang telah membantu menyelesaikan laporan ini dengan melimpahkan rahmat dan karunia-Nya. Semoga penulisan ini bermanfaat untuk pembaca serta mohon kritik dan saran untuk membangun kesempurnaan penulis di masa yang akan datang. Terima kasih.

Jakarta, 13 Agustus 2018

Penulis,
Yeyen Maryani

DAFTAR ISI

Halaman

HALAMAN JUDUL	
LEMBAR PENGESAHAN	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN TUGAS AKHIR	
LEMBAR PERNYATAAN KEASLIAN	
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL	xvi
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Pokok Permasalahan	2
1.3. Tujuan Penelitian	3
1.4. Batasan Masalah	3
1.5. Manfaat Tugas Akhir	3
1.5. Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1. Konsep Dasar Sistem	6
2.1.1. Karakteristik Sistem	6
2.1.2. Klasifikasi Sistem	8
2.2. Konsep Dasar Informasi	10
2.2.1. Kualitas Informasi	10
2.3. Konsep Dasar Sistem Informasi	11
2.4. Konsep Dasar Pembelian	13
2.4.1. Tanggung Jawab Pembelian	13

2.4.2. Tahap-Tahap Proses Pembelian	14
2.5. Konsep Dasar Bahan Baku.....	18
2.6. <i>System Development Life Cycle (SDLC)</i>	19
2.7. <i>Model Prototype</i>	23
2.7.1. <i>Prototype Evolusioner</i>	23
2.7.2. <i>Prototype Requirement</i>	24
2.8. <i>System Requirements</i>	25
2.9. <i>Flowchart</i>	26
2.9.1. Simbol-simbol <i>Flowchart</i>	26
2.9.2. <i>Flow Direction Symbols</i>	26
2.9.3. <i>Processing Symbols</i>	27
2.9.4. <i>Input-Output Symbols</i>	28
2.10. Diagram Alir Dokumen (<i>Flowmap</i>)	29
2.11. <i>Unified Modelling Language (UML)</i>	27
2.11.1. <i>Use Case Diagram</i>	34
2.11.2. <i>Activity Diagram</i>	34
2.11.3. <i>Sequence Diagram</i>	38
2.11.4. <i>Class Diagram</i>	40
2.11.5. <i>Deployment Diagram</i>	42
2.12. <i>Entity Relationship Diagram (ERD)</i>	43
2.12.1. <i>Conceptual Data Model (CDM)</i>	44
2.13. <i>Database</i>	48
2.14. Kamus Data	48
2.15. <i>Windows Navigation Diagram (WND)</i>	49
2.16. PHP (<i>Hypertext Preprocessor</i>)	50
2.17. MySQL	51
2.17.1. Keunggulan MySQL	51
2.17.2. Tipe Data MySQL	53
BAB III METODOLOGI PENELITIAN	55
3.1. Metode Penelitian	55

3.2. Jenis dan Sumber Data	55
3.3. Metode Pengumpulan Data	56
3.4. Metode Pengembangan Sistem	57
3.5. Kerangka Penelitian	58
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA	62
4.1. Sejarah Perusahaan	62
4.2. Profil Perusahaan	63
4.3. Logo Perusahaan	63
4.4. Visi, Misi dan Kebijakan Mutu Perusahaan	64
4.4.1. Visi	64
4.4.2. Misi	64
4.4.3. Kebijakan Mutu	64
4.5. Struktur Organisasi Perusahaan	65
4.5.1. Tugas dan Wewenang	66
4.6. Pemasok	69
4.7. Bahan Baku	70
4.8. Dokumen Pembelian Bahan Baku	73
4.9. Prosedur Pembelian Bahan Baku pada Divisi <i>Purchasing</i>	77
4.10. Penggambaran Sistem Informasi Pembelian Bahan Baku dengan (UML) <i>Unified Modelling Language</i>	79
4.10.1. <i>Use Case Diagram</i>	79
BAB V ANALISIS DAN PEMBAHASAN	82
5.1. Analisis Kebutuhan Sistem	82
5.2. Prosedur Sistem Informasi Pembelian Bahan Baku.....	84
5.3. Analisis Sistem Usulan	87
5.3.1. <i>Use Case Diagram</i>	87
5.3.2. <i>Activity Diagram</i>	95
5.3.3. <i>Sequence Diagram</i>	106
5.3.4. <i>Class Diagram</i>	116
5.3.5. <i>Deployment Diagram</i>	117
5.4. Pemodelan Data Sistem Usulan	117

5.4.1. <i>Entity Relationship Diagram (ERD)</i>	118
5.4.2. <i>CDM (Conceptual Data Model)</i>	119
5.4.2. Kamus Data	119
5.5. Perancangan Sistem	124
5.5.1. <i>WND (Windows Navigation Diagram)</i>	124
5.5.2. Perancangan Antarmuka Pengguna	126
5.6. Implementasi Sistem <i>Hardware</i> dan <i>Software</i>	139
BAB VI KESIMPULAN DAN SARAN	140
6.1. Kesimpulan	140
6.2. Saran	140
DAFTAR PUSTAKA	141
LAMPIRAN	

DAFTAR GAMBAR

	Halaman
Gambar II.1	Komponen Sistem Informasi 11
Gambar II.2	Penggambaran Prototipe Evolusioner 24
Gambar II.3	Penggambaran Prototipe Requirement 25
Gambar II.4	Contoh WND 50
Gambar III.1	Kerangka Penelitian 60
Gambar IV.1	Logo PT Baneex Indonesia 63
Gambar IV.2	Struktur Organisasi PT Baneex Indonesia 65
Gambar IV.3	SWB - <i>Wire Spring</i> 2.00 mm 70
Gambar IV.4	SWB - <i>Wire Spring</i> 2.00 mm 71
Gambar IV.5	SWC - <i>Wire Spring</i> 3.80 mm 71
Gambar IV.6	SWMB - <i>Wire</i> 5.00 mm 72
Gambar IV.7	SWMB - <i>Wire</i> 6.00 mm 72
Gambar IV.8	SWMB - <i>Wire</i> 7.00 mm 73
Gambar IV.9	Surat Permintaan Material 74
Gambar IV.10	<i>Purchase Order</i> 75
Gambar IV.11	Aliran Dokumen Sistem yang Berjalan 78
Gambar IV.12	<i>Use Case Diagram</i> Sistem Informasi Pembelian Bahan Baku 81
Gambar V.1	<i>Flowmap</i> Sistem Informasi Pembelian Bahan Baku Usulan .. 86
Gambar V.2	<i>Use Case</i> Usulan Sistem Informasi Pembelian Bahan Baku .. 87
Gambar V.3	<i>Activity Diagram Login</i> 96
Gambar V.4	<i>Activity Diagram</i> Mengelola Data Master 97
Gambar V.5	<i>Activity Diagram</i> Membuat Permintaan Pembelian 98
Gambar V.6	<i>Activity Diagram</i> Melakukan Persetujuan Permintaan 99
Gambar V.7	<i>Activity Diagram</i> Melakukan Konfirmasi 100
Gambar V.8	<i>Activity Diagram</i> Membuat <i>Purchase Order</i> 101
Gambar V.9	<i>Activity Diagram</i> Mencetak <i>Purchase Order</i> 102
Gambar V.10	<i>Activity Diagram</i> Mengirim <i>Purchase Order</i> 103
Gambar V.11	<i>Activity Diagram</i> Menginput Data Surat Jalan 104

Gambar V.12	<i>Activity Diagram</i> Mengecek Bahan Baku	105
Gambar V.13	<i>Sequence Diagram</i> Login	106
Gambar V.14	<i>Sequence Diagram</i> Mengelola Data Master.....	107
Gambar V.15	<i>Sequence Diagram</i> Membuat Permintaan Pembelian	108
Gambar V.16	<i>Sequence Diagram</i> Melakukan Persetujuan Permintaan	109
Gambar V.17	<i>Sequence Diagram</i> Melakukan Konfirmasi	110
Gambar V.18	<i>Sequence Diagram</i> Membuat <i>Purchase Order</i>	111
Gambar V.19	<i>Sequence Diagram</i> Mencetak <i>Purchase Order</i>	112
Gambar V.20	<i>Sequence Diagram</i> Mengirim <i>Purchase Order</i>	113
Gambar V.21	<i>Sequence Diagram</i> Menginput Data Surat Jalan.....	114
Gambar V.22	<i>Sequence Diagram</i> Mengecek Bahan Baku	115
Gambar V.23	<i>Class Diagram</i>	116
Gambar V.24	<i>Deployment Diagram</i>	117
Gambar V.25	ERD Sistem Informasi Pembelian Bahan Baku	118
Gambar V.26	<i>Conceptual Data Model</i>	119
Gambar V.27	<i>Windows Navigation Diagram</i>	125
Gambar V.28	<i>Form Login</i>	126
Gambar V.29	Menu Utama	127
Gambar V.30	Tampilan Halaman Data Master <i>User</i>	127
Gambar V.31	Tampilan Tombol Tambah Data Halaman Data Master <i>User</i>	128
Gambar V.32	Tampilan Tombol Ubah Data Halaman Data Master <i>User</i>	128
Gambar V.33	Tampilan Halaman Data Master <i>Supplier</i>	129
Gambar V.34	Tampilan Tombol Tambah Data Halaman Data Master <i>Supplier</i>	129
Gambar V.35	Tampilan Tombol Ubah Data Halaman Data Master <i>Supplier</i>	130
Gambar V.36	Tampilan Halaman Data Master Bahan Baku.....	130
Gambar V.37	Tampilan Tombol Tambah Data Halaman Data Master Bahan Baku.....	131
Gambar V.38	Tampilan Ubah Tambah Data Halaman Data Master Bahan Baku.....	131
Gambar V.39	Tampilan Halaman Permintaan Pembelian	132

Gambar V.40 Tampilan Tombol Selesai Halaman Permintaan Pembelian	132
Gambar V.41 Tampilan Halaman Permintaan Disetujui	133
Gambar V.42 Tampilan Halaman Permintaan Ditolak.....	133
Gambar V.43 Tampilan Halaman Validasi Permintaan	134
Gambar V.44 Tampilan Halaman <i>Purchase Order</i>	134
Gambar V.45 Tampilan Tombol Selesai Halaman <i>Purchase Order</i>	135
Gambar V.46 Tampilan Halaman Data <i>Purchase Order</i>	135
Gambar V.47 Tampilan Kirim <i>Purchase Order</i>	136
Gambar V.48 Tampilan Halaman Surat Jalan	136
Gambar V.49 Tampilan Tombol Selesai Halaman Surat Jalan	137
Gambar V.50 Tampilan Halaman Cek Bahan Baku	137
Gambar V.51 Tampilan Halaman Data Cek Bahan Baku Sesuai.....	138
Gambar V.52 Tampilan Halaman Data Cek Bahan Baku Tidak Sesuai.....	138

DAFTAR TABEL

	Halaman
Tabel II.1 <i>Flow Direction Symbols</i>	27
Tabel II.2 <i>Processing Symbols</i>	27
Tabel II.3 <i>Input-Output Symbols</i>	29
Tabel II.4 Simbol <i>Flowmap</i>	30
Tabel II.5 Jenis Diagram Pada UML dan Fungsi-fungsinya.....	32
Tabel II.6 Simbol-Simbol <i>Use Case Diagram</i>	34
Tabel II.7 Simbol-Simbol <i>Activity Diagram</i>	36
Tabel II.8 Simbol-Simbol <i>Sequence Diagram</i>	38
Tabel II.9 Simbol-Simbol <i>Class Diagram</i>	40
Tabel II.10 Simbol-Simbol <i>Deployment Diagram</i>	42
Tabel II.11 Simbol-simbol <i>Entity Relationship Diagram</i>	43
Tabel II.12 Simbol-Simbol <i>Conceptual Data Model (CDM)</i>	45
Tabel II.13 Simbol-Simbol Aturan <i>Conceptual Data Model (CDM)</i>	45
Tabel II.14 Jenis Data pada MySQL	53
Tabel IV.1 <i>Supplier</i> PT. Baneex Indonesia	69
Tabel IV.2 Definisi Aktor Sistem Informasi Pembelian Bahan Baku	80
Tabel IV.3 Definisi <i>Use Case</i> Pembelian Bahan Baku.....	80
Tabel V.1 Analisis Kebutuhan Sistem	82
Tabel V.2 Definisi Aktor Sistem Informasi Pembelian Bahan Baku.....	88
Tabel V.3 Definisi <i>Use Case</i> Pembelian Bahan Baku	88
Tabel V.4 <i>Use Case Description Login</i>	90
Tabel V.5 <i>Use Case</i> Mengelola Data Master.....	90
Tabel V.6 <i>Use Case</i> Membuat Permintaan Pembelian Bahan Baku	91
Tabel V.7 <i>Use Case</i> Melakukan Persetujuan Permintaan Pembelian	91
Tabel V.8 <i>Use Case</i> Melakukan Konfirmasi kepada <i>supplier</i>	92
Tabel V.9 <i>Use Case</i> Membuat <i>Purchase Order</i>	93
Tabel V.10 <i>Use Case</i> Mencetak <i>Purchase Order</i>	93
Tabel V.11 <i>Use Case</i> Mengirim <i>Purchase Order</i>	94

Tabel V.12	<i>Use Case</i> Menginput Data Surat Jalan.....	94
Tabel V.13	<i>Use Case</i> Mengecek Bahan Baku	95
Tabel V.14	Spesifikasi Tabel User.....	120
Tabel V.15	Spesifikasi Bahan Baku	120
Tabel V.16	Spesifikasi Tabel <i>Supplier</i>	121
Tabel V.17	Spesifikasi Tabel Permintaan Detail	121
Tabel V.18	Spesifikasi Tabel Permintaan.....	122
Tabel V.19	Spesifikasi Tabel Surat Jalan	122
Tabel V.20	Spesifikasi Tabel Surat Jalan Detail.....	122
Tabel V.21	Spesifikasi Tabel <i>Purchase Order</i>	123
Tabel V.22	Spesifikasi Tabel <i>Purchase Order</i> Detail	123

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di era globalisasi ini, perkembangan teknologi informasi saat ini sangat pesat. Teknologi informasi memiliki peran yang besar dalam salah satu bidang yang tak akan pernah habis untuk dibahas yaitu, industri. Dalam bidang industri, peranan teknologi sangat terkait dengan kegiatan yang ada di dalam bidang industri tersebut. Teknologi informasi yang semakin maju membuat manusia lebih mudah dalam menyelesaikan permasalahan berdasarkan pengetahuan dan pengalaman. Seiring meningkatnya kemajuan teknologi, kemampuan manusia pun juga ikut meningkat. Dengan adanya teknologi informasi ini memudahkan manusia untuk bertukar informasi. Untuk menjalankan teknologi informasi tersebut maka dibutuhkan sebuah sistem agar bisa berjalan dengan baik.

Sistem informasi memiliki peran penting dalam suatu perusahaan, dengan adanya sistem informasi perusahaan dapat menjalankan proses bisnis dengan maksimal. Sistem informasi dapat meningkatkan kinerja pada sebuah perusahaan seperti sistem yang terintegrasi, dengan adanya sistem informasi yang terintegrasi antar divisi menjadikan proses informasi menjadi lebih efisien. Salah satunya seperti kegiatan pembelian bahan baku yang apabila menggunakan sistem informasi yang terintegrasi akan menjadikan kegiatan tersebut menjadi lebih mudah.

Kegiatan pembelian adalah salah satu aktifitas perusahaan yang sangat penting, dengan adanya pembelian perusahaan dapat beroperasi dengan semestinya. Salah satu kegiatan pembelian yang memiliki kedudukan teratas adalah pembelian bahan baku. Bahan baku adalah modal utama dalam proses produksi. Bahan baku dibutuhkan untuk membuat produk-produk yang akan dijual kepada *customer*. Tanpa adanya bahan baku proses produksi tidak akan berjalan dengan semestinya.

Perusahaan industri di Indonesia tumbuh dengan pesat, salah satunya adalah PT Baneex Indonesia. PT Baneex Indonesia merupakan perusahaan manufaktur industri metal *spring*, *bending* dan *stamping* untuk komponen otomotif. Salah satu produk yang dihasilkan PT Baneex Indonesia yaitu *wire* yang merupakan rangkaian komponen *seat cushion* atau jok untuk mobil. Pada PT Baneex Indonesia memiliki beberapa divisi, salah satunya Divisi *Purchasing*. Divisi *Purchasing* memiliki tugas yang sangat penting, salah satunya bertanggung jawab terhadap semua pembelian, terutama pembelian bahan baku pada perusahaan untuk memenuhi kebutuhan bahan baku dalam proses produksi.

Untuk hal diatas, penulis melakukan penelitian khususnya pada divisi *purchasing* di PT Baneex Indonesia. Dalam menjalankan kegiatan pembelian bahan baku, di PT Baneex Indonesia masih menggunakan *Microsoft Excel* untuk mengolah data pembelian dikarenakan pada PT Baneex Indonesia belum ada sistem informasi untuk mengolah data pembelian bahan baku. Sistem yang ada pada perusahaan ini belum terintegrasi sehingga dalam penyajian informasi masih kurang maksimal. Selain itu, pegawai menyimpan dokumen berupa kertas tidak tertata dengan rapi sehingga beresiko terjadi kerusakan data, kehilangan data, dan kesulitan saat mencari data.

Berdasarkan latar belakang dan permasalahan tersebut, maka disusun tugas akhir dengan judul “ RANCANG BANGUN SISTEM INFORMASI PEMBELIAN BAHAN BAKU MENGGUNAKAN PHP 5.6.12 DAN MYSQL 5.6.26 PADA PT BANEEX INDONESIA” untuk membantu perusahaan dalam proses pembelian bahan baku sehingga perusahaan dapat berjalan lebih baik dan mempunyai keunggulan kompetitif di dunia industri.

1.1 Pokok Permasalahan

Berdasarkan latar belakang yang dipaparkan sebelumnya, maka dapat diidentifikasi permasalahan-permasalahan yang terjadi pada proses pembelian bahan baku di PT Baneex Indonesia adalah sebagai berikut:

1. Sistem pembelian bahan baku kurang terintegrasi dikarenakan dokumen dan laporan masih dibuat menggunakan *Microsoft Excel*.

2. Pegawai menyimpan dokumen tidak tertata dengan rapi sehingga beresiko terjadi kerusakan data, kehilangan data, dan kesulitan saat mencari data.

1.2 Tujuan Penelitian

Tujuan penelitian dari tugas akhir ini yang dilakukan terhadap divisi *Purchasing* PT Baneex Indonesia dalam melakukan pembelian bahan baku adalah untuk merancang dan membangun sistem informasi pembelian bahan baku yang mampu:

1. Merancang dan membangun sistem informasi pembelian bahan baku agar proses pembelian bahan baku menjadi terintegrasi dengan baik.
2. Membuat fasilitas penyimpanan data dengan menggunakan database agar dapat disimpan di satu lokasi sehingga mengurangi resiko kerusakan data, kehilangan data, dan kesulitan saat mencari data.

1.3 Batasan Masalah

Agar dalam penulisan Tugas Akhir ini lebih fokus dan lebih terarah, maka batasan masalahnya yaitu ruang lingkup yang diamati hanya sebatas mengenai proses proses permintaan bahan baku hingga pembuatan *purchase order*, tidak mencakup masalah pembayaran dan tidak membahas mengenai *return* pembelian bahan baku.

1.4 Manfaat Tugas Akhir

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

1. Bagi penulis

Sebagai pengalaman menganalisis suatu sistem yang riil di sebuah perusahaan dan diharapkan dapat memberikan solusi terhadap permasalahan yang terjadi didalam sistem tersebut.

2. Bagi perusahaan

- a. Memberikan partisipasi dalam pengembangan di dunia pendidikan.

- b. Mendapatkan saran yang diharapkan mampu menyelesaikan permasalahan yang terjadi di perusahaan khususnya masalah penerapan sistem informasi pada perusahaan.
- 3. Bagi Institusi Pendidikan
 - a. Untuk menghasilkan lulusan yang memiliki keterampilan sesuai dengan kebutuhan dunia kerja.
 - b. Menjalinkan kerja sama antara dunia pendidikan dan dunia kerja dalam rangka pengembangan ilmu.

1.5 Sistematika Penulisan

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

BAB I PENDAHULUAN

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

BAB II LANDASAN TEORI

Pada bab ini membahas tentang berbagai teori yang diperoleh dari buku-buku literatur ataupun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan pada laporan ini adalah seputar sistem informasi, *Unified Modelling Language* (UML), dan teori-teori lain yang berhubungan dengan perancangan sistem informasi.

BAB III METODOLOGI PENELITIAN

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

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini akan membahas tentang data yang telah diperoleh berdasarkan penelitian di PT Baneex Indonesia, meliputi profil perusahaan, struktur organisasi, proses bisnis sistem informasi pembelian bahan baku yang berjalan, dan dokumen yang terlibat pada proses bisnis tersebut.

BAB V ANALISIS DAN PEMBAHASAN

Pada bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis sistem yang meliputi diagram alir sistem usulan, pemodelan sistem dengan UML pemodelan data dengan kamus data, perancangan tampilan layar, perancangan hierarki menu, dan pembuatan spesifikasi sistem yang diperlukan.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran dalam penerapan sistem informasi pembelian bahan baku untuk perusahaan dalam penelitian dan pengembangan selanjutnya.

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Menurut Sutabri (2016), suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur komponen, atau variable yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu. Unsur-unsur yang mewakili suatu sistem secara umum adalah masukan (*input*), pengolahan (*processing*), dan keluaran (*output*).

Suatu sistem dapat terdiri dari bagian-bagian sistem atau subsistem. Subsistem adalah serangkaian kegiatan yang dapat ditentukan identitasnya yang berhubungan dalam suatu sistem. Sebagai contoh, sistem computer data terdiri dari subsistem perangkat keras dan subsistem perangkat lunak. Masing-masing subsistem dapat terdiri dari subsistem-subsistem yang lebih lagi atau terdiri dari komponen-komponen pendukung sistem itu sendiri. Berikut pengertian sistem menurut beberapa ahli:

1. Sistem adalah seperangkat komponen yang saling berhubungan dan saling bekerjasama untuk mencapai beberapa tujuan (Winanti, 2014).
2. Sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan (Kadir, 2014).
3. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu (Jogiyanto, 2005).
4. Sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu (Mustakini, 2009).

2.1.1 Karakteristik Sistem

Menurut Sutabri (2016) Model umum sebuah sistem terdiri dari *input*, *proses*, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat

sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bias dikatakan sebagai suatu sistem. Adapun karakteristiknya yang dimaksud adalah :

a. 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.

b. Batasan Sistem (*Boundary*)

Batasan sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

c. 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.

d. Penghubung Sistem (*Interface*)

Penghubung (*interface*) merupakan media yang menghubungkan sistem dengan subsistem lain disebut lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem

akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. 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.

f. Keluaran Sistem (*Output*)

Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi informasi yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Contoh, sistem informasi. Keluaran dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi *input* bagi subsistem lain.

g. Pengolahan Sistem (*Procces*)

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

h. 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

Menurut Sutabri (2016), sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang

berbeda untuk setiap kasus yang terjadi yang ada didalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandangan, diantaranya :

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan, sedangkan sistem fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penggajian, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contoh *human machine system* karena menyangkut penggunaan computer yang berinteraksi dengan manusia.

3. Sistem Deterministik dan Sistem Probabilistik

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistik.

4. Sistem Terbuka dan Sistem Tertutup

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

2.2 Konsep Dasar Informasi

Menurut McLeod (dalam Yakub, 2012) informasi (*information*) adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya. Informasi juga disebut data yang diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan. Para pembuat keputusan memahami bahwa informasi menjadi faktor kritis dalam menentukan kesuksesan atau kegagalan dalam suatu bidang usaha. Sistem apapun tanpa ada informasi tidak akan berguna, karena sistem tersebut akan mengalami kemacetan dan akhirnya berhenti. Informasi dapat berupa data mentah, data tersusun, kapasitas sebuah saluran informasi, dan sebagainya.

Menurut Sutabri (2016), informasi adalah sebuah istilah yang tidak tepat dalam pemakaiannya secara umum. Informasi dapat mengenai data mentah, data tersusun, kapasitas saluran komunikasi, dan lain sebagainya. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil, dan akhirnya mati. Informasi adalah data yang telah diklasifikasi atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.

2.2.1 Kualitas Informasi

Menurut Jogiyanto (dalam Yakub, 2012) kualitas dari *informasi (quality of information)* tergantung dari tiga hal yaitu ; *accurate*, *timeliness*, dan *relevance*.

- a. Relevan (*relevance*), berarti informasi tersebut mempunyai manfaat untuk pemakainya dan relevansi informasi untuk tiap-tiap orang akan berbeda-beda.
- b. Tepat waktu (*timeliness*), berarti informasi tersebut datang pada penerima tidak boleh terlambat. Informasi yang sedang sang tidak akan mempunyai nilai lagi, karena informasi merupakan landasan di dalam pengambilan keputusan.
- c. Akurat (*accurate*), berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke

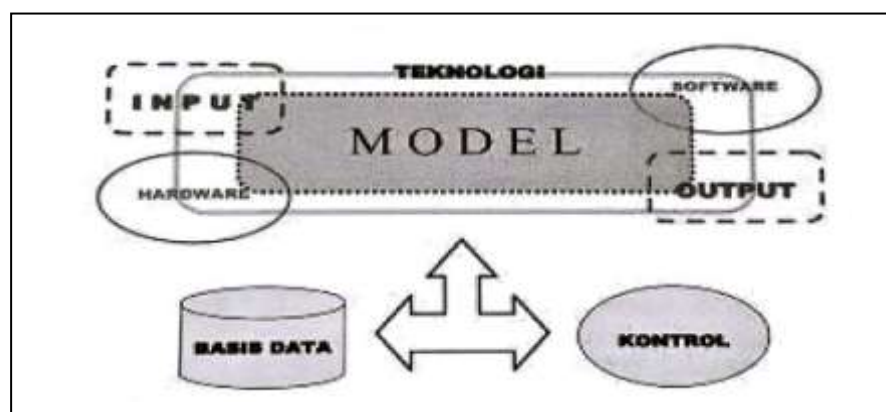
pennerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merusak informasi.

2.3 Konsep Dasar Sistem Informasi

Menurut O'Brian (dalam Yakub, 2012) sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah dan menyebarkan informasi dalam sebuah organisasi.

Menurut Azhar (dalam puspitawati, 2011) sistem infromasi merupakan komponen-komponen dari subsistem yang saling berhubungan dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mgolah data menjadi informasi.

Menurut Sutabri (2016) , sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kebiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data, dan blok kendali. Komponen sistem informasi ini digambarkan seperti Gambar II.1 berikut:



Gambar II.1 Komponen Sistem Informasi

Sumber: Sutabri (2016)

1. Blok Masukan

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

2. Blok Model

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

3. Blok Keluaran

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

4. Blok Teknologi

Teknologi merupakan “Tool Box” dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data

Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras computer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak (*software*) yang disebut DBMS (*database management system*).

6. Blok Kendali

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

2.4 Konsep Dasar Pembelian

Menurut Assauri (2008), pembelian merupakan salah satu fungsi yang penting dalam berhasilnya operasi suatu perusahaan. Fungsi ini dibebani tanggung jawab untuk mendapatkan kuantitas dan kualitas bahan-bahan yang tersedia pada waktu dibutuhkan dengan harga yang sesuai dengan harga yang berlaku. Pengawasan perlu dilakukan terhadap pelaksanaan fungsi ini, karena pembelian menyangkut investasi dana dalam persediaan dan kelancaran arus bahan ke dalam pabrik.

Menurut Irawati (2008), pembelian adalah suatu kegiatan untuk memperoleh sejumlah harta atau aktiva maupun jasa dari satu pihak untuk kelangsungan usaha atau kebutuhan yang mendasar, sehingga dilakukan pembayaran atas sejumlah uang atau jasa tersebut, untuk kelangsungan operasional perusahaan.

2.4.1 Tanggung Jawab Pembelian

Menurut Damiri (2005), dalam bukunya Manajemen Pembelian, Penerimaan, dan Penyimpanan tugas dan tanggung jawab dari divisi *purchasing* di sebuah perusahaan adalah sebagai berikut:

1. Mencari pemasok (Rekanan).
2. Menerima *purchase requisition* atau *additional stock request* atau *application for capital expenditure* dan *operating asset* dari departemen atau *outlet*.
3. Melaksanakan pembelian barang yang dibutuhkan dengan harga yang wajar dan kompetitif, kuantitas dan kualitas yang telah ditentukan dan tepat waktu.

4. Memeriksa laporan pembelian dan penerimaan harian sehingga diketahui barang-barang yang telah masuk atau belum (*purchase order status*).
5. Melaksanakan *market survey* sewaktu-waktu untuk memonitor harga-harga.
6. Mengadakan kontrak pembelian dengan pemasok untuk barang-barang tertentu bila dianggap perlu setelah dianalisis dengan cermat.
7. Membuat laporan kepada manajemen (biasanya kepada *general manager*) tentang perbandingan harga-harga (*comparison price*) antara harga pasar dengan *supplier*.
8. Membuat spesifikasi pembelian.
9. Memonitor situasi ekonomi, politik, dan kejadian-kejadian alam terutama yang dapat mempengaruhi keadaan ekonomi.
10. Mengadakan pertemuan-pertemuan secara berkala internal departemen pembelian atau dengan manajemen untuk mengevaluasi keadaan agar lebih mengoptimalkan kinerja perusahaan.

2.4.2 Tahap-Tahap Proses Pembelian

Menurut Puspitawati dan Anggadini (2011), aktivitas pembelian secara umum dapat diuraikan sebagai berikut:

1. Menentukan Kebutuhan Produk/Jasa (permintaan pembelian)

Timbulnya kebutuhan akan produk atau jasa dalam suatu perusahaan biasanya diperoleh dari dokumen permintaan pembelian (PP) yang dikirimkan bagian produksi atau bagian persediaan untuk memenuhi tersedianya produk yang diminta dalam divisinya masing-masing. Dokumen permintaan pembelian dapat dibuat secara otomatis oleh suatu sistem aplikasi tertentu dan dapat pula dipersiapkan secara manual. Setelah selesai dibuat dokumen permintaan pembelian akan divalidasi terlebih dahulu oleh supervisor dalam divisi yang membuatnya untuk selanjutnya dikirimkan ke divisi yang dituju.

2. Memilih Sumber Daya

Apabila permintaan pembelian sudah dikirimkan ke bagian pembelian maka bagian pembelian akan melakukan aktivitas pemilihan sumber daya. Aktivitas ini dilakukan bagian pembelian untuk memilih jenis sumber daya yang

dibutuhkan/diminta dan menyortir apakah permintaan tersebut dapat disetujui atau tidak. Apabila tahap ini sudah dilaksanakan bagian pembelian akan melakukan pemilihan pemasok yang akan diberikan *order* pembelian.

3. Memilih Pemasok

Proses pemilihan pemasok merupakan aktivitas pemilihan terhadap penyedia produk. Proses pemilihan ini dilakukan dengan pertimbangan untuk mendapatkan barang/jasa yang berkualitas dan harga yang telah disepakati bersama. Evaluasi terhadap pemasok harus menggunakan tenaga ahli yang berpengalaman dengan membandingkan beberapa pemasok yang sejenis sehingga dapat diperoleh pemasok yang dapat menyediakan barang sesuai dengan kriteria yang dibutuhkan, proses evaluasi pemasok dapat dilakukan baik dengan cara manual maupun dengan cara yang terkomputerisasi. Cara manual dapat dilakukan dengan menganalisis catatan/identitas data dari masing-masing pemasok. Untuk aplikasi yang berbasis komputerisasi beberapa aplikasi komputer sudah dapat melakukan aktivitas ini salah satunya adalah dengan program SAP (*System Application Product*). Aplikasi SAP menawarkan evaluasi pemasok secara otomatis untuk membantu organisasi dalam bisnis prokuremen.

4. Menerbitkan pesanan pembelian

Pesanan pembelian dapat dilakukan apabila bagian pembelian sudah mendapatkan pemasok yang akan diinginkan. Pesanan pembelian dapat dilakukan dengan cara menerbitkan dokumen pesanan pembelian yang dikirimkan pemasok ke bagian pembelian. Dokumen pesanan pembelian yang mengidentifikasi pemasok dan menginformasikan barang yang dipesan, jumlah, harga, tanggal pengiriman, jangka waktu pengiriman dan jangka waktu pembayaran yang diinginkan oleh perusahaan. Pesanan pembelian dapat dikirimkan dalam bentuk dokumen/kertas seperti surat atau fax, atau secara elektronik seperti *Electronic Data Interchange* (EDI).

5. Penerimaan barang

Setelah pemasok menyetujui *order* yang dikirimkan perusahaan, pemasok akan mengirimkan produk yang diminta kepada perusahaan. Proses pengiriman yang

baik harus disertai dengan dokumen pengirimannya yang berisi informasi barang yang dikirimkan beserta harga barang dan informasi yang dibutuhkan lainnya. Pemasok akan menyertakan barang yang dikirimkan beserta dokumen pengirimannya. Barang yang dikirimkan pemasok diterima perusahaan oleh bagian penerimaan barang. Bagian penerimaan barang akan melakukan pengecekan secara fisik jumlah barang yang dikirimkan yang dicocokkan dengan dokumen pengirimannya yang dicocokkan dengan dokumen pengirimannya. Apabila barang yang diterima telah sesuai dengan dokumen pengiriman dan dokumen pesanan pembelian maka bagian penerimaan barang akan membuat laporan penerimaan barang. Selanjutnya barang tersebut akan dikirimkan ke bagian gudang/persediaan/penyimpanan untuk disimpan. Dokumen penerimaan dibuat oleh bagian gudang ketika barang diterima dan disimpan oleh bagian gudang dengan kata lain dokumen ini dipersiapkan ketika suatu *item* barang dikirim dari sumber penerima ke sumber penyimpanan barang.

6. Verifikasi Faktur

Verifikasi faktur dilakukan ketika perusahaan akan melakukan pembayaran kepada pemasok. Faktur harus diperiksa dan dicocokkan dengan dokumen penerimaan barang dan dokumen pesanan pembeliannya. Kegiatan ini dilakukan dengan tujuan agar terciptanya pengendalian internal dalam aktivitas pengeluaran kas untuk pembayaran kepada pemasok dan memastikan bahwa biaya dan kebutuhan kuantitas sudah sesuai dan terpenuhi dengan baik.

7. Pembayaran Dengan Pemasok

Apabila ketiga dokumen tersebut sudah cocok maka perusahaan akan melakukan pembayaran dengan pemasok. Pembayaran dapat dilakukan sesuai dengan jangka waktu pembayaran dan persyaratan yang ditentukan dalam pesanan pembelian. Pembayaran diproses melalui hutang dagang dalam akuntansi keuangan.

Ada beberapa tahapan prosedur pembelian menurut Mulyadi (2001) adalah sebagai berikut:

1. Prosedur permintaan pembelian

Dalam prosedur ini, fungsi gudang mengajukan permintaan pembelian dalam formulir surat penerimaan pembelian kepada fungsi pembelian. Jika barang tidak disimpan di gudang, misalnya untuk barang-barang yang langsung dipakai, fungsi yang memakai barang mengajukan permintaan pembelian langsung ke fungsi pembelian dengan menggunakan surat permintaan pembelian.

2. Prosedur permintaan penawaran harga pemilihan pemasok

Dalam prosedur ini fungsi pembelian mengirimkan surat penawaran harga kepada para pemasok untuk memperoleh informasi mengenai harga barang dan berbagai syarat pembelian yang lain, untuk memungkinkan pemilihan pemasok yang akan ditunjukan sebagai pemasok barang yang diperlukan oleh perusahaan.

3. Prosedur order pembelian

Dalam prosedur ini fungsi pembelian mengirimkan surat order pembelian kepada pemasok yang dipilih dan memberitahukan kepada unit-unit organisasi lain dalam perusahaan (misalnya fungsi penerimaan, fungsi meminta barang dan fungsi pencatat utang) mengenai order pembelian yang sudah dikeluarkan oleh perusahaan.

4. Prosedur penerimaan barang

Dalam prosedur ini fungsi penerimaan melakukan pemeriksaan mengenai jenis, kualitas dan mutu barang yang diterima dari pemasok dan kemudian membuat laporan penerimaan barang untuk menyatakan barang dari pemasok tersebut.

5. Prosedur pencatatan utang

Dalam prosedur ini fungsi akuntansi memeriksa dokumen-dokumen yang berhubungan dengan pembelian (surat order pembelian, laporan penerimaan

barang dan faktur dari pemasok) dan menyelenggarakan pencatatan utang atau mengarsipkan dokumen sumber sebagai catatan utang.

6. Prosedur distribusi pembelian

Prosedur ini meliputi distribusi rekening yang didebet dari transaksi pembelian untuk kepentingan pembuatan laporan manajemen.

2.5 Konsep Dasar Bahan Baku

Menurut Baroto (2002), bahan baku adalah barang-barang yang terwujud seperti tembakau, kertas, plastik ataupun bahan-bahan lainnya yang diperoleh dari sumber-sumber alam atau dibeli dari pemasok, atau diolah sendiri oleh perusahaan untuk digunakan perusahaan dalam proses produksinya sendiri.

Menurut Hanggana (2006), pengertian bahan baku adalah sesuatu yang digunakan untuk membuat barang jadi, bahan pasti menempel menjadi satu dengan barang jadi. Dalam sebuah perusahaan bahan baku dan bahan penolong memiliki arti yang sangat penting, karena menjadi modal terjadinya proses produksi sampai hasil produksi.

Menurut Nafarin (2007), bahan baku merupakan bahan langsung (*direct material*), yaitu bahan yang membentuk suatu kesatuan yang tidak terpisahkan dari produk jadi. Bahan baku adalah bahan utama atau bahan pokok dan merupakan komponen utama dari suatu produk. Bahan baku biasanya lebih mudah ditelusuri dalam suatu produk dan harganya relatif tinggi dibandingkan dengan bahan pembantu. Misalnya produk kursi rotan bahan bakunya rotan. Adapun bahan pembantu dari produk kursi rotan, seperti paku, lem kayu. Bahan pembantu (*indirect material*) merupakan bahan pelengkap yang melekat pada suatu produk. Bahan pembantu biasanya tidak mudah ditelusuri dalam suatu produk dan harganya relatif rendah dibandingkan dengan bahan baku.

Adapun jenis-jenis bahan baku menurut Gunawan Adisaputro dan Marwan Asri (2011) adalah :

1. Bahan baku langsung

Bahan baku langsung atau *direct material* adalah semua bahan baku yang merupakan bagian daripada barang jadi yang di hasilkan. Biaya yang di keluarkan untuk membeli bahan baku langsung ini mempunyai hubungan yang erat dan sebanding dengan jumlah barang jadi yang di hasilkan.

2. Bahan Baku Tidak langsung

Bahan baku tidak langsung atau disebut juga dengan *indirect material*, adalah bahan baku yang ikut berperan dalam proses produksi tetapi tidak secara langsung tampak pada barang jadi yang di hasilkan. Sebagai contoh jenis dari bahan baku menurut Gunawan Adisaputro dan Marwan Asri adalah apabila barang jadi yang di hasilkan adalah meja dan kursi, maka yang merupakan bahan baku langsung dari pembuatan meja dan kursi tersebut adalah kayu, sedangkan yang termasuk kedalam bahan baku tidak langsung adalah paku dan plamir yang berfungsi sebagai perekat kayu dan dasar cat untuk kursi yang di hasilkan.

2.6 *System Development Life Cycle (SDLC)*

Menurut Dennis et al (2015), *System Development Life Cycle (SDLC)* memiliki empat perangkat fase dasar yaitu *planning*, *analysis*, *design* dan *implementation*. Berbeda proyek mungkin menekankan perbedaan bagian-bagian dari SDLC atau pendekatan fase SDLC dalam cara yang berbeda, tetapi semua proyek memiliki elemen dari empat fase tersebut. Masing-masing fase tersebut tersusun dari suatu rangkaian dari langkah-langkah, yang mengandalkan pada teknik yang menghasilkan dokumen spesifik dan *file* yang menjelaskan tentang proyek.

Dalam banyak proyek, fase-fase dan langkah-langkah SDLC diproses dalam suatu alur tahapan dimulai dari awal hingga akhir. Dalam proyek lain, tim proyek memindahkan langkah-langkah tersebut dengan teratur, secara bertahap, secara iteratif atau dalam pola lainnya. Dalam hal ini, dijelaskan fase-fase,

tindakan, beberapa teknik yang digunakan untuk menyempurnakan langkah-langkah dijelaskan secara umum.

Untuk saat ini, ada dua hal penting untuk dipahami tentang SDLC, yaitu:

1. Pertama, yang harus dipahami adalah mendapatkan pengertian secara umum dari fase-fase tersebut dan langkah-langkah yang digunakan dalam proyek sistem informasi dan beberapa teknik yang menghasilkan dokumen.
2. Kedua, hal yang penting untuk dipahami bahwa SDLC adalah proses penyempurnaan secara bertahap.

Dokumen yang dihasilkan dalam tahap analisis, memberikan ide umum dari suatu bagian dari sistem baru. Dokumen dari *deliverable* ini digunakan sebagai *input* pada tahap *design*, yang kemudian disempurnakan untuk menghasilkan dokumen yang menjelaskan secara detail dari sistem yang dibuat. Dokumen ini, akan digunakan dalam tahap implementasi untuk menghasilkan sistem yang sebenarnya. Berikut dibawah ini merupakan penjabaran dari setiap fase dalam SDLC:

A. *Planning* (Perencanaan)

Tahap *planning* adalah proses dasar untuk memahami mengapa sistem informasi harus dibangun dan menentukan bagaimana tim proyek akan membangunnya. Terdapat dua langkah yaitu:

1. Selama permulaan proyek, nilai bisnis suatu sistem terhadap organisasi diidentifikasi: apakah sistem tersebut akan menurunkan biaya atau meningkatkan pendapatan? Sebagian besar gagasan untuk sistem baru berasal dari luar Departemen IT (misalnya Departemen Pemasaran, Departemen Akuntansi) dalam bentuk permintaan sistem. Permintaan sistem menyajikan ringkasan singkat tentang kebutuhan bisnis dan ini menjelaskan bagaimana sistem yang mendukung kebutuhan akan menciptakan nilai bisnis. Departemen IT bekerja sama dengan orang atau Departemen yang menghasilkan permintaan (sponsor proyek) untuk melakukan analisis kelayakan informasi.

2. Setelah proyek disetujui, langkah selanjutnya adalah manajemen proyek. Selama manajemen proyek, manajer proyek membuat sebuah rencana kerja, memilih anggota proyek tersebut dan menentukan teknik-teknik untuk membantu tim proyek mengendalikan dan mengarahkan proyek pada keseluruhan tahap SDLC. Hasil dari manajemen proyek adalah rencana proyek yang menjelaskan bagaimana tim proyek akan mengembangkan sistemnya.

B. *Analysis* (Analisis)

Selama tahap ini, tim proyek menyelidiki sistem yang ada saat ini, mengidentifikasi peluang untuk perbaikan dan mengembangkan sistem baru. Tahap ini memiliki tiga langkah:

1. Strategi analisis yang dikembangkan untuk mengarahkan tim proyek. Seperti misalnya strategi yang termasuk untuk menganalisis sistem yang telah ada (disebut *as-is-system*) beserta masalah-masalah yang ada untuk merancang sistem (disebut sebagai *to-be-system*).
2. Langkah selanjutnya adalah pengumpulan persyaratan (misalnya, melalui wawancara atau kuesioner). Analisis dari informasi ini bersamaan dengan masukan dari sponsor proyek dan banyak orang lainnya mengarah pada pengembangan konsep untuk sistem. Konsep sistem ini kemudian digunakan sebagai dasar untuk mengembangkan seperangkat model analisis bisnis, yang menggambarkan bagaimana bisnis akan beroperasi jika sistem baru telah dikembangkan.
3. Analisis, konsep sistem dan model digabungkan menjadi dokumen yang disebut proposal sistem, yang dipresentasikan ke sponsor proyek dan pengambil keputusan utama lainnya yang memutuskan apakah proyek harus terus dilaksanakan.

C. *Design* (Perancangan)

Tahap *design* memutuskan bagaimana sistem akan beroperasi, dalam hal perangkat keras, perangkat lunak dan infrastruktur jaringan; antarmuka pengguna, formulir dan laporan; program spesifik, *database* dan *file* yang akan dibutuhkan. Meskipun sebagian besar keputusan strategis mengenai

sistem dibuat dalam pengembangan konsep sistem selama tahap analisis, langkah-langkah dalam tahap *design* menentukan secara tepat bagaimana sistem akan beroperasi. Fase *design* memiliki empat langkah:

1. Strategi *design* yang pertama kali dikembangkan. Hal tersebut menjelaskan apakah sistem akan dikembangkan oleh *programmer* perusahaan sendiri, apakah sistem akan diberikan ke perusahaan lain (perusahaan konsultan), atau apakah perusahaan akan membeli paket perangkat lunak yang ada.
2. Pengembangan *design* yang menggambarkan perangkat keras, perangkat lunak dan infrastruktur jaringan yang akan digunakan. Dalam kebanyakan kasus, sistem akan menambah atau mengubah infrastruktur yang sudah ada dalam organisasi. Desain antarmuka menentukan bagaimana pengguna akan bergerak melalui sistem (misalnya, metode navigasi seperti menu dan tombol di layar), formulir dan laporan yang akan digunakan sistem.
3. Pengembangan *database* dan spesifikasi *file*. Pengembangan ini menentukan dengan tepat data apa yang akan disimpan dan dimana akan disimpan.
4. Tim analisis mengembangkan desain program, yang akan mendefinisikan program yang perlu ditulis dan yang akan dilakukan masing-masing program.

D. *Implementation* (Implementasi)

Tahap akhir dalam SDLC adalah tahap *implementation*, dimana sistem benar-benar dibangun. *Implementation* adalah fase yang paling mendapat perhatian, karena kebanyakan sistem merupakan bagian terlama dan paling mahal dari proses pembangunan. Fase ini memiliki tiga langkah:

1. Pertama merekonstruksi sistem. Sistem ini dibangun dan diuji untuk memastikan kinerjanya sebagaimana yang telah dirancang. Karena biaya *bugs* bisa sangat besar, pengujian merupakan salah satu langkah paling kritis dalam implementasi. Sebagian besar organisasi member

lebih banyak waktu dan perhatian untuk menguji daripada menulis program.

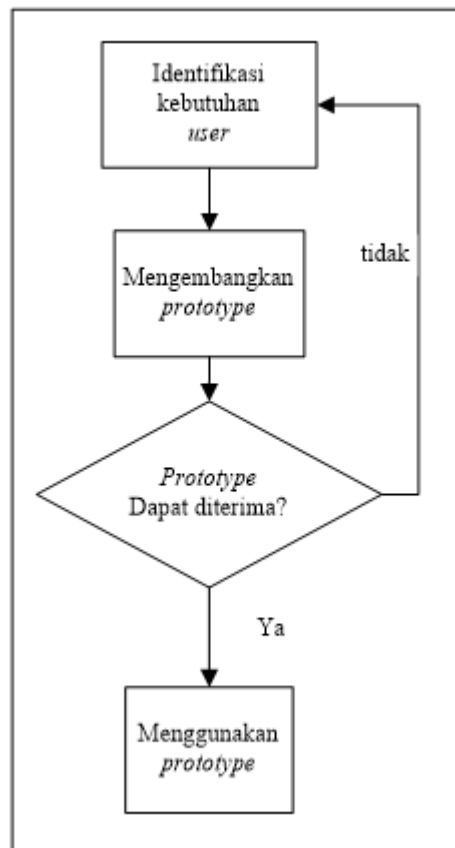
2. Instalasi sistem. Instalasi adalah proses dimana sistem lama dimatikan dan yang baru diaktifkan. Salah satu aspek terpenting adalah pengembangan rencana pelatihan untuk mengajarkan pengguna cara menggunakan sistem baru dan membantu mengelola perubahan yang disebabkan oleh sistem yang baru.
3. Menetapkan rencana dukungan untuk sistem yang dilakukan oleh tim analisis. Rencana ini biasanya mencakup tinjauan pasca-pelaksanaan formal atau informal serta cara sistematis untuk mengidentifikasi perubahan besar dan kecil yang diperlukan untuk sistem tersebut.

2.7 Model *Prototype*

Menurut McLeod (2011), *prototype* adalah suatu versi sistem potensial yang disediakan bagi pengembang dan calon pengguna yang dapat memberikan gambaran bagaimana kira-kira sistem tersebut akan berfungsi bila telah disusun dalam bentuk yang lengkap. Proses dalam memproduksi suatu *prototype* disebut *prototyping*. Tujuannya adalah menghasilkan *prototype* secepat mungkin dan memperoleh umpan balik dari pengguna yang akan memungkinkan *prototype* untuk ditingkatkan sampai sistem dianggap sempurna. Adapun jenis *prototype* terdapat dua jenis *prototype* yaitu *prototype evolusioner (evolutionary prototype)* dan *prototype requirement (requirement prototype)*.

2.7.1 *Prototype Evolusioner*

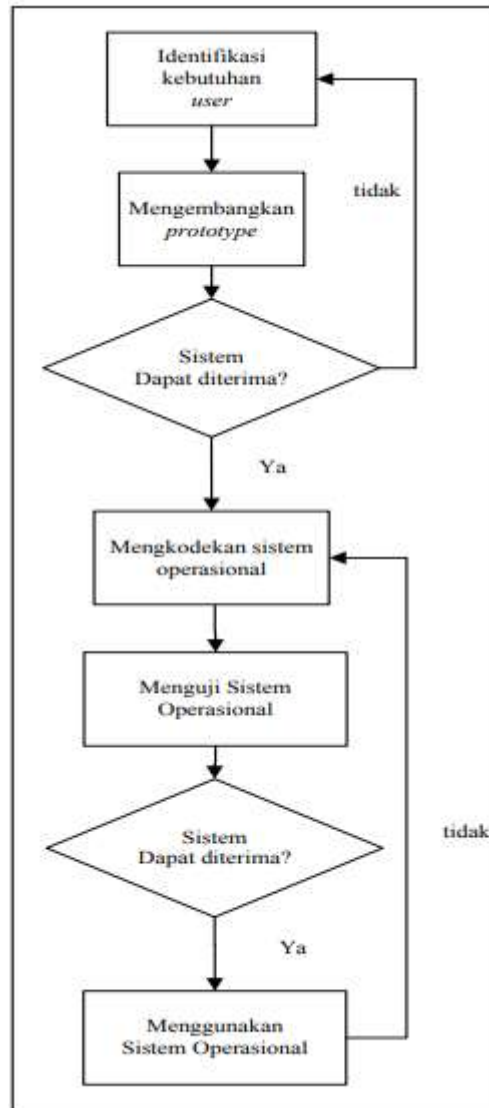
Prototype evolusioner adalah *prototype* yang terus menerus diperbaiki sampai semua kriteria sistem yang baru terpenuhi. Ada empat langkah yang diambil dalam mengembangkan suatu *prototype evolusioner* yaitu identifikasi kebutuhan pengguna, mengembangkan *prototype*, menentukan *prototype* dapat diterima atau tidak, dan penggunaan *prototype*. Adapun *prototype evolusioner* yang dimaksud dapat dilihat seperti pada Gambar 2.2 berikut (Sumber: McLeod, 2011) :



Gambar II.2 Penggambaran Prototipe Evolusioner
(Sumber: McLeod, 2011)

2.7.2 Prototipe Requirement

Prototipe *requirement* adalah suatu pengembangan untuk menentukan kebutuhan fungsional dari sistem baru pada saat para pengguna tidak mampu mengungkapkan dengan tepat apa yang mereka butuhkan. Saat kebutuhan telah ditentukan prototipe *requirement* dapat mulai dikerjakan dan proyek siap untuk mengembangkan suatu sistem yang baru.



Gambar II.3 Penggambaran Prototipe Requirement
(Sumber: McLeod, 2011)

2.8 System Requirements

System requirements atau kebutuhan sistem adalah semua aktivitas yang harus dilakukan atau didukung oleh sistem baru dan batasan-batasan yang harus dicapai sistem baru (Satzinger, 2012). *System requirements* dibagi menjadi 2 kategori, yaitu:

1. Functional Requirement

Merupakan penjelasan tentang layanan yang perlu disediakan oleh sistem, bagaimana sistem menerima dan mengolah masukan, dan bagaimana sistem

mengatasi situasi-situasi tertentu. Selain itu kadang-kadang juga secara jelas menentukan apa yang tidak dikerjakan oleh sistem. *Functional Requirement* menggambarkan *system requirement* secara detil seperti input, output dan pengecualian yang berlaku.

2. *Non-Functional Requirement*

Secara umum berisi batasan-batasan pada pelayanan atau fungsi yang disediakan oleh sistem. Termasuk di dalamnya adalah batasan waktu, batasan proses pembangunan, standar-standar tertentu. Karena berkaitan dengan kebutuhan sistem secara keseluruhan, maka kegagalan memenuhi kebutuhan jenis ini berakibat pada sistem secara keseluruhan.

2.9 *Flowchart*

Flowchart adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah *flowchart* merupakan cara penyajian dari suatu algoritma (Ladjamudin, 2013).

2.9.1 *Simbol-simbol Flowchart*

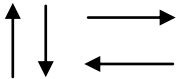
Flowchart disusun dengan simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses di dalam program. Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yakni sebagai berikut.

1. *Flow Diagram Symbols* (simbol penghubung/alur).
2. *Processing Symbols* (simbol proses).
3. *Input-Output Symbols* (simbol input-output)

2.9.2 *Flow Direction Symbols*

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga *connecting line*. Simbol-simbol tersebut adalah sebagai berikut.

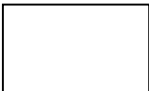
Tabel II.1 *Flow Direction Symbols*

No	Simbol	Keterangan
1.		Simbol Arus atau Flow Untuk menyatakan jalannya arus suatu proses
2.		Simbol Communication Link Untuk menyatakan bahwa adanya transisi suatu data atau informasi dari suatu lokasi ke lokasi lainnya.
3.		Simbol Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembar yang sama.
4.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembar yang berbeda.

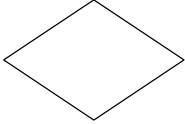
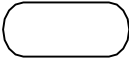
2.9.3 Processing Symbols

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses atau prosedur. Simbol-simbol tersebut adalah sebagai berikut.

Tabel II.2 *Processing Symbols*

No	Simbol	Keterangan
1.		Simbol Process Simbol yang menunjukkan pengolahan yang dilakukan komputer.
2.		Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).

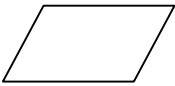
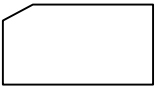
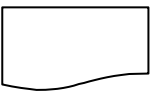
Tabel II.2 *Processing Symbols* (Lanjutan)

No	Simbol	Keterangan
3.		Simbol <i>Decision</i> atau logika Untuk menunjukan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya atau tidak.
4.		Simbol <i>Predefined Proses</i> Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5.		Simbol <i>Terminal</i> Untuk menyatakan permulaan atau akhir suatu program.
6.		Simbol <i>Keying Operation</i> Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i> .
7.		Simbol <i>Offline Storage</i> Untuk menunjukkan bahwa data dalam simbol ini. akan disimpan ke suatu media tertentu.
8.		Simbol <i>Manual Input</i> Untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i> .

2.9.4 *Input-Output Symbols*

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*. Simbol-simbol tersebut adalah sebagai berikut.

Tabel II.3 *Input-Output Symbols*

No	Simbol	Keterangan
1.		Simbol <i>Input-Output</i> Untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
2.		Simbol <i>Punched Card</i> Untuk menyatakan <i>input</i> dan <i>output</i> berasal dari kartu atau <i>output</i> ditulis ke kartu.
3.		Simbol <i>Magnetic-Tape Unit</i> Untuk menyatakan <i>input</i> berasal dari pita <i>magnetic</i> atau <i>output</i> disimpan ke pita <i>magnetic</i> .
4.		Simbol <i>Disk Storage</i> Untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
5.		Simbol <i>Document</i> Untuk menunjukkan dokumen berupa <i>input</i> dan <i>output</i> baik untuk proses manual maupun proses berbasis mekanik atau komputer.
6.		Simbol <i>Display</i> Untuk menyatakan peralatan <i>output</i> yang digunakan berupa layar (video, komputer).

(Sumber: Ladjamudin, 2013)

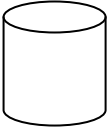
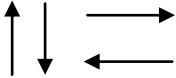
2.10 Diagram Alir Dokumen (*Flowmap*)

Menurut Soeherman dan Marion (2008), *flowmap* adalah campuran peta dan *flowchart* yang menunjukkan pergerakan benda dari satu lokasi ke lokasi lain. Seperti jumlah orang migrasi, jumlah barang yang diperdagangkan atau jumlah paket dalam jaringan. *Flowmap* menolong analisis dan *programmer* untuk memecahkan masalah kedalam segmen atau bagian yang lebih kecil dan menolong dalam menganalisis alternative-alternatif dalam pengoperasian.

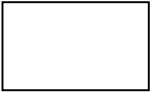
Berikut adalah beberapa petunjuk yang harus diperhatikan dalam membuat *flowmap*:

1. *Flowmap* digambarkan dari halaman atas kebawah dari kiri ke kanan.
2. Aktivitas yang digambarkan harus didefinisikan secara hati-hati dan definisi itu harus dapat dimengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus ditentukan secara jelas.
4. Setiap langkah dari aktivitas harus pada urutan yang benar.
5. Lingkup dan *range* dari aktivitas yang sedang digambarkan harus ditelusuri dengan hati-hati.
6. Gunakan simbol-simbol *flowmap* yang standar.

Tabel II.4 Simbol *flowmap*

No.	Simbol	Keterangan
1.		<u>Dokumen</u> Menunjukkan dokumen berupa dokumen <i>input</i> dan <i>output</i> pada proses manual dan proses berbasis komputer.
2.		<u>Proses Manual</u> Menunjukkan proses yang dilakukan secara manual.
3.		<u>Penyimpanan Magnetik</u> Menunjukkan media penyimpanan data informasi <i>file</i> pada proses berbasis komputer. <i>File</i> dapat disimpan pada <i>harddisk</i> , <i>disket</i> , CD dan lain-lain.
4.		<u>Arah Alir Dokumen</u> Menunjukkan arah aliran dokumen antar bagian yang terkait pada suatu sistem. Bisa dari sistem keluar ataupun dari luar ke sistem dan antar bagian diluar sistem.

Tabel II.4 Simbol *flowmap* (lanjutan)

No.	Simbol	Keterangan
5.		<u>Penghubung</u> Menunjukkan alir dokumen yang terputus atau terpisah pada halaman alir dokumen yang sama.
6.		<u>Proses Komputer</u> Menunjukkan proses yang dilakukan secara terkomputerisasi.
7.		<u>Pengarsipan</u> Menunjukkan simpanan data non-komputer informasi <i>file</i> pada proses manual. Dokumen dapat disimpan pada lemari, arsip, <i>map file</i> dan lain-lain.
8.		<u>Input Keyboard</u> Menunjukkan <i>input</i> yang dimasukkan melalui <i>keyboard</i> .
9.		<u>Penyimpanan Manual</u> Menunjukkan media penyimpanan data informasi secara manual.

(Sumber: Soeherman & Marion, 2008)

2.11 Unified Modelling Language (UML)

Menurut Dennis dkk (2015), *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 kosakata yang umum dari istilah-istilah berbasis objek dan teknik yang cukup banyak untuk memodelkan proyek pengembangan sistem dari analisis ke desain.

Diagram-diagram yang ada dalam UML terbagi menjadi dua bagian utama yaitu *Structure diagram* dan *Behavior diagram*. *Structure diagram* biasanya digunakan untuk mempresentasikan 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.

UML ini mendefinisikan suatu set dari 15 teknik untuk pemodelan sistem. Berikut dijelaskan pada Tabel II.2.

Tabel II.5 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 lainnya 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> .

Tabel II.5 Jenis Diagram Pada UML dan Fungsi-fungsinya (lanjutan)

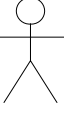
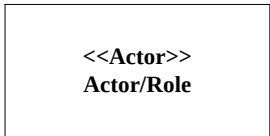
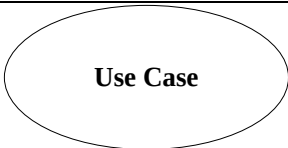
Nama Diagram	Fungsi
<i>Structure Diagrams</i>	
<i>Composite Diagram</i>	Menggambarkan struktur internal dari suatu <i>class</i> dan hubungan diantara bagian-bagian dari suatu <i>class</i> .
<i>Profile Diagram</i>	Digunakan untuk mengembangkan eksistensi UML sendiri.
<i>Behavior Diagrams</i>	
<i>Activity 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.
<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.
<i>Interaction Overview Diagram</i>	Menggambarkan suatu ikhtisar alur dari kontrol suatu proses
<i>Timing Diagram</i>	Menggambarkan interaksi yang terjadi diantara suatu set dari objek-objek dan perubahan keadaan selama perjalanan waktu.
<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

(Sumber: Dennis dkk, 2015)

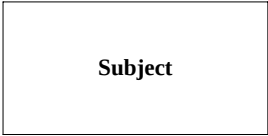
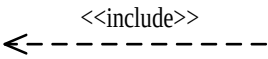
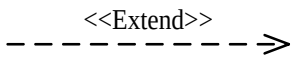
2.11.1 Use Case Diagram

Diagram *use case* dibuat untuk menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Suatu landasan pola pikir yang ditekankan dalam diagram *use case* ini adalah “apa” yang dapat diperbuat oleh sistem, dan bukan “bagaimana” sistem melakukannya. Diagram *use case* menggambarkan kebutuhan sistem dari sudut pandang *user*, yaitu proses yang dilakukan oleh sistem dalam melayani *user* yang berinteraksi dengan sistem tersebut. Simbol-simbol yang terdapat didalam *use case diagram* dijelaskan pada tabel dibawah sebagai berikut:

Tabel II.6 Simbol-Simbol *Use Case Diagram*

No	Simbol	Fungsi
1.	 Actor/Role  <<Actor>> Actor/Role	• Seseorang atau sistem yang mendapatkan keuntungan dari sistem. • Digambarkan sebagai gambar <i>stick</i> /gambar orang (<i>default</i>) atau jika bukan seorang aktor manusia, digambarkan dengan suatu kotak dengan tanda <<actor>> di dalamnya (alternatif). • Dilabelkan dengan peran. • Bisa diasosiasikan dengan <i>actor</i> lainnya berdasarkan spesialisasi. • Di letakan di luar batas sistem.
2.	 Use Case	• Merepresentasikan bagian utama dari sistem secara fungsional. • Dapat menghubungkan <i>use case</i> lain. • Dapat terhubung dengan <i>use case</i> lain. • Diletakan di dalam batas sistem. Dilabelkan dengan frasa kata kerja deskriptif.

Tabel II.6 Simbol-Simbol *Use Case Diagram* (Lanjutan)

No	Simbol	Fungsi
3.	 Subject	• Menyertakan nama subjek di dalam maupun di atas. • Merepresentasikan ruang lingkup dari subjek, sistem atau proses bisnis.
4.	 Asosiasi	• Menghubungkan suatu aktor dengan <i>use case</i> dengan interaksi antara keduanya.
5.	 <<include>>	• Merepresentasikan fungsionalitas suatu <i>use case</i> dengan <i>use case</i> lainnya. • Disimbolkan dengan anak panah dari sebuah <i>use case</i> dasar ke <i>use case</i> yang digunakan.
6.	 <<Extend>>	• Merepresentasikan perluasan (<i>extend</i>) dari <i>use case</i> lain untuk menyertakan perilaku opsional / <i>optional</i> (tidak wajib). • Disimbolkan dengan anak panah yang digambarkan dari perluasan <i>use case</i> ke <i>use case</i> dasar.
7.	 Generalisasi	• Merepresentasikan <i>use case</i> khusus ke satu <i>use case</i> yang lebih umum. • Disimbolkan dengan anak panah yang digambarkan dari <i>use case</i> khusus ke <i>use case</i> umum.

(Sumber: Dennis dkk, 2015)

2.11.2 Activity Diagram

Menurut Dennis dkk (2015), *activity diagram* digunakan untuk memodelkan perilaku dalam proses bisnis yang independen terhadap objek. Dalam banyak hal, *activity diagram* dapat dipandang sebagai data flow diagram yang digunakan pada konjungsi dengan analisis terstruktur. Walaupun begitu,

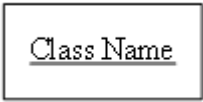
activity diagram tidak seperti data flow diagram, *activity diagram* menggunakan notasi yang mengalamatkan pemodelan paralel.

Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, 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.

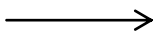
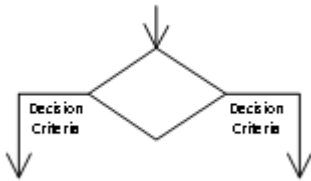
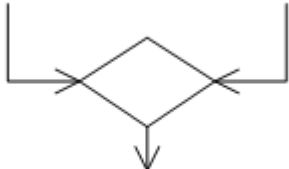
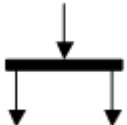
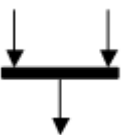
Simbol-simbol yang terdapat didalam *activity diagram* dijelaskan pada tabel dibawah sebagai berikut:

Tabel II.7 Simbol-Simbol *Activity Diagram*

No	Simbol	Nama	Deskripsi
1.		<i>Action</i>	Merupakan notasi yang simple, dan tidak untuk dianalisis perilakunya. Dinamakan sesuai aksi yg dilakukan.
2.		<i>Activity</i>	Digunakan untuk mewakili satu set aksi yang dilakukan.
3.		<i>Object Node</i>	Digunakan untuk mewakili objek yang terhubung ke satu set arus objek.

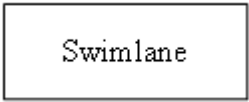
Tabel II.7 Simbol-Simbol *Activity Diagram* (Lanjutan)

No	Simbol	Nama	Deskripsi
----	--------	------	-----------

4.		<i>Control Flow</i>	Menampilkan urutan eksekusi.
5.		<i>Object Flow</i>	Menunjukkan aliran dari sebuah objek dari satu aktivitas ke aktivitas lainnya.
6.		<i>Initial Node</i>	Menggambarkan permulaan dari suatu set aktivitas atau aksi.
7.		<i>Final-activity Node</i>	Menggambarkan akhir dari setiap aktivitas.
8.		<i>Final-flow Node</i>	Menggambarkan akhir aliran control spesifik atau aliran objek.
9.		<i>Decision Node</i>	Menggambarkan suatu kondisi untuk memastikan bahwa arah panah atau arah objek hanya memiliki satu jalur ke bawah.
10.		<i>Merge Node</i>	Digunakan untuk membawa aliran keputusan yang berbeda ke satu <i>decision node</i> .
11.		<i>Fork Node</i>	Menggambarkan percabangan dari satu aliran aktivitas.
12.		<i>Join Node</i>	Menyatukan beberapa percabangan dari aliran aktivitas.

Tabel II.7 Simbol-Simbol Activity Diagram (Lanjutan)

No	Simbol	Nama	Deskripsi
----	--------	------	-----------

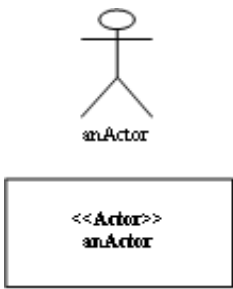
13.		<i>Swimlane</i>	Menggambarkan akhir aliran control spesifik atau aliran objek.
-----	---	-----------------	--

(Sumber: Dennis dkk, 2015)

2.11.3 *Sequence Diagram*

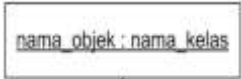
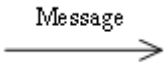
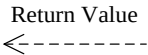
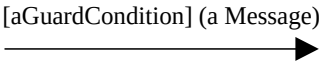
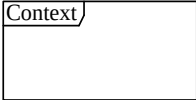
Menurut Dennis dkk (2015), *Sequence diagram* adalah salah satu dari dua jenis diagram interaksi. Mereka menggambarkan benda-benda yang berpartisipasi dalam kasus penggunaan dan pesan yang melewati antara mereka dari waktu ke waktu untuk satu *use case*. Sebuah diagram *sequence* adalah model dinamis 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. Berikut simbol yang ada pada Sequence Diagram:

Tabel II.8 Simbol-Simbol *Sequence Diagram*

No	Simbol	Nama	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.		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.

Tabel II.8 Simbol-Simbol *Sequence Diagram* (Lanjutan)

No	Simbol	Nama	Deskripsi
----	--------	------	-----------

3.		Objek	Berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan yang ditempatkan diatas diagram.
4.		<i>Execcution Occurence</i>	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.
5.		<i>Message</i>	Pesan yang menggambarkan komunikasi yang terjadi antar objek.
6.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri.
7.		<i>Guard Condition</i>	Menggambarkan tes yang harus dipenuhi untuk pesan yang akan dikirim.
8.		<i>Object Destruction</i>	Ditempatkan di akhir lifeline suatu objek untuk menunjukan akhir dari eksitensi.
9.		<i>Frame</i>	Mengindikasikan konteks dari <i>sequence diagram</i> .

(Sumber: Dennis dkk, 2015)

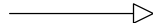
2.11.4 Class Diagram

Berdasarkan Dennis dkk (2015), sebuah *class diagram* digunakan untuk mengatur dan menyimpan informasi yang ada dalam sistem. Selama analisis, *class* mengarah kepada orang, tempat, atau sesuatu yang akan ditangkap oleh sistem. Sebuah *class* digambarkan dengan tiga buah persegi panjang, dimana di bagian atas diisikan dengan nama *class*, di bagian tengah diisi dengan *attribute*, dan di bagian bawah diisikan dengan *operation*. 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). Berikut simbol-simbol yang ada pada *class diagram*:

Tabel II.9 Simbol-Simbol *Class Diagram*

No.	Simbol	Deskripsi
1.	Class 1 -attribute1 +operation1()	– Menggambarkan orang/aktor, tempat, atau sesuatu yang dibutuhkan oleh sistem untuk mendokumentasikan dan menyimpan informasi. – Judul <i>class</i> ditulis dengan huruf tebal dan berada di tengah atas. – <i>Attribute</i> ditempatkan di tengah kotak. – <i>Operation</i> ditempatkan di bawah kotak.
2.	Nama <i>attribute</i>	Mewakili suatu properti untuk mendeskripsikan keadaan dari suatu objek. Dapat diturunkan dari <i>attribute</i> lain dengan menempatkan garis miring sebelum nama <i>attribut</i>.

Tabel II.9 Simbol-Simbol *Class Diagram* (Lanjutan)

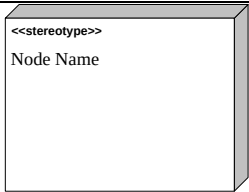
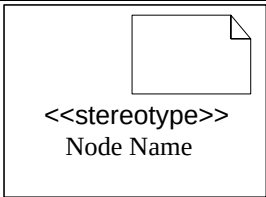
No.	Simbol	Deskripsi
3.	Nama operasi/ <i>operation</i> ()	Mewakili suatu aksi atau fungsi dari suatu <i>class</i> . Dapat diklasifikasikan sebagai konstruktor (<i>method</i> yang secara otomatis dipanggil/dijalankan pada sebuah <i>class</i>), <i>query</i> , atau perubahan operasi. Tanda kurung mengandung informasi atau parameter yang dibutuhkan untuk menampilkan operasi.
4.	<i>AssociatedWith</i> ()...* 	Mewakili sebuah hubungan antara beberapa <i>class</i> atau <i>class</i> dengan <i>class</i> . Digambarkan dengan kata kerja atau nama, yang mewakili sebuah hubungan. Tersedia diantara satu atau lebih <i>class</i> .
5.		<i>A Generalization</i> • Mewakili hubungan sejenis di antara berbagai kelas.
6.	0..* IsPartOf 1 	<i>An Aggregation</i> • Merepresentasikan hubungan yang logis antara beberapa kelas atau kelas dan itu sendiri. • Adalah bentuk khusus dari sebuah asosiasi.
7.	1..* IsPartOf 1 	<i>A Composition</i> • Merepresentasikan hubungan yang logis antara beberapa kelas atau kelas dan itu sendiri. • Adalah bentuk khusus dari sebuah asosiasi.

(Sumber: Dennis dkk, 2015)

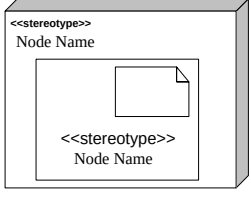
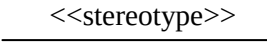
2.11.5 Deployment Diagram

Deployment diagram digunakan untuk mewakili antara komponen perangkat keras yang digunakan dalam infrastruktur fisik dari suatu sistem informasi. Misalnya, ketika merancang sistem informasi terdistribusi yang akan menggunakan jaringan area luas, *deployment diagram* dapat digunakan untuk menunjukkan hubungan komunikasi di antara berbagai *node* dalam jaringan. Mereka juga dapat digunakan untuk mewakili komponen perangkat lunak dan bagaimana mereka dikerahkan atas arsitektur fisik ke infrastruktur sistem informasi. dalam hal ini, *deployment diagram* mewakili lingkungan untuk pelaksanaan perangkat lunak.

Tabel II.10 Simbol-Simbol *Deployment Diagram*

No.	Simbol	Deskripsi
1.		<i>A Node</i> • Adalah sumber daya komputasi, misalnya, komputer klien, <i>server</i>, jaringan terpisah atau perangkat jaringan individual. • Diberi label berdasarkan nama. • Dapat berisi stereotip untuk label khusus jenis <i>node</i> yang diwakili, misalnya, perangkat, <i>workstation</i> klien, <i>server</i> aplikasi, perangkat seluler, dll.
2.		<i>An Artifact</i> • Adalah spesifikasi perangkat lunak atau basis data, misalnya, basis data atau tabel atau tampilan basis data, komponen atau lapisan perangkat lunak. • Diberi label dengan namanya. • Dapat berisi stereotip untuk jenis label khusus artefak, misalnya, file sumber, tabel basis data, file yang dapat dieksekusi, dll.

Tabel II.10 Simbol-Simbol *Deployment Diagram* (Lanjutan)

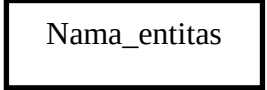
No.	Simbol	Deskripsi
3.		<i>A Node with Deployed Artifact</i> • Menggambarkan artefak yang ditempatkan pada <i>node</i> fisik.
4.		<i>An Communication Path</i> • Mewakili hubungan antara dua <i>node</i>. • Memungkinkan <i>node</i> untuk bertukar pesan. • Dapat berisi stereotip untuk secara khusus memberi label jenis jalur komunikasi yang diwakili, (misalnya, LAN, internet, serial, paralel).

(Sumber: Dennis dkk, 2015)

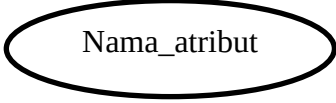
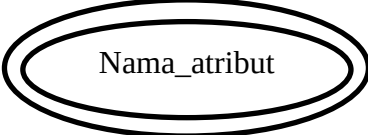
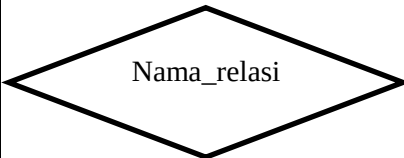
2.12 Entity Relationship Diagram (ERD)

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.12.

Tabel II.11 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.

Tabel II.11 Simbol-simbol *Entity Relationship Diagram* (Lanjutan)

No.	Simbol	Deskripsi
3.	Atribut kunci primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan.
4.	Atribut multivalai / 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 .

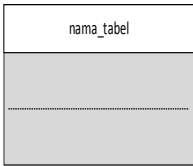
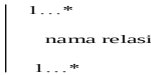
(Sumber: Rosa dan Shalahuddin, 2015)

2.12.1 *Conceptual Data Model (CDM)*

CDM (*Conceptual Data Model*) atau model konsep data merupakan konsep yang berkaitan dengan pandangan pemakai terhadap data yang disimpan dalam basis data. CDM dibuat sudah 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.12 Simbol-Simbol *Conceptual Data Model (CDM)*

No.	Simbol	Deskripsi
1.	Entitas/tabel 	Entitas atau tabel yang menyimpan data dalam basis data.
2.	Relasi 	Relasi antar tabel yang terdiri atas nama relasi dan <i>multiplicity</i> .

(Sumber: Rosa& Shalahuddin, 2015)

Aturan untuk mengubah ERD menjadi CDM secara umum adalah sebagai berikut:

Tabel II.13 Simbol-Simbol Aturan *Conceptual Data Model* (CDM)

No.	ERD	CDM						
1.	Petugas Entitas	<table><tr><th colspan="2">E2</th></tr><tr><td>PK</td><td><u>username</u></td></tr><tr><td></td><td>password nama no_petugas hak_akses</td></tr></table> Menjadi sebuah tabel tersendiri.	E2		PK	<u>username</u>		password nama no_petugas hak_akses
E2								
PK	<u>username</u>							
	password nama no_petugas hak_akses							

Tabel II.13 Simbol-Simbol Aturan *Conceptual Data Model* (CDM) (lanjutan)

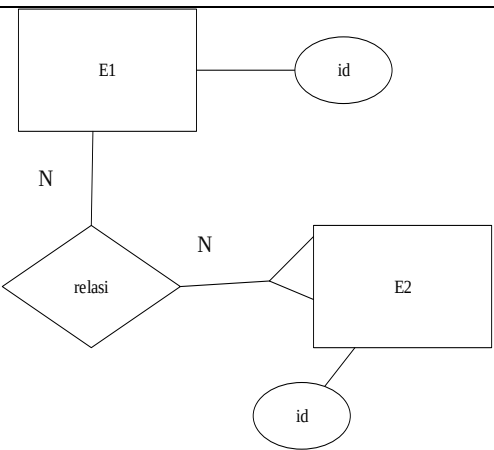
No.	ERD	CDM
-----	-----	-----

2.	pengarang Atribut <i>multivalue</i>	<table><tr><th colspan="2">E2</th></tr><tr><td>PK</td><td><u>id_pustaka</u></td></tr><tr><td>PK</td><td><u>pengarang</u></td></tr><tr><td></td><td></td></tr></table> 1..* Menjadi sebuah tabel tersendiri dengan kunci primer (<i>primary key</i>) adalah kunci primer pada entitas dan memiliki atribut dengan nama seperti pada atribut entitas.	E2		PK	<u>id_pustaka</u>	PK	<u>pengarang</u>		
E2										
PK	<u>id_pustaka</u>									
PK	<u>pengarang</u>									
3.	E1 id relasi atribut_relasi E2 id N N Relasi dengan kardinalitas <i>many to many</i>	<table><tr><th colspan="2">E2</th></tr><tr><td>PK</td><td><u>id E1</u></td></tr><tr><td>PK</td><td><u>id E2</u></td></tr><tr><td></td><td></td></tr></table> Menjadi sebuah tabel tersendiri dengan kunci primer adalah atribut yang menjadi kunci primer di kedua entitas yang direlasikannya.	E2		PK	<u>id E1</u>	PK	<u>id E2</u>		
E2										
PK	<u>id E1</u>									
PK	<u>id E2</u>									

Tabel II.13 Simbol-Simbol Aturan *Conceptual Data Model* (CDM) (lanjutan)

No.	ERD	CDM
-----	-----	-----

4.

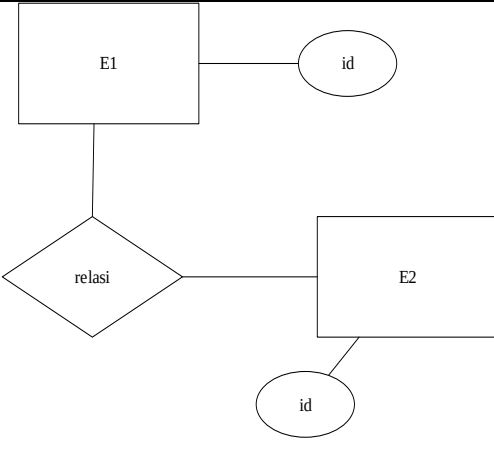


Relasi dengan kardinalitas *one to many*

E2	
PK	<u>id E1</u>
PK	<u>id E2</u>

Kunci primer entitas yang memiliki hubungan *one* akan dijadikan kunci primer di entitas yang memiliki hubungan *many* dengan kata lain, relasi tidak menjadi tabel sendiri.

5.



Relasi dengan kardinalitas *one to one*

E2	
PK	<u>id E1</u>
PK	<u>id E2</u>

Kunci primer salah satu entitas akan dijadikan kunci asing (*foreign key*) pada tabel yang lain dan kunci asing itu dijadikan kunci primer juga, dengan kata lain, relasi tidak menjadi tabel sendiri.

(Sumber: Rosa& Shalahuddin, 2015)

2.13 Database

Menurut Sutabri (2016), basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras computer dan menggunakan perangkat lunak untuk

memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak (*software*) yang disebut DBMS (*database management system*).

Menurut Jubilee Enterprise (2017), database adalah kumpulan data yang terorganisasi secara rapi. Data itu bias berbentuk banyak hal, seperti deretan nama, alamat, dan nomor telepon atau deretan kode plat nomor dan nama kota yang diwakili oleh nomor-nomor plat tersebut.

2.14 Kamus Data

Menurut Puspitawati (2011), kamus data adalah suatu penjelas tertulis mengenai data yang berada di dalam database. Kamus data pertama berbasis dokumen, kamus data itu tersimpan dalam bentuk hard copy dengan mencatat semua penjelasan data dalam bentuk tercetak. Walau sejumlah kamus berbasis dokumen masih ada, praktek yang umum sekarang adalah menggunakan kamus data berbasis computer. Pada kamus data berbasis computer, penjelasan data dimasukkan ke dalam computer dengan menggunakan data description language (DDL) dari system manajemen database, sistem kamus, atau peralatan CASE. Secara garis besar kamus data dapat didefinisikan sebagai berikut :

1. Kamus data merupakan fakta tentang data dan kebutuhan informasi dari suatu Sistem informasi.
2. Kamus data adalah daftar organisasi dari semua elem data yang ada dalam sistem secara lengkap, dengan definisi yang baku sehingga user dan analisis sistem akan memiliki pengertian sama untuk input, output, komponen penyimpanan dan perhitungannya.
3. Pada saat perancangan, kamus data digunakan untuk keperluan perancangan program.

2.15 Windows Navigation Diagram (WND)

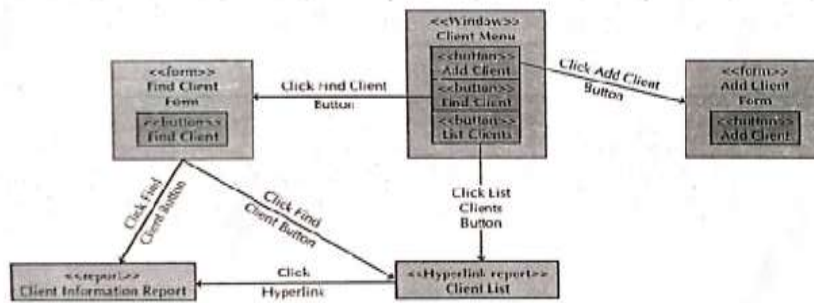
Menurut Dennis dkk (2015), desain struktur navigasi mendefinisikan komponen dasar antarmuka dan bagaimana mereka bekerja sama untuk menyediakan fungsionalitas kepada pengguna. *Windows Navigation Diagram* (WND) digunakan untuk menunjukkan bagaimana semua layar, bentuk, dan laporan yang digunakan oleh 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 PHP (*Hypertext Preprocessor*)

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 Kadir (2018), MySQL (baca : “Mai-es-kyu-el”) merupakan sistem manajemen database terkenal yang sekarang dimiliki oleh Oracle dan salah satu produknya yang bernama MySQL Community Server bersifat “Open Source”. Nama My sendiri berasal dari nama anak wanita sang pembuat kedua MySQL, yakni Michale Widenius. Adapun SQL (baca: “se-kyue-el”) yang merupakan singkatan dari “*Structured Query Language*” merupakan bahasa standar yang digunakan untuk mengakses database.

My Structured Query Language (MySQL) adalah *Relational Database Management System* (RDBMS) yang sangat cepat dan kuat. Sebuah basis data memungkinkan *user* untuk melakukan penyimpanan yang efisien, pencarian dan pengurutan data. MySQL *server* memiliki kemampuan melakukan kontrol akses terhadap data untuk memastikan bahwa setiap *user* dapat bekerja dengan sesuai, menyediakan akses yang cepat, dan meyakinkan bahwa hanya *user* yang mempunyai hak akses yang dapat mengaksesnya. Oleh karena itu, MySQL merupakan *database server* yang yang dapat digunakan banyak *user* dan banyak pekerjaan (Welling dan Thomson, 2003).

2.17.1 Keunggulan MySQL

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

1. *Portability*

Dapat berjalan stabil pada berbagai sistem operasi, di antaranya: Windows, Linux, FreeBSD, MacOS X Server, Solaris, Asigma.

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

11. *Interface*

Memiliki antarmuka terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan API.

12. *Client dan tools*

Dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi basis data sekaligus dokumen petunjuk *online*.

13. Struktur tabel

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

2.17.2 Tipe Data MySQL

MySQL mendukung banyak tipe data yang dapat disimpan pada sebuah kolom. Penjelasan singkat kegunaan masing-masing jenis data akan sangat bermanfaat dalam memilih suatu jenis data yang dipakai dalam merancang *table*. Beberapa jenis data yang tersedia pada MySQL dapat dilihat pada Tabel II.15 (Sutaji, 2012).

Tabel II.14 Jenis Data pada MySQL

Jenis Data	Keterangan
CHAR	Sebuah <i>string</i> dengan panjang tetap. Sisa Jumlah karakter yang belum terisi akan diisi dengan spasi, akan tetapi spasi ini dibuang jika data dipanggil. Jangkauan nilai M adalah 1-255 karakter
VARCHAR	<i>String</i> dengan panjang berupa variabel. M bisa mencapai 65535
DATE	Data berupa tanggal. Format tanggal dalam bentuk 'YYYY-MM-DD'
TIME	Data berupa waktu. Format waktu dalam bentuk 'HH:MM:SS'
TINYINT	Bilangan antara -128 sampai dengan +127
SMALLINT	Bilangan antara -32768 sampai dengan +32767
INT	Bilangan antara -2147683648 sampai dengan +2147683647
FLOAT	Bilangan <i>floating point</i> yang kecil (presisi tunggal). Jangkauan nilai yang diizinkan adalah -3.402823466E+38 s/d -1.175494351e-38,0 dan 1.175494351E-38 s/d 3.402823466E+38

Tabel II.13 Jenis Data pada MySQL (Lanjutan)

Jenis Data	Keterangan
DOUBLE	Bilangan <i>floating point</i> dengan ukuran normal (presisi ganda). Jangkauan nilai yang diizinkan adalah - 1.7976931348623157E+308 s/d -2.225073858507201E-308,0 dan 2.225073858507201E-308 s/d 1.7976931348623157E+308
ENUM	Sebuah <i>enumeration</i> . Sebuah objek <i>string</i> yang hanya boleh memiliki satu nilai, yang terambil dari 'value1','value2','...,NULL atau nilai spesial "" <i>error</i> . Sebuah <i>enum</i> dapat menampung 65535 pilihan nilai.
TEXT, BLOB	Sebuah TEXT atau BLOB dengan panjang karakter maksimum 65535 karakter

(Sumber: Sutaji, 2012)

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, dan memilih langkah-langkah sistematis. Metodologi penelitian mempelajari cara-cara melakukan pengamatan dengan pemikiran yang tepat secara terpadu melalui tahapan-tahapan yang disusun secara ilmiah untuk mencari, menyusun, serta menganalisis dan menyimpulkan data-data berdasarkan fakta-fakta secara ilmiah.

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

3.2 Jenis dan Sumber Data

Data yang diperlukan dalam penelitian ini ada dua jenis, yaitu data primer dan data skunder. Sumber dari data-data ini berasal dari tempat yang diamati pada praktek kerja lapangan pada PT Baneex Indonesia.

1. Data Primer

Data Primer adalah data yang diperoleh secara langsung dari perusahaan, dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data-data tersebut adalah data yang digunakan dalam proses pembelian bahan baku diantaranya struktur organisasi, analisis sistem yang telah berjalan, proses bisnis sistem saat ini dan yang akan diusulkan, dan kebutuhan pengguna sistem.

2. Data Sekunder

Data sekunder adalah data yang diperoleh dari data yang tersedia dan telah terlebih dahulu dikumpulkan dan dilaporkan oleh orang lain, buku-buku dan kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian. Data

tersebut adalah data umum perusahaan, profil perusahaan, dan struktur organisasi perusahaan.

3.3 Metode Pengumpulan Data

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

1. Studi Lapangan

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

- a. Pengamatan, tahap ini dilakukan secara langsung di divisi *purchasing* PT Baneex Indonesia dengan mengamati proses pembelian bahan baku yang berjalan, pengamatan dilakukan mulai dari proses permintaan bahan baku yang dilakukan oleh divisi *inventory*, pembelian bahan baku yang dilakukan divisi *purchasing* hingga proses penerimaan bahan baku. Melalui teknik ini, data yang dibutuhkan diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.
- b. Wawancara, Pengambilan data dengan cara berdialog dan bertanya dengan karyawan departemen terkait tentang proses bisnis yang berjalan pada sistem rekrutmen karyawan. Sebagai data yang diperlukan untuk penyusunan tugas akhir.

2. Studi Kepustakaan

Mengumpulkan data dan menambah referensi dengan membaca buku-buku, literatur, artikel di internet atau sumber tertulis lain yang berhubungan dengan judul dan permasalahan guna melengkapi data yang diperlukan dalam penulisan laporan tugas akhir ini, agar dalam praktik dan teori tidak jauh berbeda.

3.4 Metode Pengembangan Sistem

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

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

1. Melakukan wawancara dengan satu orang divisi *inventory*, satu orang divisi *purchasing* dan satu orang divisi *quality control* tentang kondisi sistem saat ini dan kebutuhan sistem yang diinginkan.

2. Membuat satu *prototype*

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

3. Menentukan apakah *prototype* dapat diterima

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

4. Menggunakan *Prototype*

Menjadikan *prototype* sebagai sistem yang dijalankan.

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 observasi, wawancara, dan studi pustaka.

2. Pokok Permasalahan

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 User

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

b. Membuat *Prototype*

Membuat sebuah prototype sesuai dengan analisis yang dilakukan. Pada tahap ini akan dirancang aplikasi tentang sistem pembelian bahan baku. Membuat prototype nantinya akan memuat tentang:

- 1) Memodelkan sistem menggunakan Unified Modeling Language (UML) yaitu menggunakan usecase diagram, activity diagram, dan sequence diagram.

- 2) Memodelkan basis data dengan menggunakan class diagram dan kamus data.
- 3) Merancang sistem dengan menggunakan WND dan rancangan antar muka.
- 4) Membuat *prototype* dengan menggunakan PHP 5.6.12 dan MySQL 5.6.26 sebagai *database*.

c. Mengevaluasi *Prototype*

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

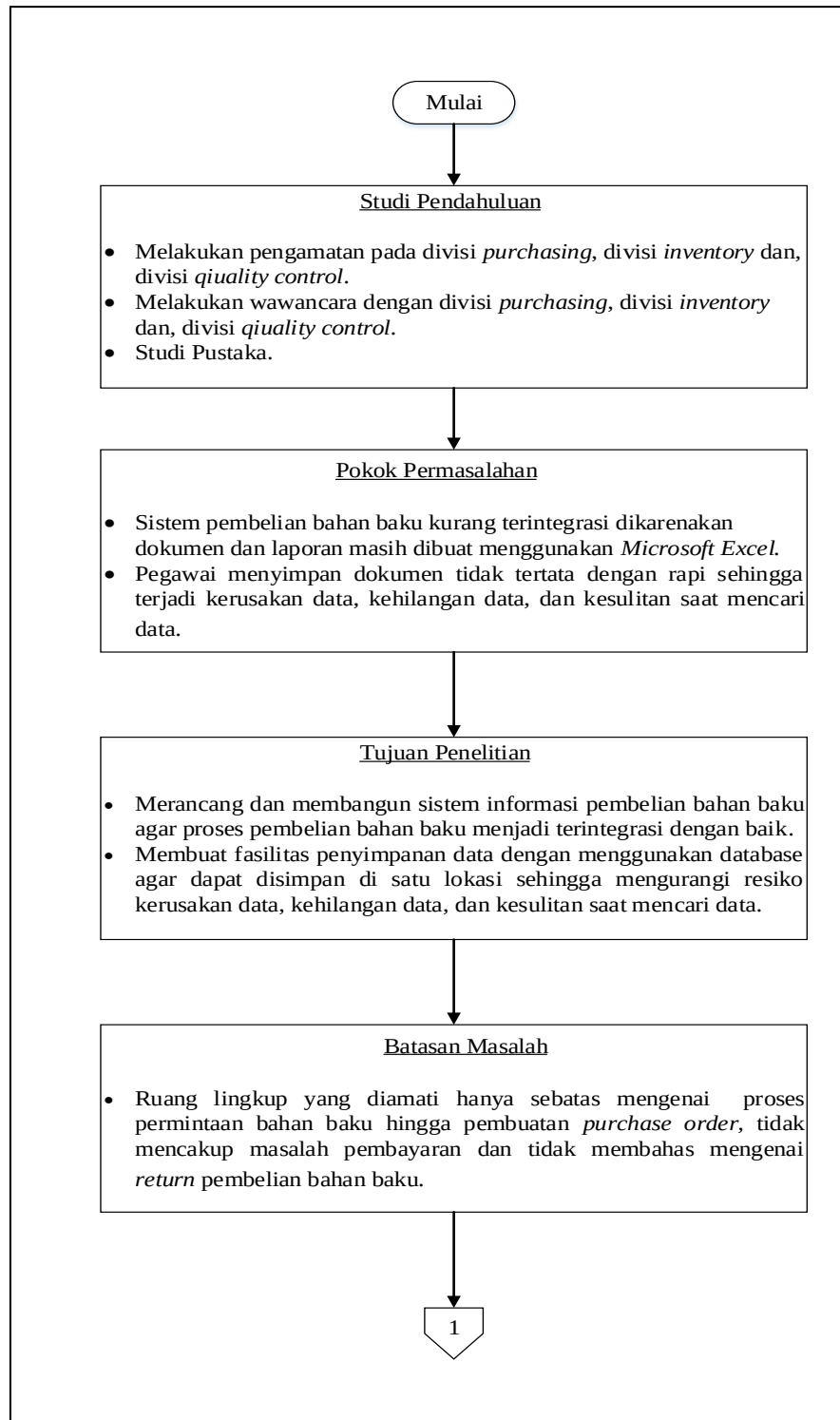
d. Menggunakan *Prototype*

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

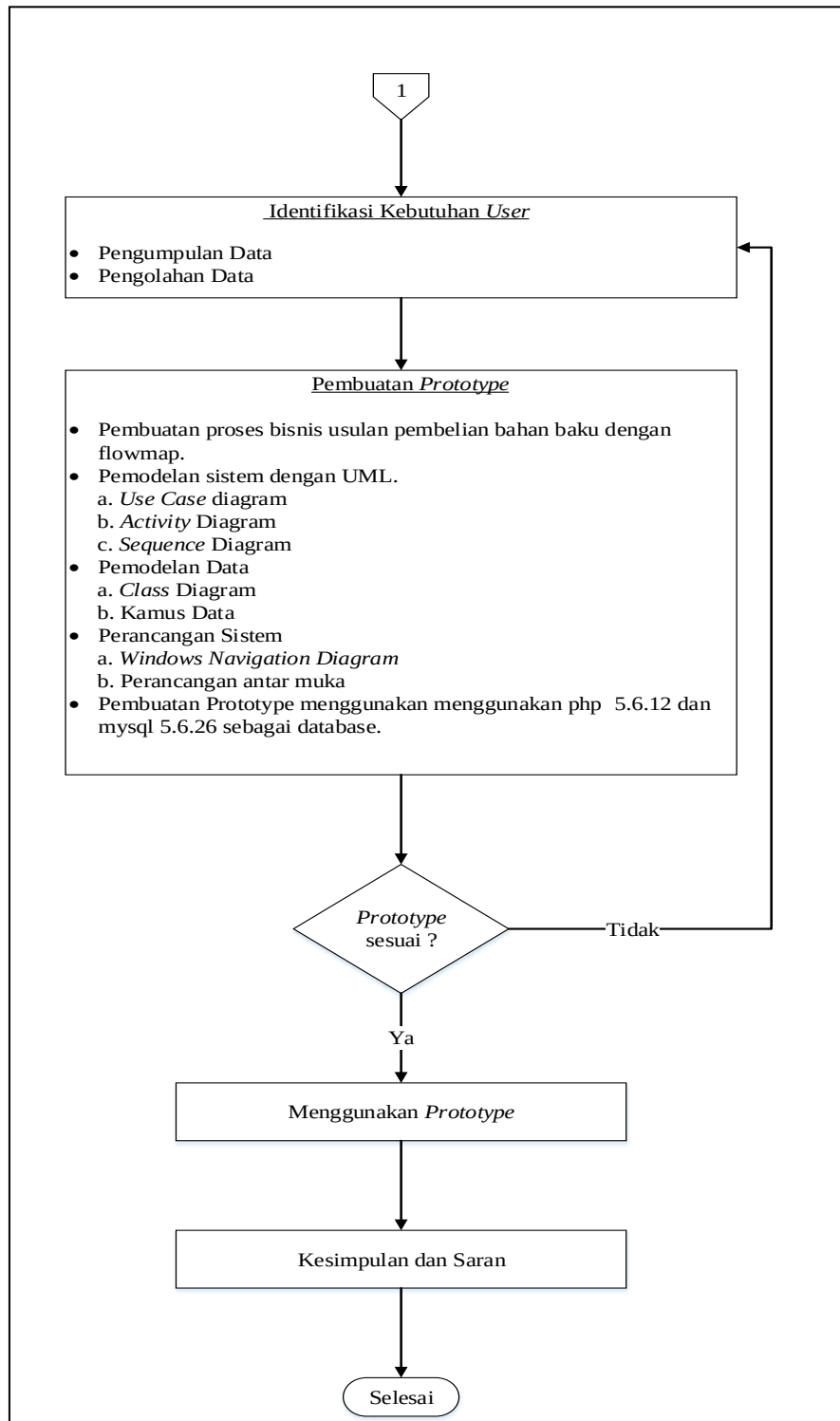
6. Kesimpulan dan Saran

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

Berikut merupakan diagram alir kerangka penelitian yang digunakan dalam laporan tugas akhir ini:



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



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

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sejarah Perusahaan

PT Baneex Indonesia merupakan perusahaan manufaktur industri metal *spring*, *bending* dan *stamping* untuk komponen otomotif. Pimpinan PT Baneex Indonesia saat ini adalah Bapak Ir. H. Tri Yono Sudarno, M.M. Perusahaan ini berdiri pada tahun 2007 dan sudah mengalami perkembangan dan perubahan pada saat ini. Pada tahun 2007, PT Baneex Indonesia berlokasi di Jalan Industri Selatan I TOB, Blok OO No. 1B Kawasan Industri Jababeka Cikarang - Bekasi 17500 dengan luas bangunan sebesar 576 M². Pada awal mulanya perusahaan mempunyai karyawan sebanyak 60 orang dengan produk yang dihasilkan adalah *spring* dan *wire*. Kemudian pada tahun 2010, PT Baneex Indonesia berpindah lokasi ke daerah Pekayon, Bekasi Selatan dengan karyawan sebanyak 10 orang dikarenakan berbagai faktor dan salah satunya yaitu *cost reduction*. Pada tahun 2012, PT Baneex Indonesia berpindah lokasi kembali di daerah Wanasari Wanayasa, Cibitung, Bekasi. Kemudian pada 2014 PT Baneex Indonesia berpindah lokasi ke daerah Toyogiri, Tambun Bekasi.

Perusahaan ini sebagai produsen dan *supplier* komponen otomotif dan elektronik khususnya komponen *seat cushion* mobil mempunyai komitmen untuk selalu memberikan produk dan layanan terbaik yang mampu memenuhi dan menterjemahkan keinginan pelanggan. Agar dapat memenuhi komitmen tersebut, maka dibuat suatu pedoman sistem manajemen mutu. Program pedoman sistem manajemen mutu ini diuraikan di dalam manual mutu yang mengacu kepada ISO 9001:2015 dan sebagai panduan dan penerapannya, maka perusahaan mencanangkan kebijakan mutu sebagai berikut:

“Kepuasan pelanggan adalah tujuan yang paling utama di dalam tanggung jawab sosial perusahaan dan merupakan dasar untuk mengembangkan perusahaan”.

4.2 Profil Perusahaan

Nama perusahaan	: PT Baneex indonesia
Alamat Head Office	: Jl.Express Blok WW 6 RT 002/RW Bojong Rawa Lumbu, Kota Bekasi
Alamat Pabrik 1	: Jl.Toyogiri RT.002 RW.03–Kp.Legon Kel. Jatimulya Kec.Tambun Selatan
Alamat Pabrik 2	: Jl. Raya Setu – Kp. Burangkeng RT 007 RW 009 Ds.Ciledug – Kec. Setu
Telepon	: 021-8201162, 0822-1351-9049
Fax	: 021-82400274
Email	: baneexindonesia@yahoo.com
Jenis Usaha	: Industri Metal <i>Spring, Bending</i> dan <i>Stamping.</i>
Produk	: Otomotif <i>Parts</i>
Status	: Perseroan Terbatas

4.3 Logo Perusahaan

PT Baneex Indonesia memiliki logo perusahaan sebagai berikut:



Gambar IV.1 Logo PT Baneex Indonesia

(Sumber : PT Baneex Indonesia, 2017)

4.4 Visi,Misi dan Kebijakan Mutu Perusahaan

PT. Baneex Indonesia adalah produsen komponen Otomotif dan Elektronik mempunyai komitmen untuk selalu memberikan produk dan layanan terbaik yang mampu memenuhi dan menterjemahkan keinginan pelanggan. Perusahaan ini mempunyai visi dan misi perusahaan sebagai berikut:

4.4.1 Visi :

“Menjadi mitra utama Perusahaan Otomotif dan Elektronik yang kompetitif dan handal dalam mutu dan layanan.”

4.4.2 Misi :

Untuk mewujudkan visi tersebut PT. Baneex Indonesia mengambil langkah-langkah dasar sebagai berikut :

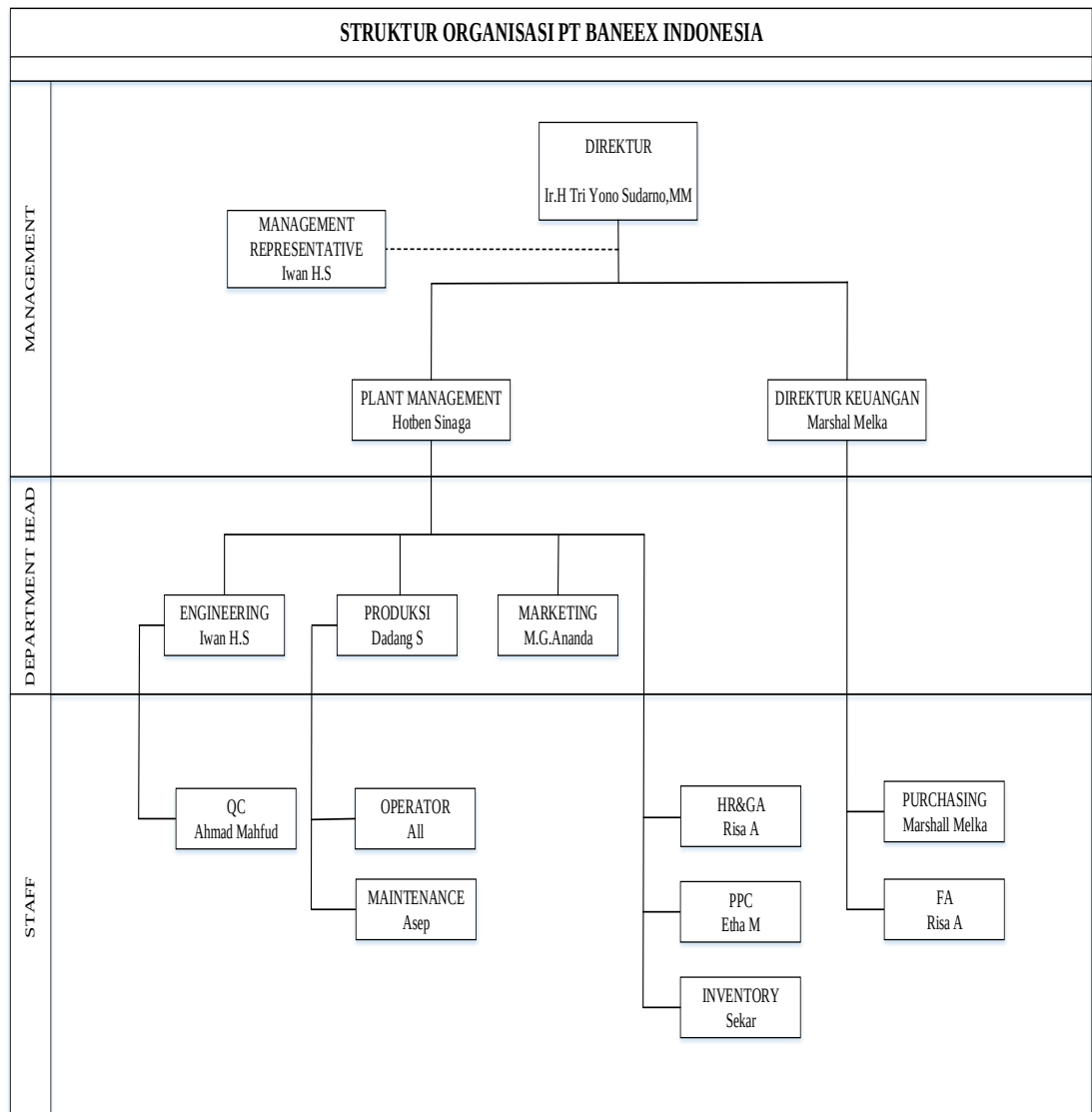
1. *Innovations*, selalu melakukan perbaikan (inovasi) secara terus menerus dan berkesinambungan baik proses teknis maupun kinerja karyawan.
2. Jaminan Mutu, menetapkan kerangka kerja proses peningkatan mutu yang sistematis dan terorganisir dengan membakukan proses guna memastikan konsistensi dan mampu telusur serta meningkatkan hubungan antar fungsi yang dapat mempengaruhi mutu.
3. Kepuasan Pelanggan, selalu memberikan produk dan layanan yang terbaik bagi pelanggan.
4. Kompetitif, meningkatkan produktivitas dan mutu produk serta layanan dengan memenuhi persyaratan pelanggan melalui kerjasama dan komunikasi yang baik, pengendalian bisnis dan penurunan pemborosan yang di akibatkan dari mutu kerja yang buruk.

4.4.3 Kebijakan Mutu

Meningkatkan produktivitas dengan perbaikan berkesinambungan dan mematuhi peraturan perundangan untuk mencapai kepuasan dan kepercayaan pelanggan.

4.5 Struktur Organisasi Perusahaan

Suatu struktur organisasi menggambarkan pembagian kerja, pelimpahan wewenang, kesatuan perintah dan tanggung jawab yang jelas. Berikut merupakan struktur organisasi PT Baneex Indonesia secara keseluruhan:



Gambar IV.2 Struktur Organisasi PT Baneex Indonesia

(Sumber: PT Baneex Indonesia, 2017)

4.5.1 Tugas dan Wewenang

Fungsi dan Uraian Tugas dari Struktur Organisasi PT Baneex Indonesia dapat dijelaskan sebagai berikut:

1. **Direktur**
 - a. Bertanggung jawab untuk memimpin perusahaan.
 - b. Mengawasi pelaksanaan aktifitas dari perusahaan secara keseluruhan meliputi aktifitas administrasi, produksi, keuangan, dan hubungannya dengan permasalahan pemasaran.
 - c. Merumuskan dan menetapkan strategi bisnis perusahaan untuk mencapai visi dan misi perusahaan.
 - d. Memilih staf-staf yang membantu kinerja perusahaan.
 - e. Menyetujui anggaran tahunan perusahaan.
 - f. Memimpin rapat umum perusahaan dalam hal meningkatkan perusahaan.
 - g. Menetapkan peraturan perusahaan, merencanakan, menetapkan sistem operasional perusahaan, menetapkan strategi pencapaian visi dan misi perusahaan.
 - h. Menetapkan kebijakan-kebijakan tentang ketentuan pelaksanaan operasional perusahaan dengan pembagian tugas yang jelas.
2. *Management Representative*
 - a. Melakukan kontrol dokumen internal dan eksternal audit.
 - b. Melakukan tindakan tinjauan manajemen.
 - c. Melakukan Tinjauan Manajemen.
3. *Plant Manager*
 - a. Melakukan pertemuan guna melakukan peninjauan ulang terhadap semua kegiatan yang telah dan sedang berjalan.
 - b. Memeriksa pencapaian program terhadap persoalan yang dihadapi serta memberikan ide-ide perbaikan.
 - c. Memeriksa pelaksanaan kegiatan di lapangan secara langsung pelaksanaan kegiatan di lapangan.

- d. Membuat laporan kinerja kerja perusahaan yang akan dilaporkan kepada pimpinan perusahaan.

4. Direktur Keuangan

Direktur Keuangan merupakan pimpinan yang menjalankan proses pemantauan dan pengambilan keputusan mengenai perihal yang berhubungan dengan keuangan di perusahaan.

5. *Engineering*

- a. Melakukan inspeksi *incoming material* dan produk jadi.
- b. Menganalisa dan melakukan perbaikan terhadap keluhan *customer*.
- c. Melakukan pengendalian alat ukur.
- d. Merencanakan dan mengembangkan *engineering* terutama sistem dan mesin-mesin produksi demi kelancaran semua kegiatan produksi secara keseluruhan.

6. Produksi

- a. Melaksanakan kegiatan produksi.
- b. Melakukan pekerjaan sesuai dengan standar kerja yang telah ditentukan.
- c. Membuat jadwal produksi sejalan dengan jadwal produksi yang diterima dari bagian PPC.
- d. Memeriksa ketersediaan bahan yang akan dipakai sesuai jadwal.
- e. Menjaga, memeriksa, dan merawat serta mengadakan evaluasi mesin, peralatan pendukung, sistem kerja, secara periodik sesuai standar yang telah ditentukan, serta secara langsung mengatasi masalah-masalah yang timbul.

7. *Marketing*

- a. Melakukan pengelolaan order produk.
- b. Melakukan evaluasi kepuasan pelanggan.
- c. Melakukan Penawaran harga dengan *supplier*.

8. *Quality Control*

Bertanggung jawab memeriksa bahan baku, monitoring, memeriksa semua proses produksi yang terlibat dalam produksi suatu produk. Mereka harus memastikan standar kualitas dipenuhi oleh setiap komponen dari produk yang disediakan oleh perusahaan.

9. *Maintenance*

Bertugas untuk perbaikan dan perawatan mesin sebagai penunjang aktivitas produksi.

10. *Operator*

Bertugas untuk mengoperasikan sebuah mesin atau peralatan suatu pabrik dengan prosedur berdasarkan instruksi dari perusahaan.

11. *HR&GA (Human Resource & General Affairs)*

Bertanggung jawab mengelola dan mengembangkan sumber daya manusia. Dalam hal ini termasuk perencanaan, pelaksanaan dan pengawasan sumber daya manusia dan pengembangan kualitas sumber daya manusia.

12. *PPC (Production Planning and Control)*

- a. Melakukan monitoring order.
- b. Menjadwalkan pengiriman produk.
- c. Menangani masalah perencanaan dan pengendalian produksi.

13. *Inventory*

- a. Membuat permintaan pembelian bahan baku.
- b. Melakukan pengendalian *stock* produk dan *stock material*.
- c. Membuat laporan rekapitulasi *stock material* dan produk.

14. *Purchasing*

- a. Melakukan seleksi *supplier* untuk di daftar ke daftar *supplier* terpilih.
- b. Membuat *Purchase Order* (PO) dari permintaan pembelian.
- c. Memonitoring pembelian.
- d. Mengevaluasi kinerja rekanan.

- e. Menjaga hubungan baik dengan bagian lain untuk menjaga budaya perusahaan yang saling mendukung untuk meningkatkan bisnis perusahaan.

15. *Finance and Accounting*

- a. Membuat draft rencana anggaran dan belanja perusahaan serta membuat rencana pengeluaran kas untuk kegiatan operasional perusahaan minggu berikutnya dan diserahkan kepada direktur keuangan.
- b. Mengendalikan transaksi pembelian, penjualan, dan *project* berdasarkan anggaran yang sudah disetujui.

4.6 Pemasok

Pemasok yang biasanya juga disebut *supplier* merupakan mitra yang penting dalam menunjang strategi perusahaan. Pengelolaan pemasok membutuhkan kemampuan negosiasi yang khusus, karena pemasok bukan bagian dari perusahaan. Pada PT Baneex Indonesia, negosiasi tersebut dilakukan oleh *team* dari divisi *purchasing*. Pemilihan pemasok dilakukan dengan hati-hati karena pemasok dapat memiliki dampak yang sangat positif atau yang sangat merugikan pada kinerja keseluruhan perusahaan. Maka dari itu, PT Baneex Indonesia harus mempunyai hubungan yang baik dengan pemasok.

Berikut merupakan daftar nama *supplier* yang bekerja sama dengan PT Baneex Indonesia:

Tabel IV.1 *Supplier* PT. Baneex Indonesia

No.	Nama <i>Supplier</i>
1.	PT Langgeng
2.	PT Mount Zugspitze
3.	PT Jhonatan

(Sumber : PT Baneex Indonesia, 2017)

4.7 Bahan Baku

Bahan baku merupakan salah satu unsur yang sangat penting dan paling aktif di dalam perusahaan yang secara terus menerus diperoleh, diubah menjadi barang yang lebih bernilai jual tinggi dan kemudian dijual kembali. Tanpa bahan baku proses pada suatu perusahaan tidak akan berjalan. PT Baneex Indonesia menghasilkan komponen untuk jok mobil dengan bahan baku besi. Bahan baku ini dibagi menjadi 2 yaitu *Coil* dan *Asetal*. Berikut beberapa jenis besi yang sering dipakai untuk memproduksi produk PT Baneex Indonesia:

1. *Wire Coil*

Wire Coil ini dibagi menjadi beberapa bagian yaitu *high carbon* dan *low carbon*. Produk yang menggunakan bahan baku *Wire Coil* ini antara lain *Wire 56G70*, *Wire 5GB0*, *Wire 56GA0*, *Wire 55G90*, *Wire 67LA0*, *Wire 56G90*, *Wire 56GB0*. Berikut ini adalah gambar bahan baku *Wire Coil*:



Gambar IV.3 SWB - *Wire Spring* 2.00 mm

(Sumber:



PT Baneex

Indonesia,2017)

Gambar IV.4 SWB - *Wire Spring* 2.00 mm

(Sumber: PT Baneex Indonesia, 2017)



Gambar

IV.5 SWC -

Wire Spring 3.80 mm

(Sumber: PT Baneex Indonesia, 2017)

2. *Wire Asental*

Wire Asental adalah bahan baku PT Baneex Indonesia dengan diameter antara 4 mm sampai dengan 7 mm. Bahan baku ini melewati proses *annealing* sehingga produk yang dihasilkan dari bahan baku ini lebih putih dan mengkilap. Produk-produk yang dihasilkan dari bahan baku ini adalah Wire AR Mid Back, Wire BL Mid Back, Wire 959, Trim Wire AR, Trim Wire AL, Trim Wire BR, Trim Wire BL, Trim Wire CR.



Gambar IV.6 SWMB - *Wire* 5.00 mm

(Sumber: PT Baneex Indonesia, 2017)



Gambar IV.7 SWMB - *Wire* 6.00 mm

(Sumber: PT Baneex Indonesia, 2017)



Gambar IV.8 SWMB - *Wire* 7.00 mm

(Sumber: PT Baneex Indonesia, 2017)

4.8 Dokumen Pembelian Bahan Baku

Dokumen yang terlibat dalam sistem informasi pembelian bahan baku adalah sebagai berikut:

1. Form Surat Permintaan Material

Surat permintaan material adalah lembar kertas yang berisikan nama bahan baku yang harus dipesan ke *supplier* dari Divisi Gudang yang ditujukan ke Divisi *Purchasing*.

SURAT PERMINTAAN MATERIAL	
TANGGAL : 25 OKTOBER 2017	
PERIHAL : PERMINTAAN PENAMBAHAN MATERIAL	
Berhubung dari PT. TS TECH meminta barang untuk part model 2 SK dan TE7X maka ada penambahan material :	
A.	SWMB ASS Ø 5 mm = 622 kg
B.	SWMB ASS Ø 6 mm = 30 kg
C.	SWMB ASS Ø 7 mm = 691 kg
Dibuat	
	
SEKAR PERTIWI	

Gambar IV.9 Surat Permintaan Material

(Sumber: PT Baneex Indonesia, 2017)

2. Form *Purchase Order*

Form *purchase order* adalah form yang dibuat oleh Divisi *Purchasing* untuk dikirim ke *supplier*. Form *Purchase Order* akan dibuat apabila surat permintaan material telah diterima oleh Divisi *Purchasing* dari Divisi *Inventory*.

 PT. BANEEX INDONESIA Jl. Imam Bonjol 2 No 2 Cikarang Bekasi - INDONESIA 17530 Phone: (021) 8201162, 832970284, Fax: (021) 82400274									
<u>PURCHASE ORDER</u>									
TO:						No. PO :			
						Tanggal PO :			
Telp : (021)82605302,82611209 Email : bambangs337@gmail.com Dear Sir, We are pleased to give you an order the following goods under the terms and condition set the forth:									
NO.	Description of Goods / Model	Quantity	Satuan	Price / Unit (Rp)	Total Amount (Rp)	FORECAST			Remarks
1			PCS		Rp -	-	-	-	
	PPN 10 %				Rp -				
Please check and confirmation of Baneex Indonesia delivery schedule and please email return to Baneex Indonesia after sign in Supplier Approval									
Time of Delivery Request		By Schedule		Grand Total (Rp)		0.00			
Transport Arrangement		-		Remarks:					
Place of Delivery		PT. Baneex Indonesia							
Term Of Payment									
Conditions Product must be in good condition, if we find theres material in Not Good condition, we will return the Material to Supplier.									
Supplier Approval :		Prepared By:				Approved :			

Gambar IV.10 Purchase Order

(Sumber: PT Baneex Indonesia, 2017)

Dalam *form* terdistribusi *purchase order* ini berisi beberapa atribut-atribut, seperti:

a. *Data Supplier*

Berisi data perusahaan pemasok yang menerima pemesanan barang dari PT Baneex Indonesia seperti: nama perusahaan, nomor telepon, dan *fax*.

b. *Nomor Purchase Order*

Menunjukkan nomor *purchase order*, nama pemasok, bulan serta tanggal *purchase order*.

c. *Tanggal Purchase Order*

Menunjukkan tanggal pemesanan barang PT Baneex Indonesia kepada pemasok.

d. *Descriptions of Goods / Model*

Menunjukkan nama barang yang dipesan oleh PT Baneex Indonesia kepada pemasok.

e. *Quantity*

Menunjukkan berapa banyak barang yang dipesan oleh PT Baneex Indonesia kepada pemasok.

f. *Satuan*

Menunjukkan satuan barang yang dipesan PT Baneex Indonesia kepada pemasok berupa satuan pieces (pcs) atau kilogram (kg).

g. *Price / Unit*

Menunjukkan harga satuan dari barang yang dipesan.

h. *Total Amount*

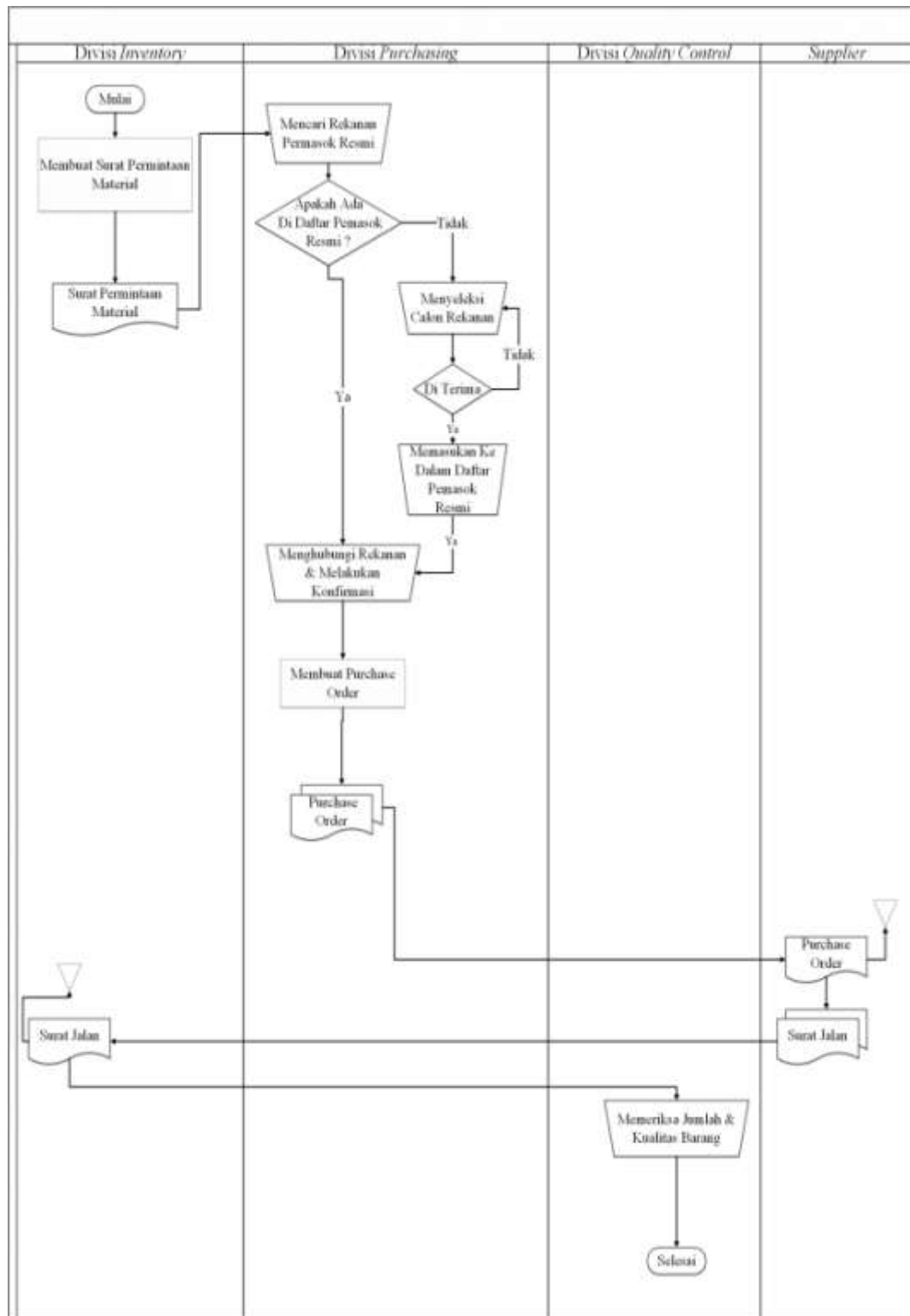
Total Amount merupakan harga total dari barang yang dipesan. *Total Amount* didapat dari *quantity* dan *price / unit*.

4.9 Prosedur Pembelian Bahan Baku pada Divisi *Purchasing*

Adapun prosedur sistem informasi dalam pembelian bahan baku pada divisi *Purchasing*:

1. Divisi *Inventory* membuat form surat permintaan material yang ditujukan kepada Divisi *Purchasing*.
2. Divisi *Purchasing* mencari rekanan pemasok resmi yang ada di dalam daftar pemasok resmi. Jika pemasok belum terdaftar ke dalam daftar pemasok resmi, maka Divisi *Purchasing* akan melakukan penyeleksian kepada calon rekanan, jika calon rekanan telah setuju dengan kesepakatan yang dibuat maka calon rekanan diterima ke dalam daftar pemasok resmi.
3. Divisi *Purchasing* membuat form *purchase order* berdasarkan surat permintaan material.
4. Divisi *Inventory* menerima surat jalan dari *supplier*.
5. Divisi *Quality Control* bertugas memeriksa jumlah barang dan memeriksa kualitas barang sesuai dengan surat jalan yang diberikan dari *supplier*, jika barang tidak sesuai maka Divisi *Quality Control* akan melakukan koordinasi dengan *supplier* untuk diambil keputusan terkait dengan barang yang tidak sesuai.

Berikut merupakan *flowmap* pembelian bahan baku pada Divisi *Purchasing* :



Gambar IV.11 Aliran Dokumen yang Berjalan

(Sumber: Hasil Analisis, 2018)

4.10 Penggambaran Sistem Pembelian Bahan Baku dengan UML (*Unified Modeling Language*)

Analisis terhadap sistem yang sedang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem, kelemahan sistem serta solusi atas masalah pada sistem tersebut. Kegiatan analisis sistem yang berjalan dengan menggunakan analisis sistem yang berorientasi pada objek-objek yang sangat diperlukan oleh sistem yang akan dirancang.

Pemodelan sistem informasi pembelian bahan baku yang berjalan berikut akan direpresentasikan dengan UML melalui diagram *use case*. Pertimbangan menggunakan diagram ini karena dapat menggambarkan sistem yang berjalan dan mudah dipahami karena menggambarkan interaksi antara aktor dengan sistem.

4.10.1 *Use Case Diagram*

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

Sistem informasi pembelian bahan baku memiliki keterkaitan antar aktor, diantaranya Divisi *Inventory*, Divisi *Purchasing*, dan Divisi *Quality Control*. Setiap aktor memiliki kepentingan-kepentingan yang berkaitan dengan sistem tersebut. Penjelasan *use case diagram* sistem informasi pembelian bahan baku yang sedang berjalan di PT Baneex Indonesia adalah sebagai berikut:

1. Definisi aktor

Pendefinisian aktor pada *use case diagram* berdasarkan sistem informasi pembelian bahan baku di PT Baneex Indonesia dapat dilihat pada tabel IV.2 berikut:

Tabel IV.2 Definisi Aktor Sistem Informasi Pembelian Bahan Baku

No	Aktor	Definisi
1	Divisi <i>Inventory</i>	Divisi <i>Inventory</i> membuat surat permintaan material dan menyerahkan surat permintaan material tersebut kepada Divisi <i>Purchasing</i> untuk dicarikan supplier dan dibuatkan <i>Purchase Order</i> .
2	Divisi <i>Purchasing</i>	Menerima surat permintaan material dari Divisi <i>Inventory</i> yang berisi bahan baku yang dibutuhkan dan membuat form <i>Purchase Order</i> .
3	Divisi <i>Quality Control</i>	Memeriksa jumlah dan kualitas barang.

(Sumber: Hasil Analisis, 2018)

2. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* pembelian bahan baku yang sedang berjalan pada Divisi *Purchasing* di PT Baneex Indonesia dapat dilihat pada tabel IV.3 berikut:

Tabel IV.3 Definisi *Use Case* Pembelian Bahan Baku

No	Nama <i>Use case</i>	Definisi
1	Membuat surat permintaan material	Proses pembuatan form surat permintaan material yang dilakukan Divisi <i>Inventory</i> yang ditujukan kepada Divisi <i>Purchasing</i> .
2	Mencari rekanan pemasok	Proses mencari dan mengecek data rekanan supplier.
3	Menghubungi rekanan	Proses yang dilakukan Divisi <i>Purchasing</i> untuk melakukan kontak dengan <i>supplier</i> .
4	Melakukan	Proses yang dilakukan Divisi <i>Purchasing</i> untuk

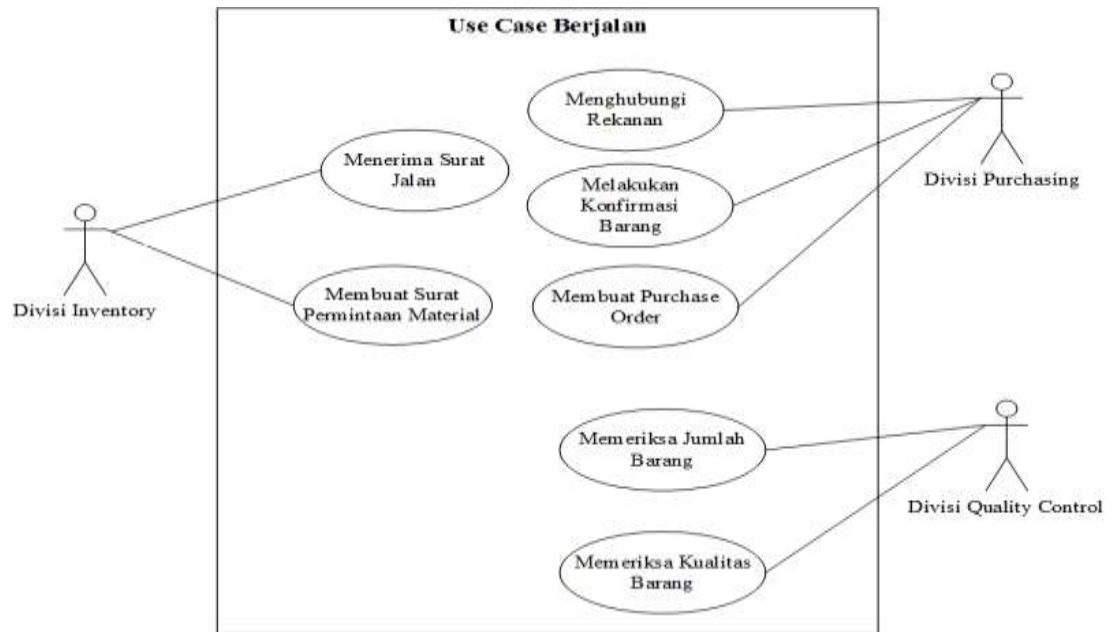
	konfirmasi barang	mengkonfirmasi barang yang dipesan.
--	-------------------	-------------------------------------

Tabel IV.3 Definisi *Use Case* Pembelian Bahan Baku (Lanjutan)

No	Nama <i>Use case</i>	Definisi
5	Membuat <i>purchase order</i>	Proses membuat form <i>purchase order</i> untuk dikirim ke <i>supplier</i> .
6	Menerima surat jalan	Proses penerimaan surat jalan dari <i>supplier</i> ke Divisi <i>Inventory</i> .
7	Memeriksa jumlah barang	Proses pemeriksaan jumlah barang dengan cara memeriksa aktual sesuai spesifikasi pesanan atau memeriksa kesesuaian aktual dengan surat.
8	Memeriksa kualitas barang	Proses pemeriksaan jumlah barang dengan cara memeriksa aktual sesuai spesifikasi pesanan atau memeriksa kesesuaian aktual dengan surat.

(Sumber: Hasil Analisis, 2018)

Use Case Diagram sistem informasi pembelian bahan baku yang sedang berjalan pada PT Baneex Indonesia terdiri dari tiga aktor sebagaimana digambarkan pada Gambar IV.12 dibawah ini :



Gambar IV.12 *Use Case* Diagram Sistem Berjalan
(Sumber: Hasil Analisis, 2018)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Berdasarkan hasil pengolahan informasi data dan analisis sistem informasi pada divisi *purchasing* PT Baneex Indonesia maka dibutuhkan sebuah perancangan aplikasi sistem informasi pembelian bahan baku. Berikut merupakan analisis kebutuhan sistem untuk aplikasi sistem informasi pembelian bahan baku:

Tabel V.1 Analisis Kebutuhan Sistem Informasi

No	Permasalahan	Kebutuhan <i>User</i>	Solusi	Kebutuhan Sistem	
				<i>Functional Requirement</i>	<i>Non-Functional Requirement</i>
1.	Pengolahan data pembelian masih belum terintegrasi dengan baik antar divisi.	Pengolahan data yang terintegrasi dan tertata rapi.	Merancang dan membangun sistem informasi pembelian bahan baku.	Sistem dapat mengelola data master.	Data master terdiri dari data <i>supplier</i> , data <i>user</i> dan data bahan baku. Proses mengelola data yaitu, menyimpan, menambah, mengubah,

					mencari, dan menghapus.
--	--	--	--	--	-------------------------

Tabel V.1 Analisis Kebutuhan Sistem Informasi (Lanjutan)

No	Permasalahan	Kebutuhan <i>User</i>	Solusi	Kebutuhan Sistem	
				<i>Functional Requirement</i>	<i>Non-Functional Requirement</i>
2.	Proses permintaan baku masih menggunakan <i>Microsoft Excel</i> .	Sistem yang dapat membantu pembuatan permintaan bahan baku menjadi lebih tertata rapi dan terkomputerisasi	Merancang dan membangun sistem informasi pembelian bahan baku.	Sistem dapat membuat permintaan bahan baku.	Data yang di kelola antara lain no permintaan, nama bahan baku, tanggal permintaan, jumlah dan satuan.
3.	Pembuatan <i>purchase order</i> masih menggunakan <i>Microsoft Excel</i> .	Sistem yang dapat membantu pembuatan <i>purchase order</i> menjadi lebih tertata rapi dan terkomputerisasi.	Merancang dan membangun sistem informasi pembelian bahan baku.	Sistem dapat membuat layanan <i>purchase order</i> .	Data yang di kelola meliputi nomer <i>purchase order</i> , tanggal <i>purchase order</i> , nama <i>supplier</i> , jumlah, satuan, dan harga barang.

4.	Proses penerimaan bahan baku tidak terkomputerisasi.	Sistem yang dapat membantu proses penerimaan bahan baku	Merancang dan membangun sistem informasi	Sistem dapat mengelola proses penerimaan bahan baku.	Data yang di kelola meliputi no surat jalan tanggal surat jalan,
----	--	---	--	--	--

Tabel V.1 Analisis Kebutuhan Sistem Informasi (Lanjutan)

No	Permasalahan	Kebutuhan <i>User</i>	Solusi	Kebutuhan Sistem	
				<i>Functional Requirement</i>	<i>Non-Functional Requirement</i>
		menjadi lebih tertata rapi dan terkomputerisasi	pembelian bahan baku		nama bahan baku, dan jumlah bahan baku
5.	Dokumen data supplier, data bahan baku, permintaan bahan baku, pembelian bahan baku hingga penerimaan bahan baku yang belum terorganisir dengan baik sehingga beresiko sulit mencari data,	Sistem mampu menyimpan data-data dokumen ke dalam database.	Membuat proses penginputan data-data dokumen tersimpan ke dalam database dan membuat laporan data-data dokumen.	Sistem dapat mengelola data pembelian bahan baku.	Proses mengelola data yaitu, menyimpan, menambah, mengubah, dan menghapus.

	kehilangan data dan kerusakan data.				
--	---	--	--	--	--

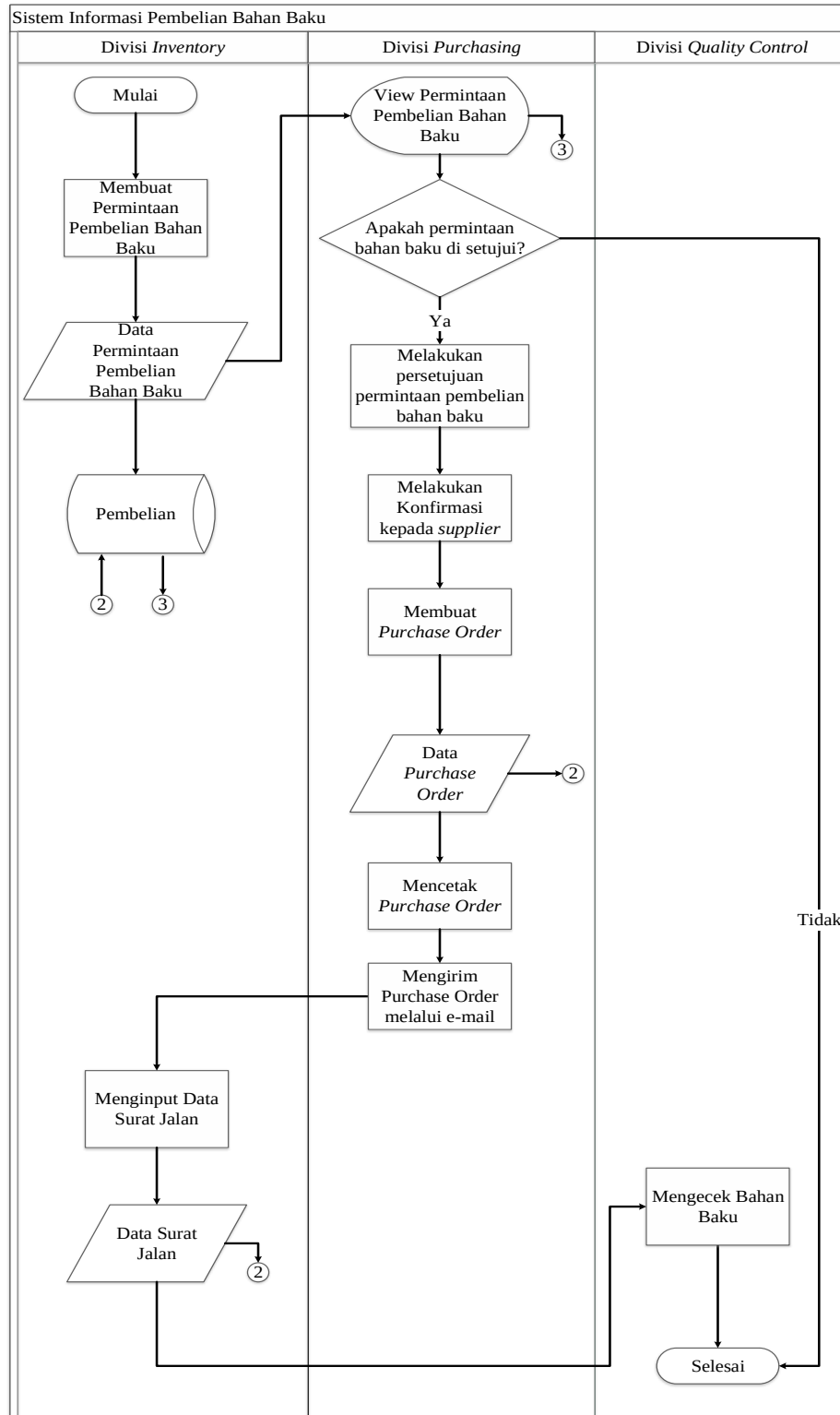
(Sumber: Hasil Analisis, 2018)

5.2 Prosedur Sistem Informasi Pembelian Bahan Baku

Prosedur sistem informasi pembelian bahan baku yang diusulkan melalui tahapan berikut :

1. Staff divisi *Inventory* membuat permintaan pembelian bahan baku, kemudian data tersimpan ke dalam *database*.
2. Staff divisi *Purchasing* menerima permintaan pembelian bahan baku di sistem informasi pembelian bahan baku.
3. Apabila permintaan pembelian bahan baku disetujui oleh staff divisi *Purchasing*, maka staff divisi *Purchasing* akan membuat *purchase order* dan tersimpan ke dalam *database*.
4. Kemudian staff divisi *Purchasing* mencetak *purchase order*.
5. Staff divisi *Purchasing* mengirim *purchase order* melalui email.
6. Staff divisi *Inventory* menerima surat jalan dari *supplier* kemudian data surat jalan diinput kemudian tersimpan ke dalam *database*.
7. Staff divisi *Quality Control* melakukan pengecekan bahan baku yang telah dikirim dari *supplier* kemudian tersimpan ke dalam *database*.

Berikut merupakan *Flowmap* sistem informasi pembelian bahan baku usulan yang dapat dilihat pada Gambar V.1 :



Gambar V.1 *Flowmap* Sistem Informasi Pembelian Bahan Baku Usulan

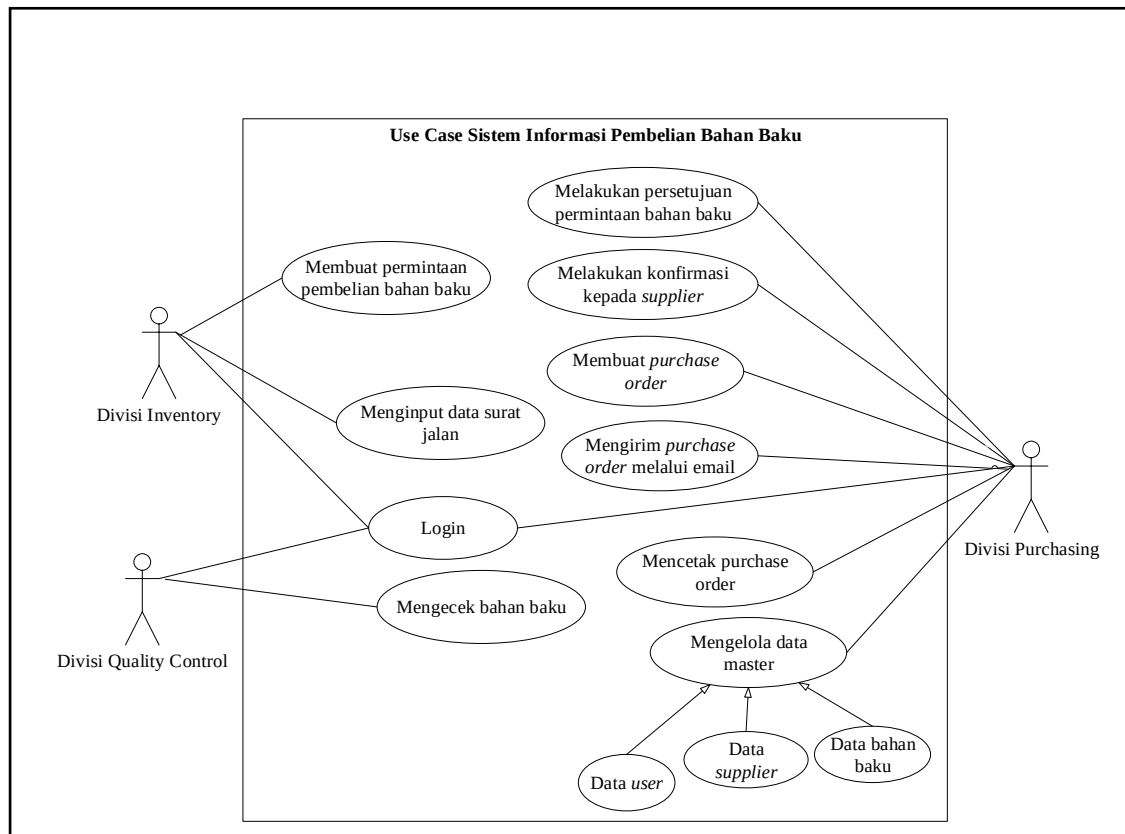
Sumber: Hasil Analisis Data (2018)

5.3 Analisis Sistem Usulan

Analisis proses sistem informasi pembelian bahan baku di PT Baneex Indonesia menggunakan pemodelan sistem *Unified Modelling Language* (UML), berikut akan dimodelkan analisis menggunakan beberapa model yaitu *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *deployment diagram*. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi pembelian bahan baku yang akan dibangun.

5.3.1 Use case diagram

Use Case Diagram sistem informasi pembelian bahan baku yang diusulkan pada PT Baneex Indonesia digambarkan pada Gambar V.2 dibawah ini :



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

(Sumber: Hasil Analisis, 2018)

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* berdasarkan usulan sistem informasi pembelian bahan baku dapat dilihat pada tabel berikut :

Tabel V.2 Definisi Aktor Sistem Informasi Pembelian Bahan Baku

No	Aktor	Definisi
1	Divisi <i>Inventory</i>	Divisi <i>Inventory</i> bertugas menginput data permintaan pembelian bahan baku dan mengelola data surat jalan.
2	Divisi <i>Purchasing</i>	Divisi <i>Purchasing</i> bertugas melakukan persetujuan permintaan pembelian bahan baku, menghubungi rekanan, melakukan konfirmasi bahan baku, menginput data <i>purchase order</i> , dan mencetak <i>purchase order</i> .
3	Divisi <i>Quality Control</i>	Divisi <i>Quality Control</i> bertugas mengecek bahan baku.

(Sumber: Hasil Analisis, 2018)

2. Definisi Use Case

Pendefinisian *use case* pada *use case diagram* pembelian bahan baku yang sedang berjalan pada Divisi *Purchasing* di PT Baneex Indonesia dapat dilihat pada tabel V.3 berikut:

Tabel V.3 Definisi Use Case Pembelian Bahan Baku

No	Nama Use case	Definisi
1	<i>Login</i>	Proses melakukan <i>login</i> pada aplikasi.
2	Mengelola data master	Proses generalisasi yang meliputi tiga proses pengelolaan data master yaitu data <i>user</i> , data supplier dan data bahan baku.
3	Mengelola data <i>user</i>	Proses mengelola data <i>user</i> yang terdiri dari tambah, ubah, dan hapus pada data <i>user</i> .

4	Mengelola data <i>supplier</i>	Proses mengelola data <i>supplier</i> yang terdiri dari tambah, ubah, dan hapus pada data <i>supplier</i> .
5	Mengelola data bahan baku	Proses mengelola data bahan baku yang terdiri dari tambah, ubah, dan hapus pada data bahan baku.

Tabel V.3 Definisi *Use Case* Pembelian Bahan Baku (Lanjutan)

No	Nama <i>Use case</i>	Definisi
6	Membuat permintaan pembelian bahan baku	Proses input data permintaan pembelian bahan baku yang dilakukan <i>staff</i> divisi <i>Inventory</i> yang ditujukan kepada <i>staff</i> divisi <i>Purchasing</i> .
7	Melakukan persetujuan permintaan pembelian bahan baku	Proses yang dilakukan <i>staff</i> divisi <i>purchasing</i> meyetujui permintaan pembelian pembelian bahan baku yang diajukan.
8	Melakukan Konfirmasi kepada <i>supplier</i>	Proses yang dilakukan <i>staff</i> divisi <i>purchasing</i> melakukan konfirmasi barang kepada <i>supplier</i> .
9	Membuat <i>purchase order</i>	Proses pembuatan <i>purchase order</i> untuk membuat <i>purchase order</i> .
10	Mencetak <i>purchase order</i>	Proses untuk mencetak data <i>purchase order</i> pada <i>supplier</i> .
11	Mengirim <i>purchase order</i> melalui email	Proses untuk mengirim <i>purchase order</i> pada <i>supplier</i> .
12	Menginput data surat jalan	Proses input data surat jalan sebagai data penerimaan bahan baku.
13	Mengecek Bahan Baku	Proses pengecekan bahan baku sebagai data penerimaan bahan baku.

Sumber: Hasil Analisis Data (2018)

3. Skenario Use Case

Skenario jalannya masing-masing use case pada sistem informasi pembelian bahan baku usulan dapat dilihat pada poin berikut:

a. *Use Case Login*

Berikut adalah skenario *use case login* yang terdapat dalam sistem informasi pembelian bahan baku:

Tabel V.4 *Use Case Description Login*

Nama <i>Use Case</i>	<i>Login</i>
Aktor	Divisi <i>Inventory</i> , Divisi <i>Purchasing</i> dan Divisi <i>Quality Control</i> .
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan <i>login</i> sesuai dengan hak akses dari user.
<i>Normal Flow Events:</i>	1. <i>User</i> membuka sistem 2. <i>User</i> mengisi <i>username</i> dan <i>password</i> pada <i>login form</i>. 3. Sistem akan memvalidasi data <i>username</i> dan <i>password</i>.. 4. Jika data yang dimasukan benar maka akan muncul halaman utama. 5. Jika data yang dimasukan salah maka akan muncul pesan bahwa data yang dimasukan tidak benar.

Sumber: Hasil Analisis Data (2018)

b. *Use case Mengelola Data master*

Berikut adalah skenario *use case* mengelola data master yang terdapat dalam sistem informasi pembelian bahan baku :

Tabel V.5 *Use Case Mengelola Data Master*

Nama <i>Use Case</i>	Mengelola Data Master
----------------------	-----------------------

Aktor	Divisi <i>Purchasing</i> .
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan <i>login</i> sesuai dengan hak akses dari user.
<i>Normal Flow Events:</i>	1. Divisi <i>Purchasing</i> memilih menu data master. 2. <i>User</i> memilih menu sub data master yang terdiri dari data <i>user</i>, data <i>supplier</i>, dan data bahan baku. 3. <i>User</i> dapat menambah data, mengubah data, menghapus data, dan menyimpan data ke dalam <i>database</i>.. 4. Sistem akan memperbaharui data pada <i>database</i>.

Sumber: Hasil Analisis Data (2018)

c. *Use Case* Membuat Permintaan Pembelian Bahan Baku

Berikut adalah skenario *use case* menginput data permintaan pembelian bahan baku :

Nama <i>Use Case</i>	Membuat Permintaan Pembelian Bahan Baku
Aktor	Divisi <i>Inventory</i>
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses membuat permintaan pembelian bahan baku
<i>Normal Flow Events:</i>	1. Divisi <i>Inventory</i> memilih menu permintaan. 2. <i>User</i> memilih sub menu permintaan bahan baku. 3. Sistem menampilkan <i>form</i> permintaan bahan baku. 4. <i>User</i> menginput data permintaan bahan baku pada <i>form</i> permintaan bahan baku. 5. Klik tombol tambah data kemudian sistem akan menyimpan data permintaan ke dalam <i>database</i>. 6. Kemudian klik tombol selesai, lalu menginput tgl permintaan.

	7. Klik simpan kemudian data tersimpan ke dalam <i>database</i> .
--	---

Tabel V.6 *Use Case* Membuat Permintaan Pembelian Bahan Baku

Sumber: Hasil Analisis Data (2018)

d. *Use Case* Melakukan Persetujuan Permintaan Pembelian Bahan Baku

Berikut adalah skenario *use case* melakukan persetujuan permintaan pembelian bahan baku :

Tabel V.7 *Use Case* Melakukan Persetujuan Permintaan Pembelian

Nama <i>Use Case</i>	Melakukan Persetujuan Permintaan Pembelian Bahan Baku
Aktor	Divisi <i>Purchasing</i> .
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses menginput data permintaan pembelian bahan baku
<i>Normal Flow</i> <i>Events:</i>	1. Divisi <i>Purchasing</i> mendapatkan notifikasi permintaan pembelian yang telah diajukan. 2. Divisi <i>Purchasing</i> memilih menu pembelian. 3. Divisi <i>Purchasing</i> memilih sub menu validasi. 4. Jika klik validasi maka akan akan menampilkan data permintaan yang di setujui, jika klik tolak maka akan akan menampilkan data tolak.

Sumber: Hasil Analisis Data (2018)

e. *Use Case* Melakukan Konfirmasi kepada *supplier*

Berikut adalah skenario *use case* melakukan konfirmasi barang kepada *supplier* yang terdapat dalam sistem informasi pembelian bahan baku :

Tabel V.8 *Use Case* Melakukan Konfirmasi kepada *supplier*

Nama <i>Use Case</i>	Melakukan Konfirmasi kepada <i>supplier</i>
Aktor	Divisi <i>Purchasing</i> .
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses divisi <i>purchasing</i> melakukan konfirmasi dengan <i>supplier</i> .

<i>Normal Flow Events:</i>	1. Divisi <i>Purchasing</i> memilih menu pembelian, lalu memilih sub menu validasi permintaan. 2. Sistem menampilkan halaman data validasi permintaan. 3. Setelah melakukan persetujuan, maka akan muncul data permintaan yang di setujui. 4. Kemudian klik kirim, maka data bahan baku yang telah di setujui akan terkirim.
----------------------------	---

Sumber: Hasil Analisis Data (2018)

f. *Use Case Membuat Purchase Order*

Berikut adalah skenario *use case* membuat *purchase order* yang terdapat dalam sistem informasi pembelian bahan baku :

Tabel V.9 *Use Case Membuat Purchase Order*

Nama <i>Use Case</i>	Membuat <i>Purchase Order</i>
Aktor	Divisi <i>Purchasing</i> .
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses membuat <i>Purchase Order</i> .
<i>Normal Flow Events:</i>	5. Divisi <i>Purchasing</i> memilih menu pembelian, lalu memilih sub menu <i>purchase order</i>. 6. Sistem menampilkan <i>form purchase order</i>. 7. Divisi <i>Purchasing</i> membuat <i>purchase order</i> kemudian klik tambah, maka data <i>purchase order</i> akan tersimpan ke <i>database</i>.

Sumber: Hasil Analisis Data (2018)

g. *Use Case Mencetak Purchase Order*

Berikut adalah skenario *use case* mencetak *purchase order* yang terdapat dalam sistem informasi pembelian bahan baku :

Tabel V.10 *Use Case Mencetak Purchase Order*

Nama Use Case	Mencetak <i>Purchase Order</i>
Aktor	Divisi <i>Purchasing</i> .
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses mencetak <i>purchase order</i> .
<i>Normal Flow Events:</i>	1. Divisi <i>Purchasing</i> memilih menu pembelian. 2. Divisi <i>Purchasing</i> memilih sub menu data <i>purchase order</i>. 3. Divisi <i>Purchasing</i> klik detail kemudian detail <i>purchase order</i> akan muncul. 4. Kemudian Divisi <i>Purchasing</i> mencetak <i>purchase order</i>

Sumber: Hasil Analisis Data (2018)

h. Use Case Mengirim *Purchase Order*

Berikut adalah skenario *use case* mengirim *purchase order* melalui email yang terdapat dalam sistem informasi pembelian bahan baku :

Tabel V.11 Use Case Mengirim *Purchase Order*

Nama Use Case	Mengirim <i>Purchase Order</i>
Aktor	Divisi <i>Purchasing</i> .
Deskripsi Use Case	<i>Use case</i> ini menggambarkan proses mengirim <i>purchase order</i> melalui email.
<i>Normal Flow Events:</i>	1. Divisi <i>Purchasing</i> memilih menu pembelian , kemudian memilih sub menu kirim <i>purchase order</i>. 2. Sistem menampilkan halaman kirim <i>purchase order</i>. 3. Divisi <i>Purchasing</i> mengisi email tujuan, <i>subject</i> dan file yang akan dikirim. 4. Sistem menampilkan halaman kirim <i>purchase order</i>.

Sumber: Hasil Analisis Data (2018)

a. Use Case Menginput Data Surat Jalan

Berikut adalah skenario *use case* menginput data surat jalan yang terdapat dalam sistem informasi pembelian bahan baku :

Tabel V.12 *Use Case* Menginput Data Surat Jalan

Nama <i>Use Case</i>	Mengelola Data Surat Jalan
Aktor	Divisi <i>Inventory</i>
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses menginput surat jalan.
<i>Normal Flow Events:</i>	1. Divisi <i>Inventory</i> memilih menu permintaan , kemudian memilih sub menu surat jalan. 2. Sistem menampilkan form surat jalan. 3. Divisi <i>Inventory</i> menginput data surat jalan kemudian tersimpan ke dalam <i>database</i>.

Sumber: Hasil Analisis Data (2018)

b. *Use Case* Mengecek Bahan Baku

Berikut adalah skenario *use case* mengelola data pengecekan bahan baku yang terdapat dalam sistem informasi pembelian bahan baku :

Tabel V.13 *Use Case* Mengecek Bahan Baku

Nama <i>Use Case</i>	Mengelola Data Pengecekan Bahan Baku
Aktor	Divisi <i>Quality Control</i> .
Deskripsi <i>Use Case</i>	<i>Use case</i> ini menggambarkan proses mengelola data hasil pengecekan bahan baku yang dikirimkan <i>supplier</i> .
<i>Normal Flow Events:</i>	1. Divisi <i>Quality Control</i> memilih menu pengecekan. 2. Divisi <i>Quality Control</i> memilih sub menu cek bahan baku. 3. Sistem menampilkan form cek bahan baku. 4. Jika data bahan baku sesuai dengan surat jalan maka klik sesuai, jika bahan baku tidak sesuai dengan surat jalan maka klik tidak sesuai.

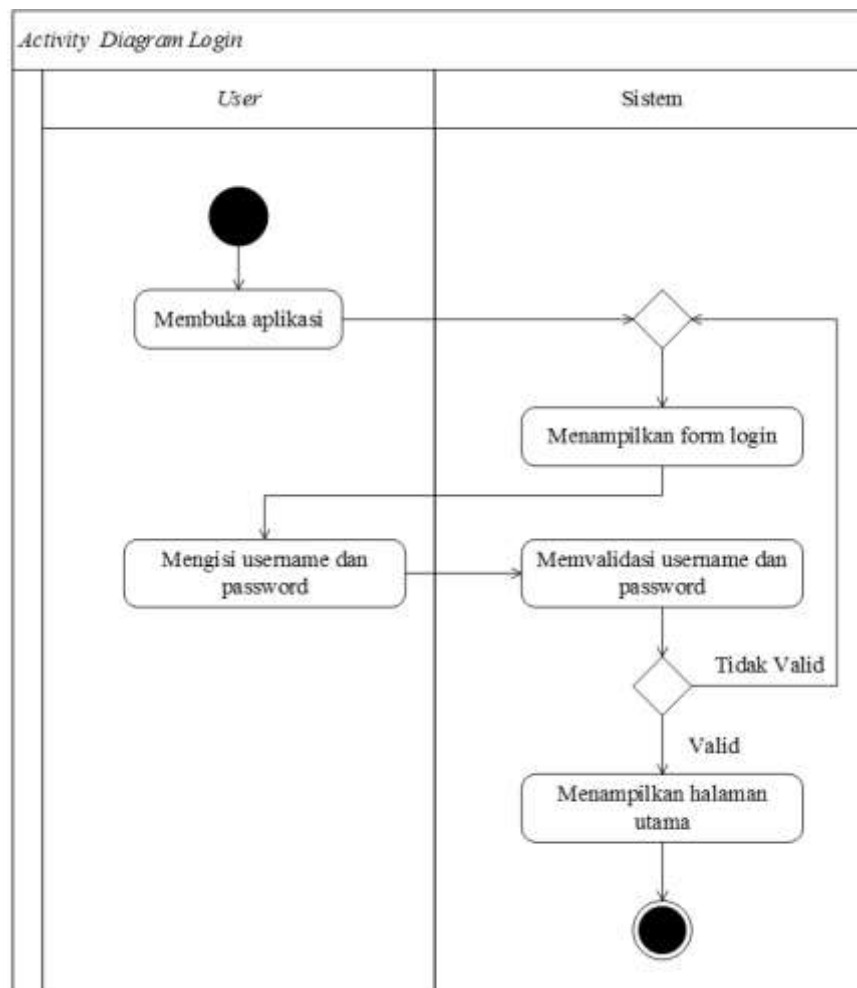
Sumber: Hasil Analisis Data (2018)

5.3.2 Activity Diagram

Activity Diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada di dalam sistem. Agar lebih memahami sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem informasi pembelian bahan baku.

1. Activity Diagram Login

Activity diagram login ini menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu Divisi *Purchasing*, *Divisi Inventory*, dan Divisi *Quality Control* untuk dapat masuk ke dalam Sistem Informasi Pembelian Bahan Baku. *Activity diagram login* dapat dilihat pada Gambar V.3 sebagai berikut:

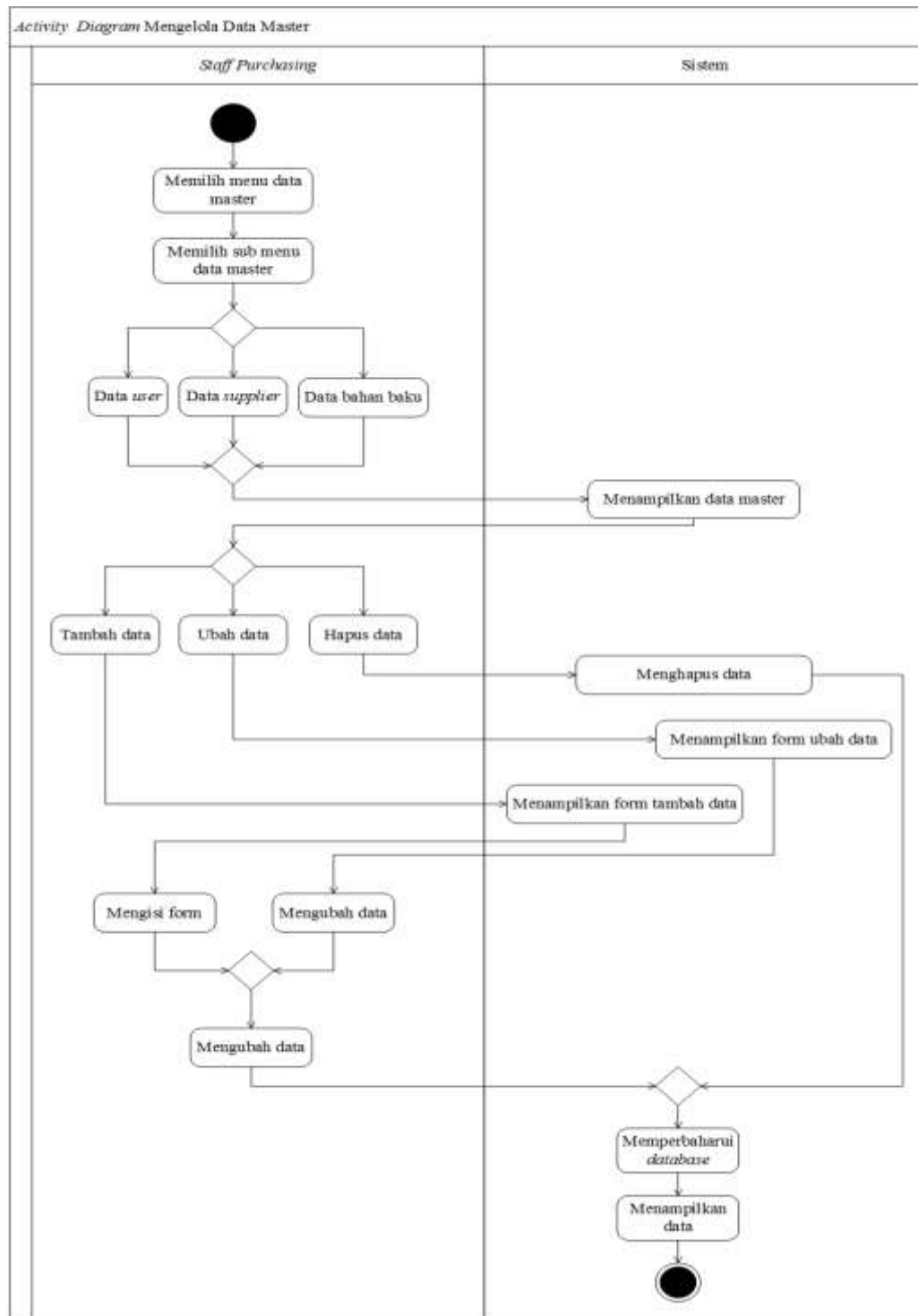


Gambar V.3 Activity Diagram Login

(Sumber: Hasil Analisis Data, 2018)

2. Activity Diagram Mengelola Data Master

Activity diagram mengelola data master ini menjelaskan alur aktivitas yang dilakukan oleh Divisi *Purchasing*. *Activity diagram* mengelola data master dapat dilihat pada Gambar V.4 sebagai berikut:

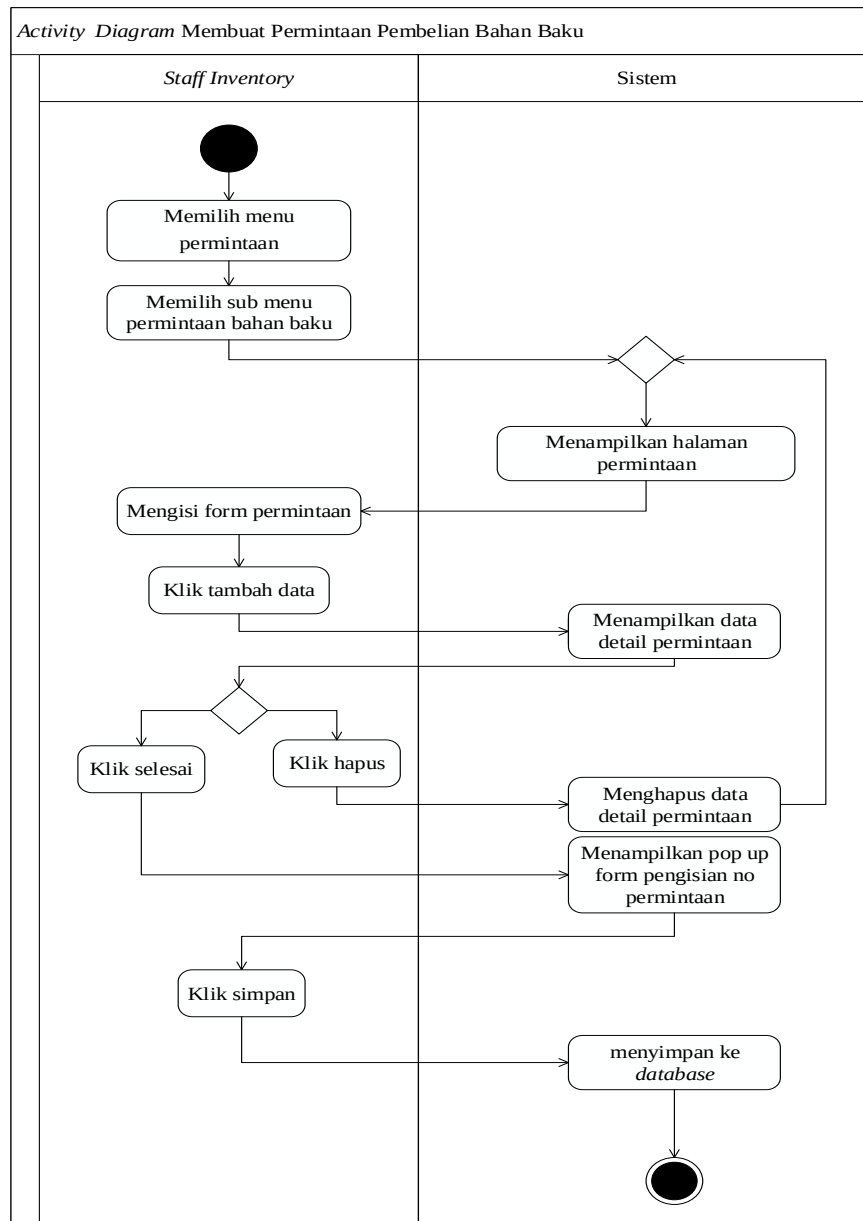


Gambar V.4 Activity Diagram Mengelola Data Master

(Sumber: Hasil Analisis Data, 2018)

3. Activity Diagram Membuat Permintaan Pembelian

Activity diagram membuat permintaan pembelian bahan baku berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada saat membuat permintaan pembelian. Activity diagram membuat permintaan pembelian yang diusulkan dapat dilihat pada Gambar V.5 sebagai berikut:

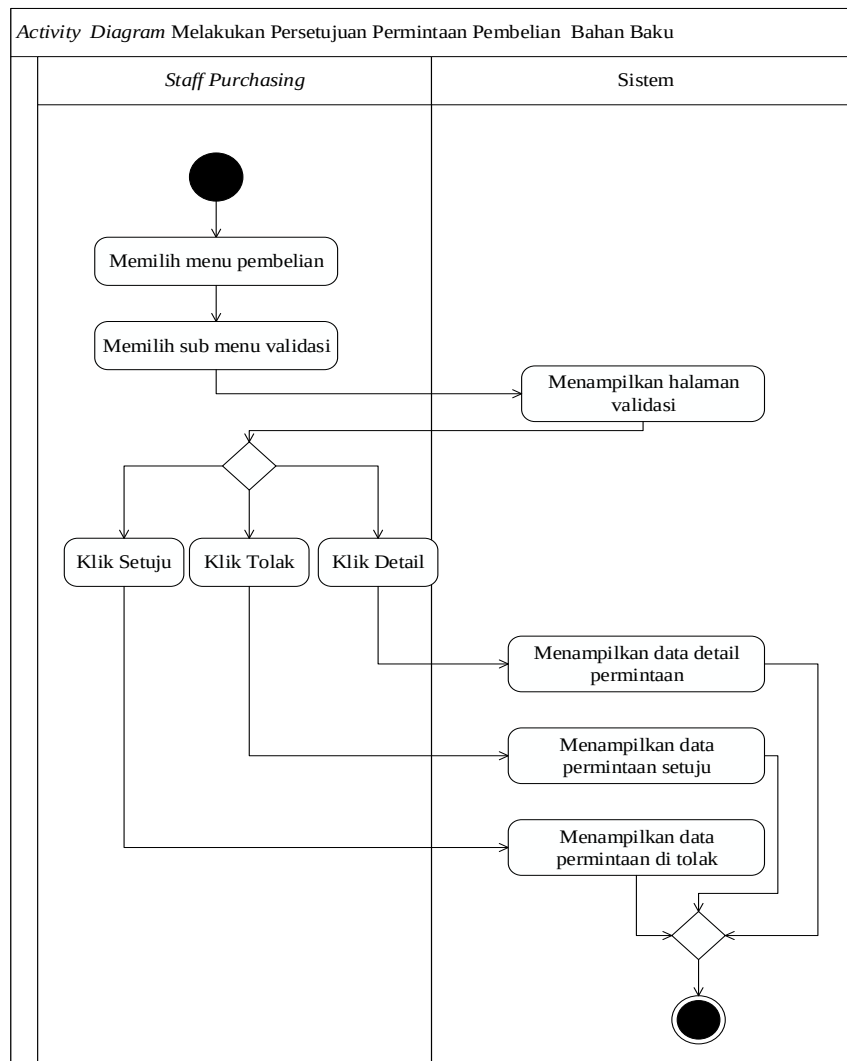


Gambar V.5 Activity Diagram Membuat Permintaan Pembelian

(Sumber: Hasil Analisis Data, 2018)

4. *Activity diagram* Melakukan Persetujuan Permintaan Pembelian

Activity diagram melakukan persetujuan permintaan berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada persetujuan permintaan pembelian. *Activity diagram* melakukan persetujuan permintaan pembelian yang diusulkan dapat dilihat pada Gambar V.6 sebagai berikut:



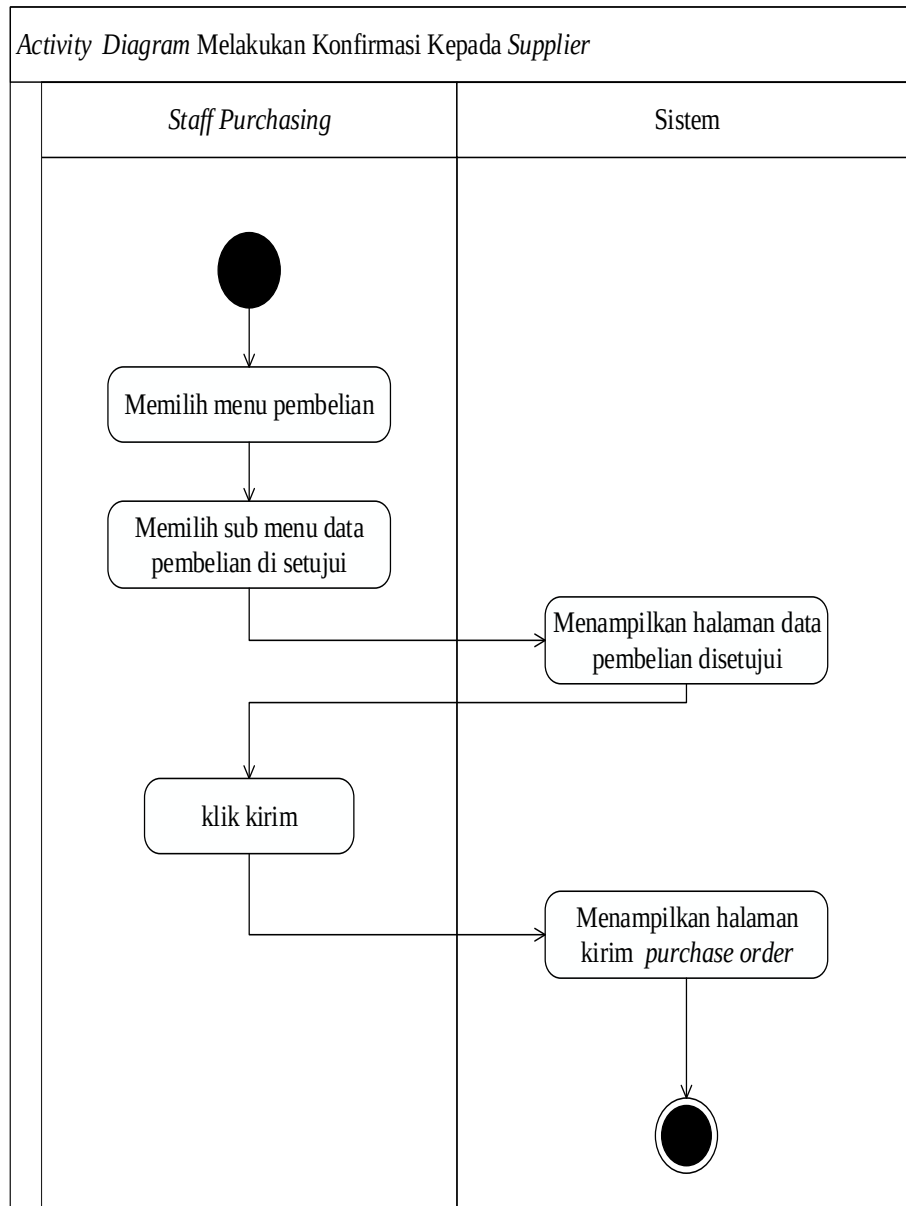
Gambar V.6 *Activity Diagram* Melakukan Persetujuan Permintaan

(Sumber: Hasil Analisis Data, 2018)

5. *Activity Diagram* Melakukan Konfirmasi Kepada *Supplier*

Activity diagram melakukan konfirmasi kepada *supplier* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada saat melakukan konfirmasi kepada *supplier*. *Activity*

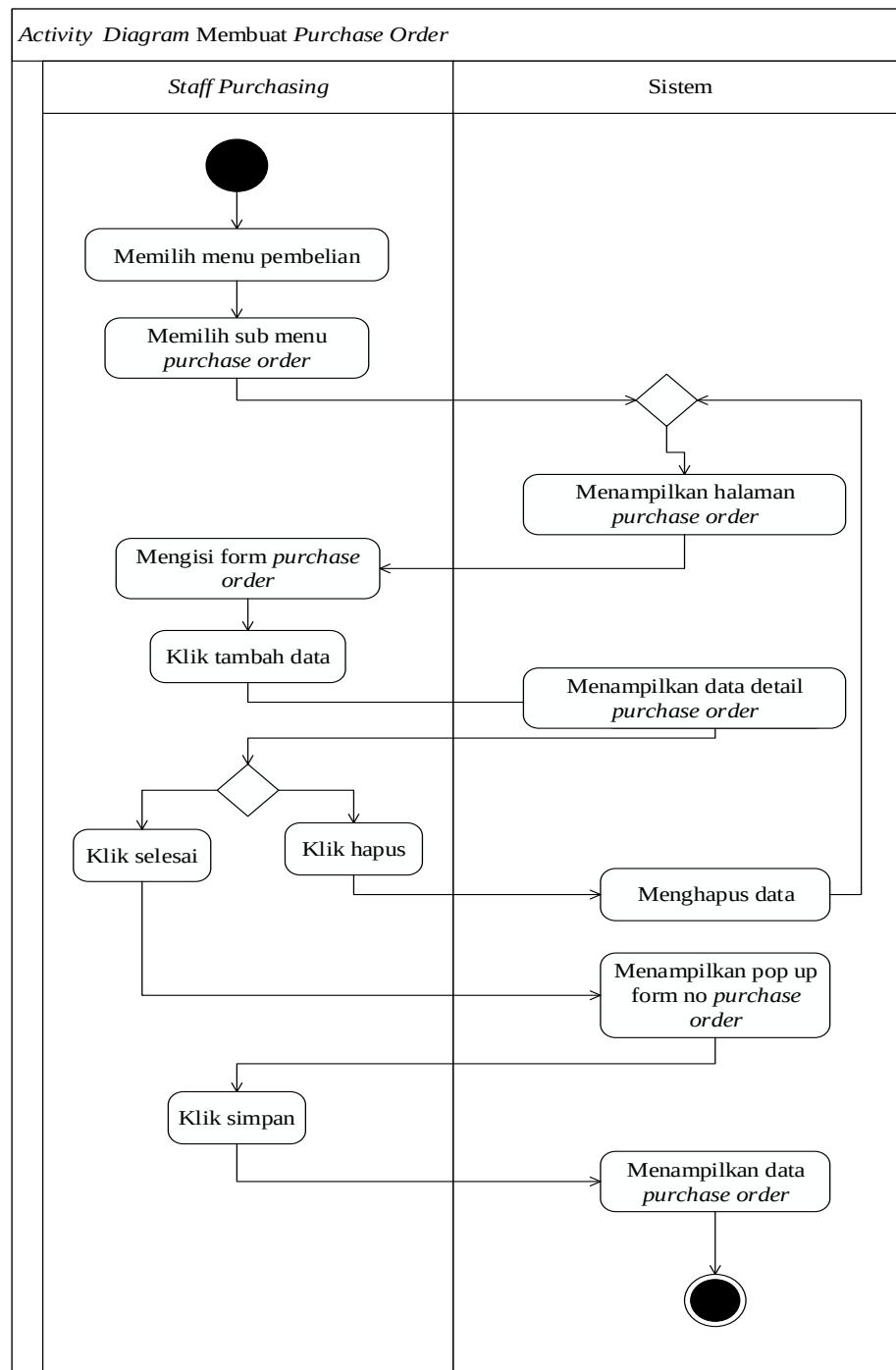
diagram melakukan konfirmasi kepada *supplier* yang diusulkan dapat dilihat pada Gambar V.7 sebagai berikut:



Gambar V.7 Activity Diagram Melakukan konfirmasi
(Sumber: Hasil Analisis Data, 2018)

6. Activity Diagram Membuat Purchase Order

Activity diagram membuat *purchase order* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada saat menginput *purchase order*. Activity diagram menginput data *purchase order* yang diusulkan dapat dilihat pada Gambar V.8 sebagai berikut:

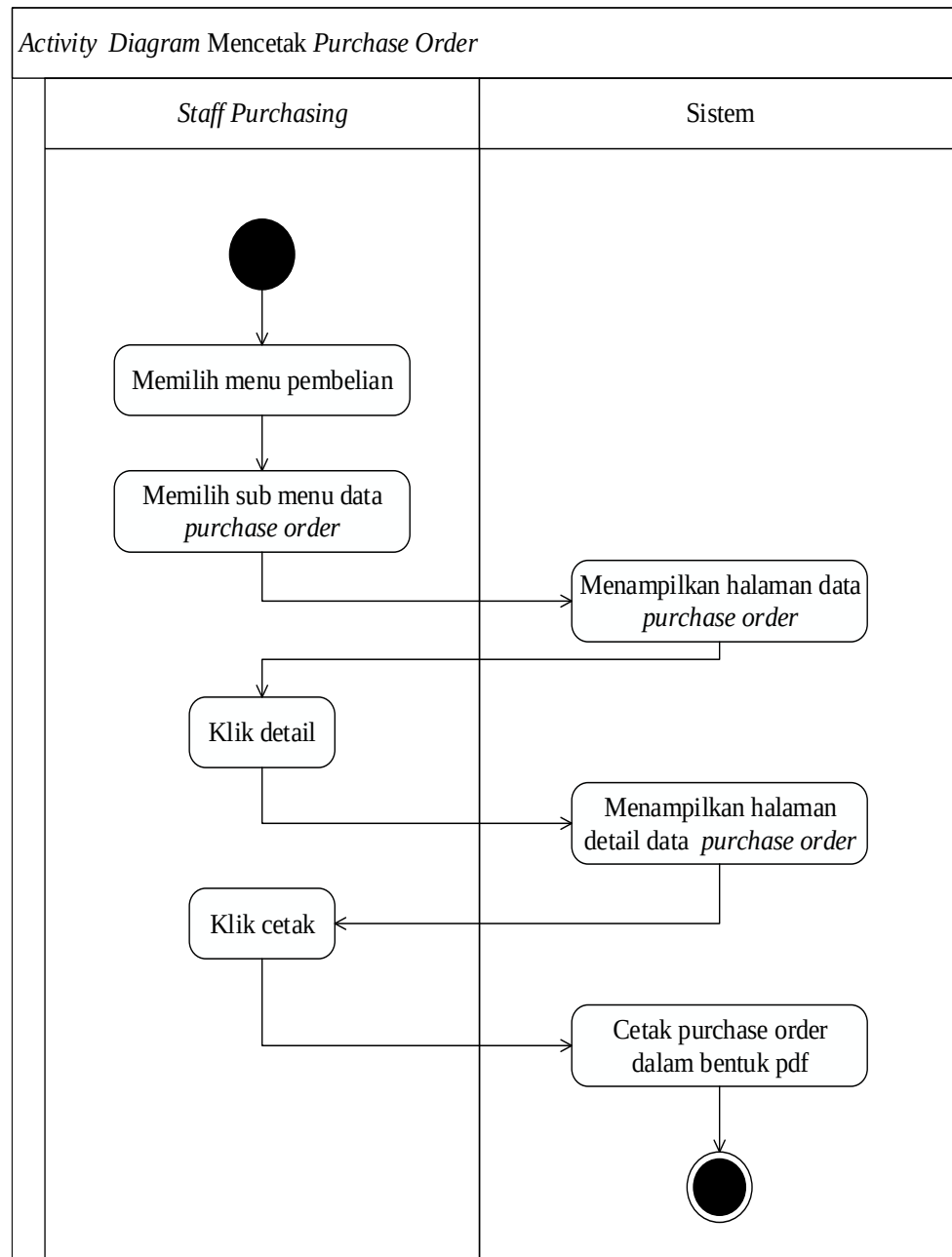


Gambar V.8 Activity Diagram Membuat Purchase Order

(Sumber: Hasil Analisis Data, 2018)

7. Activity Diagram Mencetak Purchase Order

Activity diagram mencetak *purchase order* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada mencetak *purchase order*. *Activity diagram* mencetak *purchase order* yang diusulkan dapat dilihat pada Gambar V.9 sebagai berikut:

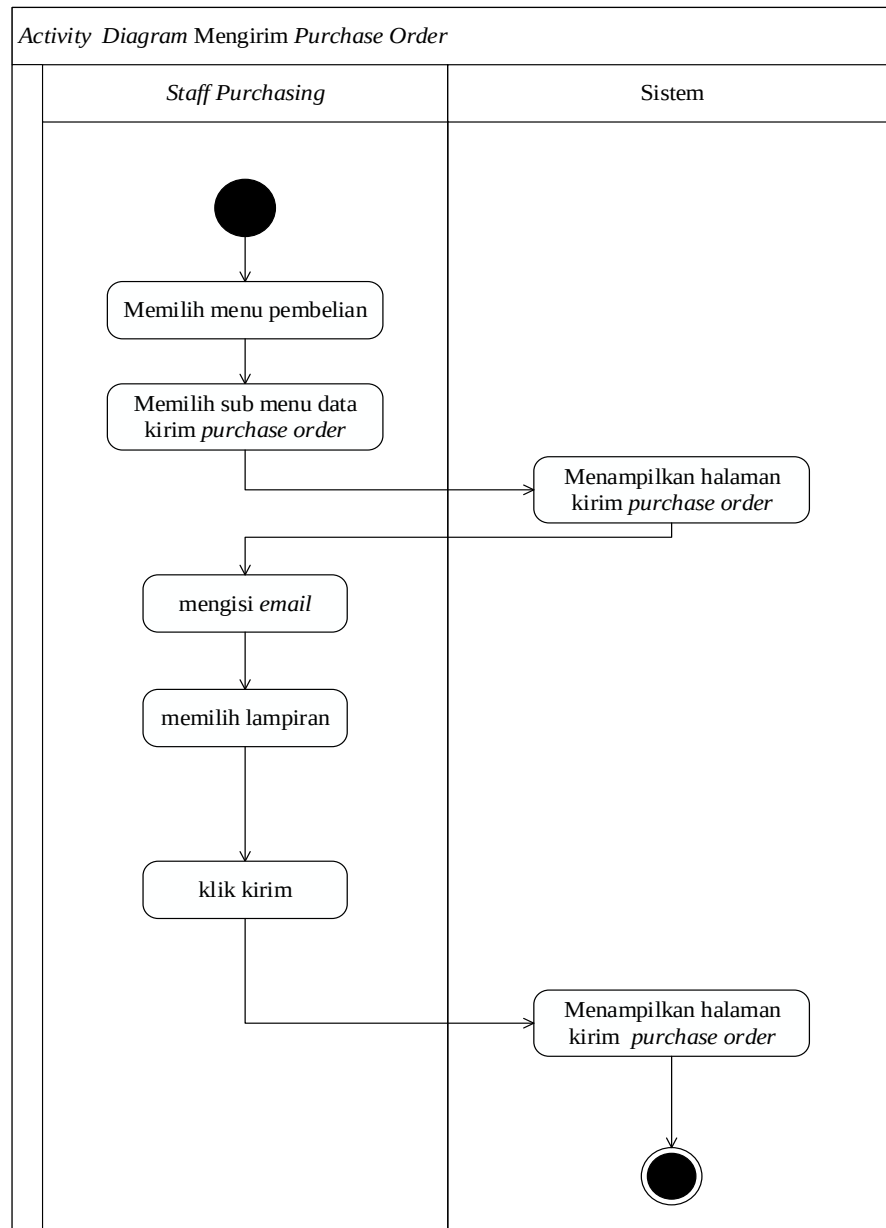


Gambar V.9 *Activity Diagram* Mencetak *Purchase Order*

(Sumber: Hasil Analisis Data, 2018)

8. *Activity Diagram* Mengirim *Purchase Order*

Activity diagram mengirim *purchase order* berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada mengirim *purchase order*. *Activity diagram* mengirim *purchase order* yang diusulkan dapat dilihat pada Gambar V.10 sebagai berikut:

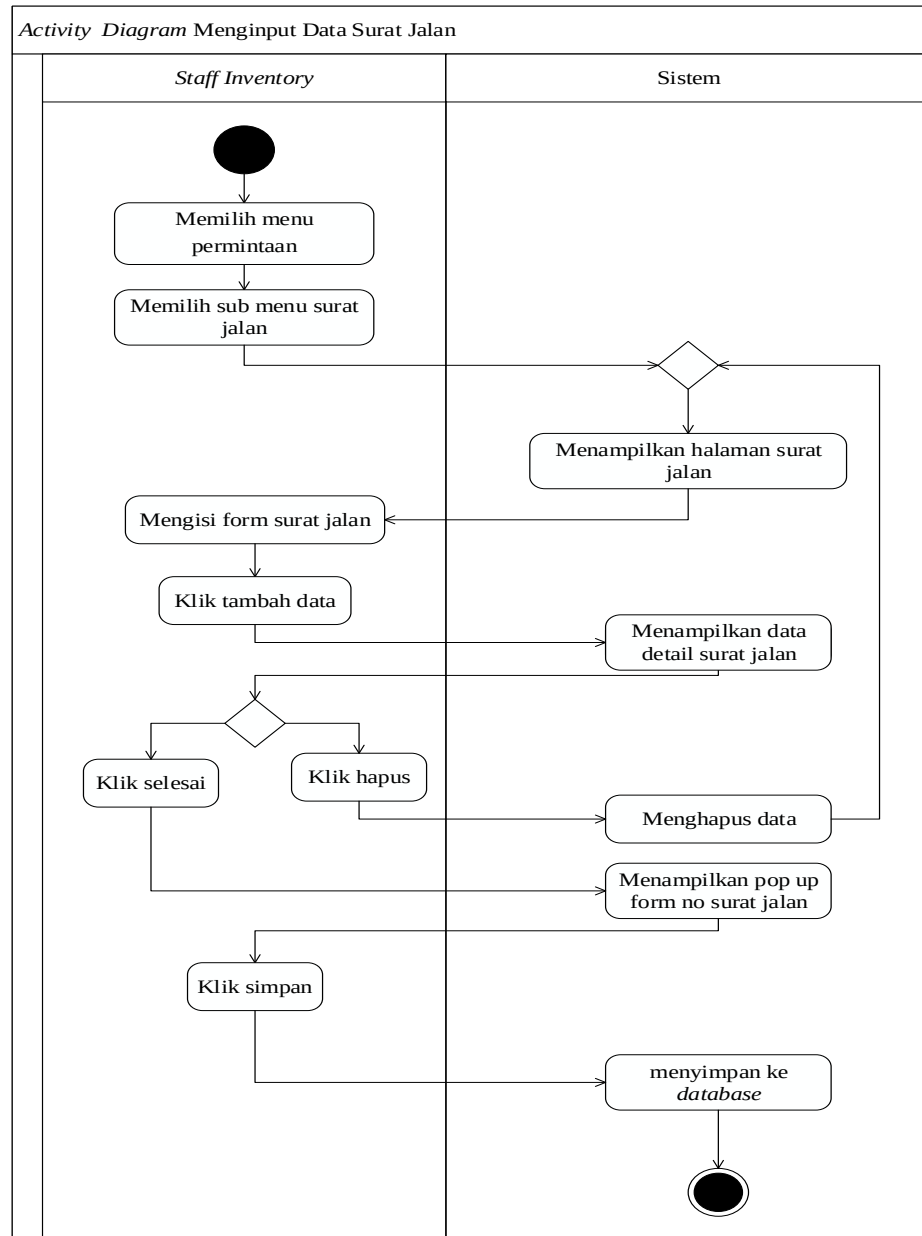


Gambar V.10 Activity Diagram Mengirim Purchase Order

(Sumber: Hasil Analisis Data, 2018)

9. Activity Diagram Menginput Data Surat Jalan

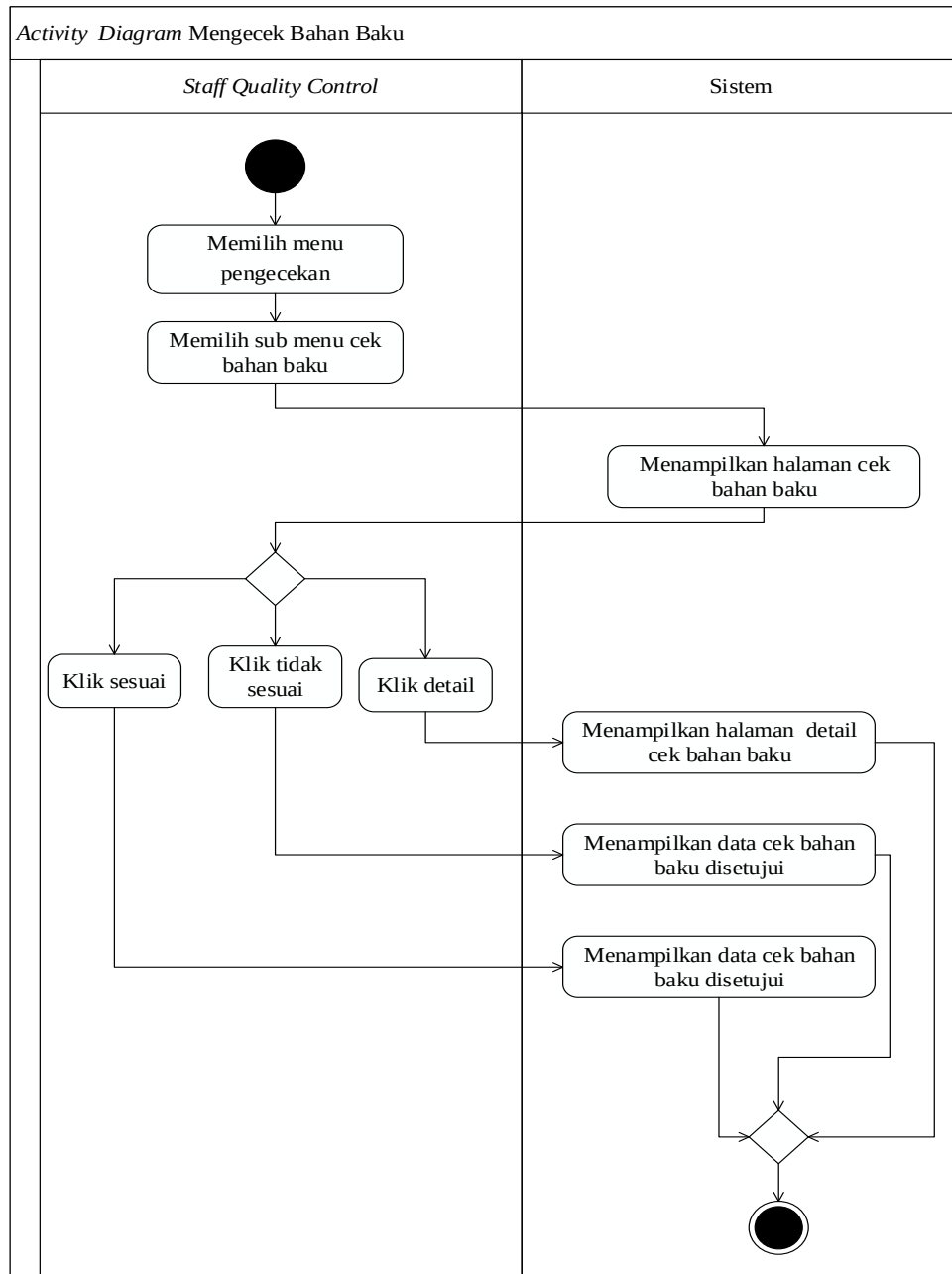
Activity diagram menginput data surat jalan berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada menginput data surat jalan. *Activity diagram* menginput data surat jalan yang diusulkan dapat dilihat pada Gambar V.11 sebagai berikut:



Gambar V.11 *Activity Diagram* Mengelola Data Surat Jalan
(Sumber: Hasil Analisis Data, 2018)

10. *Activity Diagram* Mengecek Bahan Baku

Activity diagram mengecek bahan baku berikut ini menunjukkan fungsi-fungsi yang dapat dilakukan pada mengecek bahan baku. *Activity diagram* mengecek bahan baku yang diusulkan dapat dilihat pada Gambar V.12 sebagai berikut:



Gambar V.12 *Activity Diagram* Mengecek Bahan Baku

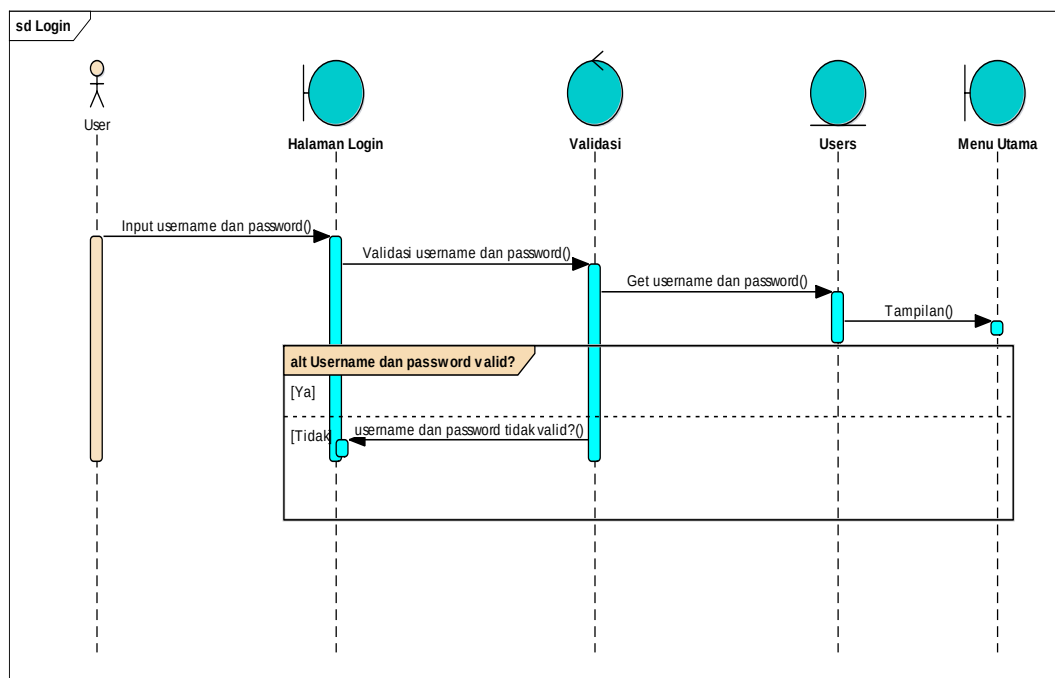
(Sumber: Hasil Analisis Data, 2018)

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 pembelian bahan baku usulan:

1. Sequence Diagram Login

Sequence diagram login menjelaskan sebuah *sequence diagram* dalam proses *login*. Proses ini dilakukan oleh *user* yaitu divisi *purchasing*, divisi *inventory* dan divisi *quality control* sebelum masuk sistem. Adapun *sequence diagram* dari *use case login* dapat dilihat pada Gambar V.13.



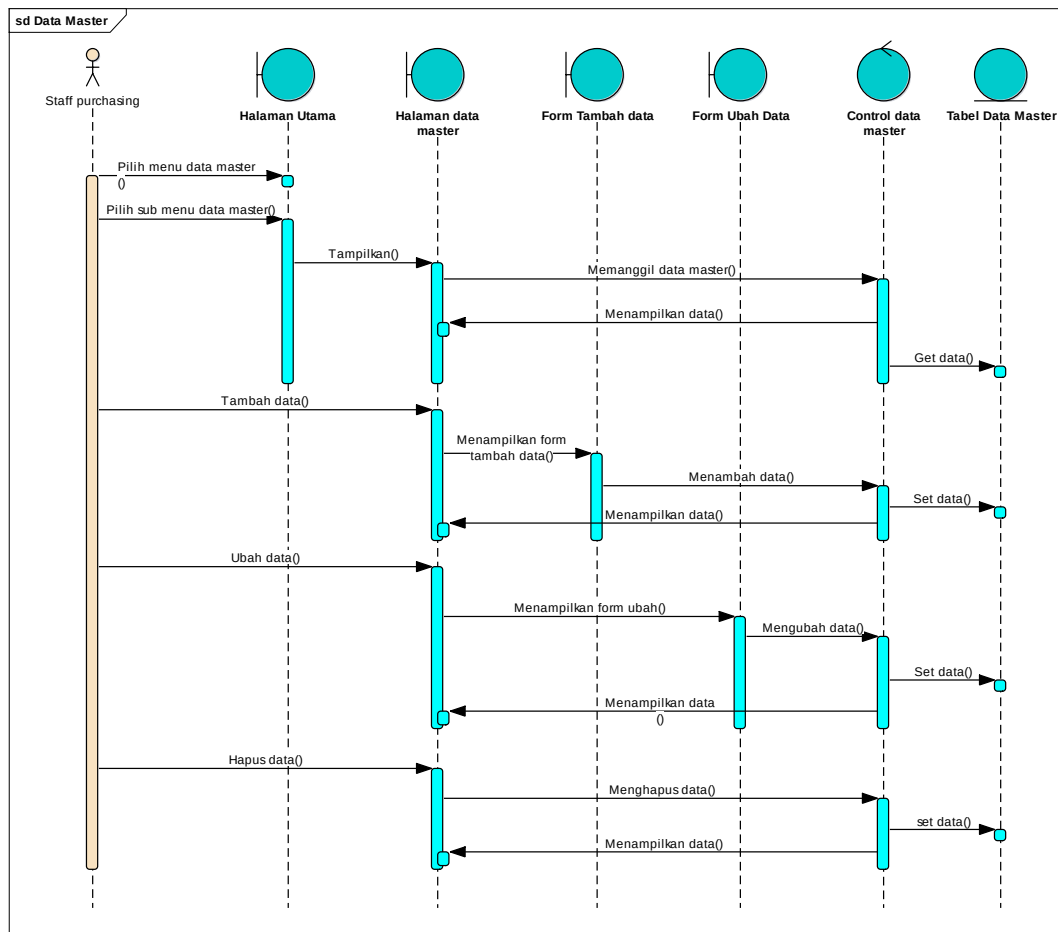
Gambar V.13 Sequence Diagram Login

(Sumber: Hasil Analisis Data, 2018)

2. Sequence Diagram Mengelola Data Master

Sequence diagram mengelola data master ini menjelaskan interaksi yang dilakukan Divisi *Purchasing* dengan sistem agar data yang ada dalam *database* selalu diperbarui. Terdapat tiga data master dalam sistem informasi pembelian bahan baku yaitu data master *user*, data master

supplier dan data master bahan baku. *Sequence diagram* proses mengelola data master dapat dilihat pada Gambar V.14.

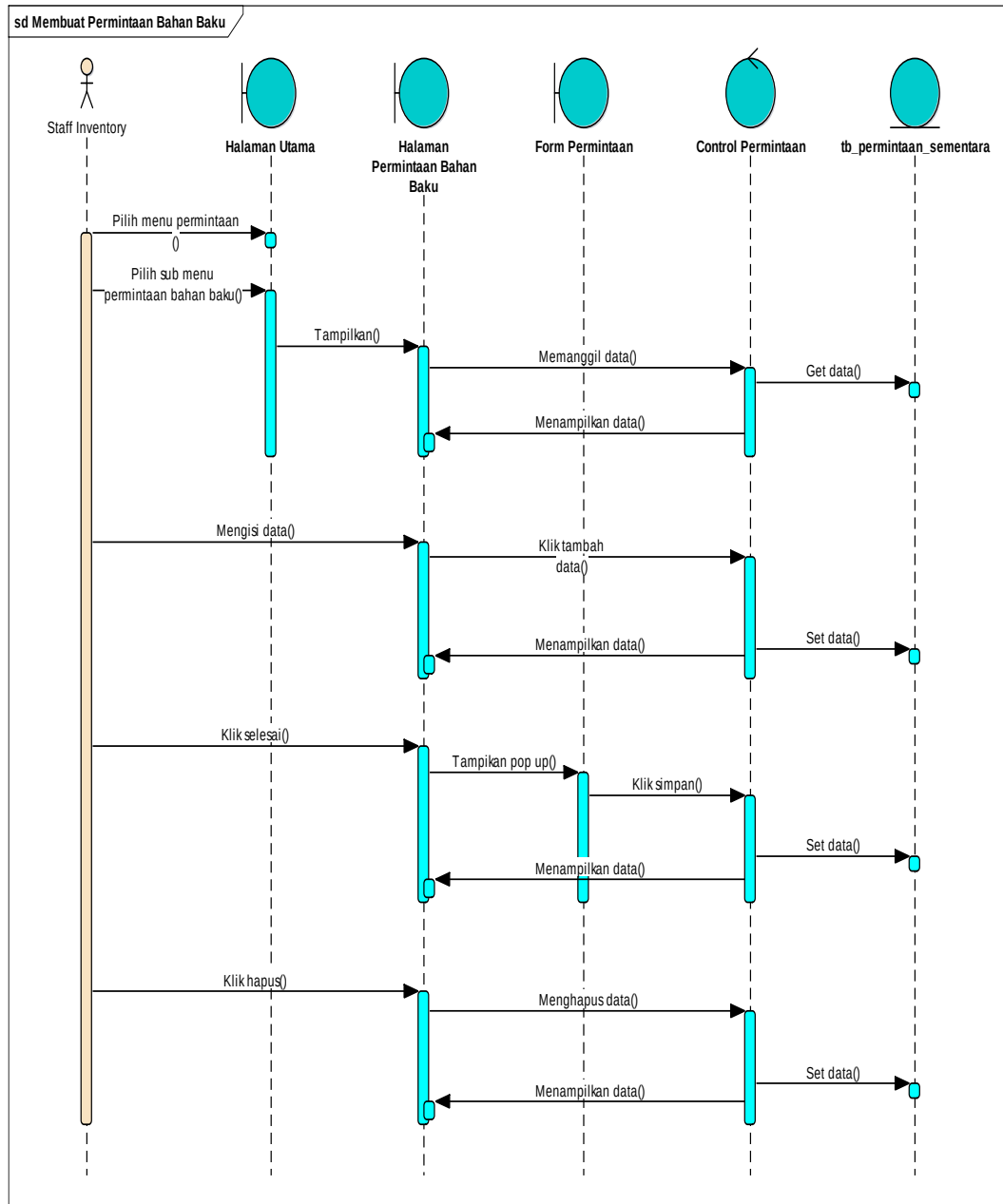


Gambar V.14 *Sequence Diagram* Mengelola Data Master

(Sumber: Hasil Analisis Data, 2018)

3. *Sequence Diagram* Membuat Permintaan Pembelian Bahan Baku

Sequence diagram membuat permintaan pembelian bahan baku menjelaskan sebuah *sequence diagram* dalam proses membuat permintaan pembelian bahan baku yang dimasukkan oleh Divisi *Inventory*. Adapun *sequence diagram* dari *use case* membuat permintaan pembelian bahan baku dapat dilihat pada Gambar V.15.

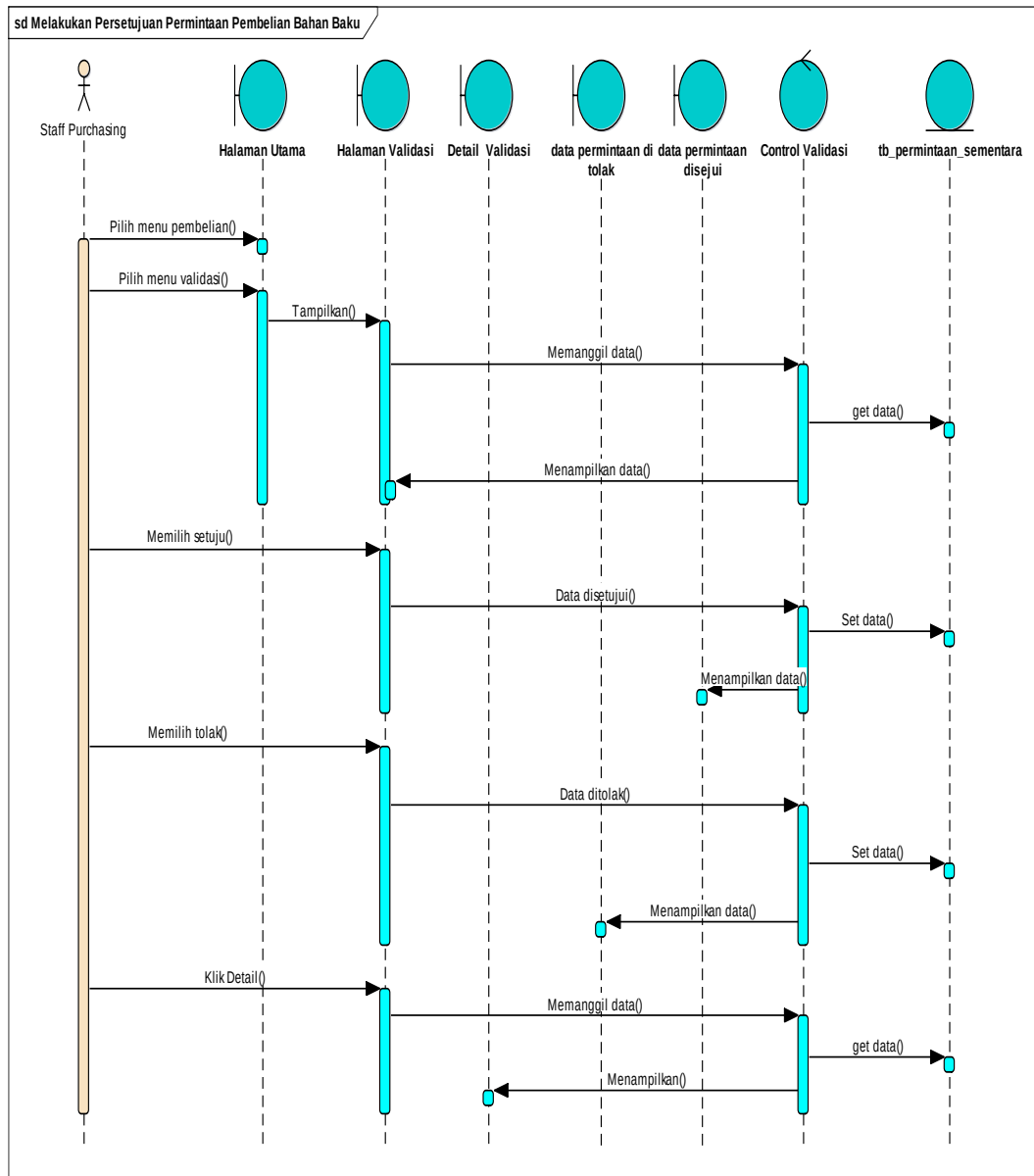


Gambar V.15 *Sequence Diagram* Membuat Permintaan Pembelian

(Sumber: Hasil Analisis Data, 2018)

4. *Sequence Diagram* Melakukan Persetujuan Permintaan Pembelian

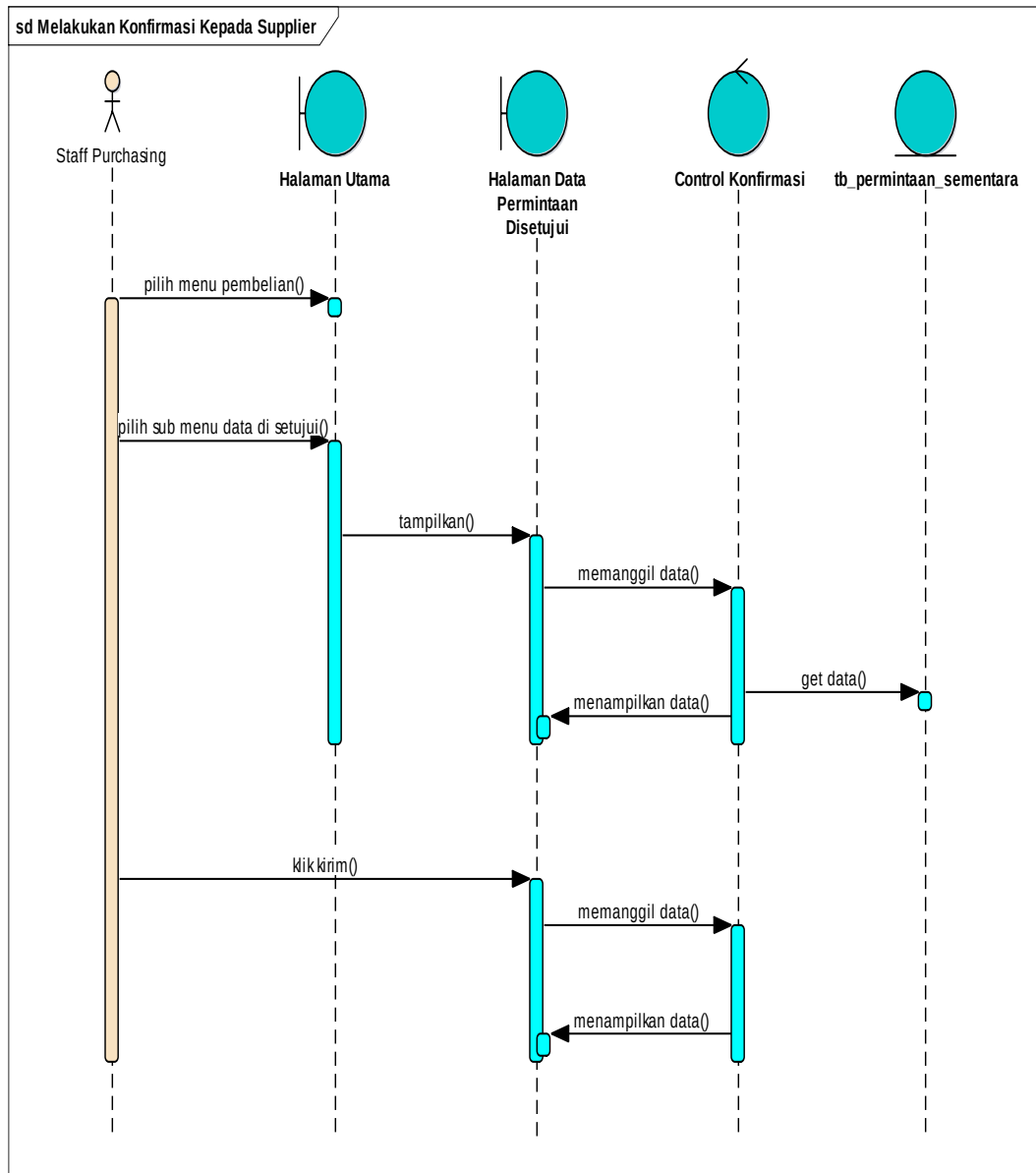
Sequence diagram melakukan persetujuan permintaan pembelian menjelaskan sebuah *sequence diagram* dalam proses persetujuan permintaan pembelian yang dilakukan oleh Divisi *Purchasing*. Adapun *sequence diagram* dari *use case* melakukan persetujuan permintaan pembelian dapat dilihat pada Gambar V.16



Gambar V.16 *Sequence Diagram* Melakukan Persetujuan Permintaan (Sumber: Hasil Analisis Data, 2018)

5. *Sequence Diagram* Melakukan Konfirmasi Kepada Supplier

Sequence diagram melakukan konfirmasi menjelaskan sebuah *sequence diagram* dalam proses melakukan konfirmasi yang dilakukan oleh Divisi *Purchasing*. Adapun *sequence diagram* dari *use case* melakukan konfirmasi dapat dilihat pada Gambar V.17.

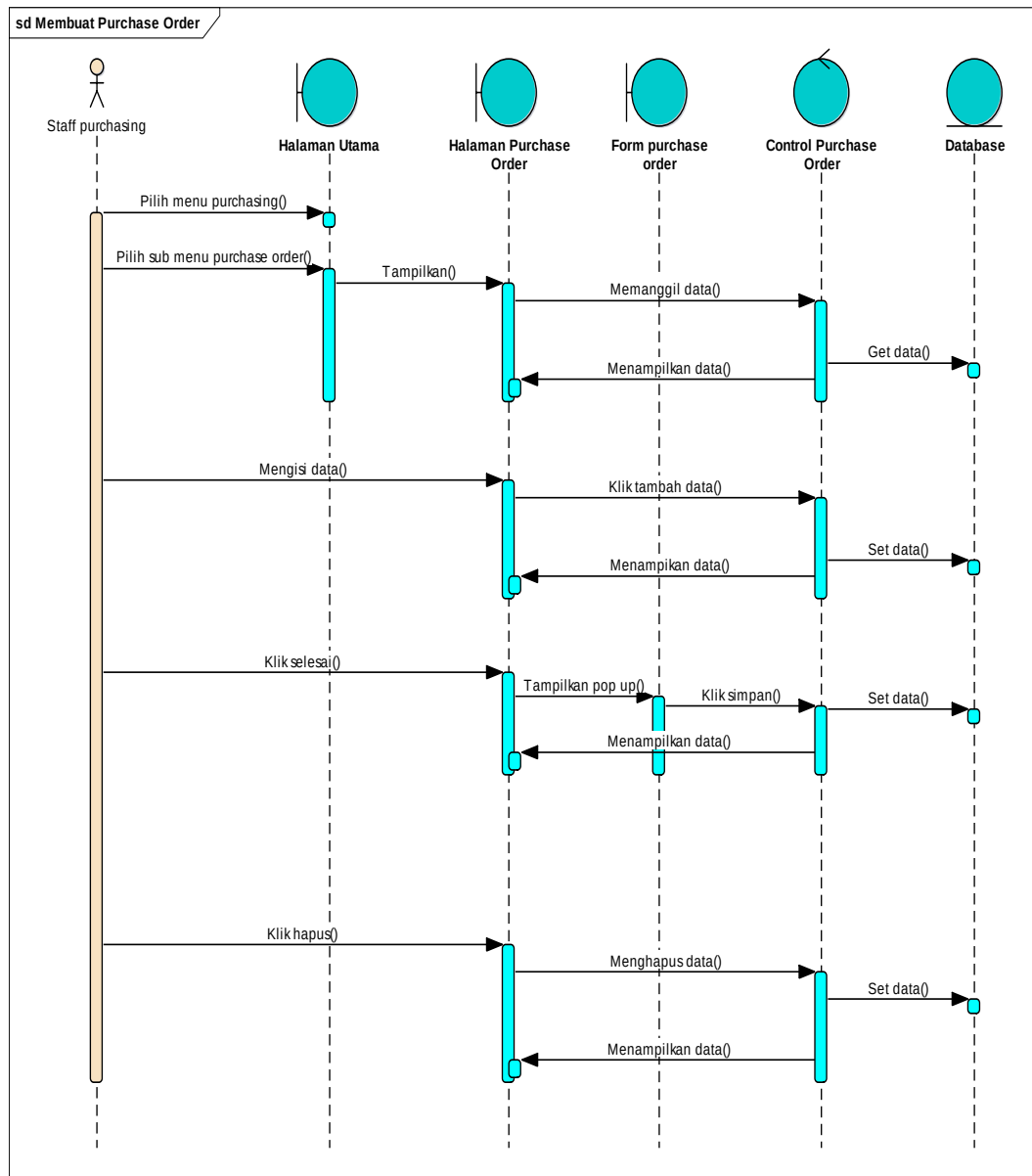


Gambar V.17 Sequence Diagram Melakukan Konfirmasi

(Sumber: Hasil Analisis Data, 2018)

6. Sequence Diagram Membuat Purchase Order

Sequence diagram membuat purchase order menjelaskan sebuah sequence diagram dalam proses membuat purchase order yang dilakukan oleh Divisi Purchasing. Adapun sequence diagram dari use case membuat purchase order dapat dilihat pada Gambar V.18.

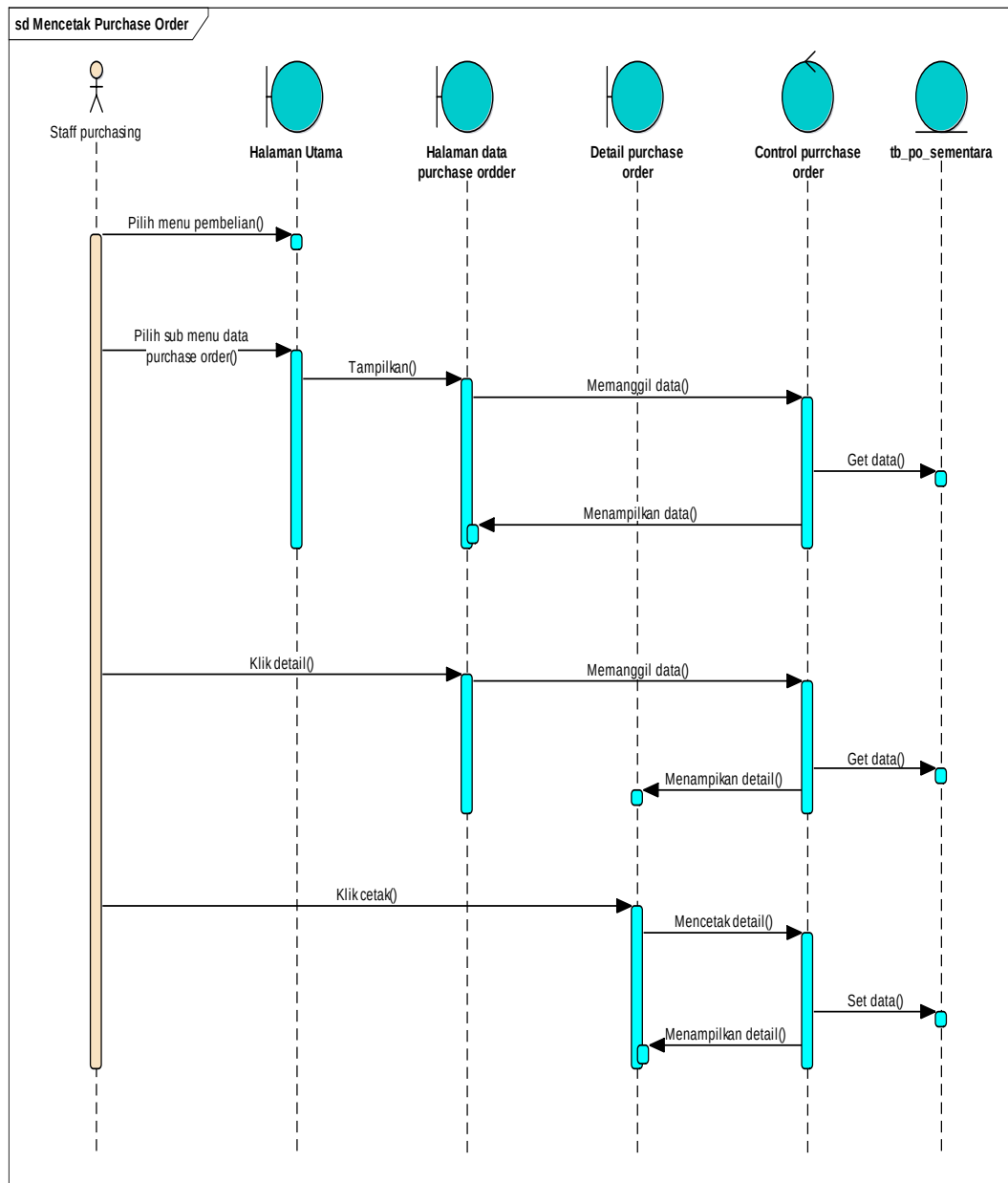


Gambar V.18 Sequence Diagram Membuat Purchase Order

(Sumber: Hasil Analisis Data, 2018)

7. Sequence Diagram Mencetak Purchase Order

Sequence diagram mencetak *purchase order* menjelaskan sebuah *sequence diagram* dalam proses mencetak *purchase order* yang dilakukan oleh Divisi *Purchasing*. Adapun *sequence diagram* dari *use case* mencetak *purchase order* dapat dilihat pada Gambar V.19.



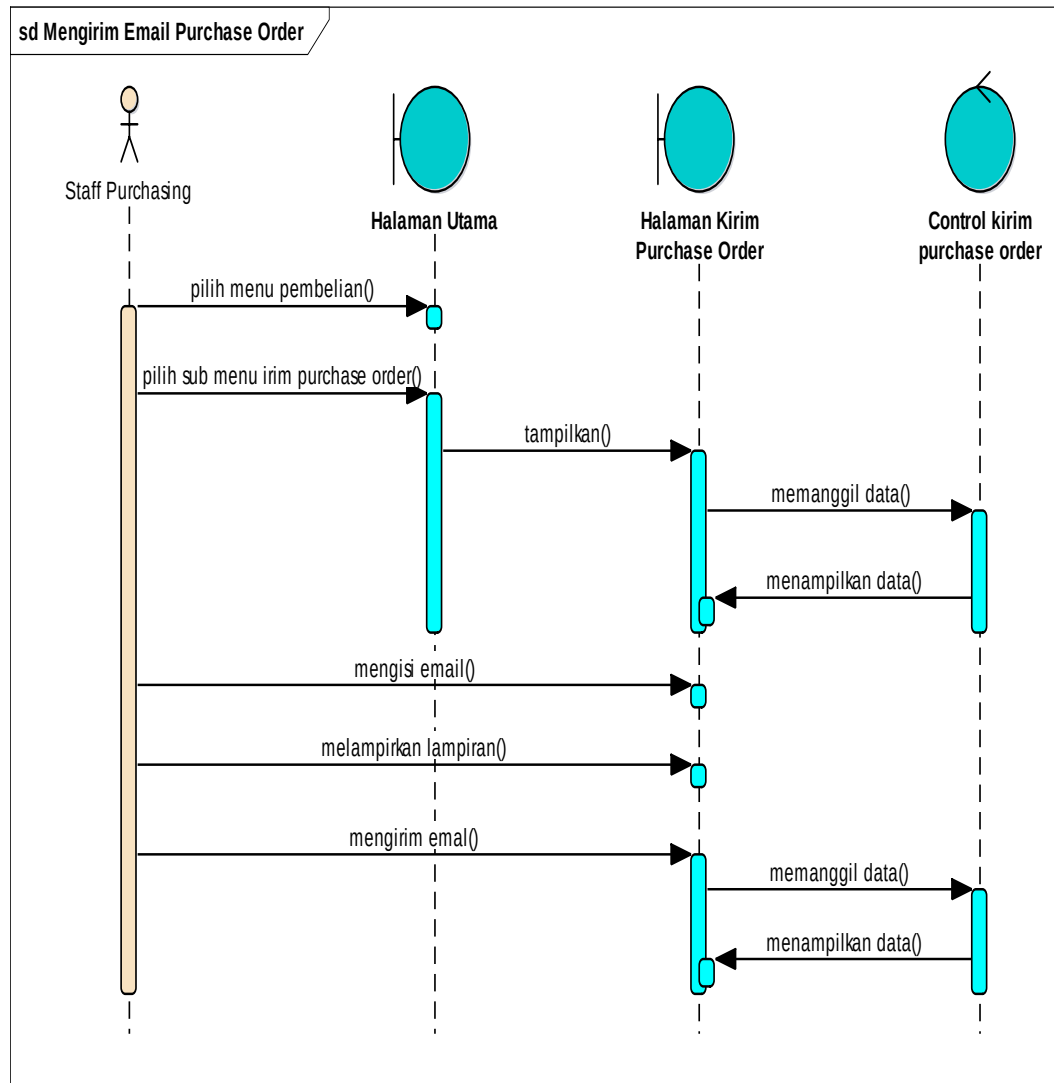
Gambar V.19 *Sequence Diagram* Mencetak *Purchase Order*

(Sumber: Hasil Analisis Data, 2018)

8. *Sequence Diagram* Mengirim *Purchase Order*

Sequence diagram mengirim *purchase order* menjelaskan sebuah *sequence diagram* dalam proses mengirim *purchase order* yang dilakukan oleh Divisi *Purchasing*.

Adapun *sequence diagram* dari *use case* mengirim *purchase order* dapat dilihat pada Gambar V.20.

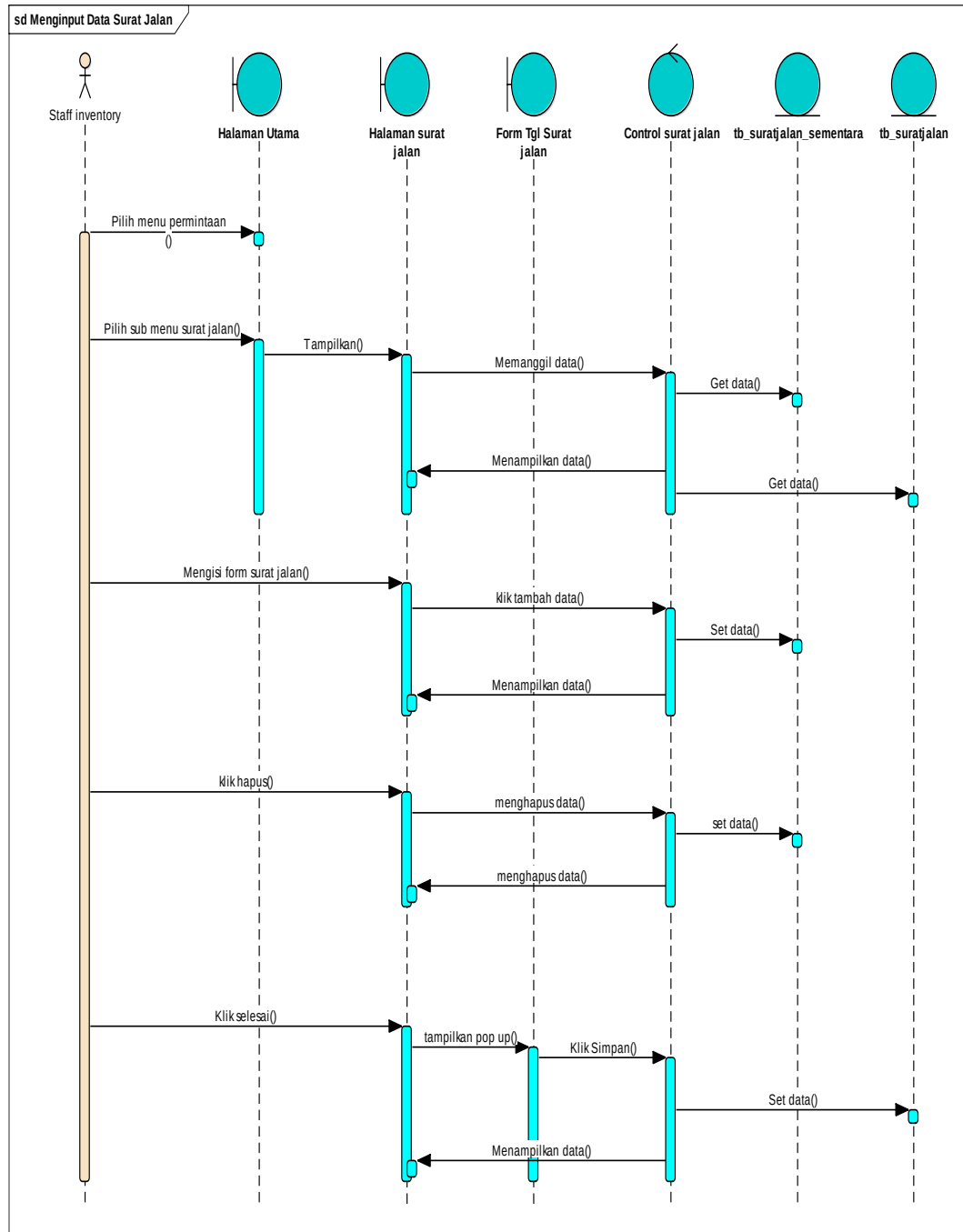


Gambar V.20 *Sequence Diagram* Mengirim *Purchase Order*

(Sumber: Hasil Analisis Data, 2018)

9. *Sequence Diagram* Menginput Surat Jalan

Sequence diagram menginput surat jalan menjelaskan sebuah *sequence diagram* dalam proses meenginput surat jalan yang dilakukan oleh Divisi *Inventory*. Adapun *sequence diagram* dari *use case* menginput surat jalan dapat dilihat pada Gambar V.21.

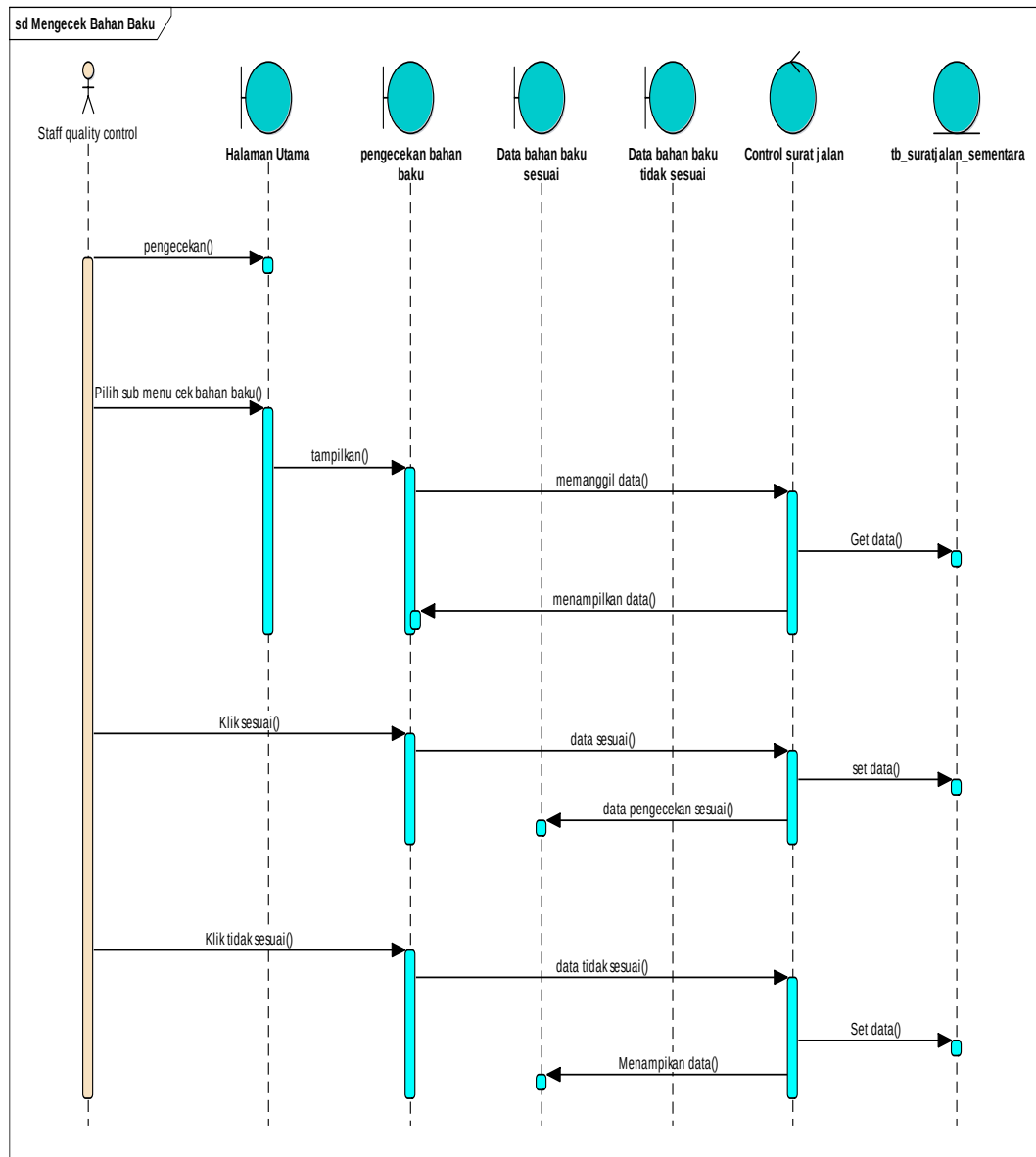


Gambar V.21 *Sequence Diagram* Menginput Data Surat Jalan

(Sumber: Hasil Analisis Data, 2018)

10. *Sequence Diagram* Mengecek Bahan Baku

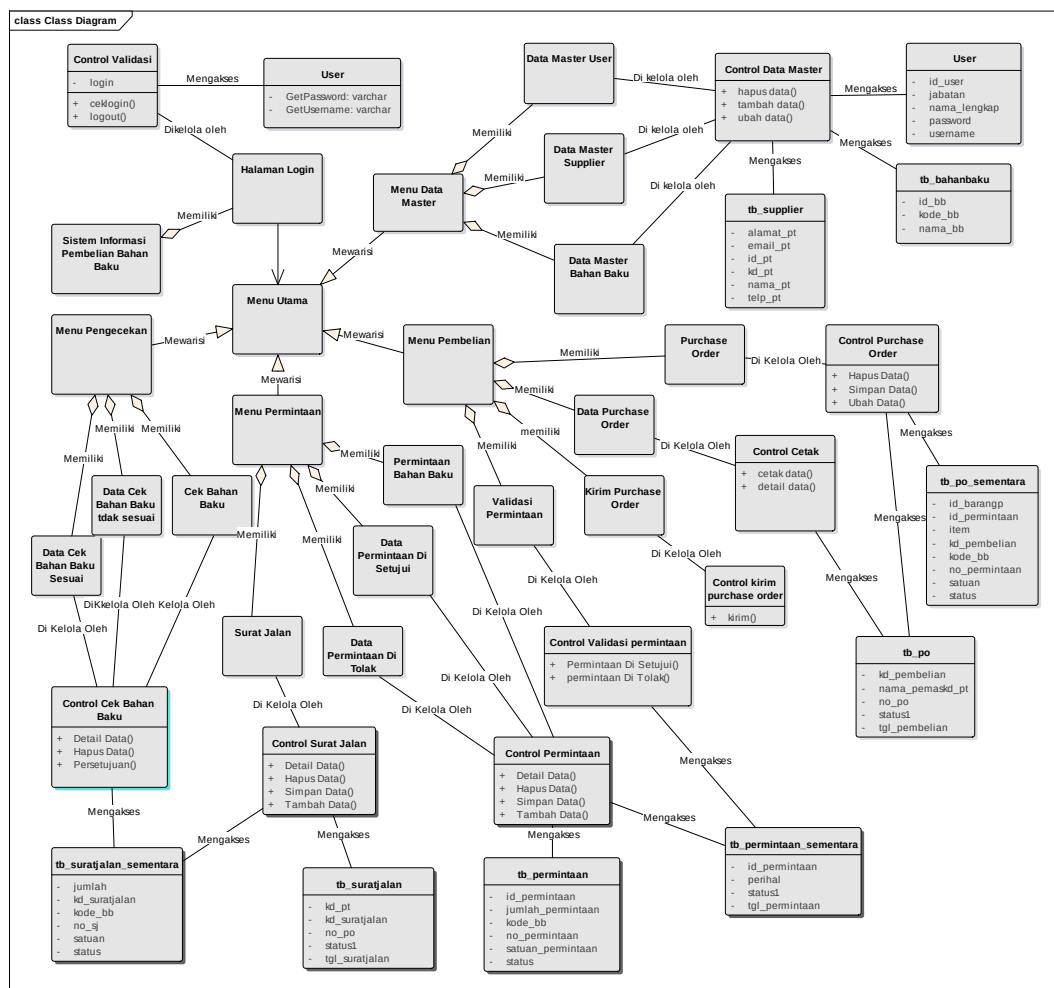
Sequence diagram mengecek bahan baku menjelaskan sebuah *sequence diagram* dalam proses mengecek bahan baku yang dilakukan oleh Divisi *Quality Control*. Adapun *sequence diagram* dari *use case* mengecek bahan baku dapat dilihat pada Gambar V.22.



Gambar V.22 *Sequence Diagram* Mengecek Bahan Baku

(Sumber: Hasil Analisis Data, 2018)

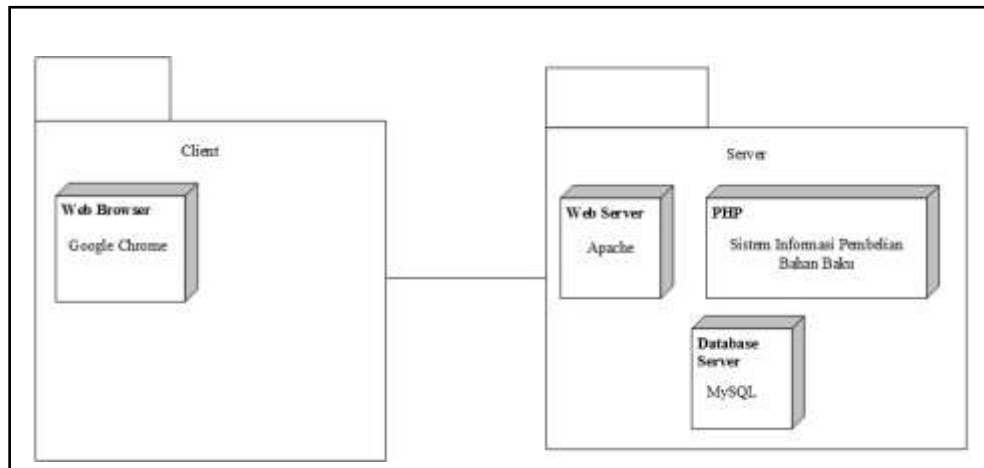
Class diagram pada usulan sistem informasi pembelian *spare part* mesin *stamping* digunakan untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat, untuk membangun sistem *class diagram* memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem.



(Sumber: Hasil Analisis, 2018)

5.3.5 *Deployment diagram*

Deployment diagram pada usulan sistem informasi pembelian bahan baku digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. Berikut merupakan *deployment diagram* sistem informasi pembelian bahan baku.



Gambar V.24 *Deployment Diagram*
(Sumber: Hasil Analisis, 2018)

Berikut ini adalah penjelasan Gambar V.20 *deployment diagram* sistem informasi pembelian bahan baku:

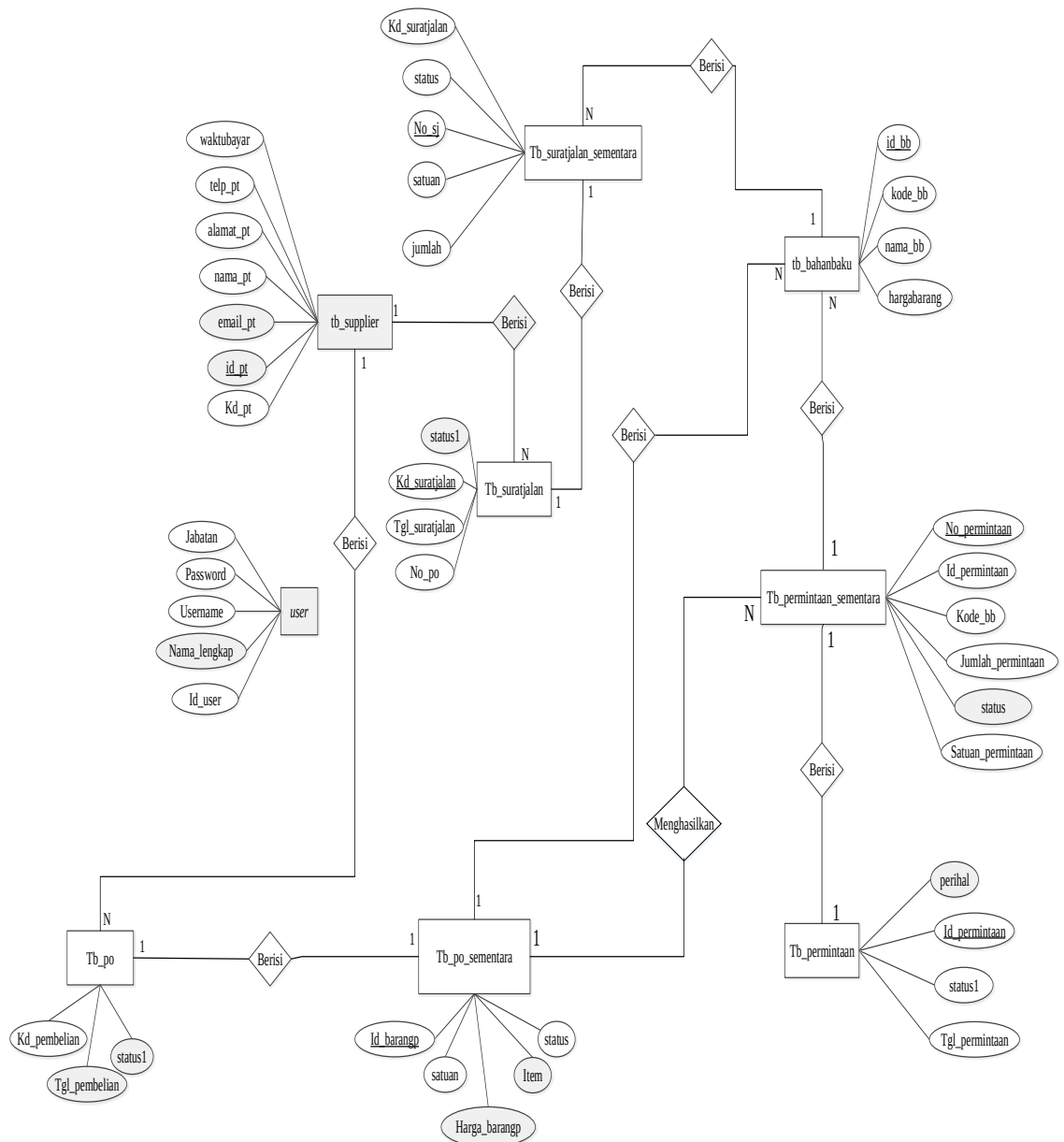
1. *Client* adalah computer *client* yang harus terinstall sebuah *web browser* untuk menjelaskan aplikasi sistem informasi pembelian bahan baku dan terhubung dengan *server*.
2. *Server* aplikasi sistem informasi pembelian bahan baku yang terdiri dari *web server* (*apache*), *application server* (sistem informasi pembelian bahan baku) dan *database server* (*purchasing*).

5.4 **Pemodelan Data Sistem Usulan**

Pemodelan data pada sistem informasi pembelian bahan baku di PT Baneex Indonesia menggunakan dua cara yaitu *Entity Relationship Diagram* (ERD) yang digunakan untuk menjelaskan hubungan antar data dalam basis data dan kamus data yang digunakan untuk menjelaskan isi dari *database* yang digunakan dalam sistem usulan.

5.4.1 Entity Relationship Diagram (ERD)

ERD digunakan sebagai suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi dalam sistem usulan. ERD digambarkan dengan simbol dan notasi yang berbeda, berikut merupakan ERD dari sistem informasi pembelian bahan baku.

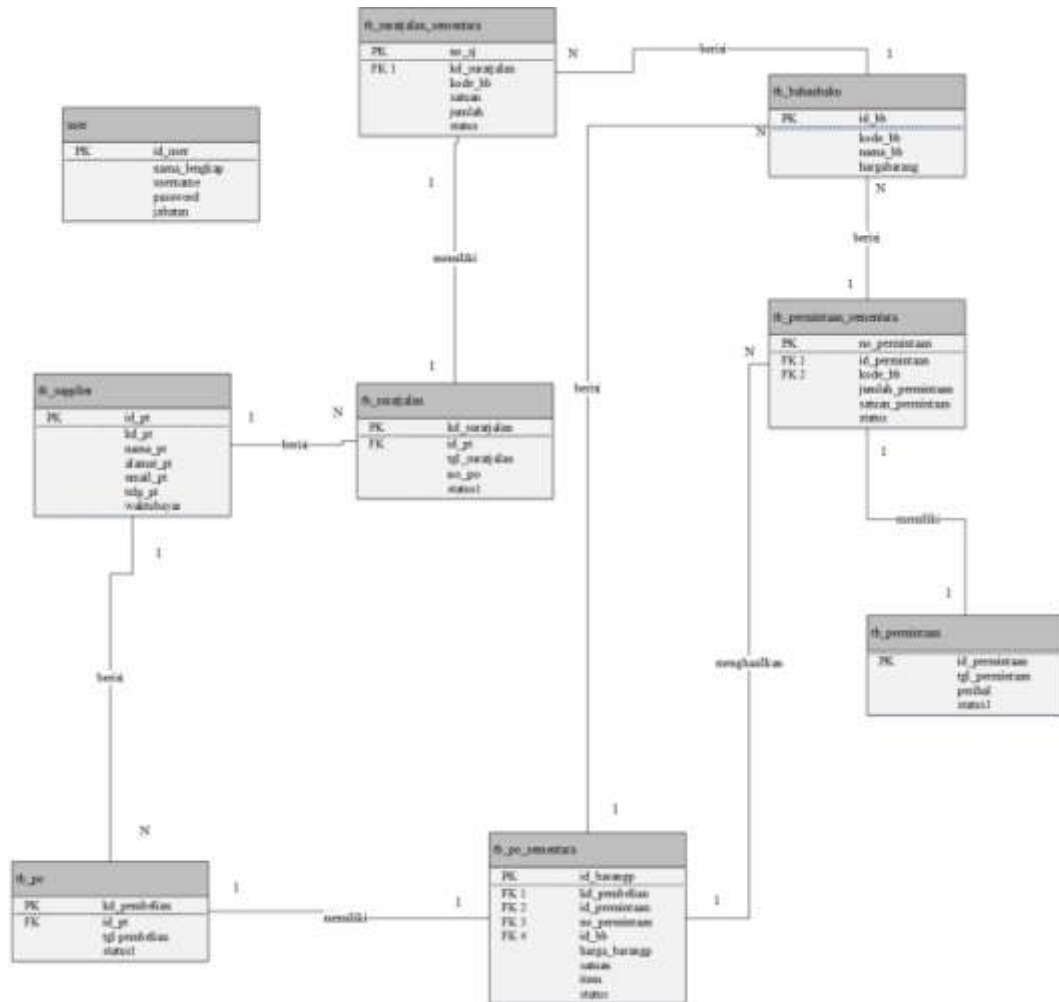


Gambar V.25 ERD Sistem Informasi Pembelian Bahan Baku

(Sumber: Hasil Analisis, 2018)

5.4.2 Conceptual Data Model (CDM)

CDM (*Conceptual Data Model*) sistem informasi pembelian bahan baku yang diusulkan terdapat pada Gambar V.26.



Gambar V.26 Conceptual Data Model

(Sumber: Hasil Analisis, 2018)

5.4.3 Kamus Data

Kamus menggambarkan sebuah katalog data yang terdapat didalam sistem dengan maksud untuk mendefinisikan data yang mengalir di dalam sistem dengan lengkap sehingga user dan analis sistem mempunyai pengertian yang sama tentang

input, output, dan komponen data store. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan dan basis data. Berikut adalah kamus data sistem informasi pembelian bahan baku:

1. Spesifikasi tabel *user*

- a. Nama tabel : *user*
- b. Fungsi : Untuk menyimpan data *user*/pengguna
- c. Tipe : File data *master*

Tabel V.14 Spesifikasi Tabel User

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id User	<i>id_user</i>	Int	15	<i>Primary key</i>
2	Nama Lengkap	<i>nama_lengkap</i>	Char	20	
3	<i>Username</i>	<i>username</i>	Varchar	15	
4	<i>Password</i>	<i>password</i>	Varchar	20	
5	Jabatan	<i>jabatan</i>	Varchar	20	

(Sumber: Hasil Analisis, 2018)

2. Spesifikasi tabel bahan baku

- a. Nama tabel : *tb_bahanbaku*
- b. Fungsi : Untuk menyimpan bahan baku
- c. Tipe : File data *master*

Tabel V.15 Spesifikasi Bahan Baku

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID BB	<i>id_bb</i>	Varchar	20	<i>Primary key</i>
2	Kode Bahan Baku	<i>kode_bb</i>	Varchar	20	
3	Nama Bahan Baku	<i>nama_bb</i>	Varchar	30	
4	Harga Bahan Baku	<i>hargabarang</i>	Int	15	

(Sumber: Hasil Analisis, 2018)

3. Spesifikasi tabel *supplier*

- a. Nama tabel : *tb_supplier*

- b. Fungsi : Untuk menyimpan data *supplier*
- c. Tipe : File data *master*

Tabel V.16 Spesifikasi Tabel *Supplier*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id <i>Supplier</i>	id_pt	Varchar	20	<i>Primary key</i>
2	Kode <i>Supplier</i>	kd_pt	Varchar	10	
3	Nama <i>Supplier</i>	nama_pt	Varchar	20	
4	Alamat	alamat_pt	Varchar	200	
5	E-mail	email_pt	Varchar	20	
6	Telepon	telp_pt	Varchar	15	
7	Waktu Bayar	waktubayar	Varchar	20	

(Sumber: Hasil Analisis, 2018)

4. Spesifikasi Tabel Permintaan Detail

- a. Nama tabel : tb_ permintaan_sementara
- b. Fungsi : Untuk menyimpan data permintaan
- c. Tipe : File data transaksi

Tabel V.17 Spesifikasi Tabel Permintaan Detail

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	No Permintaan	no_permintaan	Int	10	<i>Primary key</i>
2	Id Permintaan	id_permintaan	Varchar	20	<i>Foreign Key</i>
3	Kode Bahan Baku	id_bb	Varchar	20	<i>Foreign Key</i>
4	Satuan	satuan_permintaan	Varchar	5	
5	Jumlah	jumlah_permintaan	Varchar	10	
6	Status	status	Varchar	10	

(Sumber: Hasil Analisis, 2018)

5. Spesifikasi tabel permintaan

- d. Nama tabel : tb_ permintaan
- e. Fungsi : Untuk menyimpan data permintaan
- f. Tipe : File data transaksi

Tabel V.18 Spesifikasi Tabel Permintaan

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id Permintaan	id_permintaan	Varchar	20	<i>Primary key</i>
2	Tanggal Permintaan	tgl_permintaan	Date		
3	Status	status1	Varchar	20	

(Sumber: Hasil Analisis, 2018)

6. Spesifikasi tabel surat jalan

- a. Nama tabel : tb_suratjalan
- b. Fungsi : Untuk menyimpan data surat jalan
- c. Tipe : File data transaksi

Tabel V.19 Spesifikasi Tabel Surat Jalan

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Kode Surat Jalan	kd_suratjalan	Varchar	20	<i>Primary key</i>
2	Tanggal surat	tgl_suratjalan	Date		
3	Nama Pemasok	id_pt	Varchar	20	<i>Foreign key</i>
4	No PO	no_po	Varchar	30	
5	Status	status1	Varchar	20	

(Sumber: Hasil Analisis, 2018)

7. Spesifikasi tabel surat jalan detail

- a. Nama tabel : tb_suratjalan_sementara
- b. Fungsi : Untuk menyimpan data detail surat jalan
- c. Tipe : File data transaksi

Tabel V.20 Spesifikasi Tabel Surat Jalan Detail

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	No Surat Jalan	No_sj	Int	5	<i>Primary key</i>

Tabel V.20 Spesifikasi vTabel Surat Jalan Detail (Lanjutan)

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
2	Kode Surat Jalan	kd_suratjalan	Varchar	20	<i>Foreign key</i>
3	Nama Bahan Baku	id_bb	Varchar	20	<i>Foreign key</i>

4	Satuan	satuan	Varchar	15	
5	Jumlah	jumlah	Varchar	10	
6	Status	status	Varchar	20	

(Sumber: Hasil Analisis, 2018)

8. Spesifikasi tabel *purchase order*

- a. Nama tabel : tb_po
- b. Fungsi : Untuk menyimpan data *purchase order*
- c. Tipe : File data transaksi

Tabel V.21 Spesifikasi Tabel *Purchase Order*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Kode Pembelian	Kd_pembelian	Varchar	20	<i>Primary key</i>
2	Nama Pemasok	id_pt	Varchar	20	<i>Foreign key</i>
3	Tanggal <i>purchase order</i>	Tgl_pembelian	Date		
4	Status1	Status	Varchar	20	

(Sumber: Hasil Analisis, 2018)

9. Spesifikasi tabel *purchase order detail*

- a. Nama tabel : tb_po_sementara
- b. Fungsi : Untuk menyimpan detail data *purchase order*
- c. Tipe : File data transaksi

Tabel V.22 Spesifikasi Tabel *Purchase Order Detail*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	ID Barang	Id_barangp	int	10	<i>Primary key</i>

2	Kd_pembelian	Kd_pembelian	Varchar	20	<i>Foreign key</i>
3	Id permintaan	id_permintaan	Varchar	20	<i>Foreign key</i>
4	No Permintaan	no_permintaan	Int	10	<i>Foreign key</i>
5	Nama Bahan Baku	id_bb	Varchar	20	<i>Foreign key</i>
6	Satuan	Satuan_pembelian	Varchar	5	
7	Harga Bahan Baku	Harga_barangp	Varchar	20	
8	Jumlah	item	Varchar	20	
9	status	status	Varchar	20	

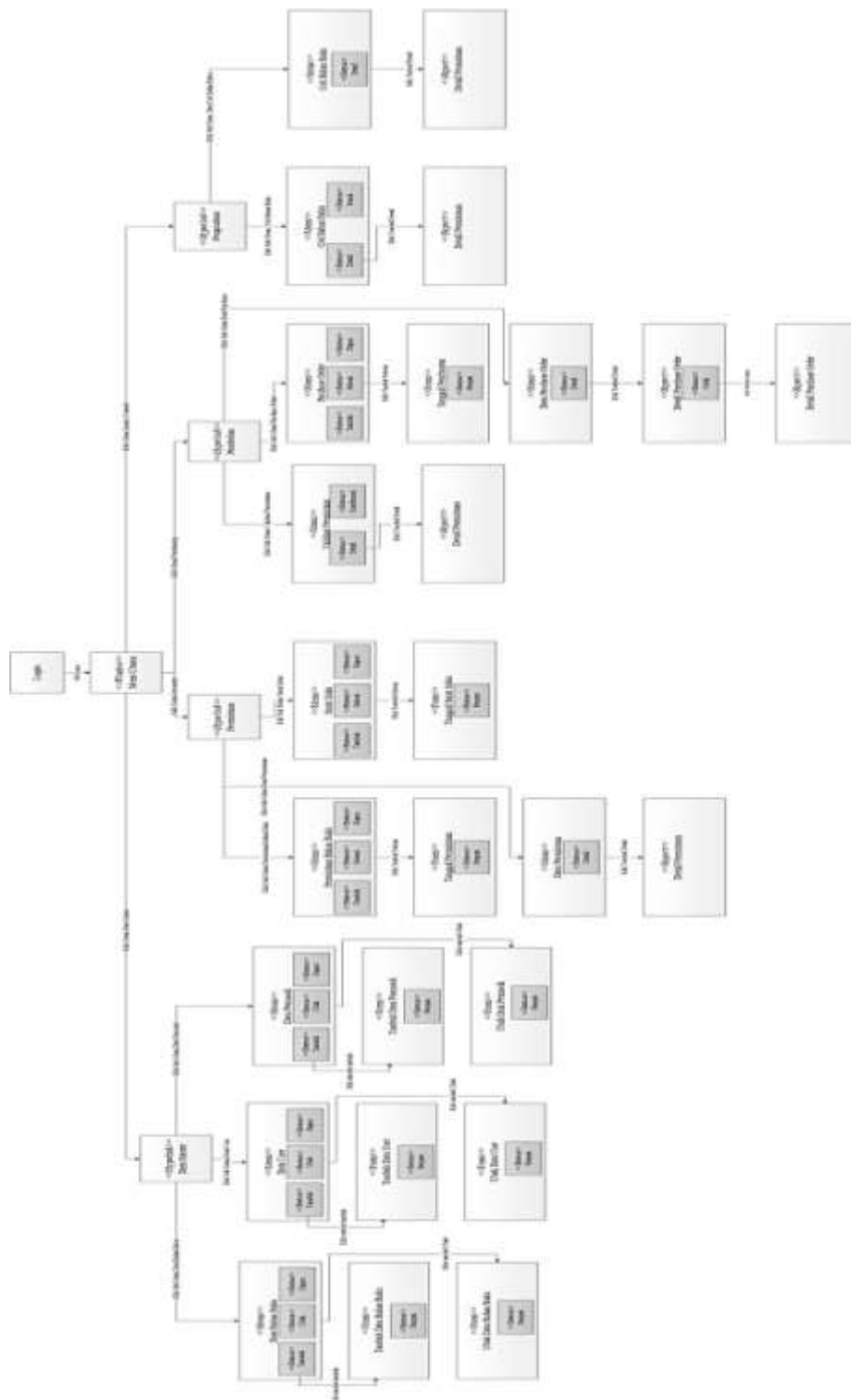
(Sumber: Hasil Analisis, 2018)

5.5 Perancangan Sistem

Perancangan sistem pada sistem usulan digunakan sebagai ilustrasi yang menggambarkan bagaimana aplikasi yang dibuat berjalan dan juga menggambarkan navigasi dari aplikasi tersebut dengan *tools Windows Navigation Diagram*(WND) serta menggambarkan tampilan antar muka dari aplikasi.

5.5.1 Windows Navigation Diagram (WND)

Windows Navigation Diagram pada sistem usulan digunakan untuk menunjukan bagaimana navigasi dari halaman-halaman yang terdapat di dalam aplikasi, berikut merupakan *Windows Navigation Diagram* sistem informasi pembelian bahan baku:



Gambar V.27 Windows Navigation Diagram

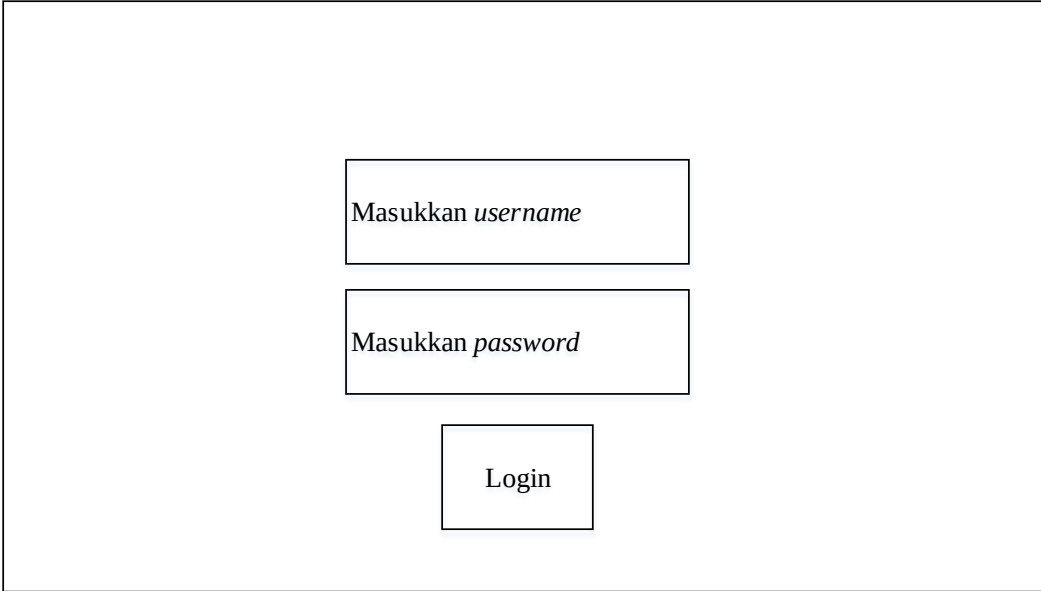
(Sumber: Hasil Analisis, 2018)

5.5.2 Perancangan Antarmuka Pengguna

Perancangan antarmuka dalam sistem usulan merupakan penggambaran antarmuka aplikasi yang hendak dibuat, antarmuka merupakan bagian yang berhubungan langsung dengan pengguna, berikut merupakan rancangan antarmuka sistem informasi pembelian bahan baku:

1. *Form Login*

Form login merupakan *form* yang digunakan pengguna sebagai jalan masuk ke dalam aplikasi dengan cara memasukan *username* dan *password* yang terdapat didalam *form login* tersebut, rancangan *form login* dapat dilihat pada gambar berikut:



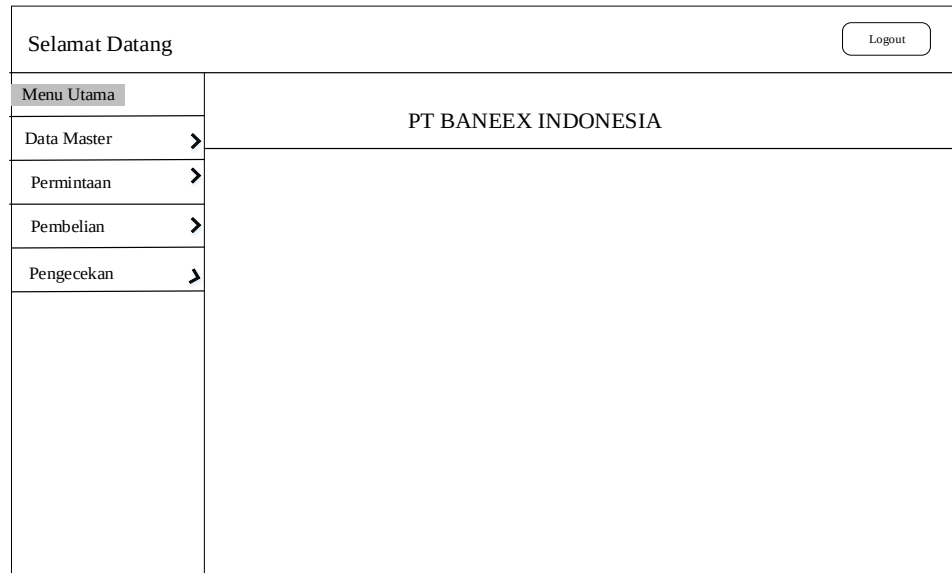
The diagram illustrates a login form interface. It consists of three rectangular input fields stacked vertically, each with a light blue background and a black border. The top field is labeled "Masukkan *username*", the middle field is labeled "Masukkan *password*", and the bottom field is a button labeled "Login". The fields are centered within a larger rectangular frame.

Gambar V.28 *Form Login*

(Sumber: Hasil Analisis, 2018)

2. Tampilan Halaman Menu Utama

Tampilan halaman menu utama adalah tampilan yang pertama kali muncul ketika user sukses melakukan *login*. Rancangan halaman menu utama dapat dilihat pada Gambar V.33.



Gambar V.29 Menu Utama

(Sumber: Hasil Analisis, 2018)

3. Tampilan Halaman Data Master *User*

Tampilan halaman data master *user* merupakan tampilan yang muncul ketika user memilih sub menu data *user*, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data master *user*, berikut tampilan halaman data master *user*:

Selamat Datang Logout

Menu Utama
Data Master ▾

Data User

Data Supplier
Data Bahan Baku
Permintaan >
Pembelian >
Pengecekan >

Data User

Show Entries
Search

No	Nama Lengkap	Username	Password	Level	Opsi
					✎ 🗑

Showing 1 to 2 of 2 entries

Previous
1
Next

+ Tambah Data

Gambar V.30 Tampilan Halaman Data Master *User*

(Sumber: Hasil Analisis, 2018)

Selamat Datang Logout

Menu Utama
Data Master ▾

Data User

Data Supplier
Data Bahan Baku
Permintaan >
Pembelian >
Pengecekan >

Tambah DataUser ✕

Nama Lengkap
Username
Password
Jabatan

Bersih

Tambah

Gambar V.31 Tampilan Tombol Tambah Data Halaman Data Master *User*

(Sumber: Hasil Analisis, 2018)

Selamat Datang Logout

Menu Utama
Data Master
Data User
Data Supplier
Data Bahan Baku
Permintaan
Pembelian
Pengecekan

Edit Data User
Nama Lengkap
Username
Password
Jabatan
Bersih
Simpan

Show
No

+

Gambar V.32 Tampilan Tombol Ubah Data Halaman Data Master *User*
(Sumber: Hasil Analisis, 2018)

4. Tampilan Halaman Data Master *Supplier*

Tampilan halaman data master *supplier* merupakan tampilan yang muncul ketika user memilih sub menu data supplier, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data master *supplier*, berikut tampilan halaman data master *supplier*:

Selamat Datang Logout

Menu Utama
Data Master
Data User
Data Supplier
Data Bahan Baku
Permintaan
Pembelian
Pengecekan

Data Supplier
Show
10
Entries
Search

No	Nama Lengkap	Username	Password	Level	Opsi

Showing 1 to 2 of 2 entries
Previous
1
Next

+
Tambah Data

Gambar V.33 Tampilan Halaman Data Master *Supplier*

(Sumber: Hasil Analisis, 2018)

The screenshot shows a web application interface. At the top, there is a header bar with 'Selamat Datang' on the left and a 'Logout' button on the right. Below the header is a sidebar menu with the following items: 'Menu Utama', 'Data Master' (with a dropdown arrow), 'Data User', 'Data Supplier' (highlighted), 'Data Bahan Baku', 'Permintaan' (with a right arrow), 'Pembelian' (with a right arrow), and 'Pengecekan' (with a right arrow). The main content area displays a modal window titled 'Tambah Data Suplier' with a close button (X) in the top right corner. The modal contains five text input fields: 'Kode Suplier', 'Nama Suplier', 'Alamat Suplier', 'E-mail Suplier', and 'Telepon Suplier'. At the bottom of the modal are two buttons: 'Bersih' and 'Tambah'. To the left of the modal, there is a 'Show' button and a 'No' button. To the right of the modal, there is a 'Show' button and a 'Next' button. At the bottom of the modal, there is a 'Tambah Data' button with a plus icon.

Gambar V.34 Tampilan Tombol Tambah Data Halaman Data Master *Supplier*

(Sumber: Hasil Analisis, 2018)

The screenshot shows the same web application interface as Gambar V.33, but with the modal window titled 'Edit Data Suplier' instead of 'Tambah Data Suplier'. The modal contains four text input fields: 'Nama Suplier', 'Alamat Suplier', 'E-mail Suplier', and 'Telepon Suplier'. At the bottom of the modal are two buttons: 'Bersih' and 'Simpan'. To the left of the modal, there is a 'Show' button and a 'No' button. To the right of the modal, there is a 'Show' button and a 'Next' button. At the bottom of the modal, there is a 'Tambah Data' button with a plus icon. The sidebar menu is the same as in Gambar V.33, but with 'Inventory' (with a dropdown arrow) added below 'Data Bahan Baku'.

Gambar V.35 Tampilan Tombol Ubah Data Halaman Data Master *Supplier*

(Sumber: Hasil Analisis, 2018)

5. Tampilan Halaman Data Master Bahan Baku

Tampilan halaman data master bahan baku merupakan tampilan yang muncul ketika user memilih sub menu data bahan baku, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data master bahan baku, berikut tampilan halaman data master bahan baku :

Gambar V.36 Tampilan Halaman Data Master Bahan Baku

(Sumber: Hasil Analisis, 2018)

Gambar V.37 Tampilan Tombol Tambah Data Halaman Data Master Bahan Baku

(Sumber: Hasil Analisis, 2018)

The screenshot displays a web application interface. At the top, a header bar contains 'Selamat Datang' on the left and a 'Logout' button on the right. Below the header is a sidebar menu with the following items: 'Menu Utama', 'Data Master' (with a dropdown arrow), 'Data User', 'Data Supplier', 'Data Bahan Baku' (highlighted), 'Permintaan' (with a right arrow), 'Pembelian' (with a right arrow), and 'Pengecekan' (with a right arrow). The main content area is titled 'Data' and features a modal window titled 'Edit Data Bahan Baku' with a close button (X). The modal contains two input fields: 'Kode Bahan Baku' and 'Nama Bahan Baku'. Below these fields is a 'Tambah' button. To the left of the modal, there is a table with columns 'No' and 'Nama'. Below the table, it says 'Showing 1 to 2 of 2 entries'. At the bottom of the main content area, there is a '+ Tambah Data' button and a pagination control with 'Previous', '1', and 'Next' buttons.

Gambar V.38 Tampilan Tombol Ubah Data Halaman Data Master Bahan Baku

(Sumber: Hasil Analisis, 2018)

6. Tampilan Halaman Permintaan Pembelian

Halaman permintaan pembelian merupakan halaman yang tampil ketika user memilih menu sub menu permintaan pembelian pada halaman ini akan ditampilkan *form* permintaan yang akan dilakukan, berikut tampilan halaman permintaan pembelian:

Selamat Datang		Logout													
Menu Utama	Permintaan Pembelian Bahan Baku Nama Bahan Baku Satuan Jumlah Tambah Data Show 10 Entries Search <table border="1"> <thead> <tr> <th>Nama Bahan Baku</th> <th>Satuan</th> <th>Jumlah</th> <th>Opsi</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table> Showing 1 to 2 of 2 entries Previous 1 Next Selesai			Nama Bahan Baku	Satuan	Jumlah	Opsi								
Nama Bahan Baku				Satuan	Jumlah	Opsi									

Gambar V.39 Tampilan Halaman Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

Pada tampilan halaman permintaan pembelian diatas terdapat dua tombol yaitu tambah data, selesai, dan hapus jika tombol tambah data diklik maka data akan tampil di tabel yang berada di samping form input. Jika klik tombol selesai maka tampilannya seperti ini:

Selamat Datang		Logout													
Menu Utama	Permintaan Pembelian Bahan Baku Nama Bahan Baku Satuan Jumlah Tambah Show 10 Entries Search <table border="1"> <thead> <tr> <th>Nama Bahan Baku</th> <th>Satuan</th> <th>Jumlah</th> <th>Opsi</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td></tr> <tr><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table> Showing 1 to 2 of 2 entries Previous 1 Next Selesai			Nama Bahan Baku	Satuan	Jumlah	Opsi								
Nama Bahan Baku				Satuan	Jumlah	Opsi									

Gambar V.40 Tampilan Tombol Selesai Halaman Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

7. Tampilan Halaman Permintaan Disetujui

Halaman permintaan data permintaan merupakan halaman yang tampil ketika user memilih menu permintaan lalu memilih sub menu permintaan disetujui pada halaman ini akan

ditampilkan list permintaan yang disetujui oleh divisi *purchasing*, berikut tampilan halaman permintaan disetujui:

PT Baneex Indonesia		Logout	
Menu Utama	Data Permintaan Disetujui		
Data Master >			
Permintaan v	10 Entries		Search
Permintaan Pembelian			
Permintaan Disetujui	No	No Permintaan	Nama Bahan Baku
Permintaan Ditolak			Satuan
Surat Jalan			Status
Pembelian v			Ops
Pengecekan v			Detail
			Previous 1 Next

Gambar V.41 Tampilan Halaman Permintaan Disetujui

(Sumber: Hasil Analisis, 2018)

8. Tampilan Halaman Permintaan Ditolak

Halaman permintaan data permintaan merupakan halaman yang tampil ketika user memilih menu permintaan lalu memilih sub menu permintaan ditolak oleh Divisi *Purchasing*:

PT Baneex Indonesia		Logout	
Menu Utama	Data Permintaan Ditolak		
Data Master >			
Permintaan v	10 Entries		Search
Permintaan Pembelian			
Permintaan Disetujui	No	No Permintaan	Nama Bahan Baku
Permintaan Ditolak			Satuan
Surat Jalan			Status
Pembelian v			Ops
Pengecekan v			Detail
			Previous 1 Next

Gambar V.42 Tampilan Halaman Permintaan Ditolak

(Sumber: Hasil Analisis, 2018)

9. Tampilan Halaman Validasi Permintaan

Halaman validasi permintaan merupakan halaman yang ditujukan kepada divisi *purchasing* untuk melihat permintaan yang diajukan namun belum disetujui, berikut halaman validasi permintaan:

Selamat Datang		Logout											
Menu Utama													
Data Master	10 ▼ Entries Search												
Permintaan													
Pembelian	<table border="1"> <thead> <tr> <th>No</th> <th>No. Permintaan</th> <th>Tanggal Permintaan</th> <th>Status</th> <th>Opsi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td> Detail Validasi </td> </tr> </tbody> </table>			No	No. Permintaan	Tanggal Permintaan	Status	Opsi					Detail Validasi
No	No. Permintaan	Tanggal Permintaan	Status	Opsi									
				Detail Validasi									
Validasi													
Purchase Order													
Data Purchase Order													
Pengecekan	Previous 1 Next												

Gambar V.43 Tampilan Halaman Validasi Permintaan
(Sumber: Hasil Analisis, 2018)

10. Tampilan Halaman *Purchase Order*

Halaman *purchase order* merupakan halaman yang muncul ketika user memilih sub menu pembelian, pada halaman ini menampilkan form yang siap diinput, berikut tampilannya:

Selamat Datang		Logout	
Menu Utama			
Data Master	10 ▼ Entries Search		
Permintaan			
Pembelian	---Pilih Bahan Baku--- ▼		
Validasi			
Purchase Order	Nama Bahan Baku Satuan Jumlah Harga Showing 1 to 2 of 2 entries Selesai Tambah Data		
Data Purchase Order			
Pengecekan	Previous 1 Next		

Gambar V.44 Tampilan Halaman *Purchase Order*
(Sumber: Hasil Analisis, 2018)

Pada tampilan halaman *purchase order* diatas terdapat dua tombol yaitu tambah data, selesai, dan hapus jika tombol tambah data diklik maka data akan tampil di tabel yang berada di samping form input. Jika klik tombol selesai maka tampilannya seperti ini:

PT Baneex Indonesia		Logout
Menu Utama		
Data Master		
Permintaan		
Pembelian		
Validasi		
Purchase Order		
Data Purchase Order		
Pengecekan		

Nama Bahan Baku
Show 10 Entries
Search

Pilih Bahan Baku

Form Input Tanggal

Jumlah
Ops

Satuan

Tanggal Pembuatan

Nama Pemasok

Pilih Pemasok

Simpan

Tambah Data

Previous
1
Next

Gambar V.45 Tampilan Tombol Selesai Halaman *Purchase Order*

(Sumber: Hasil Analisis, 2018)

11. Tampilan Halaman Data *Purchase Order*

Halaman data *purchase order* merupakan halaman yang muncul ketika user memilih sub menu pembelian, pada halaman ini menampilkan tabel data *purchase order* yang telah selesai di input., berikut tampilannya:

Selamat Datang		Logout
Menu Utama		
Data Master		
Permintaan		
Pembelian		
Validasi		
Purchase Order		
Data Purchase Order		
Kirim Purchase Order		
Pengecekan		

10 Entries
Search

No	No. PO	Nama Bahan Baku	Satuan	Status	Ops
					Detail

Previous
1
Next

Gambar V.46 Tampilan Halaman Data *Purchase Order*

(Sumber: Hasil Analisis, 2018)

12. Tampilan Halaman Kirim *Purchase Order*

Halaman kirim *purchase order* merupakan halaman yang muncul ketika user memilih sub menu pembelian, pada halaman ini menampilkan form yang siap diinput, berikut tampilannya:

Selamat Datang Logout	
Menu Utama	Purchase Order Email : Subject : Attachment : Choose File No File Chosen Kirim
Data Master >	
Permintaan >	
Pembelian ▼	
Validasi	
Purchase Order	
Data Purchase Order	
Kirim Purchase Order	
Pengecekan >	

Gambar V.47 Tampilan Halaman Kirim *Purchase Order*
(Sumber: Hasil Analisis, 2018)

13. Tampilan Halaman Surat Jalan

Halaman surat jalan merupakan halaman yang muncul ketika user memilih sub menu permintaan, pada halaman ini menampilkan form yang siap diinput, berikut tampilannya:

Selamat Datang Logout									
Menu Utama	Surat Jalan Nama Bahan Baku ---Pilih Bahan Baku--- Show 10 Entries Search Satuan Jumlah Tambah Data <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 33%;">Nama Bahan Baku</th> <th style="width: 16%;">Satuan</th> <th style="width: 16%;">Jumlah</th> <th style="width: 35%;">Ops</th> </tr> </thead> <tbody> <tr><td> </td><td> </td><td> </td><td> </td></tr> </tbody> </table> Showing 1 to 2 of 2 entries Previous 1 Next Selesai	Nama Bahan Baku	Satuan	Jumlah	Ops				
Nama Bahan Baku		Satuan	Jumlah	Ops					
Data Master >									
Permintaan ▼									
Permintaan Pembelian									
Data Permintaan									
Surat Jalan									
Pembelian >									
Pengecekan >									

Gambar V.48 Tampilan Halaman Surat Jalan
(Sumber: Hasil Analisis, 2018)

Selamat Datang		Logout	
Menu Utama	Surat Jalan		
Data Master			
Permintaan	Nama Bahan Baku Show 10 Entries Search		
Permintaan Pembelian	---Pilih Bahan Baku--- Satuan Jumlah Tambah	Form Input Tanggal Nomor Surat Jalan Tanggal Surat Jalan Nama Pemasok ---Pilih Pemasok--- Simpan	Jumlah Opsi Previous 1 Next
Data Permintaan			
Surat Jalan			
Pembelian			
Pengecekan			

Gambar V.49 Tampilan Tombol Selesai Halaman Surat Jalan
(Sumber: Hasil Analisis, 2018)

14. Tampilan Halaman Cek Bahan Baku

Halaman cek bahan baku merupakan halaman yang muncul ketika user memilih sub menu pengecekan , pada halaman ini menampilkan form yang siap diinput, berikut tampilannya:

Selamat Datang		Logout											
Menu Utama	Pengecekan Bahan Baku												
Data Master													
Permintaan	10 Entries Search												
Pembelian													
Pengecekan	<table border="1"> <thead> <tr> <th>No</th> <th>No. Surat Jalan</th> <th>Tanggal Surat Jalan</th> <th>Status</th> <th>Opsi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td>Detail Sesuai</td> </tr> </tbody> </table>			No	No. Surat Jalan	Tanggal Surat Jalan	Status	Opsi					Detail Sesuai
No	No. Surat Jalan	Tanggal Surat Jalan	Status	Opsi									
				Detail Sesuai									
Cek Bahan Baku													
Data Cek Bahan Baku													
	Previous 1 Next												

Gambar V.50 Tampilan Halaman Cek Bahan Baku
(Sumber: Hasil Analisis, 2018)

15. Tampilan Halaman Data Cek Bahan Baku Sesuai

Halaman data cek bahan baku sesuai merupakan halaman yang muncul ketika user memilih sub menu pengecekan , pada halaman ini menampilkan data cek bahan baku.

Selamat Datang		Logout											
Menu Utama	Data Pengecekan Bahan Baku Sesuai 10 ▾ Entries Search <table border="1"> <thead> <tr> <th>No</th> <th>No. Surat Jalan</th> <th>Tanggal Surat Jalan</th> <th>Status</th> <th>Opsi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td>Detail</td> </tr> </tbody> </table> Previous 1 Next			No	No. Surat Jalan	Tanggal Surat Jalan	Status	Opsi					Detail
No				No. Surat Jalan	Tanggal Surat Jalan	Status	Opsi						
							Detail						
Data Master													
Permintaan													
Pembelian													
Pengecekan ▾													
Cek Bahan Baku													
Data Cek Bahan Baku Sesuai													
Data Cek Bahan Baku Tidak Sesuai													

Gambar V.51 Tampilan Halaman Data Cek Bahan Baku Sesuai

(Sumber: Hasil Analisis, 2018)

16. Tampilan Halaman Data Cek Bahan Baku Tidak Sesuai

Halaman data cek bahan baku tidak sesuai merupakan halaman yang muncul ketika user memilih sub mnu pengecekan , pada halaman ini menampilkan data cek bahan baku.

Selamat Datang		Logout											
Menu Utama	Data Pengecekan Bahan Baku Sesuai 10 ▾ Entries Search <table border="1"> <thead> <tr> <th>No</th> <th>No. Surat Jalan</th> <th>Tanggal Surat Jalan</th> <th>Status</th> <th>Opsi</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> <td>Detail</td> </tr> </tbody> </table> Previous 1 Next			No	No. Surat Jalan	Tanggal Surat Jalan	Status	Opsi					Detail
No				No. Surat Jalan	Tanggal Surat Jalan	Status	Opsi						
							Detail						
Data Master													
Permintaan													
Pembelian													
Pengecekan ▾													
Cek Bahan Baku													
Data Cek Bahan Baku Sesuai													
Data Cek Bahan Baku Tidak Sesuai													

Gambar V.52 Tampilan Halaman Data Cek Bahan Baku Tidak Sesuai

(Sumber: Hasil Analisis, 2018)

5.6 Implementasi Sistem *Software* dan *Hardware*

Tahap ini adalah tahap pengkodean program menggunakan bahasa pemrograman PHP dengan *Sublime Text 3* sebagai aplikasi editor. Setiap *interface* berisikan kode

program agar program dapat dijalankan sesuai dengan fungsinya. Untuk mendukung kebutuhan implementasi sistem diperlukan suatu spesifikasi perangkat lunak (*software*) maupun perangkat keras (*hardware*). Adapun spesifikasinya sebagai berikut:

1. Analisis Kebutuhan *Software*

- a. Sistem Operasi : *Microsoft Windows 7*
- b. *Database Server* : MySQL versi 5.6.26
- c. Bahasa Pemrograman : PHP 5.6.12
- d. *Web Browser* : *Google Chrome*

2. Analisis Kebutuhan *Hardware*

- a. *Processor* : Minimal *Processor Pentium IV*
- b. RAM : Minimal RAM 1024 MB
- c. *Harddisk* : Minimal *Harddisk* 100 GB
- d. *Mouse, Keyboard, Monitor* sebagai peralatan antar muka.
- e. *Printer* sebagai media pencetakan dokumen.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

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

1. Aplikasi sistem informasi pembelian bahan baku ini dapat mengintegrasikan antar divisi sehingga dapat memudahkan Divisi *Inventory* untuk melakukan pembuatan permintaan bahan baku, Divisi *Purchasing* untuk melakukan pembuatan *purchase order*, dan Divisi *Quality Control* untuk melakukan pengecekan bahan baku.
2. Aplikasi sistem informasi pembelian bahan baku ini terintegrasi dengan *database* sehingga data dapat tersimpan dengan aman dan mengurangi kehilangan data, kerusakan data serta dapat mempermudah saat mencari data.

6.2 Saran

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

1. Untuk penerapan sistem baru menggantikan sistem yang lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialisasi penggunaan sistem ini kepada divisi yang terkait.
2. Sebaiknya dilakukan pemeliharaan aplikasi secara berkala, sehingga aplikasi dapat berjalan dengan baik.

DAFTAR PUSTAKA

- Asri, Marwan dan Adisaputro , Gunawan. 2011. *Anggaran Perusahaan*. Yogyakarta : BPFE – Yogyakarta.
- Baroto, Teguh. 2002. *Perencanaan dan pengendalian produksi*. Jakarta: Ghalia Indonesia.
- Damiri, J. 2005. *Manajemen Pembelian, Penerimaan, dan Penyimpanan*. Yogyakarta: Graha Ilmu.
- Dennis, Alan., Barbara Haley., dan David Tegarden. 2015. *System Analysis and Design, Fifth Edition*. America: Willey.inc.
- Hanggana, Sri. 2006. *Prinsip Dasar Akuntansi Biaya*. Surakarta : Mediatama.
- HM, Jogiyoanto. 2005. *Analisis dan Desain Sistem Informasi : Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*. Yogyakarta : Andi.
- Irawati, Susan, 2008. *Akuntansi Dasar 1 & 2*. Bandung : Pustaka.
- Jubilee, Enterprise. 2017. *Mengolah Database dengan MS Ecel 2016*. Jakarta: PT Elex Media Komputindo.
- Kadir, Abdul. 2014. *Pengenalan Sistem Informasi Edisi Revisi*. Yogyakarta: Andi.
- Kadir, Abdul. 2018. *Pemograman Android & Database*. Jakarta : PT Elex Media Komputindo.
- Ladjamudin, Al-Bahra Bin. 2013. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Graha Ilmu.
- M. Nafarin. 2007. *Penganggaran Perusahaan, Edisi Ketiga*. Jakarta : Salemba Empat.
- Marliana B, Winanti, S.Si. M, Si. 2014. *Sistem Informasi Manajemen*. Bandung.
- McLeod, Raymond., dan Gerge. 2011. *Sistem Informasi Manajemen*. Jakarta: Salemba Empat.
- Mulyadi, 2001. *Sistem Akuntansi*. Edisi Ketiga . Jakarta : Salemba Empat
- Mustakini, Jogiyoanto Hartono. 2009. *Sistem Informasi Teknologi*, Yogyakarta : Andi offset.
- Puspitawati, Lilis., dan Anggadini, Sri Dewi. 2011. *Sistem Informasi Akuntansi*. Yogyakarta: Graha Ilmu.
- Rosa, A.S, dan Shalahuddin, M. 2015. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.

Soeherman, Bonnie dan Pinontoan, Marion.2008.*Designing Information System*.Jakarta: Elex Media Komputindo.

Satzingeretal, Jackson dan Burd. 2012. *System Analysis and Desaign in a Changing World*. USA: Course Technology.

Sutabri, Tata. 2016. *Sistem Informasi Manajemen Edisi Revisi*. Yogyakarta : Andi.

Sutaji, Deni.2012.*Sistem Inventory Mini Market dengan PHP dan JQuery*.Yogyakarta: Lokomedia.

Welling, L dan Thomson, L.2003.*PHP and MySql Web Developnent, Second Edition*.Indianapolis: Sams Publishing.

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

LAMPIRAN

LAMPIRAN A

TAMPILAN PROGRAM

1. Tampilan Login



The login page features a header image with the text "PT BANEEX INDONESIA" and a background of a hand holding a magnifying glass over a document. Below the header, there are two input fields: "Username" with the placeholder "Masukkan Username" and "Password" with the placeholder "Masukkan Password". A green "Login" button is positioned below the password field.

2. Tampilan Menu Utama



3. Tampilan Data *User*

SELAMAT DATANG Log Out

Menu Utama

Data Master

Data User

Data Supplier

Data Bahan Baku

Pembelian

Data User

Show 10 entries Search

No.	Nama Lengkap	Username	Password	Jabatan	opsi
1	muhammad	purchasing	purchasing	purchasing	 
2	sakar	inventory	inventory	inventory	 
3	ahmad	quality	quality	quality	 

Showing 1 to 3 of 3 entries

Previous 1 Next

+ Tambah Data

4. Tampilan Data Supplier

SELAMAT DATANG Log Out

Menu Utama

Data Master

Data User

Data Supplier

Data Bahan Baku

Pembelian

Data Supplier

Show 10 entries Search

No.	Kode Supplier	Nama Supplier	Alamat	E-mail	Telepon	Waktu Bayar	opsi
1	MZ	PT MOUNT ZUDSPITZ	TRUMBEE BLOK 15 NO 7 DELTA SURAB	mounta@gmail.com	031-2275-0900	45 days	 
2	INS	PT LANGSUNG	Kawasan Industri MM 200, Jalan Sumblo Blok A3, Mekamwang, Cikarang, Gandamalak, Cikarang Bar., Bekasi, Jawa Barat 17520	kinggeng@gmail.com	021-8865236	30 days	 
3	JHD	PT JHDATRA	BEKASI	jhandan@gmail.com	021-8837368	35 days	 

Showing 1 to 3 of 3 entries

Previous 1 Next

+ Tambah Data

5. Tampilan Data Bahan Baku

SELAMAT DATANG log Out

Menu Utama

Data Master

Data User

Data Supplier

Data Bahan Baku

Permintaan

Data Bahan Baku

Show: 4 entries Search:

No.	Kode bahan baku	Nama bahan baku	Harga bahan baku	Opel
1	B0304	SWWD ASS DIAMETER 4 MM	Rp. 10.000	 
2	B0305	SWWD ASS DIAMETER 5 MM	Rp. 11.000	 
3	B0306	SWWD ASS DIAMETER 6 MM	Rp. 12.000	 
4	B0307	SWWD ASS DIAMETER 7 MM	Rp. 13.000	 

Showing 1 to 4 of 4 entries

Previous 1 Next

+ Tambahkan Data

6. Tampilan Halaman Permintaan Bahan Baku

SELAMAT DATANG log Out

Menu Utama

Permintaan

Permintaan BB

Permintaan Stokup

Permintaan Stokuk

Surat Jalan

Permintaan Pembelian

Show: 0 entries Search:

Nama Bahan Baku:

Satuan:

Jumlah:

Nama bahan baku: * Satuan: Jumlah: Opel:

No data available in table

Showing 0 to 0 of 0 entries

Previous Next

+ Tambahkan Data

7. Tampilan Halaman Permintaan Disetujui



8. Tampilan Halaman Permintaan Ditolak



9. Tampilan Halaman Surat Jalan

Selamat Datang

Log Out

Menu Utama

Pembelian

Pembelian BB

Pembelian Stoknya

Pembelian Stokas

Surat Jalan

Surat Jalan

Names Bahan Baku

---Pilih Bahan Baku---

Satuan

---Pilih Satuan---

Jumlah

Tambah Data

Show 10 entries

Search

Names bahan baku	Satuan	Jumlah	Opsi
No data available in table			

Showing 0 to 0 of 0 entries

Previous

Next

10. Tampilan Halaman Validasi Permintaan

SELAJART DATANG

Menu Utama

Doko Master

Pembelian

Validasi Permintaan

Purchase Order

Data Purchase Order

Kirim Purchase Order

Logout

Validasi Permintaan

Show 10 per page

Search

No.	Nomor Permintaan	Bahan Baku	Jumlah	Status	Ops
1.	SP00-005	SWME ASSI DIAMETER 5 MM	500	proses	Selesai Get Solution Cancel

Showing 1 to 1 of 1 entries

Previous

Next

11. Tampilan Halaman *Purchase Order*

Menu Utama

Data Master

Pembelian

Validasi Permisian

Purchase Order

Data Purchase Order

Kirim Purchase Order

Pembuatan Purchase Order

Tambah Data

Nama Bahan Baku

—Pilih Bahan Baku—

Satuan

Jumlah Barang

Harga Barang

Show 10 entries

Search

No.	Nama bahan baku	Satuan	Jumlah	Harga	Total	Ops
No items available in table						

Showing 0 to 0 of 0 entries

Previous Next

Submit

12. Tampilan Halaman Data Purchase Order

SELAMAT DATANG

Log Out

Menu Utama

Data Master

Pembelian

Validasi Permisian

Purchase Order

Data Purchase Order

Kirim Purchase Order

Data Purchase Order

Show 10 entries

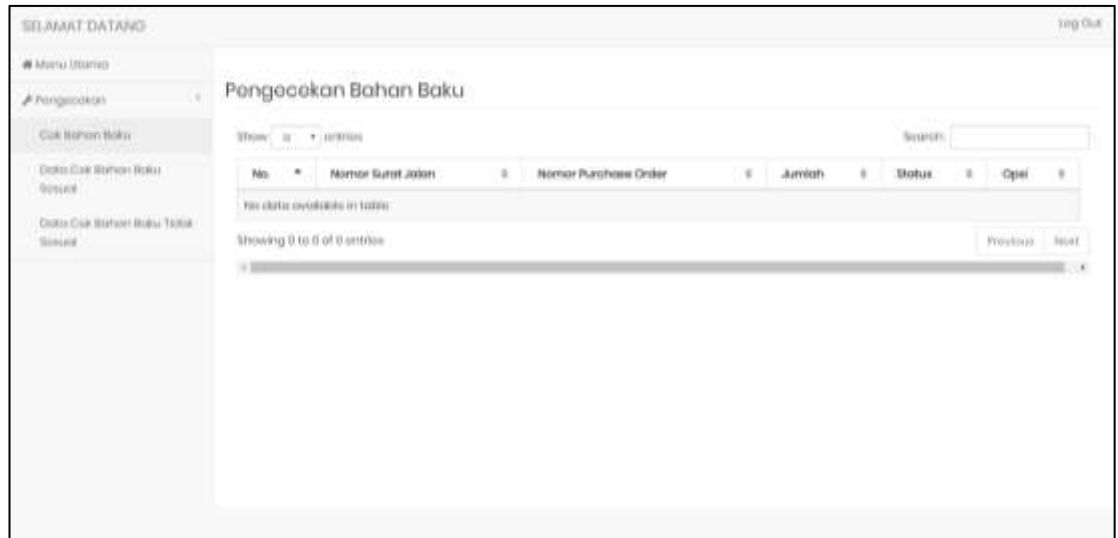
Search

No.	Nomor Purchase Order	Nama Supplier	Tanggal Pembelian	Ops
1	0002/M2/SP/K/2008	PT MOUNT JUCSPRTZE	2008-09-08	Tambah
2	0003/M2/SP/K/2008	PT MOUNT JUCSPRTZE	2008-09-08	Tambah
3	0004/M2/SP/K/2008	PT JHONATAN	2008-09-08	Tambah

Showing 1 to 3 of 3 entries

Previous 1 Next

13. Tampilan Halaman Cek Bahan Baku



14. Tampilan Halaman Data Cek Bahan Baku Sesuai



15. Tampilan Halaman Data Cek Bahan Baku Tidak Sesuai



KODE PROGRAM

1. Login.php

```
<?php
@session_start();
include "../koneksi.php";
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <title>PT BANEEX INDONESIA</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
<!--
=====-->
    <link rel="icon" type="image/png" href="images/logobb.png"/>
<!--
=====
-->
    <link rel="stylesheet" type="text/css"
href="vendor/bootstrap/css/bootstrap.min.css">
<!--
=====-->
    <link rel="stylesheet" type="text/css" href="fonts/font-awesome-
4.7.0/css/font-awesome.min.css">
<!--
=====-->
```

```
<link rel="stylesheet" type="text/css" href="fonts/Linearicons-Free-v1.0.0/icon-font.min.css">
```

```
<!--
```

```
=====-->
```

```
<link rel="stylesheet" type="text/css"
href="vendor/animate/animate.css">
```

```
<!--
```

```
=====-->
```

```
<link rel="stylesheet" type="text/css" href="vendor/css-
hamburgers/hamburgers.min.css">
```

```
<!--
```

```
=====
```

```
-->
```

```
<link rel="stylesheet" type="text/css"
href="vendor/animation/css/animation.min.css">
```

```
<!--
```

```
=====
```

```
-->
```

```
<link rel="stylesheet" type="text/css"
href="vendor/select2/select2.min.css">
```

```
<!--
```

```
=====
```

```
-->
```

```
<link rel="stylesheet" type="text/css"
href="vendor/daterangepicker/daterangepicker.css">
```

```
<!--
```

```
=====
```

```
-->
```

```
<link rel="stylesheet" type="text/css" href="css/util.css">
```

```
<link rel="stylesheet" type="text/css" href="css/main.css">
```

```
<!--
=====
-->
</head>
<body>
    <div class="limiter">
        <div class="container-login100">
            <div class="wrap-login100">
<div class="login100-form-title" style="background-image: url(images/bg-
01.jpg);">

    <span class="login100-form-title-1">
        PT BANEEX INDONESIA
    </span>
</div>

<form action="" method="post" class="login100-form validate-form" >

<div class="wrap-input100 validate-input m-b-26" data-validate="Username is
required">

    <span class="label-input100">Username</span>

    <input class="input100" placeholder="Masukan Username" name="user"
    type="username" >
<span class="focus-input100"></span>

    </div>

<div class="wrap-input100 validate-input m-b-18" data-validate = "Password is
required">

    <span class="label-input100">Password</span>

    <input class="input100" placeholder="Masukan Password" name="pass"
    type="password">
```

```

        <span class="focus-input100"></span>
    </div>

    <div class="container-login100-form-btn">

        <button class="login100-form-btn" name="login">

            Login

        </button>

    </div>

</form>

<?php>

if(isset($_POST['login'])){

    $user    = $_POST['user'];

    $pass    = $_POST['pass'];

    if($user == "" || $pass == "") {

?>

<script type="text/javascript">alert("Username atau Password tidak boleh
kosong");</script>

<?php

    } else {

        $data_user = mysql_query("select * from user where username = '$user' and
        password = '$pass'");

        $r        = mysql_fetch_array($data_user);

        $username = $r['username'];

        $password = $r['password'];

        $jabatan  = $r ['jabatan'];

```

```

        if ($user == $username && $pass == $password) {

$_SESSION['jabatan'] = $jabatan;

header('location:../index.php');

        } else{

?>

<script type="text/javascript">alert("Username atau Password salah") ; </script>
<?php

}

}

}

?>

                </div>

            </div>

        </div>

<!--
=====-->

        <script src="vendor/jquery/jquery-3.2.1.min.js"></script>

<!--
=====-->

        <script src="vendor/animation/js/animation.min.js"></script>

<!--
=====-->

        <script src="vendor/bootstrap/js/popper.js"></script>
        <script src="vendor/bootstrap/js/bootstrap.min.js"></script>

<!--
=====-->

        <script src="vendor/select2/select2.min.js"></script>

```

```

<!--
=====-->
    <script src="vendor/daterangepicker/moment.min.js"></script>
    <script src="vendor/daterangepicker/daterangepicker.js"></script>
<!--
=====-->
    <script src="vendor/countdowntime/countdowntime.js"></script>
<!--
=====-->
    <script src="js/main.js"></script>
</body>
</html>

```

2. Index.php

```

<?php
    @session_start();
    include "koneksi.php";
    if(!isset($_SESSION['jabatan'])){
        header('location:Login/login.php');
    }
    ?>
<?php
    ob_start();
    require_once('config/+koneksi.php');
    // include "config/+koneksi.php";
    require_once('models/database.php');
    $connection = new Database($host, $user, $pass, $database);
    ?>
<!DOCTYPE html>
<html lang="en">

```

```
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1">
  <link rel="icon" type="image/png" sizes="16x16"
href="assets/img/logobb.png">
  <title>PT BANEEX INDONESIA</title>
  <!-- Bootstrap Core CSS -->
  <link href="assets/vendor/bootstrap/css/bootstrap.min.css"
rel="stylesheet">
  <!-- MetisMenu CSS -->
  <link href="assets/vendor/metisMenu/metisMenu.min.css"
rel="stylesheet">
  <!-- DataTables CSS -->
  <link href="assets/vendor/datatables-plugins/dataTables.bootstrap.css"
rel="stylesheet">
  <!-- DataTables Responsive CSS -->
  <link href="assets/vendor/datatables-responsive/dataTables.responsive.css" rel="stylesheet">
  <!-- Custom CSS -->
  <link href="assets/dist/css/sb-admin-2.css" rel="stylesheet">
  <!-- Morris Charts CSS -->
  <link href="assets/vendor/morrisjs/morris.css" rel="stylesheet">
  <!-- Custom Fonts -->
  <link href="assets/vendor/font-awesome/css/font-awesome.min.css"
rel="stylesheet" type="text/css">
  <!-- FullCalendar -->
  <link href="assets/vendor/fullcalendar/dist/fullcalendar.min.css"
rel="stylesheet">
```

```

    <link href="assets/vendor/fullcalendar/dist/fullcalendar.print.css"
rel="stylesheet" media="print">
    <link rel="stylesheet" type="text/css" href="Kirimpo/vendor/css-
hamburgers/hamburgers.min.css">
<!--
=====-->
    <link rel="stylesheet" type="text/css"
href="Kirimpo/vendor/ansition/css/ansition.min.css">
<!--
=====-->
    <link rel="stylesheet" type="text/css"
href="Kirimpo/vendor/select2/select2.min.css">
<!--
=====-->
    <link rel="stylesheet" type="text/css"
href="Kirimpo/vendor/daterangepicker/daterangepicker.css">
<!--
=====-->
    <link rel="stylesheet" type="text/css" href="Kirimpo/css/util.css">
    <link rel="stylesheet" type="text/css" href="Kirimpo/css/main.css">
    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file://
-->
    <!--[if lt IE 9]>
        <script
src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
        <script
src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></scrip
t>
    <![endif]-->

```

```

</head>
<body>
    <div id="wrapper">

        <!-- Navigation -->
        <nav class="navbar navbar-default navbar-static-top"
role="navigation" style="margin-bottom: 0">
            <div class="navbar-header">
                <a class="navbar-brand"
href="index.php?page=dashboard">SELAMAT DATANG</a>
            </div>
            <!-- /.navbar-header -->
            <div class="top-menu">
                <ul class="nav pull-right top-menu">
                    <li><a class="logout" href="logout.php">Log Out</a></li>
                </ul>
            </div>
            <div class="navbar-default sidebar" role="navigation">
                <div class="sidebar-nav navbar-collapse">
                    <ul class="nav" id="side-menu">
                        <li>
                            <a href="?page=dashboard"><i class="fa fa-home"></i>
Menu Utama</a>
                        </li>
                    </ul>
                </div>
                <?php
                    $jabatan = $_SESSION['jabatan'] == 'purchasing';
                    if ($jabatan) {
                        ?>
                        <li>
                            <a href="#"><i class="fa fa-tasks"></i> Data Master<span class="fa
arrow"></span></a>

```

```

<ul class="nav nav-second-level">
    <li>
        <a href="?page=data_user">Data User</a>
    </li>
    <li>
        <a href="?page=supplier">Data Supplier</a>
    </li>
    <li>
        <a href="?page=bahanbaku">Data Bahan Baku</a>
    </li>
</ul>
<!-- /.nav-second-level -->
</li>

```

```

<?php
}
?>
<?php
$jabatan = $_SESSION['jabatan'] == 'inventory';
if ($jabatan) {
    ?>

```

```

<li>
    <a href="#"><i class="fa fa-sitemap fa-fw"></i>
    Permintaan<span class="fa arrow"></span></a>

```

```

<ul class="nav nav-second-level">
    <?php
    $jabatan = $_SESSION['jabatan'] == 'inventory';
    if ($jabatan) {
        ?>
        <li>
            <a href="?page=p_bahanbaku">Permintaan

```

```

BB</a>

```

```

        </li>
        <li>
            <a href="?page=data_validasi">Permintaan Disetujui</a>
        </li>
        <li>
            <a href="?page=data_validasitolak">Permintaan Ditolak</a>
        </li>
        <li>
            <a href="?page=surat_jalan">Surat Jalan</a>
        </li>
    <?php
    }
?>
</ul>
<!-- /.nav-second-level -->
</li>
<?php
}
?>
<?php
$jabatan = $_SESSION['jabatan'] == 'purchasing';
if ($jabatan) {
    ?>
    <li>
        <a href="#"><i class="fa fa-files-o fa-fw"></i>
Pembelian <span class="fa arrow"></span></a>
        <ul class="nav nav-second-level">
            <li>
                <a href="?page=validasi_pbb">Validasi
Permintaan</a>
            </li>

```

```

        <li>
            <a href="?page=purchase_order">Purchase
Order</a>

        </li>
        <li>
            <a href="?page=data_po">Data Purchase Order</a>
        </li>
        <li>
            <a href="?page=kirim_po">Kirim Purchase
Order</a>

        </li>
    </ul>
    <!-- /.nav-second-level -->
</li>
<?php
}
?>
<?php
$jabatan = $_SESSION['jabatan'] == 'quality';
if ($jabatan) {
    ?>
    <li>
        <a href="#"><i class="fa fa-wrench"></i>
Pengecekan<span class="fa arrow"></span></a>
        <ul class="nav nav-second-level">
            <li>
                <a href="?page=cek_bahanbaku">Cek Bahan Baku</a>
            </li>
            <li>
                <a href="?page=data_cekbb">Data Cek Bahan Baku
Sesuai</a>

```

```
</li>
<li>
  <a href="?page=datacekbbtolak">Data Cek Bahan Baku Tidak
  Sesuai</a>
```

```
</li>
</ul>
<!-- /.nav-second-level -->
</li>
<?php
}
?>
</ul>
</div>
<!-- /.sidebar-collapse -->
</div>
<!-- /.navbar-static-side -->
</nav>
<?php
if (@$_GET['page'] == 'dashboard' || @$_GET['page'] == '') {
  include "views/dashboard.php" ;
} else if (@$_GET['page'] == 'data_user') {
  include "views/data_user.php" ;
} else if (@$_GET['page'] == 'supplier') {
  include "views/supplier.php" ;
} else if (@$_GET['page'] == 'bahanbaku') {
  include "views/bahanbaku.php" ;
} else if (@$_GET['page'] == 'p_bahanbaku') {
  include "views/p_bahanbaku.php" ;
} else if (@$_GET['page'] == 'cek_bahanbaku') {
  include "views/cek_bahanbaku.php" ;
} else if (@$_GET['page'] == 'surat_jalan') {
```

```
include "views/surat_jalan.php" ;
} else if (@$_GET['page'] == 'purchase_order') {
    include "views/purchase_order.php" ;
} else if (@$_GET['page'] == 'validasi_pbb') {
    include "views/validasi_pbb.php" ;
} else if (@$_GET['page'] == 'detailvalidasi_pbb') {
    include "views/detailvalidasi_pbb.php" ;
} else if (@$_GET['page'] == 'data_validasi') {
    include "views/data_validasi.php" ;
} else if (@$_GET['page'] == 'detaildatavalidasi') {
    include "views/detaildatavalidasi.php" ;
} else if (@$_GET['page'] == 'data_po') {
    include "views/data_po.php" ;
} else if (@$_GET['page'] == 'data_cekbb') {
    include "views/data_cekbb.php" ;
} else if (@$_GET['page'] == 'detailcekbb') {
    include "views/detailcekbb.php" ;
} else if (@$_GET['page'] == 'detaildatacekbb') {
    include "views/detaildatacekbb.php" ;
} else if (@$_GET['page'] == 'data_validasitolak') {
    include "views/data_validasitolak.php" ;
} else if (@$_GET['page'] == 'datacekbbtolak') {
    include "views/datacekbbtolak.php" ;
} else if (@$_GET['page'] == 'detailpo') {
    include "views/detailpo.php" ;
} else if (@$_GET['page'] == 'detailpopdf') {
    include "views/detailpo-pdf.php" ;
} else if (@$_GET['page'] == ' kirim_po') {
    include "views/kirim_po.php" ;
}
}
```

```

?>
</div><!-- /#page-wrapper -->
</div><!-- /#wrapper -->
<!-- jQuery -->
<script src="assets/vendor/jquery/jquery.min.js"></script>
<!-- Bootstrap Core JavaScript -->
<script src="assets/vendor/bootstrap/js/bootstrap.min.js"></script>
<!-- Metis Menu Plugin JavaScript -->
<script src="assets/vendor/metisMenu/metisMenu.min.js"></script>
<!-- Morris Charts JavaScript -->
<script src="assets/vendor/raphael/raphael.min.js"></script>
<script src="assets/vendor/morrisjs/morris.min.js"></script>
<script src="assets/data/morris-data.js"></script>
<!-- Custom Theme JavaScript -->
<script src="assets/dist/js/sb-admin-2.js"></script>
<!-- DataTables JavaScript -->
<script
src="assets/vendor/datatables/js/jquery.dataTables.min.js"></script>
<script src="assets/vendor/datatables-
plugins/dataTables.bootstrap.min.js"></script>
<script src="assets/vendor/datatables-
responsive/dataTables.responsive.js"></script>
<script>
$(document).ready(function() {
    $('#dataTables-example').DataTable({
        responsive: true
    });
});
</script>
</body>
</html>

```

3. Data_user.php

```
<?php
include "models/m_datauser.php";
$user = new User($connection);
if(@$_GET['act'] == ") {
?>

<div id="page-wrapper">
    <div class="row">
        <div class="col-lg-15">
            <h3 class="page-header">Data User</h3>
        </div>
        <!-- /.col-lg-12 -->
    </div>
    <div class="row">
        <div class="col-lg-15">
            <div class="table-responsive">
                <table class="table table-striped table-bordered table-hover"
id="dataTables-example">
                    <thead>
                        <tr>
                            <th align="center"></th>
                            <th> Nama Lengkap </th>
                            <th> Username </th>
                            <th> Password </th>
                            <th> Jabatan </th>
                            <th> opsi </th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php
                            $no = 1;
```

```

$stampil = $user->tampil();
while($data = $stampil->fetch_object())
{
?>
<tr>
    <td align="center"><?php echo $no++; ?></td>
    <td><?php echo $data->nama_lengkap; ?></td>
    <td><?php echo $data->username; ?></td>
    <td><?php echo $data->password; ?></td>
    <td><?php echo $data->jabatan; ?></td>
    <td align="center">
        <a id="edit_user" data-toggle="modal" data-target="#edit"
data-id="<?php echo $data->id_user; ?> " data-nama="<?php echo $data-
>nama_lengkap;?>" data-usern="<?php echo $data->username; ?>" data-
pwd="<?php echo $data->password; ?>" data-jbtn="<?php echo $data-
>jabatan; ?>" >
            <button class="btn btn-info btn-xs"><i class="fa fa-edit">
</i> </button>
            </a>
            <a href="?page=data_user&act=del&id=<?php echo $data-
>id_user; ?>" onclick="return confirm('Apakah anda yakin ingin
menghapus data ini')">
                <button class="btn btn-danger btn-xs"><i class="fa fa-trash-
o"> </i></button>
            </a>
        </td>
    </tr>
<?php
} ?>
</tbody>
</table>

```

```

</div>

<button type="button" class="btn btn-info" data-toggle="modal" data-
target="#tambah"><i class="fa fa-plus"></i> Tambah Data</button>

<div id="tambah" class="modal fade" role="dialog">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <h4 class="modal-title">Tambah Data User</h4>
      </div>
      <form action="" method="post" enctype="multipart/form-
data">
        <div class="modal-body">
          <div class="form-group">
            <label class="control-label" for="id_user"></label>
            <input type="hidden" name="id_user" class="form-
control" id="id_user" required="">
          </div>
          <div class="form-group">
            <label class="control-label" for="nama_lengkap"> Nama
Lengkap</label>
            <input type="text" name="nama_lengkap" class="form-control"
id="nama_lengkap" required="">
          </div>
          <div class="form-group">
            <label class="control-label"
for="username">Username</label>
            <input type="text" name="username" class="form-control"
id="username" required>

```

```
</div>
    <div class="form-group">
        <label class="control-label" for="password">Password</label>
        <input type="text" name="password" class="form-control" id="password"
required>
    </div>
    <div class="form-group">
        <label class="control-label"
for="jabatan">Jabatan</label>
        <input type="text" name="jabatan" class="form-control" id="jabatan"
required>
    </div>
</div>
    <div class="modal-footer">
        <button type="reset" class="btn btn-danger"> Bersih</button>
        <input type="submit" class="btn btn-success" name="tambah"
value="Simpan">
    </div>
</form>
<?php
if(@$_POST['tambah']) {
    $id_user= $connection->conn->real_escape_string($_POST['id_user']);
$nama_lengkap = $connection->conn-
>real_escape_string($_POST['nama_lengkap']);
$username = $connection->conn-
>real_escape_string($_POST['username']);
$password = $connection->conn-
>real_escape_string($_POST['password']);
$jabatan = $connection->conn->real_escape_string($_POST['jabatan']);
$user->tambah( $id_user, $nama_lengkap, $username, $password,
$jabatan);
```

```

header("location: ?page=data_user");
}
?>

</div>
</div>
</div>

<div id="edit" class="modal fade" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close" data-dismiss="modal">&times;</button>

<h4 class="modal-title">Edit User</h4>
</div>
<form action="" method="post" enctype="multipart/form-data">
<div class="modal-body" id="modal-edit">
<div class="form-group">
<label class="control-label" for="id_user"></label>
<input type="hidden" name="id_user" class="form-control" id="id_user"
required>

</div>
<div class="form-group">
<label class="control-label" for="nama_lengkap"> Nama
Lengkap</label> <input type="text" name="nama_lengkap" class="form-
control" id="nama_lengkap" required>
</div>
<div class="form-group">
<label class="control-label" for="username">Username</label>
<input type="text" name="username" class="form-control" id="username"
required>

```

```

        </div>
        <div class="form-group">
            <label class="control-label" for="password">Password</label>
            <input type="text" name="password" class="form-
control" id="password" required>
        </div>
        <div class="form-group">
            <label class="control-label" for="jabatan">Jabatan</label>
            <input type="text" name="jabatan" class="form-
control" id="jabatan" readonly>
        </div>
    </div>
    <div class="modal-footer">
        <input type="submit" class="btn btn-success" name="edit"
value="Simpan">
    </div>
</form>
<?php
    if(@$_POST['edit']) {
        $id_user= $connection->conn->real_escape_string($_POST['id_user']);
        $nama_lengkap = $connection->conn-
>real_escape_string($_POST['nama_lengkap']);
        $username = $connection->conn-
>real_escape_string($_POST['username']);
        $password = $connection->conn-
>real_escape_string($_POST['password']);
        $jabatan = $connection->conn->real_escape_string($_POST['jabatan']);
        $user->edit( $id_user, $nama_lengkap, $username, $password, $jabatan);
        header("location: ?page=data_user");
    }
?>

```

```

        </div>
    </div>
</div>
<!-- jQuery -->
<script src="assets/vendor/jquery/jquery.min.js"></script>
    <script type="text/javascript">
$(document).on("click", "#edit_user", function() { //karena yang di klik id-
edit maka namanya #edit_bbaku sesuai dengan id-edit di deklarasi atas
    var id_user = $(this).data('id');
    var nama_lengkap = $(this).data('nama');
    var username1 = $(this).data('usern');
    var password1 = $(this).data('pwd');
    var jabatan1 = $(this).data('jbtn');
    $("#edit #id_user").val(id_user);
    $(".modal-body #nama_lengkap").val(nama_lengkap);
    $(".modal-body #username").val(username1);
    $(".modal-body #password").val(password1);
    $(".modal-body #jabatan").val(jabatan1);
})

/*
$(document).ready(function(e) {
    $("#form").on("submit", (function(e) {
        e.preventDefault();
        $.ajax({
            url : 'models/proses_edit_suser.php',
            type : 'POST',
            data :new FormData(this),
            contentType : false,
            cache : false,
            processData : false,

```

```

        success : function(msg) {
            $('.table').html(msg);
        }
    });
    });
    }); /*
</script>
</div>
</div>
<?php
} elseif(@$_GET['act'] == 'del') {
    $user->hapus($_GET['id']);
    header("location: ?page=data_user");
}
?>

```

4. Bahanbaku.php

```

<?php
include "models/m_bahanbaku.php";
$bbaku = new bahanbaku($connection);
if(@$_GET['act'] == "") {
?>

<?php
$hasil = $bbaku->ambilkode();
$data = $hasil->fetch_array();
$kodecust = $data['kode'];
$noUrut = substr($kodecust, 4, 4);
$char = "BB-";
$newID = $char.sprintf("%03s" , $noUrut++ +1);

```

?>

```
<div id="page-wrapper">
  <div class="row">
    <div class="col-lg-15">
      <h3 class="page-header">Data Bahan Baku</h3>
    </div>
    <!-- /.col-lg-12 -->
  </div>
  <div class="row">
    <div class="col-lg-12">
      <div class="table-responsive">
        <table class="table table-striped table-bordered table-hover"
id="dataTables-example">
          <thead>
            <tr>
              <th align="center">No.</th>
              <th> Kode bahan baku </th>
              <th> Nama bahan baku </th>
              <th> Harga bahan baku </th>
              <th> Opsi </th>
            </tr>
          </thead>
          <tbody>
            <?php
              $no = 1;
              $stampil = $bbaku->tampil();
              while($data = $stampil->fetch_object())
              {
                ?>
                <tr>
                  <td align="center"><?php echo $no++; ?></td>
```

```

        <td><?php echo $data->kode_bb; ?></td>
        <td><?php echo $data->nama_bb; ?></td>
        <td>Rp. <?php echo number_format($data->hargabarang ,
0, ".", "."); ?></td>
        <td align="center">
            <a id="edit_bbaku" data-toggle="modal" data-target="#edit"
data-id="<?php echo $data->id_bb; ?> " data-kode="<?php echo $data-
>kode_bb;?>" data-nama="<?php echo $data->nama_bb; ?>" data-
harga="<?php echo $data->hargabarang; ?>" >
                <button class ="btn btn-info btn-xs"><i class="fa fa-edit">
</i> </button>
            </a>
            <a href="?page=bahanbaku&act=del&id=<?php echo $data-
>id_bb; ?>" onclick="return confirm('Yakin ingin menghapus data ini')">
                <button class ="btn btn-danger btn-xs"><i class ="fa fa-trash-
o"> </i> </button>
            </a>
        </td>
    </tr>
</tbody>
<?php
} ?>
</table>
</div>

<button type="button" class="btn btn-info" data-toggle="modal" data-
target="#tambah"><i class="fa fa-plus"></i>Tambah Data</button>
<div id="tambah" class="modal fade" role="dialog">
    <div class="modal-dialog">
        <div class="modal-content">
            <div class="modal-header">

```

```

        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <h4 class="modal-title">Tambah Data Bahan Baku</h4>
    </div>
    <form action="" method="post" enctype="multipart/form-
data">
        <div class="modal-body">
            <div class="form-group">
                <label class="control-label" for="id_bb"></label>
                <input type="hidden" name="id_bb" class="form-
control" id="id_bb" value="<?php echo $newID; ?>" required>
            </div>
            <div class="form-group">
                <label class="control-label" for="kode_bb">Kode
Bahan Baku</label>
                <input type="text" name="kode_bb" class="form-
control" id="kode_bb" required>
            </div>
            <div class="form-group">
                <label class="control-label" for="nama_bb">Nama
Bahan Baku</label>
                <input type="text" name="nama_bb" class="form-
control" id="nama_bb" required>
            </div>
            <div class="form-group">
                <label class="control-label" for="hargabarang">Harga
Barang</label>
                <input type="text" name="hargabarang" class="form-
control" id="hargabarang" required>
            </div>
        </div>
    </div>

```

```

        <div class="modal-footer">
            <button type="reset" class="btn btn-danger">
Bersih</button>
            <input type="submit" class="btn btn-success"
name="tambah" value="Simpan">
        </div>
    </form>

    <?php
        if(@$_POST['tambah']) {
            $id_bb = $connection->conn->real_escape_string($_POST['id_bb']);
            $kode_bb = $connection->conn->real_escape_string($_POST['kode_bb']);
            $nama_bb = $connection->conn->real_escape_string($_POST['nama_bb']);
            $hargabarang = $connection->conn->real_escape_string($_POST['hargabarang']);
            $bbaku->tambah( $id_bb, $kode_bb, $nama_bb, $hargabarang);

            header("location: ?page=bahanbaku");
        }
    ?>

    </div>
</div>
</div>
    <div id="edit" class="modal fade" role="dialog">
        <div class="modal-dialog">
            <div class="modal-content">
                <div class="modal-header">
                    <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <h4 class="modal-title">Edit Data Bahan Baku</h4>
            </div>

```

```

        <form action="" method="post"
enctype="multipart/form-data">
        <div class="modal-body" id="modal-edit">
        <div class="form-group">
            <label class="control-label" for="id_bb"></label>
<input type="hidden" name="id_bb" class="form-control" id="id_bb"
value="<?php echo $newID; ?>" required >
        </div>
        <div class="form-group">
            <label class="control-label" for="kode_bb">Kode Bahan
Baku</label>
            <input type="text" name="kode_bb" class="form-
control" id="kode_bb" required="">
        </div>
        <div class="form-group">
            <label class="control-label" for="nama_bb">Nama Bahan Baku
</label>
            <input type="text" name="nama_bb" class="form-
control" id="nama_bb" required>
        </div>
        <div class="form-group">
            <label class="control-label" for="hargabarang">Harga Barang</label>
            <input type="text" name="hargabarang" class="form-
control" id="hargabarang" required>
        </div>
        </div>
        <!-- </div> -->
        <div class="modal-footer">
<input type="submit" class="btn btn-primary" name="edit" value="Simpan">

        </div>
    </form>

```

```

        <?php
            if(@$_POST['edit']) {
                $id_bb = $connection->conn->real_escape_string($_POST['id_bb']);
                $kode_bb = $connection->conn->real_escape_string($_POST['kode_bb']);
                $nama_bb = $connection->conn->real_escape_string($_POST['nama_bb']);
                $hargabarang = $connection->conn->real_escape_string($_POST['hargabarang']);

                $bbaku->edit($id_bb, $kode_bb, $nama_bb, $hargabarang);

                header("location: ?page=bahanbaku");
            }
        ?>

```

```

        </div>
    </div>
</div>

```

```

<script src="assets/vendor/jquery/jquery.min.js"></script>
<script type="text/javascript">
    $(document). on("click", "#edit_bbaku", function() { //karena
yang di klik id-edit maka namanya #edit_bbaku sesuai dengan id-edit di
deklarasi atas

```

```

        var idbb=$(this).data('id');
        var kodebb=$(this).data('kode');
        var namabb=$(this).data('nama');
        var hargabb=$(this).data('harga');
        $("#edit #id_bb").val(idbb);
        $(".modal-body #kode_bb").val(kodebb);
        $(".modal-body #nama_bb").val(namabb);
        $(".modal-body #hargabarang").val(hargabb);

```

```

    })

    /*
$(document).ready(function(e) {
    $("#form").on("submit", (function(e) {
        e.preventDefault();
        $.ajax({
            url : 'models/proses_edit_supplier.php',
            type : 'POST',
            data :new FormData(this),
            contentType : false,
            cache : false,
            processData : false,
            success : function(msg) {
                $('.table').html(msg);
            }
        });
    }));
}); */
</script>
</div>
</div>

<?php
} elseif(@$_GET['act'] == 'del') {
    $bbaku->hapus($_GET['id']);
    header("location: ?page=bahanbaku");
}
?>

```

LAMPIRAN C

WAWANCARA

Pewawancara : Bagaimana proses pembelian bahan baku di PT Baneex Indonesia?

Narasumber : Proses pembelian bahan baku di PT Baneex Indonesia masih menggunakan excel yang diawali dengan divisi *Inventory* membuat surat permintaan material yang ditujukan kepada divisi *Purchasing* kemudian setelah surat permintaan material diterima oleh divisi *Purchasing*, maka divisi *Purchasing* akan menerbitkan *purchase order* yang ditujukan kepada *supplier*.

Pewawancara : Bagaimana maksud No Purchase Order yang tertera di form PO tersebut?

Narasumber : No yang paling depan adalah urutan nomernya, kemudian MZ adalah kode pemasoknya, setelahnya SP adalah singkatan dari kata *supplier*, kemudian angka romawi melambangkan bulan saat melakukan *purchase order* dan angka 17 di akhir adalah tahun berapa *purchase order* dibuat.

Pewawancara : Berapa kali *purchase order* keluar dari baneex ke *supplier*?

Narasumber : Kalau untuk itu tergantung kebutuhan materialnya.

Pewawancara : Bagaimana proses penerimaan bahan baku di PT Baneex Indonesia?

Narasumber : Awalnya bahan baku datang kemudian *supplier* memberikan surat jalan ke divisi *Inventory* kemudian yang mengecek materialnya adalah divisi *Quality Control*.

Pewawancara : Bagaimana dengan penyimpanan dokumen-dokumen yang ada, apakah sudah terarsip dengan baik?

Narasumber : Belum terarsip dengan baik. Karena data masih disimpan secara manual.