

**RANCANG BANGUN SISTEM INFORMASI
PEMBELIAN *SPARE PART* MESIN *STAMPING*
MENGUNAKAN CODEIGNITER 3.1.4 DAN MYSQL
5.6.20 DENGAN FITUR BOT PADA APLIKASI PESAN
TELEGRAM DI PT NUSA INDAH JAYA UTAMA**

TUGAS AKHIR

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

**OLEH
RUSDIANSYAH
1314039**



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

ABSTRAK

PT Nusa Indah Jaya Utama merupakan sebuah industri manufaktur yang bergerak dibidang otomotif dengan melakukan produksi komponen-komponen otomotif baik roda dua maupun roda empat, PT Nusa Indah Jaya Utama menggunakan beberapa mesin stamping dengan kapasitas tekanan berbeda-beda. Divisi *purchasing* bertanggung jawab terhadap segala pembelian yang dilakukan perusahaan. Kegiatan pembelian *spare part* mesin *stamping* dilakukan secara manual dengan nota pembelian lalu proses pemesanan barang dilakukan melalui telepon tanpa menggunakan dokumen *purchase order*, proses validasi atau penyetujuan dilakukan secara manual yang dapat menghambat proses tersebut jika manajer tidak berada ditempat, perusahaan juga tidak memiliki data *spare part* dan data *supplier* yang dibutuhkan dalam melakukan proses pembelian. Sistem informasi pembelian *spare part* mesin *stamping* ditujukan untuk membantu proses pembelian sehingga dapat terintegrasi antar bagian serta penyimpanan dokumen secara terkomputerisasi yang dapat memudahkan proses pengelolaan informasi. Pengembangan sistem yang dilakukan adalah menggunakan metode *waterfall*. Pemodelan sistem menggunakan *Unified Modelling Language* (UML). Pemodelan data menggunakan *Entity Relationship Diagram* (ERD), *Conceptual Data Model* (CDM), dan kamus data. Perancangan sistem menggunakan *Windows Navigation Diagram* (WND) dan antarmuka pengguna. Pengujian sistem menggunakan *Functional Testing*. Pembuatan aplikasi menggunakan CodeIgniter 3.1.4 sebagai *framework* PHP dan MySQL 5.6.20 sebagai *database*. Pembuatan fitur tambahan bot pada aplikasi pesan Telegram menggunakan PHP 5.5.15 sebagai bahasa pemrograman *engine* bot. Aplikasi sistem informasi pembelian *spare part* mesin *stamping* memiliki *database* dan menyediakan fungsi pengelolaan data dan penyimpanan dokumen serta proses validasi atau penyetujuan permintaan pembelian secara *mobile* melalui aplikasi pesan Telegram. Masih terdapat banyak ruang untuk pengembangan aplikasi sistem informasi pembelian *spare part* mesin *stamping* ini antara lain membuat fitur pembayaran kepada *supplier* dan membuat fitur pemilihan *supplier*, serta pengembangan integrasi antara aplikasi pesan Telegram dan aplikasi sistem informasi pembelian *spare part* mesin *stamping*, agar dapat dilakukan pengolahan data melalui aplikasi pesan Telegram.

Kata Kunci : sistem informasi, pembelian *spare part* mesin *stamping*, metode *waterfall*, CodeIgniter, MySQL, bot Telegram.

DAFTAR ISI

	Halaman
Halaman Judul.....	i
Lembar Persetujuan Dosen Pembimbing.....	ii
Lembar Bimbingan Praktik Tugas Akhir.....	iii
Lembar Pernyataan Keaslian.....	iv
Abstrak	v
Kata Pengantar	vi
Daftar Isi.	viii
Daftar Gambar.	xii
Daftar Tabel	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Pokok Permasalahan.....	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah.....	4
1.5 Manfaat Tugas Akhir.....	4
1.6 Sistematika Penulisan	5
BAB II LANDASAN TEORI	7
2.1 Rancang Bangun	7
2.2 Sistem.....	7
2.2.1 Karakteristik Sistem.....	8
2.2.2 Jenis Sistem.....	11
2.3 Informasi	12
2.3.1 Siklus dan Fungsi Informasi.....	13
2.3.2 Nilai Informasi	14
2.3.3 Kualitas Informasi	16
2.4 Sistem Informasi	18
2.5 Pembelian.....	20

2.5.1	Tugas dan Tanggung Jawab Divisi Purchasing.....	21
2.5.2	Proses Pembelian.....	21
2.5.3	Tujuan Pembelian.....	24
2.6	<i>Spare Part</i>	25
2.7	<i>System Development Life Cycle (SDLC)</i>	26
2.8	<i>Model Waterfall</i>	29
2.9	<i>System Requirements</i>	30
2.10	<i>Flowchart</i>	31
2.11	<i>Unified Modelling Language (UML)</i>	32
2.11.1	<i>Use Case Diagram</i>	34
2.11.2	<i>Activity Diagram</i>	35
2.11.3	<i>Sequence Diagram</i>	37
2.11.4	<i>Class Diagram</i>	39
2.11.5	<i>Deployment Diagram</i>	40
2.12	<i>Database</i>	41
2.13	<i>Entity Relationship Diagram (ERD)</i>	42
2.14	<i>Conceptual Data Model (CDM)</i>	44
2.15	Kamus Data	47
2.16	<i>Windows Navigation Diagram</i>	47
2.17	PHP	48
2.18	<i>CodeIgniter</i>	49
2.19	MySQL.....	50
2.19.1	Keunggulan MySQL	50
2.19.2	Tipe Data MySQL	52
2.20	Bot Telegram.....	53
2.21	<i>Functional Testing</i>	54

BAB III METODOLOGI PENELITIAN 55

3.1	Metodologi 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	Tentang Perusahaan	62
4.2	Profil Perusahaan	64
4.3	Visi dan Misi Perusahaan.....	64
4.4	Struktur Organisasi Perusahaan	65
4.5	<i>Job Description</i>	67
4.6	Mesin-mesin Produksi Perusahaan	69
4.7	<i>Spare Part</i>	70
4.8	Dokumen Pembelian <i>Spare Part</i>	73
4.9	Prosedur Pembelian <i>Spare Part</i>	76
BAB V ANALISIS DAN PEMBAHASAN.....		78
5.1	Analisis Kebutuhan Pengguna	78
5.2	Prosedur Usulan Sistem Informasi Pembelian Sparepart Mesin Stamping	81
5.3	Analisis Sistem Usulan	83
5.3.1	<i>Use Case Diagram</i>	83
5.3.2	<i>Activity Diagram</i>	92
5.3.3	<i>Sequence Diagram</i>	100
5.3.4	<i>Class Diagram</i>	108
5.3.5	<i>Deployment Diagram</i>	109
5.4	Pemodelan Data Sistem Usulan	110
5.4.1	<i>Entity Relationship Diagram (ERD)</i>	110
5.4.2	<i>Conceptual Data Model (CDM)</i>	111
5.4.3	Kamus Data.....	112
5.5	Perancangan Sistem	117
5.5.1	<i>Windows Navigation Diagram (WND)</i>	117
5.5.2	Perancangan Antarmuka Pengguna.....	119

5.5.3	Fitur Bot Telegram.....	135
5.6	Implementasi Sistem <i>Software</i> dan <i>Hardware</i>	141
5.7	<i>Functional Testing</i>	141
BAB IV	KESIMPULAN DAN SARAN.....	149
6.1	Kesimpulan	149
6.2	Saran.....	149
	DAFTAR PUSTAKA.....	151

DAFTAR GAMBAR

	Halaman
Gambar II.1 Gambaran Sistem Terbuka	12
Gambar II.2 Gambaran Sistem Tertutup	12
Gambar II.3 Siklus Informasi.....	14
Gambar II.4 Pilar kualitas informasi.....	17
Gambar II.5 Alur Proses Pembelian.....	24
Gambar II.6 Penggambaran Metode <i>Waterfall</i>	30
Gambar II.7 UML Diagram	33
Gambar II.8 Contoh <i>Windows Navigation Diagram</i>	48
Gambar III.1 Kerangka Penelitian	60
Gambar IV.1 Lambang PT Nusa Indah Jaya Utama.....	63
Gambar IV.2 Struktur Organisasi PT Nusa Indah Jaya Utama.....	66
Gambar IV.3 Hasil Analisis Kerusakan Oleh Divisi <i>Maintenance</i>	73
Gambar IV.4 Nota Permintaan Pembelian.....	74
Gambar IV.5 Surat Penawaran <i>SparePart</i>	75
Gambar IV.6 Surat Jalan	75
Gambar IV.7 Aliran Data Sistem Informasi Pembelian <i>Spare Part</i>	77
Gambar V.1 <i>Flowmap</i> usulan	82
Gambar V.2 <i>Use Case Diagram</i> usulan.....	84
Gambar V.3 <i>Activity Diagram Login</i>	92
Gambar V.4 <i>Activity Diagram</i> Mengelola Data master	93
Gambar V.5 <i>Activity Diagram</i> Permintaan Pembelian	94
Gambar V.6 <i>Activity Diagram</i> Penyetujuan Permintaan Pembelian	95
Gambar V.7 <i>Activity Diagram</i> Membuat PO.....	96
Gambar V.8 <i>Activity Diagram</i> Mencetak PO	97
Gambar V.9 <i>Activity Diagram</i> Mengirim PO	98
Gambar V.10 <i>Activity Diagram</i> Kelola Surat Jalan.....	99
Gambar V.11 <i>Activity Diagram</i> Kelola Laporan Pembelian	100

Gambar V.12 <i>Sequence Diagram Login</i>	101
Gambar V.13 <i>Sequence Diagram</i> Mengelola Data Master.....	102
Gambar V.14 <i>Sequence Diagram</i> Membuat Permintaan Pembelian	103
Gambar V.15 <i>Sequence Diagram</i> Validasi	104
Gambar V.16 <i>Sequence Diagram</i> Mengelola PO	105
Gambar V.17 <i>Sequence Diagram</i> Cetak PO	106
Gambar V.18 <i>Sequence Diagram</i> Mengirim PO	106
Gambar V.19 <i>Sequence Diagram</i> Kelola Surat Jalan	107
Gambar V.20 <i>Sequence Diagram</i> Kelola Laporan	108
Gambar V.21 <i>Class Diagram</i> Sistem Usulan	109
Gambar V.22 <i>Deployment Diagram</i>	110
Gambar V.23 <i>Entity Relationship Diagram</i>	111
Gambar V.24 <i>Conceptual Data Model</i>	112
Gambar V.25 <i>Windows Navigation Diagram</i>	118
Gambar V.26 <i>Form Login</i>	119
Gambar V.27 Tampilan Halaman Utama	120
Gambar V.28 Tampilan Halaman Data Master <i>Supplier</i>	120
Gambar V.29 Tampilan <i>Form Input</i> Data Master <i>Supplier</i>	121
Gambar V.30 Tampilan Halaman Data Master <i>Spare part</i>	121
Gambar V.31 Tampilan <i>Form Input</i> Data Master <i>Spare part</i>	122
Gambar V.32 Tampilan Halaman Buat Permintaan	122
Gambar V.33 Tampilan <i>Form</i> Permintaan Pembelian.....	123
Gambar V.34 Tampilan <i>Form</i> Konfirmasi Permintaan Pembelian	123
Gambar V.35 Tampilan Detail Permintaan Pembelian.....	124
Gambar V.36 Tampilan Halaman Permintaan Ditolak	124
Gambar V.37 Tampilan Halaman Permintaan Disetujui	125
Gambar V.38 Tampilan Halaman Detail Permintaan Disetujui.....	125
Gambar V.39 Tampilan Halaman Penyetujuan Permintaan Pembelian	126
Gambar V.40 Tampilan Halaman Detail Penyetujuan Permintaan Pembelian...	126
Gambar V.41 Tampilan Form Alasan Tolak Permintaan	127
Gambar V.42 Tampilan Halaman Permintaan Disetujui	127

Gambar V.43 Tampilan Halaman Detail Permintaan Disetujui.....	128
Gambar V.44 Tampilan Halaman Buat PO	128
Gambar V.45 Tampilan <i>Form</i> Buat PO	129
Gambar V.46 Tampilan <i>Form</i> PO Detail	129
Gambar V.47 Tampilan <i>Form</i> Konfirmasi PO	130
Gambar V.48 Tampilan Halaman Cetak	130
Gambar V.49 Tampilan Detail PO Keluar	131
Gambar V.50 <i>Purchase Order</i>	131
Gambar V.51 Halaman Kirim PO	132
Gambar V.52 Halaman Surat Jalan	132
Gambar V.53 Halaman <i>Input</i> Surat Jalan	133
Gambar V.54 <i>Form</i> Konfirmasi <i>Input</i> Surat Jalan	133
Gambar V.55 <i>Form</i> Konfirmasi Detail Surat Jalan	134
Gambar V.56 <i>Form</i> Cetak Laporan	134
Gambar V.57 Laporan Pembelian <i>Spare part</i>	135
Gambar V.58 Profil Bot	136
Gambar V.59 Interaksi Awal Bot.....	137
Gambar V.60 Perintah /pending.....	137
Gambar V.61 Perintah /valid.....	138
Gambar V.62 Perintah /tolak.....	138
Gambar V.63 Perintah /detail.....	139
Gambar V.64 Perintah /detail_rusak	139
Gambar V.65 Perintah /validasi	140
Gambar V.66 Perintah /status	140
Gambar V.67 Notifikasi Permintaan Pembelian.....	141

DAFTAR TABEL

	Halaman
Tabel II.1 Contoh <i>Input-Process-Output</i>	8
Tabel II.2 Simbol-simbol <i>Flowchart</i>	31
Tabel II.3 Simbol-Simbol <i>Use case Diagram</i>	34
Tabel II.4 Simbol-Simbol <i>Activity Diagram</i>	35
Tabel II.5 Simbol-Simbol <i>Sequence Diagram</i>	37
Tabel II.6 Simbol Jenis-jenis <i>Class</i> pada <i>Sequence Diagram</i>	39
Tabel II.7 Simbol-Simbol <i>Class Diagram</i>	39
Tabel II.8 Simbol-Simbol <i>Deployment Diagram</i>	41
Tabel II.9 Simbol-simbol <i>Entity Relationship Diagram</i>	42
Tabel II.10 Simbol-Simbol <i>Conceptual Data Model (CDM)</i>	44
Tabel II.11 Simbol-Simbol Aturan <i>Conceptual Data Model (CDM)</i>	44
Tabel II.12 Contoh Kamus Data Untuk Tabel Pemasok.....	47
Tabel II.13 Jenis Data pada MySQL.....	52
Tabel IV.1 Data Mesin PT Nusa Indah Jaya Utama	70
Tabel IV.2 Data <i>Spare part</i> PT Nusa Jaya Indah Utama	71
Tabel V.1 <i>System Requirement</i>	78
Tabel V.2 Definisi Aktor.....	85
Tabel V.3 Definisi <i>Use case</i>	85
Tabel V.4 Deskripsi <i>Use case</i> Login	89
Tabel V.5 Deskripsi <i>Use case</i> Mengelola Data Master	87
Tabel V.6 Deskripsi <i>Use case</i> Mengelola Permintaan Pembelian	87
Tabel V.7 Deskripsi <i>Use case</i> Penyetujuan Permintaan Pembelian	88
Tabel V.8 Deskripsi <i>Use case</i> Mengelola PO	89
Tabel V.9 Deskripsi <i>Use case</i> Mencetak PO	90
Tabel V.10 Deskripsi <i>Use case</i> Mengirim PO	90
Tabel V.11 Deskripsi <i>Use case</i> Mengelola Surat Jalan	91
Tabel V.12 Deskripsi <i>Use case</i> Mencetak <i>Purchase Order</i>	91

Tabel V.13 Spesifikasi Tabel User.....	113
Tabel V.14 Spesifikasi Tabel <i>Spare part</i>	113
Tabel V.15 Spesifikasi Tabel <i>Supplier</i>	114
Tabel V.16 Spesifikasi Tabel Permintaan.....	114
Tabel V.17 Spesifikasi Tabel Permintaan Detail	115
Tabel V.18 Spesifikasi Tabel Surat.....	115
Tabel V.19 Spesifikasi Tabel Surat Detail	116
Tabel V.20 Spesifikasi Tabel <i>Purchase Order</i>	116
Tabel V.21 Spesifikasi Tabel <i>Purchase Order</i> Detail	117
Tabel V.22 <i>Functional Test Use Case Login</i>	142
Tabel V.23 <i>Functional Test Use Case</i> Mengelola Data <i>Supplier</i>	143
Tabel V.24 <i>Functional Test Use Case</i> Mengelola Data <i>Spare Part</i>	144
Tabel V.25 <i>Functional Test Use Case</i> Mengelola Permintaan Pembelian	145
Tabel V.26 <i>Functional Test Use Case</i> Penyetujuan Permintaan Pembelian ...	145
Tabel V.27 <i>Functional Test Use Case</i> Mengelola PO	146
Tabel V.28 <i>Functional Test Use Case</i> Mencetak PO	147
Tabel V.29 <i>Functional Test Use Case</i> Kirim PO.....	147
Tabel V.30 <i>Functional Test Use Case</i> Mengelola Surat Jalan.....	148
Tabel V.31 <i>Functional Test Use Case</i> Mengelola Laporan Pembelian	148

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dunia saat ini memiliki teknologi informasi yang sangat maju bila dibandingkan dengan sepuluh tahun yang lalu. Ini menandakan perkembangan teknologi informasi yang sangat pesat di seluruh dunia, perkembangan teknologi ini ditandai dengan semakin mudah dan cepatnya penyebaran maupun akses terhadap informasi. Perkembangan teknologi informasi tidak dapat dilepaskan dari perkembangan sistem informasi yang merupakan sebuah kombinasi pemanfaatan teknologi informasi tersebut. Perkembangan teknologi informasi dan selanjutnya sistem informasi sangat mempengaruhi kehidupan manusia baik dari segi ekonomi, politik, sosial, dan budaya. Dengan pemanfaatan yang baik perkembangan sistem informasi dapat memberikan kemudahan dalam segala sendi kehidupan manusia. Dalam dunia bisnis atau ekonomi pemanfaatan sistem informasi dapat memberikan sebuah kemudahan dalam hal integrasi informasi pada sebuah perusahaan.

Sebuah perusahaan memerlukan adanya sebuah integrasi informasi untuk memudahkan perusahaan dalam melakukan proses bisnisnya agar dapat tetap bertahan di dalam persaingan dunia bisnis yang semakin ketat. Untuk itu, diperlukan sebuah pemanfaatan sistem informasi yang dapat membantu efektifitas peredaran arus informasi dalam sebuah perusahaan, dengan pemanfaatan sistem informasi yang baik sebuah perusahaan dapat mengintegrasikan informasi antar divisi di dalam perusahaannya sehingga terbentuk sebuah kesatuan informasi yang dapat menunjang proses bisnis perusahaan, hal ini dapat menguntungkan perusahaan dari segi efektifitas kerja divisi di dalam perusahaan sehingga hasil produksi perusahaan dapat meningkat. Hasil produksi sebuah perusahaan akan sangat bergantung terhadap divisi-divisi pendukungnya, seperti informasi

kecacatan barang yang diproduksi, informasi perbaikan mesin maupun informasi pembelian yang dilakukan divisi *purchasing*.

Divisi *purchasing* merupakan sebuah divisi yang bertanggung jawab terhadap segala pembelian yang dibutuhkan perusahaan demi terpenuhinya kebutuhan dalam menjalankan proses produksi perusahaan. Dalam menjalankan sebuah proses pembelian divisi *purchasing* membutuhkan sebuah landasan informasi untuk mengeksekusi proses pembelian, seperti informasi permintaan pembelian *spare part* mesin yang diminta oleh divisi *maintenance*, informasi permintaan pembelian ini harus berisi detail yang dibutuhkan agar proses pembelian dapat berjalan dengan baik seperti nama *spare part*, ukuran *spare part*, spesifikasi *spare part* dan lain sebagainya.

Salah satu perusahaan menengah di Indonesia bernama PT Nusa Indah Jaya Utama merupakan sebuah perusahaan yang bergerak dibidang otomotif yang memproduksi *stamping parts* untuk kendaraan roda empat dan roda dua. Dalam struktur organisasi PT Nusa Indah Jaya Utama terdapat sebuah divisi yang bertugas melakukan berbagai macam pembelian yang bernama divisi *purchasing*. Divisi ini bertanggung jawab terhadap semua proses pembelian kebutuhan perusahaan dari bahan baku hingga *spare part* mesin produksi. Dalam menjalankan proses pembelian divisi *purchasing* masih menggunakan cara manual yaitu menggunakan kertas nota yang tidak memiliki detail yang baik untuk melakukan pembelian sehingga dapat menimbulkan kesalahan dalam proses pembelian. Proses validasi atau penyetujuan permintaan juga dilakukan secara manual oleh manajer *purchasing* hal ini dapat memperlambat proses pembelian jika manajer *purchasing* tidak berada ditempat untuk melakukan validasi. Dalam hubungan pembelian kepada *supplier* PT Nusa Indah Jaya Utama tidak memiliki dokumen *purchase order* selain itu divisi *purchasing* PT Nusa Indah Jaya Utama juga tidak melakukan pendataan terhadap *spare part* yang dibutuhkan perusahaan serta tidak melakukan pendataan terhadap *supplier* yang bekerja sama dengan perusahaan sehingga hal-hal ini dapat memperlambat proses pembelian yang berdampak pada terhambatnya proses produksi perusahaan.

Berdasarkan latar belakang dan permasalahan tersebut, maka disusun tugas akhir dengan judul “RANCANG BANGUN SISTEM INFORMASI PEMBELIAN *SPARE PART* MESIN *STAMPING* MENGGUNAKAN CODEIGNITER 3.1.4 DAN MYSQL 5.6.20 DENGAN FITUR BOT PADA APLIKASI PESAN TELEGRAM DI PT NUSA INDAH JAYA UTAMA” untuk membantu perusahaan dalam proses pembelian *spare part* mesin *stamping* sehingga proses pembelian *spare part* mesin *stamping* perusahaan dapat berjalan lebih baik.

1.2 Pokok Permasalahan

Berdasarkan latar belakang yang dipaparkan sebelumnya, maka dapat diidentifikasi permasalahan-permasalahan yang terjadi di PT Nusa Indah Jaya Utama adalah sebagai berikut:

1. Dokumen nota pembelian sebagai *purchase requisition* tidak memiliki detail yang baik sehingga dapat menjadi celah kekeliruan dalam pembelian *spare part* oleh divisi *purchasing* dan perusahaan tidak memiliki *purchase order* atau surat pemesanan yang ditujukan kepada *supplier* yang dapat menyebabkan terjadinya kekeliruan jika pemesanan hanya dilakukan melalui kontak telepon, pembelian juga menjadi tidak terdokumentasi dengan baik karena tidak terdapatnya *purchase order*.
2. Tidak terdapat pendataan *spare part* dan pendataan *supplier* yang baik sehingga proses pembelian dilakukan berulang-ulang dengan mencari kontak *supplier* melalui kartu nama sehingga memperlambat proses pembelian.
3. Proses validasi atau penyetujuan dilakukan secara manual yang dapat memperlambat proses pembelian karena seringkali manajer *purchasing* tidak berada di perusahaan.

1.3 Tujuan Penelitian

Tujuan penelitian dari tugas akhir ini yang dilakukan terhadap divisi *purchasing* PT Nusa Indah Jaya Utama dalam melakukan pembelian *spare part*

mesin *stamping* adalah untuk merancang dan membangun sistem informasi pembelian *spare part* mesin *stamping* yang mampu:

1. Mengintegrasikan antar bagian khususnya bagian yang terlibat dalam proses pembelian *spare part* mesin *stamping* yaitu divisi *purchasing* dan divisi *maintenance*, agar terdapat integrasi atau kesatuan informasi antar divisi dan meminimalisir kesalahan yang mungkin terjadi saat proses pembelian berjalan dengan menyediakan *form* permintaan pembelian dan *purchase order* yang baik sehingga proses pembelian menjadi lebih terstruktur dan proses produksi perusahaan berjalan lancar.
2. Menyediakan fasilitas penyimpanan dokumen-dokumen pembelian dan data master yang berhubungan dengan pembelian menggunakan database agar dapat disimpan di satu lokasi.
3. Menyediakan fitur validasi yang bersifat *mobile* dan dapat digunakan dimanapun serta terhubung langsung dengan *database* pembelian perusahaan sehingga dapat membantu proses pembelian.

1.4 Batasan masalah

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

1. Tempat melakukan penelitian adalah di PT Nusa Indah Jaya Utama.
2. Penelitian dilakukan pada divisi *purchasing* selama satu bulan terhitung sejak tanggal 7 Agustus 2017 sampai dengan 8 September 2017.
3. Ruang lingkup yang diamati sebatas pada divisi *purchasing* dalam proses pembelian *spare part* mesin *stamping* hingga proses penerimaan *spare part* di tangan perusahaan namun tidak membahas proses pemilihan *supplier*, pengembalian *spare part* dan pembayaran yang dilakukan perusahaan.

1.5 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 berjalan di sebuah perusahaan dan diharapkan dapat memberikan solusi terhadap permasalahan yang terjadi di dalam 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.

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada di dalamnya. Adapun sistematika penulisan Tugas Akhir ini 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 tugas akhir ini adalah seputar rancang bangun, konsep dasar sistem, sistem informasi, konsep dasar pembelian, konsep pengembangan sistem, *tools* pemodelan sistem dengan *Unified Modelling Language* (UML), *Entity Relationship Diagram* (ERD), *Conceptual Data Model*, Kamus Data, *Windows Navigation Diagram*, PHP, CodeIgniter, MySQL, dan Bot Telegram.

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 Nusa Indah Jaya Utama, meliputi profil perusahaan, struktur organisasi, proses bisnis sistem pembelian *spare part* mesin *stamping* 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 *Unified Modelling Language* (UML). Pemodelan data dengan *Entity Relationship Diagram* (ERD), *Conceptual Data Model*, dan Kamus Data. Perancangan sistem menggunakan *Windows Navigation Diagram*, perancangan antarmuka pengguna. Pembuatan spesifikasi sistem yang diperlukan dan perancangan bot Telegram.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran dalam penerapan sistem informasi pembelian *spare part* mesin *stamping* untuk perusahaan dalam penelitian dan pengembangan selanjutnya.

BAB II

LANDASAN TEORI

2.1 Rancang Bangun

Menurut McLeod (2011) rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisis dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan. Rancangan sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru.

Perancangan adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik. Sedangkan pengertian bangun atau pembangunan sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian. Bangun sistem adalah membangun sistem informasi dan komponen yang didasarkan pada spesifikasi desain. Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil analisis ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut ataupun memperbaiki sistem yang sudah ada.

2.2 Sistem

Dalam pengertian umum, sebuah sistem adalah sekumpulan benda yang memiliki hubungan di antara mereka. Dengan kata lain sistem juga merupakan sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Dalam bukunya Djahir dan Pratita (2014) mengutip beberapa pendapat ahli sebagai berikut:

1. Sistem adalah kumpulan atau grup dari subsistem atau bagian atau komponen apapun baik fisik ataupun nonfisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu. (Susanto, 2002: 18).

2. Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. McLeod dan Schell (2004: 9).
3. Menurut Von Bartalanfy, sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi di antara unsur-unsur tersebut dengan lingkungan.
4. Menurut Raporot, sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain.

Lebih lanjut menurut Puspitawati dan Anggadini (2011), Pendekatan sistem yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai berikut:

sistem adalah kumpulan dari elemen-elemen untuk mencapai suatu tujuan tertentu. Sesuatu dapat dikatakan sistem apabila memenuhi dua syarat:

1. Memiliki bagian-bagian yang saling berintegrasi dengan maksud untuk mencapai suatu tujuan, bagian-bagian itu dinamakan sub-sistem.
2. Harus memenuhi tiga unsur *Input-Process-Output*.

Tabel II.1 Contoh *Input-Process-Output*

<i>Input</i>	<i>Process</i>	<i>Output</i>
Data pembelian, Data barang yang dibeli, <i>purchase requisition</i>	Melakukan pencarian penjual, membuat <i>purchase order</i>	Laporan pembelian

(Sumber: Puspitawati dan Anggadini, 2011)

2.2.1 Karakteristik Sistem

Karakteristik sistem menurut Susanto (2004) yang dikutip dari buku sistem informasi akuntansi oleh Puspitawati dan Anggadini (2011), adalah adanya tujuan sistem, batas sistem, subsistem, hubungan sistem, lingkungan sistem, dan *input*, proses, dan *output*. Untuk lebih jelasnya akan diuraikan sebagai berikut:

1. Tujuan sistem

Tujuan sistem merupakan target atau sasaran akhir yang ingin dicapai oleh suatu sistem. Agar target tersebut dapat tercapai, maka target atau sasaran tersebut harus diketahui terlebih dahulu ciri-ciri atau kriterianya. Ciri-ciri atau

kriteria dapat juga digunakan sebagai tolak ukur dalam menilai keberhasilan suatu sistem dan menjadi dasar dilakukannya suatu pengendalian. Jadi kalau melihat kepada alasan diatas maka kriteria suatu tujuan itu mutlak adanya, tanpa ada kriteria yang jelas maka akan sulit dilakukan pengendalian karena sulit dilakukan pengukuran sehingga pada akhirnya sistem tidak akan mencapai sasaran.

2. Batas Sistem

Batas sistem merupakan garis abstraksi yang memisahkan antara sistem dan lingkungannya. Batas sistem ini bagi umat manusia sangat relatif dan tergantung pada tingkat pengetahuan dan situasi kondisi yang dirasakan oleh orang yang melihat sistem tersebut.

3. Subsistem

Subsistem merupakan komponen atau bagian dari suatu sistem, bisa fisik atau abstrak. Suatu subsistem akan memiliki subsistem yang lebih kecil dan seterusnya. Istilah yang menggambarkan bagian dari suatu sistem tidak harus selalu subsistem istilah lain mungkin adalah komponen, elemen atau unsur. Subsistem istilah yang digunakan untuk menunjukan bagian dari sistem pada saat sistem tersebut menjadi konteks pembahasan. Pada saat konteks pembahasan berpindah ke subsistem maka substansi tersebut menjadi sistem dan didalam sistem ini juga terdapat subsistem yang lebih kecil, sedang sistem yang sebelumnya di mana subsistem yang kini menjadi sistem berada, disebut sebagai supersistem. Dengan beralihnya konteks yang menjadi pembahasan atau dengan beralihnya bidang yang dianggap sebagai sistem maka beralih pula batas-batasnya.

4. Hubungan dan Hierarki Sistem

Hubungan sistem adalah hubungan yang terjadi antara subsistem dengan subsistem lainnya yang setingkat atau antara subsistem dengan sistem yang lebih besar. Ada dua macam hubungan sistem, yaitu hubungan horizontal dan hubungan vertikal.

5. *Input-Proses-Output*

Tiga komponen sistem fungsi/subsistem adalah *Input-Proses-Output*, fungsi ini juga menunjukkan bahwa sistem sebagai proses tidak bisa berdiri sendiri harus terdapat *input* dan *output*. Tiga komponen tersebut dapat dijabarkan sebagai berikut:

- a. *Input* merupakan segala bentuk masukan ke dalam suatu sistem, *input* bervariasi bisa berupa energi, manusia, data, model, dan lain-lain.

Input merupakan pemicu bagi sistem untuk melakukan proses. *Input* dapat diklasifikasikan kedalam tiga kategori sebagai berikut:

- 1) *Serial input* merupakan *input* yang diperoleh sebagai hasil atau *output* sistem sebelumnya.
- 2) *Probable input* merupakan potensi *input* yang dapat digunakan oleh suatu sistem. Suatu sistem harus dapat menentukan *input* mana yang sesuai untuk menghasilkan *output* yang diharapkan.
- 3) *Feedback* dan pengendalian, *feedback* adalah data atau informasi tentang kinerja (*output* dari suatu sistem) Suatu *output* akan terdiri dari:
 - a) Jumlah yang dihasilkan oleh suatu sistem untuk periode waktu tertentu.
 - b) Waktu yang diperlukan diperlukan oleh suatu sistem untuk menyelesaikan satu produk atau jasa tertentu.
 - c) Kualitas dari produk dan jasa yang dihasilkan.

Terdapat beberapa jenis *feedback* seperti, *feedback input* yaitu merupakan *input* jenis ke tiga, *input* ini merupakan bagian *output* dari sistem yang sama yang digunakan sebagai *control*. *Feedback input* ini dikelompokkan lagi menjadi dua yaitu:

- 1) *Negative feedback input* digunakan sebagai alat kontrol untuk memperkecil. *Negative feedback* ini terus berlangsung sampai terjadi titik keseimbangan dimana antara *output* yang diharapkan dan realisasi terjadi kesesuaian dengan tingkat penyimpangan yang dapat diterima.

2) *Positive feedback input* digunakan sebagai alat kontrol untuk memperbesar. *Positive feedback* ini terus berlangsung sampai terjadi titik keseimbangan dimana antara *output* yang diharapkan dan realisasi terjadi kesesuaian dengan tingkat penyimpangan yang dapat diterima.

b. Fungsi kedua yaitu Proses yang merupakan perubahan dari *input* menjadi *output*. Proses mungkin berupa perakitan yang menghasilkan satu macam *output* dari berbagai macam *input* yang disusun berdasarkan aturan tertentu.

c. Fungsi ketiga yaitu *Output*. *Output* adalah hasil dari suatu proses yang merupakan tujuan dari keberadaan sistem. *Output* dapat diklasifikasikan menjadi tiga macam yaitu:

- 1) *Output* yang langsung diberikan ke konsumen untuk dikonsumsi.
- 2) *Output* suatu sistem yang dikonsumsi oleh suatu subsistem yang lain dalam sistem yang sama dalam suatu siklus produksi.
- 3) *Output* yang merupakan bagian dari *output* secara keseluruhan yang dapat dikonsumsi oleh sistem yang lain atau oleh sistem yang bersangkutan, namun menjadi tidak berguna apabila dibuang ke lingkungan.

6. Lingkungan sistem

Lingkungan sistem adalah faktor-faktor diluar sistem yang mempengaruhi sistem. Lingkungan sistem ada dua macam yaitu lingkungan eksternal (lingkungan yang berada diluar sistem) dan lingkungan internal (lingkungan yang berada didalam suatu sistem).

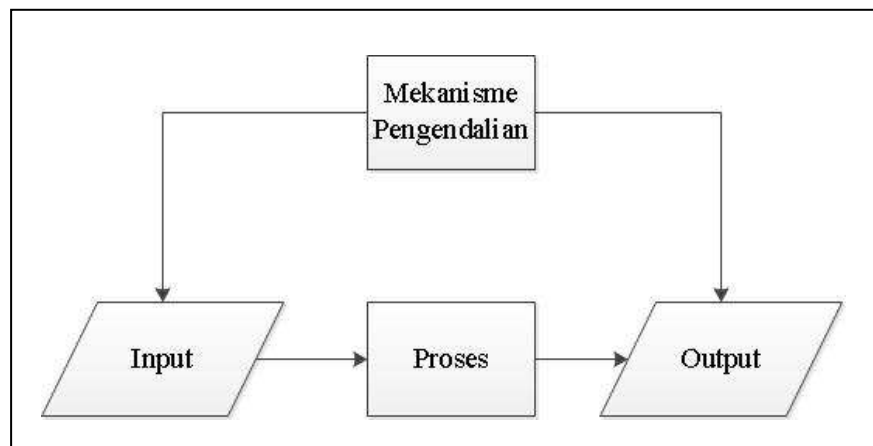
2.2.2 Jenis Sistem

Menurut McLeod (2001) yang dikutip oleh Rusdiana dan Irfan (2014). Jenis sistem secara umum terdiri atas sistem terbuka dan sistem tertutup. Sistem terbuka adalah sistem yang tidak memiliki sasaran, pengendalian mekanis dan umpan balik. Adapun sistem yang tertutup adalah sebuah sistem yang memiliki sasaran, pengendalian mekanis dan umpan balik. Secara sederhana, sistem terbuka dapat digambarkan sebagai berikut:



Gambar II.1 Gambaran Sistem Terbuka
(Sumber: Rusdiana dan Irfan, 2014)

Sistem tertutup adalah sistem yang tidak berinteraksi secara langsung dengan lingkungannya melalui arus sumber daya. Skema sistem tertutup dapat dilihat pada gambar di bawah ini.



Gambar II.2 Gambaran Sistem Tertutup
(Sumber: Rusdiana dan Irfan, 2014)

2.3 Informasi

Menurut R.J. Beishon yang dikutip Sunyoto (2014) informasi adalah diinterpretasikan, barangkali, lebih luas daripada biasanya, yang mencakup isyarat dan data yang diterima seorang manajer sehari-hariannya, apakah itu bersangkutan dengan pekerjaan atau tidak. Pendapat lain menurut Eilon dikutip Sunyoto (2014) informasi adalah sebagai pernyataan yang menjelaskan suatu peristiwa atau suatu objek atau suatu konsep, sedemikian rupa sehingga membantu kita untuk membedakan dari yang lain. Sementara menurut Sutabri (2004) Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Informasi dikelompokkan menjadi 3 bagian, yaitu:

1. Informasi Strategis

Informasi ini digunakan untuk mengambil keputusan jangka panjang, mencakup informasi eksternal, rencana perluasan perusahaan, dan sebagainya.

2. Informasi Taktis

Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.

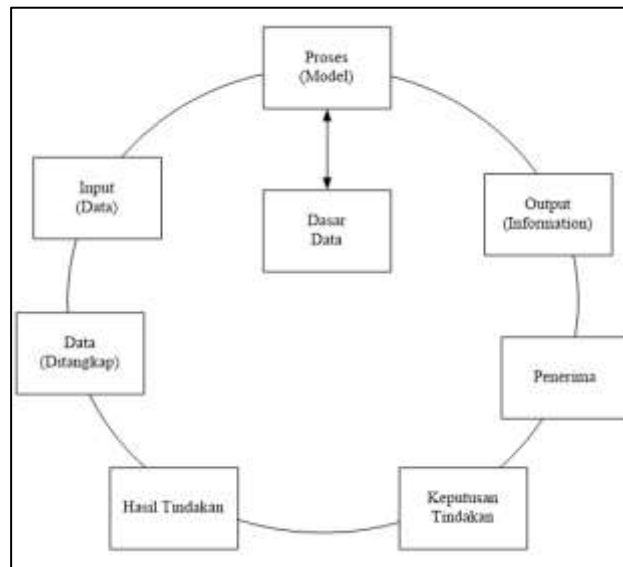
3. Informasi Teknis

Informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stock, retur penjualan dan laporan kas harian.

2.3.1 Siklus dan Fungsi Informasi

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan.

Menurut Sutabri (2004), data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus informasi ini dapat digambarkan sebagai berikut:



Gambar II.3 Siklus Informasi
Sumber: Sutabri (2004)

Fungsi utama dari informasi adalah menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi. Informasi yang disampaikan kepada pemakai mungkin merupakan hasil dari data yang dimasukkan kedalam pengolahan. Akan tetapi dalam kebanyakan pengambilan keputusan yang kompleks, informasi hanya dapat menambah kemungkinan kepastian atau mengurangi bermacam-macam pilihan. Informasi yang disediakan bagi pengambil keputusan memberikan suatu kemungkinan faktor risiko pada tingkat-tingkat pendapatan yang berbeda.

2.3.2 Nilai Informasi

Menurut Sutabri (2004), nilai dari informasi ditentukan dari dua hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai apabila manfaat yang diperoleh lebih berharga dibandingkan dengan biaya untuk mendapatkannya. Berikut ini adalah nilai informasi berdasarkan atas 10 (sepuluh) sifat, yaitu:

1. Mudah diperoleh

Sifat ini menunjukkan kemudahan dan kecepatan untuk memperoleh informasi. Kecepatannya dapat diukur, misalnya 1 menit versus 24 jam. Akan tetapi berapa nilainya bagi pemakai informasi sulit untuk mengukurnya.

2. Luas dan lengkap

Sifat ini menunjukkan kelengkapan isi informasi. Hal ini tidak hanya mengenai volumenya, akan tetapi juga mengenai keluaran informasinya. Sifat ini sangat kabur dan karena itu sulit untuk mengukurnya.

3. Ketelitian

Sifat ini berhubungan dengan tingkat kebebasan dari kesalahan keluaran informasi. Pada volume data yang besar biasanya terdapat dua jenis kesalahan, yakni kesalahan pencatatan dan kesalahan perhitungan.

4. Kecocokan

Sifat ini menunjukkan seberapa baik keluaran informasi dalam hubungannya dengan permintaan para pemakai. Isi informasi harus ada hubungannya dengan masalah yang sedang dihadapi sedangkan semua keluaran yang lainnya tidak berguna. Sifat ini sulit mengukurnya.

5. Ketepatan waktu

Sifat ini berhubungan dengan waktu yang dilalui, yang lebih pendek dari siklus untuk mendapatkan informasi. Masukan pengolahan dan pelaporan keluaran kepada para pemakai biasanya tepat waktu. Dalam beberapa hal, ketepatan waktu dapat diukur. Misalnya berapa banyak penjualan dapat ditingkatkan dengan menanggapi permintaan pelanggan mengenai ketersediaan barang-barang inventaris.

6. Kejelasan

Sifat ini menunjukkan tingkat kejelasan informasi. Informasi hendaknya terbebas dari istilah-istilah yang tidak jelas.

7. Keluwesan

Sifat ini berhubungan dengan apakah informasi tersebut dapat digunakan untuk membuat lebih dari satu keputusan, tetapi apakah juga dapat digunakan untuk

lebih dari seorang pengambil keputusan. Sifat ini sulit mengukurnya, akan tetapi dalam beberapa hal dapat diukur dengan suatu nilai tertentu.

8. Dapat dibuktikan

Sifat ini menunjukkan sejauh mana informasi itu dapat diuji oleh beberapa pemakai hingga sampai didapatkan kesimpulan yang sama.

9. Tidak ada prasangka

Sifat ini berhubungan dengan ada tidaknya keinginan untuk mengubah informasi tersebut guna mendapatkan kesimpulan yang telah diarahkan sebelumnya.

10. Dapat diukur

Sifat ini menunjukkan hakikat informasi yang dihasilkan oleh sistem informasi formal. Meskipun kabar angin, desas-desus, dugaan-dugaan, klenik, dan lainnya juga sering dianggap sebagai informasi, namun hal-hal tersebut berada di luar lingkup pembahasan.

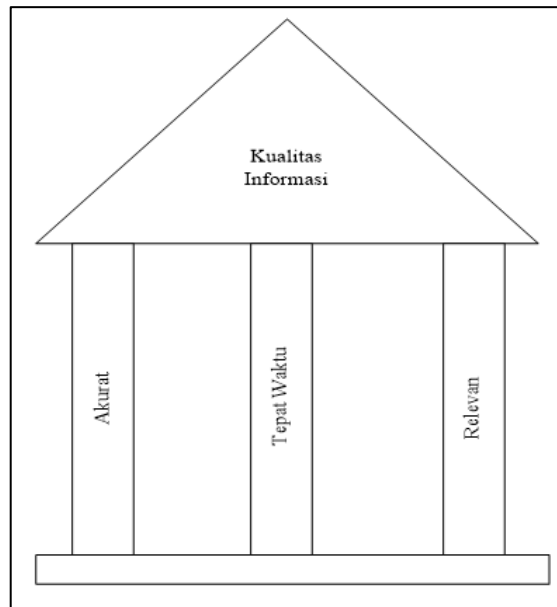
Informasi bernilai sempurna apabila pengambil keputusan dapat mengambil keputusan secara optimal dalam setiap hal, dan bukan keputusan yang rata-rata akan menjadi optimal dan untuk menghindari kejadian-kejadian yang akan mendatangkan kerugian.

2.3.3 Kualitas Informasi

Informasi bervariasi dalam kualitasnya karena adanya kesalahan yang timbul, dimana kesalahan dalam hal ini merupakan persoalan yang sangat penting untuk dikendalikan. Menurut Sunyoto (2014) kesalahan-kesalahan dalam informasi dapat disebabkan oleh:

1. Metode pengukuran dan pengumpulan data yang salah.
2. Tidak mengikuti prosedur pengolahan data yang benar.
3. Data hilang atau tidak terolah.
4. Kesalahan mencatat dan mengoreksi data.
5. File induk yang salah.
6. Kesalahan dalam prosedur pengolahan data.
7. Kesalahan yang disengaja.

Dalam buku Jogiyanto (2005) John Burch dan Gary Grudnitski menggambarkan kualitas dari informasi dengan bentuk bangunan yang ditunjang oleh tiga buah pilar.



Gambar II.4 Pilar kualitas informasi
(Sumber: Jogiyanto, 2005)

1. Akurat

Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi.

2. Tepat pada waktunya

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

3. Relevan

Berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lain berbeda. Misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan

perusahaan adalah kurang relevan dan akan lebih relevan bila ditunjukkan kepada ahli teknik perusahaan.

2.4 Sistem Informasi

Dalam bukunya yang berjudul Analisis dan Perancangan Sistem Informasi menggunakan model terstruktur dan UML Muslihudin dan Oktafianto (2016) mengutip beberapa pendapat ahli yang mendeskripsikan sistem informasi sebagai berikut:

1. Sistem informasi merupakan suatu kumpulan dari komponen-komponen dalam organisasi yang berhubungan dengan proses penciptaan aliran informasi. (Yakub, 2012)
2. Sistem informasi merupakan perangkat prosedur yang terorganisasi dengan sistematis, bila dilaksanakan akan menyediakan informasi yang dapat dimanfaatkan dalam proses pembuatan keputusan. (Nuraida, 2008)
3. Sistem informasi adalah sekumpulan komponen yang saling bekerja sama, yang digunakan untuk mencatat data, mengolah data dan menyajikan informasi untuk para pembuat keputusan agar dapat membuat keputusan dengan baik. (Winarno, 2006)

Menurut Sutabri (2004) sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari komponen *input*, komponen model, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran. Berikut ini adalah penjelasan dari masing-masing komponen yang terlibat didalamnya, yaitu:

1. Komponen *input*

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

2. Komponen model

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

3. Komponen *output*

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

4. Komponen teknologi

Teknologi merupakan *tool box* dalam sistem informasi, Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

5. Komponen *hardware*

Hardware berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung *database* atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

6. Komponen *software*

Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari hardware untuk menciptakan suatu informasi.

7. Komponen basis data

Merupakan blok yang berisi definisi basis data yang disediakan untuk menyimpan data-data yang akan disimpan dalam media penyimpan. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

8. Komponen kontrol

Merupakan sebuah komponen yang bertugas mendefinisikan bagaimana kontrol terhadap sistem dilakukan sehingga sistem dapat berjalan dengan baik. Dalam blok kontrol ini misalnya didefinisikan bagaimana melindungi data yang ada di *database* agar selalu sama dengan kenyataan yang dicatat.

2.5 Pembelian

Pembelian adalah sebuah aktivitas bisnis yang merupakan salah satu aktivitas utama di dalam sebuah perusahaan untuk mendukung jalannya roda bisnis perusahaan. Sebuah perusahaan tidak dapat memenuhi segala kebutuhan bisnisnya sendiri sehingga sebuah perusahaan membutuhkan perusahaan lain sebagai pemasok kebutuhan perusahaan tersebut. dalam hal ini aktivitas pembelian menjadi sebuah aktivitas yang sangat penting, aktivitas pembelian bukan lagi hanya aktivitas membeli suatu barang namun aktivitas ini melibatkan analisis pemasok dan juga analisis kebutuhan produk, karena produk yang dibeli harus berkualitas dan tepat guna serta dengan harga yang sesuai. Para ahli mencoba mendeskripsikan pengertian pembelian sebagai berikut:

1. Menurut Bodnar (2004) Aktivitas pembelian dapat disebut dengan *prokuremen*. *Prokuremen* merupakan suatu proses bisnis yang diawali dengan pemilihan sumber daya, aktivitas pembuatan *order*, dan perolehan barang dan jasa dari pemasok yang dilakukan oleh perusahaan. dari definisi tersebut maka dapat disimpulkan bahwa *prokuremen* dapat disamakan dengan aktivitas pembelian. (Puspitawati dan Anggadini, 2011).
2. Menurut Lenders, Fearon, Flynn, Johnson yang dikutip oleh Indrajit, Djokopranoto (2005), *purchasing describes the process of buying: learning the need, locating and selecting a supplier, negotiating prices and other pertinent terms, and following up to ensure delivery.*
3. Menurut tim penerbit PPM dan Asosiasi Logistik Indonesia (2011) dalam buku Panduan & Direktori Logistik Indonesia, pembelian pada dasarnya merupakan pesan yang dikirim oleh suatu perusahaan kepada konsumen dengan

mengatakan, “Kami telah setuju dengan syarat-syaratnya, maka kirimlah barang kepada kami dan kami akan membayarnya.

2.5.1 Tugas dan Tanggung Jawab Divisi *Purchasing*

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.

2.5.2 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 *prokurement*.

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

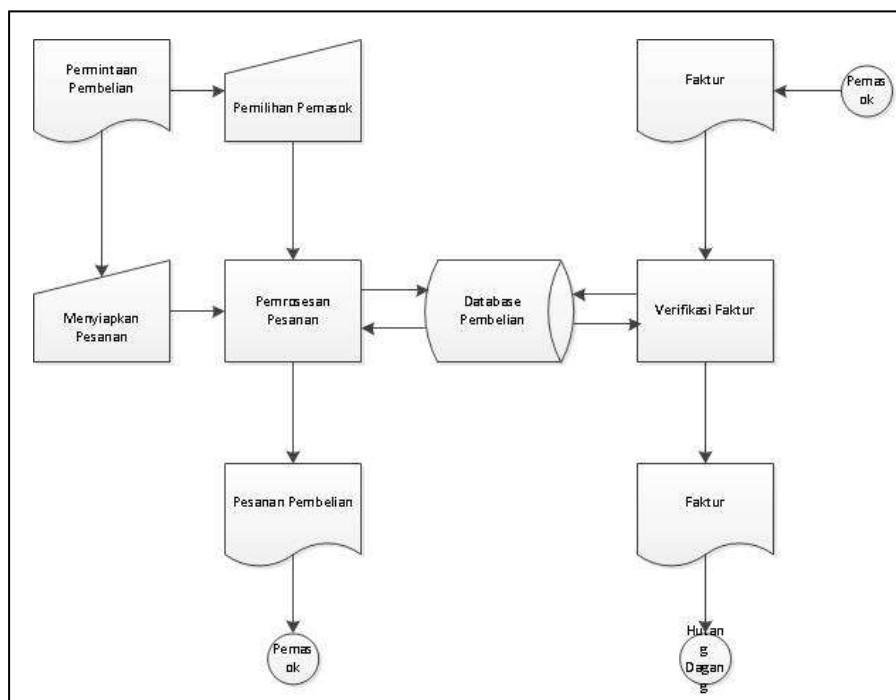
Verifikasi faktur dilakukan ketika perusahaan akan melakukan pembayaran kepada pemasok. Faktur harus diperiksa dan dicocokkan dengan dokumen

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.

Berikut merupakan alur dari proses pembelian menurut Puspitawati dan Anggadini (2011):



Gambar II.5 Alur Proses Pembelian
(sumber: Puspitawati dan Anggadini, 2011)

2.5.3 Tujuan Pembelian

Salah satu tujuan dari pembelian di sebuah perusahaan yaitu mendukung proses bisnis perusahaan agar berjalan optimal dengan menyediakan kebutuhan-

kebutuhan dari proses bisnis. Menurut tim penerbit PPM dan Asosiasi Logistik Indonesia (2011) dalam buku Panduan & Direktori Logistik Indonesia tujuan dari pembelian atau prokuremen secara keseluruhan adalah menjamin bahwa sebuah perusahaan memiliki pasokan bahan yang terpercaya. Dengan tujuan pokok tersebut, dapat dikembangkan daftar sasaran langsung sebagai berikut:

1. Mengatur aliran barang yang terpercaya dan terus menerus di perusahaan.
2. Bekerja dekat dengan departemen pengguna, mengembangkan hubungan dan memahami kebutuhan mereka.
3. Mencari pemasok yang bagus, bekerja bersama mereka dan mengembangkan hubungan yang menguntungkan.
4. Membeli barang yang tepat dan memastikan bahwa barang-barang tersebut memiliki kualitas yang dapat diterima, datang pada waktu dan tempat yang diinginkan, dan memenuhi prasyarat lainnya.
5. Menegosiasikan harga dan syarat yang laik.

2.6 Spare Part

Spare part atau suku cadang atau onderdil merupakan sebutan dari komponen-komponen yang menyusun suatu mesin, setiap komponen pada sebuah mesin mempunyai fungsi tersendiri yang secara khusus telah didesain. Sebuah suku cadang mempunyai jangka waktu penggunaan sebelum suku cadang menjadi aus, maka dari itu diperlukan pengecekan secara berkala untuk mengetahui tingkat keausan pada suku cadang.

Menurut Kamus Besar Bahasa Indonesia suku cadang adalah alat-alat (dalam peralatan teknik) yang merupakan bagian dari mesin. Sedangkan menurut Menurut Indrajit dan Djokopranoto (2003) menyatakan Suku cadang atau *spare part* adalah suatu alat yang mendukung pengadaan barang untuk keperluan peralatan yang digunakan dalam proses produksi.

Berdasarkan definisi diatas dapat disimpulkan bahwa *spare part* atau suku cadang merupakan bagaian dari sebuah mesin (peralatan teknik) yang berfungsi untuk mendukung proses produksi perusahaan.

Berdasarkan penggunaannya, suku cadang dapat dibagi menjadi tiga jenis. Indrajit dan Djokopranoto (2003) mengklasifikasikan suku cadang ke dalam tiga jenis sebagai berikut:

1. Suku cadang habis pakai (*consumable parts*)

Suku cadang jenis ini adalah suku cadang untuk pemakaian biasa, yaitu yang akan aus dan rusak, kerusakan suku cadang ini dapat terjadi sewaktu-waktu. Oleh karena itu, pengaturan persediaannya haruslah sedemikian rupa sehingga sewaktu-waktu diperlukan haruslah selalu tersedia, atau dapat diadakan dalam waktu singkat sehingga tidak mengganggu jalannya peralatan.

2. Suku cadang pengganti (*replacement parts*)

Suku cadang jenis ini adalah suku cadang yang penggantianannya biasanya dilakukan pada waktu *overhaul*, yaitu pada waktu diadakan perbaikan besar-besaran. Waktu *overhaul* ini biasanya dapat dijadwalkan sesuai dengan rekomendasi pabrik pembuat peralatan tersebut. Selain waktu *overhaul* yang dapat dijadwalkan, suku cadang yang perlu diganti dapat juga diperkirakan dengan cukup akurat. Oleh karena itu, biasanya jenis suku cadang ini tidak disimpan dalam persediaan, kecuali untuk peralatan vital.

3. Suku cadang jaminan (*insurance parts*)

Suku cadang jenis ini adalah suku cadang yang biasanya tidak pernah rusak, tetapi dapat rusak, dan apabila rusak dapat menghentikan operasi dan produksi. Suku cadang jaminan ini biasanya bentuknya besar, harganya mahal, dan waktu pembuatannya lama.

2.7 *System Development Life Cycle (SDLC)*

Menurut Dennis (2015) *System Development Life Cycle (SDLC)* memiliki empat perangkat fase dasar yaitu *planning*, *analysis*, *design*, dan *implementation*. Beberapa proyek dalam membangun sebuah sistem informasi mungkin dapat berbeda-beda sesuai dengan caranya masing-masing, tetapi hampir semua proyek memiliki elemen dari empat fase tersebut. Masing-masing fase tersebut tersusun dari beberapa langkah-langkah yang menghasilkan *deliverable* atau hasil kegiatan

seperti beberapa dokumen spesifik dan file yang menjelaskan pemahaman tentang proyek.

Dalam beberapa proyek, tahapan dan langkah-langkah SDLC diproses dalam suatu alur tahapan, dimulai dari awal hingga akhir. Dalam proyek lain, tim proyek terus berpindah dari fase ke fase tersebut secara teratur, bertahap, iteratif, atau dalam bentuk pola lainnya. Dalam SDLC, dijelaskan fase-fase, tindakan, serta beberapa teknik yang digunakan untuk menyempurnakan langkah-langkah yang dijelaskan secara umum.

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 merupakan penjabaran dari setiap fase dalam SDLC:

1. *Planning* (Perencanaan)

Tahap *planning*/perencanaan adalah proses dasar yang menjelaskan mengapa sebuah sistem informasi harus dibangun dan menentukan bagaimana sebuah tim proyek akan membangunnya. Terdapat dua langkah dalam tahap ini yaitu:

- a. Inisiasi proyek, pada tahap ini 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 area Departemen IT (misalnya dari Departemen Pemasaran, Departemen Akuntansi) dalam suatu bentuk *system request* (permintaan sistem). Permintaan sistem menyajikan ringkasan singkat tentang kebutuhan sistem dalam suatu bisnis dan menjelaskan bagaimana suatu sistem akan menciptakan nilai bisnis. Departemen IT bekerja sama dengan departemen lain yang menghasilkan permintaan (disebut sponsor proyek) untuk melakukan analisis kelayakan sistem informasi.

System request (permintaan sistem) dan analisis kelayakan disajikan dan dipresentasikan kepada pihak komite persetujuan *Information System*

(komite pengarah), yang menentukan apakah proyek tersebut harus diambil atau tidak.

- b. Setelah proyek disetujui, proyek memasuki manajemen proyek. Selama manajemen proyek, manajer proyek membuat sebuah rencana kerja, membentuk anggota staff proyek, dan anggota teknisi lainnya di tempat, untuk membantu tim proyek mengendalikan dan mengarahkan proyek pada keseluruhan tahap SDLC.

2. *Analysis* (Analisis)

Analisis menurut Muslihudin dan Oktafianto (2016) merupakan tahapan awal dalam pengembangan sistem dan merupakan tahap fundamental yang sangat menentukan kualitas sistem yang akan dikembangkan. Tahap analisis dalam SDLC menjelaskan siapa yang akan memakai sistem, apa yang sistem kerjakan serta kapan dan dimana sistem akan digunakan. Selama pada tahap ini, tim proyek menyelidiki sistem yang ada saat ini, mengidentifikasi peluang untuk melakukan perbaikan, dan mengembangkan sistem baru. Tahap ini memiliki tiga langkah:

- a. Langkah yang pertama adalah Strategi Analisis. Langkah ini dikembangkan untuk mengarahkan usaha dari tim proyek untuk menganalisis sistem yang telah ada (disebut sebagai *as-is system*) beserta masalah-masalah yang ada dan untuk merancang suatu sistem yang baru (disebut sebagai *to-be system*).
- b. Langkah selanjutnya adalah mengumpulkan kebutuhan-kebutuhan sistem (melalui wawancara atau kuesioner). Pada tahap ini terdapat juga masukan analisis informasi dari sponsor proyek dan pihak lainnya yang mengarah pada pengembangan konsep untuk sistem baru. Konsep sistem ini kemudian digunakan sebagai dasar untuk mengembangkan seperangkat model analisis bisnis, yang menggambarkan bagaimana bisnis akan beroperasi jika sistem baru dikembangkan.
- c. Kemudian Analisis, konsep sistem, dan model digabungkan menjadi dokumen yang disebut proposal sistem, yang dipresentasikan ke sponsor proyek dan pengambil keputusan utama lainnya (misalnya, anggota komite persetujuan) yang memutuskan apakah proyek tetap terus dilanjutkan.

3. *Design* (Perancangan)

Tahap *design* atau perancangan memutuskan bagaimana sistem akan beroperasi, mulai dari hal yang berkaitan dengan perangkat keras (*hardware*), perangkat lunak (*software*) dan infrastruktur jaringan seperti:

- a. Antarmuka pengguna (*interface*),
- b. Formulir,
- c. Laporan,
- d. Program spesifik: *database*, dan *file-file* lainnya yang akan dibutuhkan.

Meskipun sebagian besar keputusan strategis mengenai sistem dibuat dalam pengembangan konsep sistem selama tahap analisis, langkah-langkah dalam tahap desain menentukan secara tepat bagaimana sistem akan beroperasi.

4. *Implementation* (Implementasi)

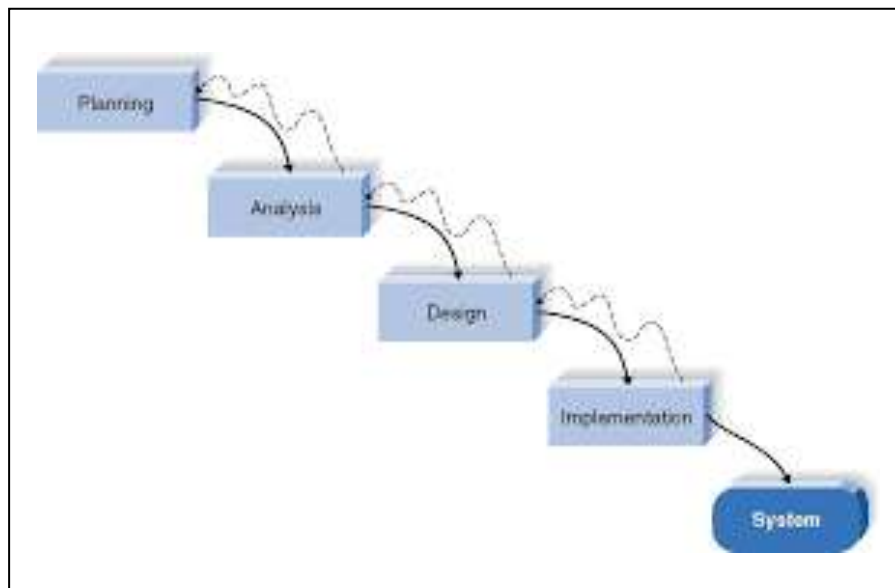
Tahap akhir dalam SDLC adalah tahap implementasi, di mana sistem benar-benar dibangun (atau dibeli, dalam kasus paket *desaign software*). Fase ini sangat penting, karena kebanyakan sistem merupakan bagian terlama dan paling mahal dari proses pengembangan.

2.8 **Model Waterfall**

Menurut Dennis (2010) penggunaan metode pengembangan *waterfall*, seorang analis dan *user* memproses pengembangan secara bertahap dari satu fase ke fase berikutnya, setiap fase biasanya berlangsung cukup lama dan setiap fase yang dilewati akan di presentasikan kepada sponsor untuk mendapatkan persetujuan, jika sponsor belum menyetujui suatu fase maka pengembangan sistem tidak dapat dilanjutkan ke fase berikutnya. Metodologi ini menyerupai air terjun atau *waterfall* karena bergerak maju dari satu fase ke fase berikutnya secara bertahap seperti cara kerja air terjun, walaupun dalam SDLC memungkinkan untuk kembali ke fase sebelumnya namun, hal ini akan sulit dilakukan dalam metode *waterfall*.

Keuntungan yang didapat dari pengembangan dengan metode *waterfall* yaitu, dapat mengidentifikasi kebutuhan sistem jauh sebelum proses pemrograman berlangsung sehingga meminimalisasi perubahan yang dapat terjadi pada

kebutuhan sistem saat proyek berjalan. Kelemahan dari model *waterfall* adalah desain harus benar-benar ditentukan sebelum pemrograman dimulai dan lamanya proses yang dibutuhkan untuk menyelesaikan sistem. Berikut penggambaran model *waterfall* menurut Dennis (2010)



Gambar II.6 Penggambaran Metode *Waterfall*
(sumber: Dennis, 2010)

2.9 *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 requirements* atau kebutuhan fungsional. *Functional requirements* 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 detail seperti *input*, *output* dan pengecualian yang berlaku. Contoh: Sistem mampu menginput dan menampilkan data bahan baku.

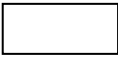
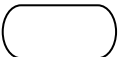
2. *Non-functional requirements* atau kebutuhan non-fungsional. *Non-functional requirements* secara umum berisi batasan-batasan pada pelayanan atau fungsi yang disediakan oleh sistem. Termasuk di dalamnya adalah batasan waktu, batasan proses pembangunan, standar-standar tertentu. Karena berkaitan dengan kebutuhan sistem secara keseluruhan, maka kegagalan memenuhi kebutuhan jenis ini berakibat pada sistem secara keseluruhan. Contoh: Hak akses admin seleksi penerimaan pegawai dapat melakukan semua kegiatan pada sistem sementara hak akses untuk kepala departemen hanya bisa melihat hasil akhir seleksi.

2.10 Flowchart

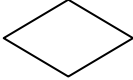
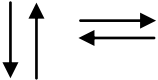
Untuk menggambarkan sebuah proses agar mudah dipahami oleh orang lain maka dibutuhkan alat bantu yang berbentuk diagram alir (*flowchart*). *Flowchart* menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga *flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu. Diagram alir ini selain dibutuhkan sebagai alat komunikasi, juga diperlukan sebagai dokumentasi (Sitorus, 2015).

Tujuan dari *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas menggunakan simbol-simbol. Simbol simbol yang terdapat dalam sebuah *flowchart* dijelaskan pada tabel berikut:

Tabel II.2 Simbol-simbol *Flowchart*

Simbol	Nama	Keterangan
	Terminal	Menyatakan permulaan atau akhir suatu program.
	<i>Input-Output</i>	Meyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
	<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.

Tabel II.2 Simbol-simbol *Flowchart* (lanjutan)

Simbol	Nama	Keterangan
	<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya atau tidak.
	<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
	<i>Off-line Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
	<i>Predafined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
	<i>Punched Card</i>	Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu.
	<i>Document</i>	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak di kertas.
	<i>Flow</i>	Menyatakan jalannya arus suatu proses

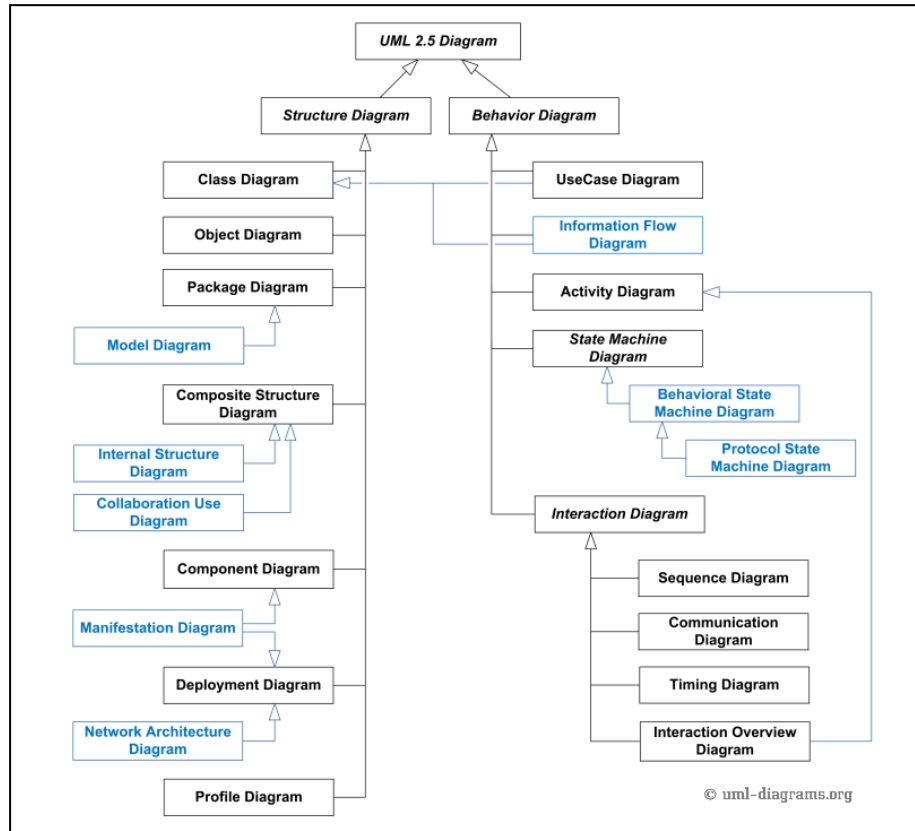
(Sumber: Sitorus, 2015)

2.11 Unified Modelling Language (UML)

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

Menurut Dennis (2015), 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.



Gambar II.7 UML Diagram
(Sumber: uml-diagrams.org, 2017)

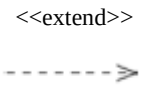
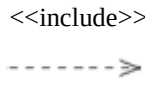
Berikut ini penjelasan singkat dari pembagian kategori UML menurut website uml-diagrams.org:

1. *Structure diagrams*, yaitu kumpulan diagram yang menunjukkan struktur statis sistem dan bagian-bagiannya pada tingkat abstraksi dan implementasi yang berbeda dan bagaimana bagian-bagian itu saling terkait satu sama lain. Elemen dalam diagram struktur mewakili konsep sistem yang berarti, dan mungkin mencakup konsep abstrak, dunia nyata dan implementasi.
2. *Behavior diagrams*, yaitu kumpulan diagram yang menunjukkan perilaku dinamis objek dalam suatu sistem, yang dapat digambarkan sebagai rangkaian perubahan sistem dari waktu ke waktu

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.3 Simbol-Simbol *Use case Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Use case</i>	Merepresentasikan bagian utama dari sistem secara fungsional.
2		<i>Actor</i>	Seseorang atau sistem yang mendapatkakan keuntungan dari sistem
3		<i>Association</i>	Menghubungkan suatu actor dengan <i>use case</i> ..
4		<i>Extend</i>	Merepresentasikan eksistensi dari <i>use case</i> untuk menyertakan perilaku optional
5		<i>Include</i>	Mewakili dimasukkannya satu <i>use case</i> ke <i>use case</i> yang lain.
6		<i>Subject</i>	Menyertakan nama subjek didalam maupun diatas

Tabel II.3 Simbol-Simbol *Use case Diagram* (lanjutan)

No	Simbol	Nama	Deskripsi
7		<i>Generalization</i>	Merepresentasikan <i>use case</i> khusus ke yang lebih umum.

(Sumber: Dennis, 2015)

2.11.2 Activity Diagram

Activity diagram digunakan untuk memodelkan perilaku dalam proses bisnis yang independen terhadap objek. Dalam banyak hal, *activity diagram* dapat dipandang sebagai *data flow diagram* canggih yang digunakan pada konjungsi dengan analisis terstruktur. Walaupun begitu, *activity diagram* tidak seperti data flow diagram, *activity diagram* menggunakan notasi yang mengalamatkan pemodelan paralel. (Dennis, 2010)

Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

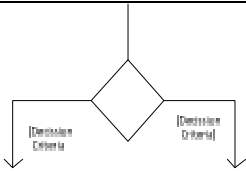
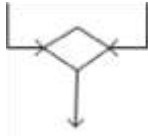
1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
3. Rancangan pengujian dimana setiap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

Berikut adalah simbol-simbol yang ada pada *Activity Diagram*:

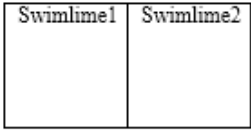
Tabel II.4 Simbol-Simbol *Activity Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Action</i>	Merupakan notasi yang simple, dan tidak untuk dianalisis perilakunya. Dinamakan sesuai aksi yg dilakukan.

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

No	Simbol	Nama	Deskripsi
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.
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-Flow Node</i>	Menggambarkan akhir dari setiap aktivitas.
8.		<i>Decision Node</i>	Untuk mewakili suatu kondisi pengujian, yang bertujuan untuk memastikan bahwa <i>control flow</i> atau <i>object flow</i> hanya menuju ke satu arah.
9.		<i>Merge Node</i>	Untuk menyatukan kembali <i>decision path</i> yang dibuat dengan menggunakan <i>decision node</i> .
10.		<i>Fork node</i>	Untuk memisahkan perilaku menjadi serangkaian aktivitas yang berjalan secara parallel atau bersamaan.
11.		<i>Join Node</i>	Untuk menyatukan kembali serangkaian aktivitas yang berjalan secara parallel atau bersamaan.

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

No	Simbol	Nama	Deskripsi
12.		<i>Swimlane</i>	Untuk membagi sebuah <i>activity diagram</i> menjadi beberapa kolom, dengan tujuan menempatkan aktivitas tertentu pada individu yang bertanggung jawab atas terlaksananya aktivitas tersebut.

(Sumber: Dennis, 2010)

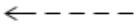
2.11.3 *Sequence Diagram*

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

Tabel II.5 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.		Garis Hidup/ <i>lifeline</i>	Menyatakan kehidupan suatu objek.

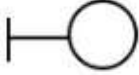
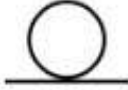
Tabel II.5 Simbol-Simbol *Sequence Diagram* (lanjutan)

No	Simbol	Nama	Deskripsi
3.		<i>Object</i>	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 secara langsung.
7.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri.

(Sumber: Dennis, 2010)

Selain simbol-simbol diatas dalam buku *Secure Software Design*, Richardson dan Thies (2013) mendefinisikan *sequence diagram* memiliki *class* dengan fungsi yang masing-masing berbeda, berikut *class* yang terdapat dalam *sequence diagram* menurut Richardson dan Thies:

Tabel II.6 Simbol Jenis-jenis *Class* pada *Sequence Diagram*

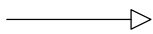
No	Simbol	Deskripsi
1		<i>Boundary Class</i> bertanggung jawab terhadap penanganan interaksi antara aktor dengan sistem.
2		<i>Entity Class</i> merupakan simbol penyimpanan, objek yang dihasilkan sebagian besar berupa data dalam sistem.
3		<i>Control Class</i> merupakan koordinator dari sistem, setidaknya harus terdapat satu <i>control class</i> dalam setiap <i>use case</i> .

(Sumber: Richardson dan Thies, 2013)

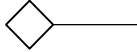
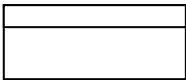
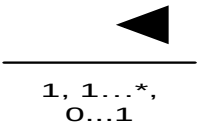
2.11.4 *Class Diagram*

Sebuah diagram kelas adalah model statis yang menunjukkan kelas dan hubungan antar kelas yang tetap konstan dalam sistem dari waktu ke waktu. Sebuah *class* merupakan deskripsi dari sekumpulan objek yang memiliki properti (*attribute*), operasi (*method*), relasi (*association*), dan tingkah laku (*behavior*) yang sama. Sebuah class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda /fungsi). (Dennis, 2010). Berikut simbol-simbol yang ada pada diagram kelas:

Tabel II.7 Simbol-Simbol *Class Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Generalization</i>	Merupakan sebuah <i>taxonomic relationship</i> antara <i>class</i> yang lebih umum dengan <i>class</i> yang lebih khusus

Tabel II.7 Simbol-Simbol *Class Diagram* (lanjutan)

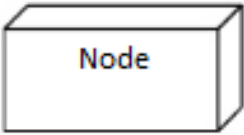
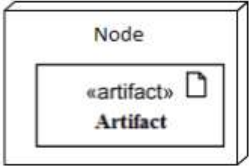
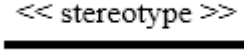
No	Simbol	Nama	Deskripsi
2		<i>Aggregation</i>	Menggambarkan suatu <i>class</i> terdiri dari <i>class</i> lain atau suatu <i>class</i> adalah bagian dari <i>class</i> lain.
3		<i>Class</i>	Kelas pada struktur sistem .
4		<i>Association</i>	Asosiasi yang menghubungkan <i>class</i> dengan <i>class Multiplicity</i> .

(Sumber: Dennis, 2010)

2.11.5 *Deployment Diagram*

Deployment diagram digunakan untuk mewakili hubungan antara komponen-komponen *hardware* yang digunakan dalam infrastruktur fisik sistem informasi. Misalnya, ketika membuat suatu sistem informasi terdistribusi yang akan menggunakan jaringan luas, *deployment diagram* dapat digunakan untuk menunjukkan hubungan komunikasi antara node yang berbeda dalam jaringan. *Deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. *Deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. Dalam hal ini, *deployment diagram* mewakili lingkungan pembuatan *software* (Dennis, 2015). Berikut simbol-simbol yang digunakan pada *deployment diagram*:

Tabel II.8 Simbol-Simbol *Deployment Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Node</i>	Untuk menggambarkan sumber daya komputasi dalam sebuah sistem (misalnya, computer klien, server, jaringan yang terpisah, atau individu perangkat jaringan).
2		<i>Artifact</i>	Untuk menggambarkan spesifikasi dari <i>software</i> atau <i>database</i> , misalnya <i>file</i> sumber, tabel <i>database</i> , <i>executable file</i> .
3		<i>Node with a Deployed Artifact</i>	Untuk menggambarkan <i>artifact</i> yang ditempatkan pada node fisik. Mendukung pemodelan distribusi perangkat lunak melalui jaringan.
4		<i>Communication Path</i>	Untuk menggambarkan hubungan antara dua node untuk bertukar pesan.

(Sumber: Dennis, 2015)

2.12 Database

Basis Data terdiri dari 2 kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya (Fathansyah, 2012). Sebagai satu kesatuan istilah, Basis data (*Database*) sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

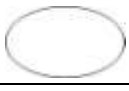
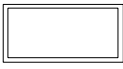
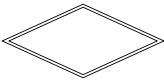
1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel yang saling berhubungan yang disimpan dalam media penyimpanan elektronis.

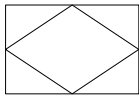
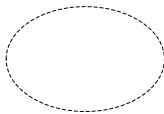
2.13 Entity Relationship Diagram (ERD)

Menurut Rosa dan Shalahuddin (2015) ERD adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional, simbol-simbol yang digunakan dalam ERD, yaitu:

Tabel II.9 Simbol-simbol Entity Relationship Diagram

No	Simbol	Nama	Deskripsi
1		<i>Strong Entity</i>	Karakteristik atau tipe entitas yang tidak bergantung pada entitas lain.
2		<i>Attribute</i>	Properti atau karakteristik tipe entitas.
3		<i>Multivalued Attribute</i>	Karakteristik tipe entitas yang dapat memiliki nilai lebih dari satu.
4		<i>Relationship</i>	Hubungan antar entitas.
5		<i>Weak Entity</i>	Entitas yang bergantung pada entitas lain.
6		<i>Identifying Entity</i>	Entitas yang merupakan induk dari <i>weak entity</i> .

Tabel II.9 Simbol-simbol *Entity Relationship Diagram* (lanjutan)

No	Simbol	Nama	Deskripsi
7		<i>Associative Entity</i>	Hanya bisa terdapat antara 2 entitas dan menggambarkan hubungan diantara 2 entitas tersebut.
8		<i>Derived Attribute</i>	Suatu atribut yang dihasilkan dari atribut yang lain.

(Sumber: Hoffer, 2010)

Di dalam ERD terdapat *multiplicity*, *multiplicity* merupakan suatu ketentuan (*constraint*) yang mengatur hubungan antara objek-objek entitas. Oleh karena itu, *multiplicity* merupakan kebijakan atau aturan bisnis (*business rule*) yang ditetapkan oleh pemakai atau organisasi, yang dapat berbeda dari satu organisasi dengan organisasi lain. Lebih lanjut, *multiplicity* dapat dibagi menjadi dua batasan atau aturan, yaitu:

1. Batasan kardinalitas

Batasan kardinalitas (*cardinality constraint*) yang menentukan jumlah maksimum objek entitas lain yang dapat berasosiasi dengan suatu objek entitas tertentu.

2. Batasan partisipasi

Batasan partisipasi (*participation constraint*) yang menentukan apakah semua atau sebagian objek entitas berpartisipasi pada suatu jenis hubungan entitas.

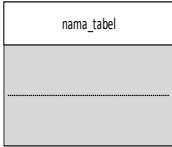
Perbandingan kardinalitas yang dilihat dari dua arah hubungan disebut dengan rasio kardinalitas atau derajat hubungan, terdapat tiga jenis rasio kardinalitas sebagai berikut: (Pahlevi, 2013)

1. *One-to-one* (1:1), jika nilai kardinalitas pada kedua arah 1.
2. *One-to-many* (1:N) atau *many-to-one* (N:1), jika nilai kardinalitas pada salah satu arah satu (1) dan arah lainnya banyak (N).
3. *Many-to-many* (N:N), jika nilai kardinalitas pada kedua arah banyak (N).

2.14 Conceptual Data Model

Menurut Rosa dan Shalahuddin (2015) CDM (*Conceptual Data Model*) atau model konsep data adalah 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. Berikut adalah simbol-simbol yang ada pada CDM:

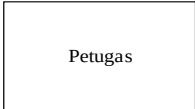
Tabel II.10 Simbol-Simbol *Conceptual Data Model*

Simbol	Deskripsi
Entitas/tabel 	Entitas atau tabel yang menyimpan data dalam basis data
Relasi 	Relasi antar tabel yang terdiri atas nama relasi dan <i>multiplicity</i>

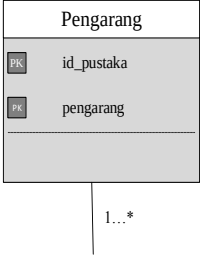
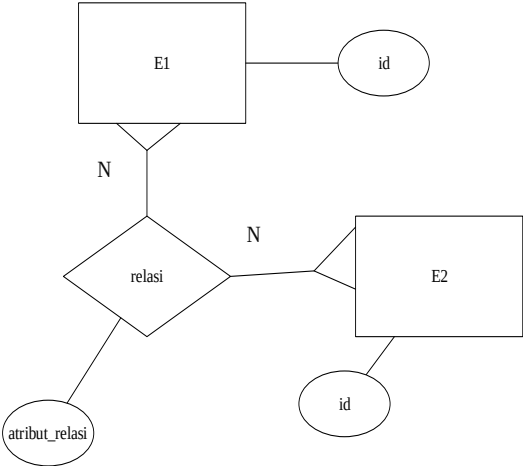
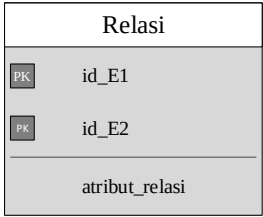
(Sumber: Rosa dan Shalahuddin, 2015)

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

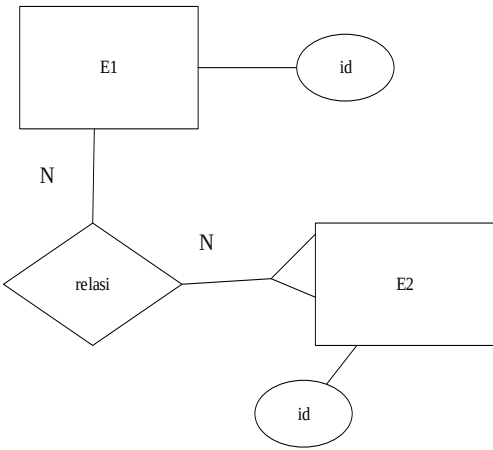
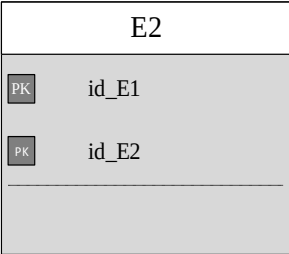
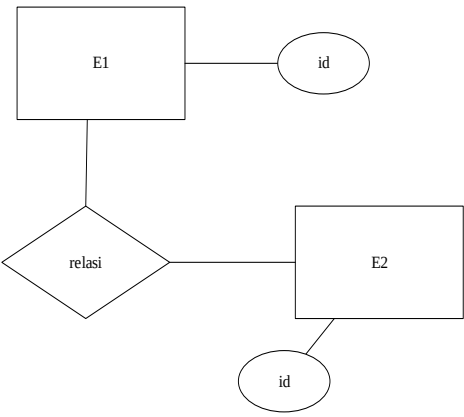
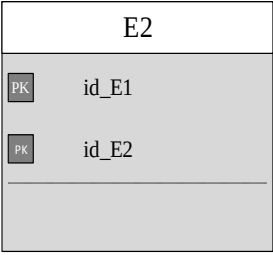
Tabel II.11 Simbol-Simbol Aturan *Conceptual Data Model*

ERD	CDM
 Entitas	 Menjadi sebuah tabel tersendiri

Tabel II.11 Simbol-Symbol Aturan *Conceptual Data Model* (Lanjutan)

 Atribut <i>multivalue</i>	 Menjadi sebuah tabel tersendiri dengan kunci primer (primary key) adalah kunci primer pada entitas dan memiliki atribut dengan nama seperti pada atribut entitas
 Relasi dengan kardinalitas <i>many to many</i>	 Menjadi sebuah tabel tersendiri dengan kunci primer adalah atribut yang menjadi kunci primer di kedua entitas yang direlasikannya.

Tabel II.11 Simbol-Simbol Aturan *Conceptual Data Model* (Lanjutan)

ERD	CDM
 Relasi dengan kardinalitas <i>one to many</i>	 Kunci primer entitas yang memiliki hubungan <i>one</i> akan dijadikan kunci primer di entitas yang memiliki hubungan <i>many</i> dengan kata lain, relasi tidak menjadi tabel sendiri
 Relasi dengan kardinalitas <i>one to one</i>	 Kunci primer salah satu entitas akan dijadikan kunci asing (<i>foreign key</i>) pada tabel yang lain dan kunci asing itu dijadikan kunci primer juga, dengan kata lain, relasi tidak menjadi tabel sendiri

(Sumber: Rosa dan Shalahuddin, 2015)

2.15 Kamus Data

Menurut Jogiyanto (2005) kamus data (*data dictionary*) adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data diharapkan analisis sistem dapat mendefinisikan data yang mengalir dalam sistem dengan lengkap. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem. Kamus data dapat berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan di dalam sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses. Contoh dari kamus data adalah sebagai berikut:

Tabel II.12 Contoh Kamus Data Untuk Tabel Pemasok

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

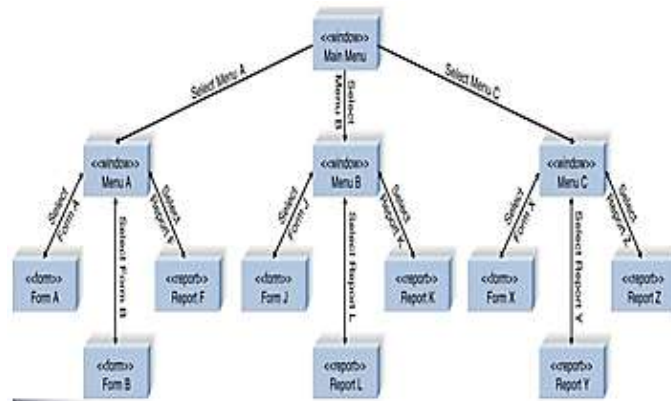
(Sumber: Jogiyanto, 2005)

2.16 Windows Navigation Diagram

Windows Navigation Diagram merupakan *statechart* diagram khusus yang berfokus pada *user interface*. *Windows Navigation Diagram* (WND) menurut Dennis (2010) adalah sebagai berikut:

1. Menunjukkan bagaimana semua layar, formulir, dan laporan terkait.
2. Menunjukkan bagaimana pengguna bergerak dari satu ke yang lain.
3. Seperti diagram keadaan untuk *user interface*
 - a. Kotak mewakili komponen.
 - b. Panah mewakili transisi.
 - c. Stereotipe menunjukkan tipe antarmuka.

Contoh dari *Windows Navigation Diagram* dapat dilihat pada gambar berikut:



Gambar II.8 Contoh *Windows Navigation Diagram*

(Sumber: Dennis, 2010)

2.17 PHP

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1994. PHP (PHP: Hypertext Preprocessor) adalah bahasa *server-sidescripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Karena PHP merupakan *server-side scripting* maka sintaks dan perintah perintah PHP akan dieksekusi di server kemudian hasilnya dikirimkan ke browser dalam format HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh *user* sehingga keamanan dalam *web* lebih terjamin (Sutaji, 2012). Untuk menjalankan sistem PHP dibutuhkan 3 komponen, diantaranya adalah sebagai berikut (Sutaji, 2012):

1. *Webserver*
2. Program PHP
3. *Database Server*

Adapun kelebihan-kelebihan dari bahasa PHP diantaranya adalah sebagai berikut:

1. PHP mudah dibuat dan kecepatan akses tinggi.
2. PHP dapat berjalan dalam *webserver* yang berbeda dalam sistem operasi yang berbeda pula.

3. PHP diterbitkan secara gratis.
4. PHP merupakan bahasa yang dapat diletakkan dalam tag HTML.
5. Sistem *database* yang didukung PHP cukup banyak.
6. PHP termasuk *server side programming*.

2.18 CodeIgniter

CodeIgniter Menurut Hakim (2010) CodeIgniter adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis PHP dibanding jika menulis semua kode program dari awal. CodeIgniter pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc. (<http://ellislab.com>), sebuah perusahaan yang memproduksi CMS (Content Management System) yang cukup handal, yaitu *Expression Engine* (<http://www.expressionengine.com>). Saat ini CodeIgniter dikembangkan oleh Expression Engine Development Team.

Adapun beberapa keuntungan menggunakan CodeIgniter, diantaranya:

1. Gratis

CodeIgniter berlisensi dibawah Apache/BSD opensource.

2. Ditulis menggunakan PHP 4

Meskipun CodeIgniter dapat berjalan di PHP 5, namun sampai saat ini kode program CodeIgniter masih dibuat dengan menggunakan PHP 4.

3. Berukuran kecil

Ukuran CodeIgniter yang kecil merupakan keunggulan tersendiri. Dibanding dengan *framework* lain yang berukuran besar.

4. Menggunakan konsep MVC

CodeIgniter menggunakan konsep MVC yang memungkinkan pemisahan layer *application-logic* dan *presentation*.

5. Memiliki paket *library* yang lengkap

CodeIgniter mempunyai *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh sebuah aplikasi berbasis *web*, misalnya mengakses *database*, mengirim email, memvalidasi form, menangani session dan sebagainya.

6. *Extensible*

Sistem dapat dikembangkan dengan mudah menggunakan *plugin* dan *helper*, atau dengan menggunakan *hooks*.

7. Tidak memerlukan *template engine*

Meskipun CodeIgniter dilengkapi dengan *template parser* sederhana yang dapat digunakan, tetapi hal ini tidak mengharuskan kita untuk menggunakannya.

8. Dokumentasi lengkap dan jelas

Dari sekian banyak *framework*, CodeIgniter adalah satu-satunya *framework* dengan dokumentasi yang lengkap dan jelas.

9. Komunitas

Komunitas CodeIgniter saat ini berkembang pesat. Salah satu komunitasnya bisa dilihat di (<http://CodeIgniter.com/forum/>).

2.19 MySQL

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). MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam basis data sejak lama, yaitu SQL (*Structure Query Language*) (Sutaji, 2012).

2.19.1 Keunggulan MySQL

MySQL memiliki beberapa keunggulan yang dapat dimanfaatkan pengguna, 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 terenkripsi.

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. Struktur tabel

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

2.19.2 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.13 (Sutaji, 2012).

Tabel II.13 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

Tabel II.13 Jenis Data pada MySQL (lanjutan)

Jenis Data	Keterangan
ENUM	Sebuah <i>enumeration</i> . Sebuah obyek string 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
TEXT, BLOB	Sebuah TEXT atau BLOB dengan panjang karakter maksimum 65535 karakter
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
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

(Sumber: Sutaji, 2012)

2.20 Bot Telegram

Bot adalah aplikasi pihak ketiga yang dijalankan di dalam aplikasi Telegram, Telegram Bots adalah akun khusus yang tidak memerlukan nomor telepon tambahan untuk disiapkan.. Pengguna dapat berinteraksi dengan bot dengan mengirim pesan, perintah, dan permintaan sebaris kepada bot. Bot dapat dikontrol melalui permintaan HTTPS ke API bot yang disediakan oleh pihak Telegram. *User* dapat berinteraksi dengan dua cara yaitu:

1. Kirim pesan dan perintah ke bot dengan membuka obrolan dengan bot atau dengan menambahkannya ke grup. Ini berguna untuk bot *chatting* atau bot berita

2. Kirim permintaan langsung melalui *input field* Telegram dengan cara mengetik nama bot dan query yang diinginkan dan bot akan mengirimkan konten yang diminta.

Pesan, perintah, dan permintaan yang dikirim oleh pengguna diteruskan ke perangkat lunak yang berjalan di server developer. Server perantara dari Telegram menangani semua enkripsi dan komunikasi dengan API Telegram untuk pengguna. pengguna berkomunikasi dengan server ini melalui antarmuka HTTPS sederhana yang menawarkan versi sederhana dari API Telegram, yang disebut antarmuka API Bot. (<https://core.Telegram.org/bots>)

Bot Telegram mempunyai sebuah *command* global yang digunakan semua *developer* yang dijadikan syarat sebagai *command* dasar oleh Telegram, *global command* tersebut antara lain:

1. /start

Digunakan untuk mulai berinteraksi dengan pengguna, dengan mengirimkan pesan selamat datang, perintah ini juga dapat digunakan untuk mengirimkan parameter tambahan kepada bot.

2. /help

Memberikan penjelasan bantuan tentang fungsi bot dan apa yang dapat dilakukan oleh bot dan memberikan beberapa *list* perintah yang dapat diberikan kepada bot

3. /setting

Mengembalikan pengaturan bot untuk pengguna yang mengirim perintah tersebut dan menyarankan perintah untuk mengedit pengaturan.

2.21 Functional Testing

Functional testing merupakan bagian dari black box testing, menurut Simarmata (2010) functional testing adalah pengujian yang dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan. Pengujian fungsional meliputi seberapa baik sistem melaksanakan fungsinya, termasuk perintah-perintah pengguna, manipulasi data, pencarian dan proses bisnis, pengguna layar, dan integrasi.

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Didalam melakukan sebuah penelitian diperlukan sebuah cara yang menjadi standar yang digunakan dalam semua disiplin ilmu agar penelitian yang dihasilkan dapat dipertanggung jawabkan maka dari itu diperlukan sebuah metodologi penelitian untuk membantu peneliti dalam melakukan penelitian.

Metodologi penelitian sendiri merupakan tata cara atau kerangka kerja yang digunakan untuk melaksanakan sebuah penelitian. 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 Nusa Indah Jaya Utama.

1. Data Primer

Data Primer adalah data yang diperoleh secara langsung dari perusahaan, dimana pengumpulan data atau informasi dilakukan secara langsung dari objek yang diteliti. Data tersebut adalah data yang digunakan dalam proses pembelian *spare part* mesin *stamping* diantaranya nota permintaan pembelian, analisis sistem yang sedang berjalan, proses bisnis sistem saat ini, 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

Dalam sebuah penelitian dibutuhkan data sebagai bahan uji dan bahan pertimbangan, untuk mengumpulkan data dibutuhkan sebuah metode yang efektif untuk memudahkan peneliti didalam proses pengumpulan data. Dalam penulisan laporan ini penulis menggunakan beberapa metode pengumpulan data sebagai berikut:

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* dan divisi *maintenance* PT Nusa Indah Jaya Utama dengan mengamati proses pembelian *spare part* mesin *stamping* yang berjalan, pengamatan dilakukan mulai dari proses permintaan pembelian *spare part* yang dilakukan divisi *maintenance*, pembelian *spare part* yang dilakukan divisi *purchasing* hingga proses penerimaan *spare part*. 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 divisi terkait tentang proses bisnis yang berjalan pada sistem pembelian *spare part* mesin *stamping*. 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

Permasalahan yang terjadi pada sistem pembelian *spare part* mesin *stamping* di PT Nusa Indah Jaya Utama telah teridentifikasi pada tahap sebelumnya dan akan menjadi dasar pengembangan sistem yang akan dilakukan. Pada penelitian ini untuk mengatasi masalah yang terjadi pada sistem maka akan diadakan sebuah pengembangan sistem. Sistem dikembangkan menjadi lebih terkomputerisasi, menggunakan media Telegram dan berbasis objek. Pengembangan sistem dilakukan dengan menggunakan metode *waterfall*, pengembangan sistem dengan metode *waterfall* terdiri dari tahap *planning*, *analysis*, *design*, dan implementasi.

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

1. *Planning*

Tahap perencanaan merupakan tahap dimana pengembang merencanakan untuk membangun suatu sistem baru dengan memahami konteks bisnis yang berlangsung. Sistem yang dibuat diharapkan dapat membantu proses bisnis yang berjalan.

2. *Analysis*

Tahap analisis dilakukan untuk menganalisis kebutuhan-kebutuhan yang dibutuhkan perusahaan dengan melakukan observasi dan wawancara. Pada tahap ini akan dilakukan analisis data untuk memenuhi kebutuhan sistem baru.

3. *Design*

Tahap ini merupakan tahap dimana dilakukan sebuah desain dari sistem yang akan dibuat menggunakan *tools* perancangan yang tersedia, pada tahap ini akan dibuat desain alur sistem dengan simbol-simbol yang disesuaikan untuk mencapai tujuan yang telah ditetapkan pada proses *planning* dan *analysis*.

4. *Implementation*

Pada tahap ini pengembang membangun sebuah sistem dengan desain yang telah dirancang pada tahap sebelumnya, pada tahap ini desain yang telah dibuat akan diterapkan pada sebuah aplikasi, tahap ini disertai dengan proses pengkodean program.

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. Identifikasi masalah

Pada tahap ini dilakukan penelitian terhadap permasalahan yang terjadi di perusahaan khususnya divisi *purchasing* dalam proses pembelian *spare part* mesin *stamping*, dalam proses pembelian *spare part* mesin *stamping* terdapat beberapa aktivitas yang menyebabkan terhambatnya proses pembelian.

3. Tujuan penelitian

Tujuan penelitian dimaksudkan untuk menjabarkan tujuan akhir yang hendak dicapai dari sistem yang akan dibuat.

4. Batasan masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik mengenai arah pemecahannya dan dapat agar dapat mencapai tujuan yang telah ditetapkan sebelumnya.

5. Penerapan Metode *Waterfall*

a. *Planing*

Proses perencanaan diawali dengan memahami konteks bisnis dari aplikasi yang akan dibuat, serta melakukan perencanaan dari sistem yang akan dibuat dengan menentukan *input* dan *output* dari sistem.

b. *Analysis*

Menganalisis kebutuhan user terhadap program yang akan dibuat dengan mengolah data yang didapat dari proses observasi dan wawancara, dari data

yang didapat, penulis dapat menggambarkan fungsionalitas dari sebuah sistem.

c. *Design*

Pada tahap ini akan dirancang sebuah aplikasi sistem pembelian *spare part* mesin *stamping*. Pada tahap *design* nantinya akan memuat proses sebagai berikut:

- 1) Memodelkan sistem menggunakan *Unified Modeling Language* (UML) yaitu menggunakan *usecase diagram*, *activity diagram*, *sequence diagram*, *class diagram*, dan *deployment diagram*.
- 2) Memodelkan basis data dengan menggunakan *Entity Relationship Diagram* (ERD), *Conceptual Data Model* (CDM), dan kamus data.
- 3) Merancang sistem dengan menggunakan *Windows Navigation Diagram* dan rancangan antar muka dan perancangan bot Telegram.

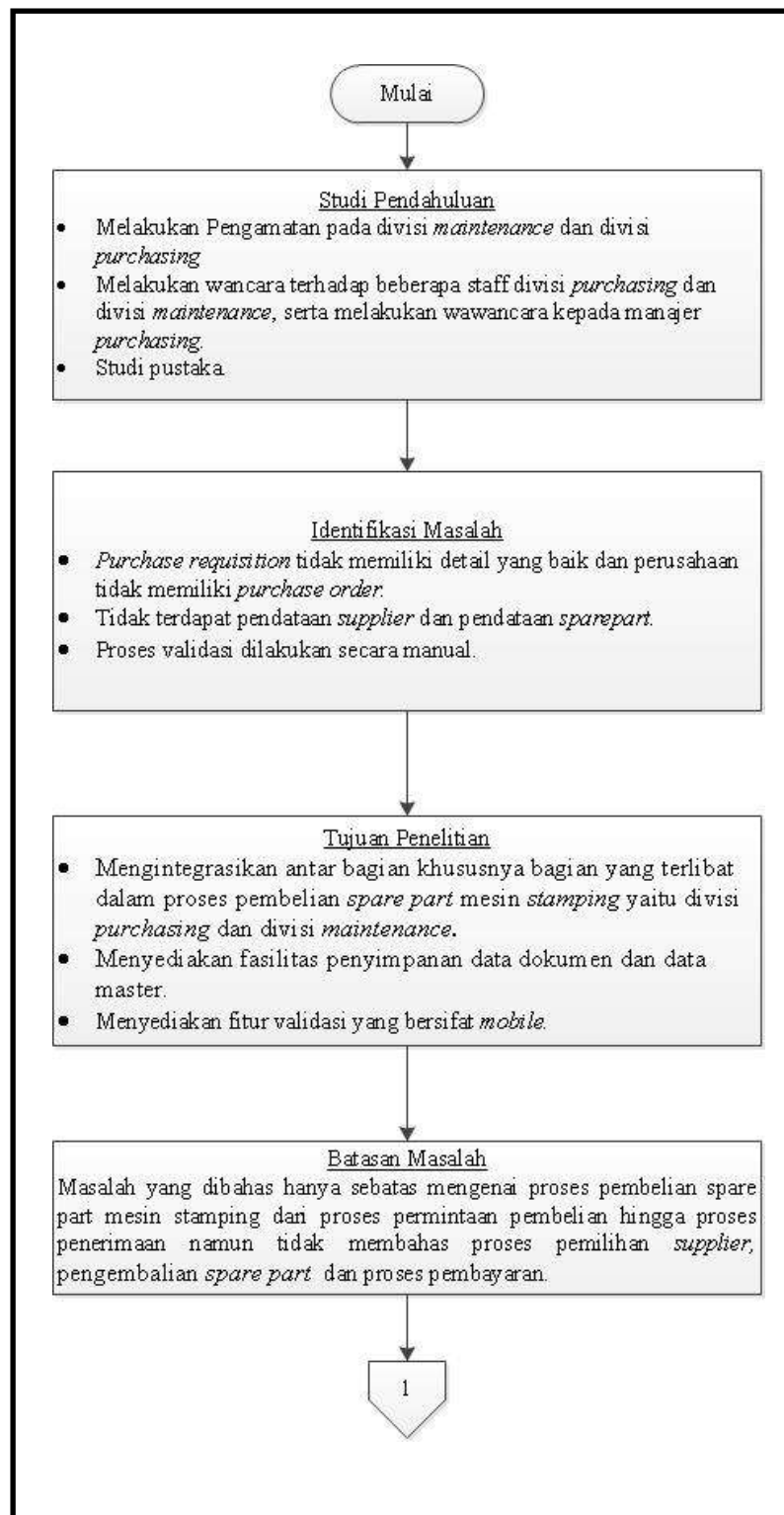
d. *Implementation*

Dalam tahap ini dokumentasi dari tahap sebelumnya akan dituangkan ke sebuah aplikasi dengan melakukan pengkodean untuk membuat sebuah aplikasi sistem pembelian *spare part* mesin *stamping* dengan menggunakan CodeIgniter 3.1.4 dan MySQL 5.6.20 sebagai *database*. Dan melakukan pengkodean Bot Telegram sebagai media penyalur informasi.

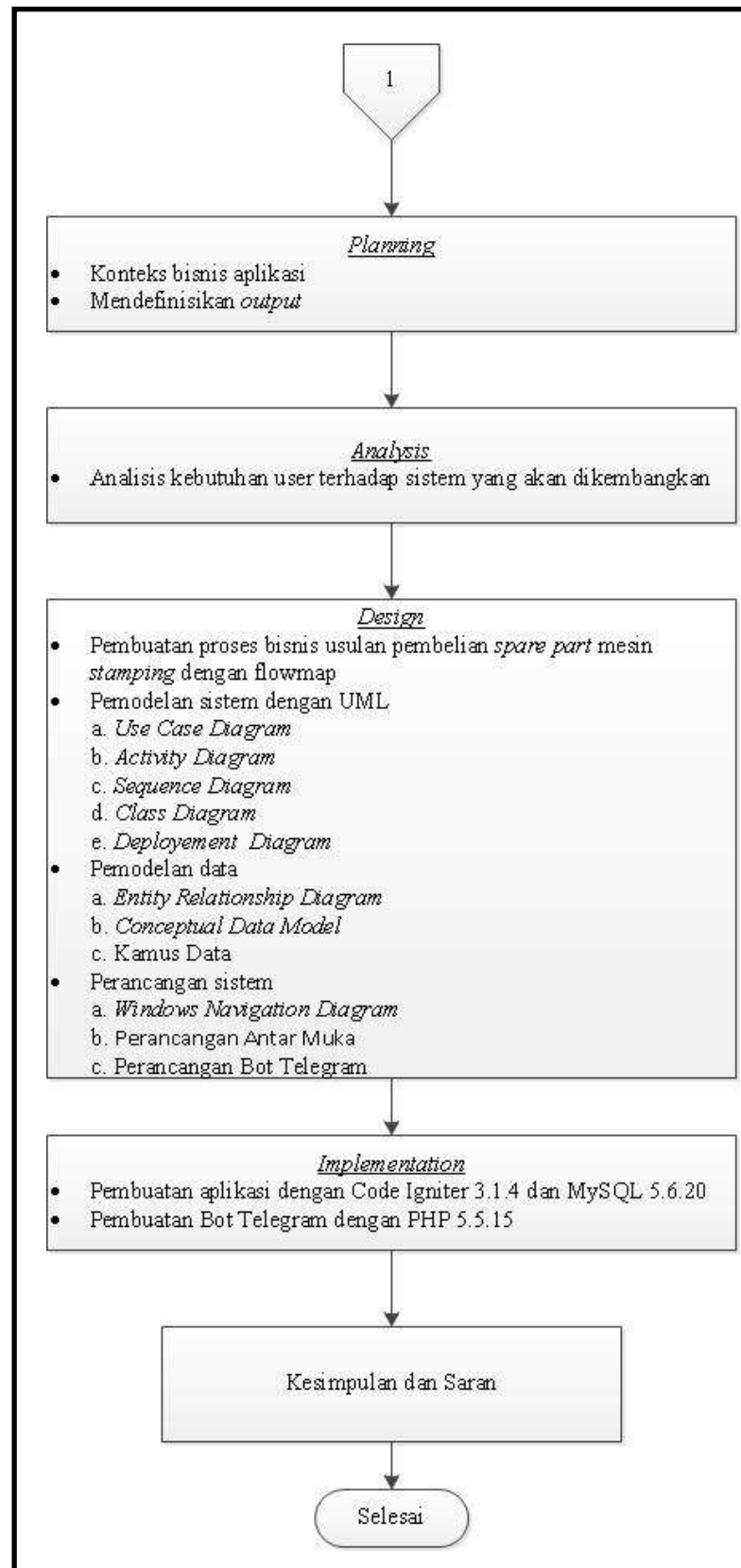
6. Kesimpulan dan Saran

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 Tentang Perusahaan

PT Nusa Indah Jaya Utama didirikan pada tahun 1974 oleh H. Muhammad Kusnadi pada awal berdiri perusahaan ini masih berbentuk CV dengan nama CV Nusa Indah dan bergerak dalam penanganan limbah besi dari kapal-kapal tua. Pada tahun 1976 CV Nusa Indah bekerjasama dengan PT Mitsubishi Kramayudha Motors dalam penanganan limbah industri yg berupa material logam yang berasal dari produksi *stamping press* dan dikelola untuk menjadi bahan baku peleburan baja yang ada di kawasan industri Pulo Gadung dan Tegal dengan dengan jumlah limbah logam industri kurang lebih 700 ton dalam satu bulan, limbah industri tersebut dijual ke peleburan logam untuk bahan baku besi siku. selain itu, CV Nusa Indah juga memberikan *men power (outsourcing)* kepada PT Mitsubishi Kramayudha Motors. Pada tahun 1978 CV Nusa Indah telah menjadi rekanan tetap PT Mitsubishi Kramayudha Motors khususnya di bagian penanganan pengerjaan *part* komponen mobil *colt*.

Pada tahun 1991 atas kebijakan pemerintah untuk setiap industri otomotif yang ada di Indonesia khususnya PMA (penanam modal asing) diwajibkan memberikan *support* kepada industri dalam negeri dalam rangka untuk berkembangnya pembuatan komponen lokal. Sehingga CV Nusa Indah ditunjuk oleh PT Mitsubishi Kramayudha Motors untuk melakukan pekerjaan yang menggunakan mesin PP ukuran 25 ton, 40 ton, 60 ton dan 80 ton yaitu untuk produksi *clip harness*, parabola dan kulkas merek Mitsubishi. Pada tahun 1991 CV Nusa Indah pindah lokasi di jalan pulogebang dengan luas area 1350m² dan menjadi *home industry* dalam pengerjaan komponen Mitsubishi, setelah itu CV Nusa Indah semakin berkembang sehingga mendapat kepercayaan dan bekerja sama dengan PT Daihatsu Motor Indonesia, PT Isuzu Motor Indonesia dan PT Inti Pantja Press Industry. Lalu pada tahun 2007 CV Nusa Indah berubah bentuk menjadi Perseroan Terbatas (PT) dan menambah nama perusahaan menjadi PT

Nusa Indah Jaya Utama berdasarkan keputusan menteri Hukum dan Hak Asasi Manusia Republik Indonesia dan juga merubah anggaran dasar perseroan dengan Nomor: AHU-12765.AH.01.02. Tahun 2012. Kemudian Surat Izin Usaha Perdagangan (SIUP) untuk PT Nusa Indah Jaya Utama sendiri keluar pada tahun 2013 dengan Nomor: 510/656-BPPT/PM/IX/2013. Dan perusahaan berpindah lokasi ke Jalan Laskar 49 Pekayon, Bekasi Selatan.

Selain masih menjadi *vendor* utama untuk PT Mitsubishi Kramayudha Motors, PT Nusa Indah Jaya Utama juga menjalin kerjasama dengan ASTRA Group. Pada tahun 2013 PT Nusa Indah Jaya Utama mulai menggunakan mesin *medium press* yang didatangkan dari China dan Korea dalam mengerjakan pesanan PT Garmak Motor, mobil *Chevrolet* serta motor TVS buatan India yang komponennya sebagian besar dibuat PT Nusa Indah jaya utama. Pada tahun 2014 masuk mesin *high press* dengan kapasitas 200 – 315 ton untuk menjadi *vendor* PT Gemala Kempa Daya dan PT Fuji Teknik Indonesia. Berikut adalah lambang PT Nusa Indah Jaya Utama:



Gambar IV.1 Lambang PT Nusa Indah Jaya Utama
(Sumber: PT Nusa Indah Jaya Utama, 2017)

4.2 Profil Perusahaan

Nama Perusahaan	: PT Nusa Indah Jaya Utama
Lokasi	: Jl. Laskar Raya No. 49 RT. 003 RW. 002 Kel. Pekayon Jaya, Kec. Bekasi Selatan
Telepon	: 021-82411782 / 021-8201008
Fax	: 021-82411782
Direktur Utama	: Bpk. H. M. Kusnadi
Email	: PTniju@gmail.com
Aktivitas Bisnis	: <i>Stamping</i> dan <i>Manufacturing</i>
Status	: Perseroan Terbatas
Produksi	: Komponen Otomotif
Perizinan	: 1. SIUP: 510/656-BPPT/PM/IX/2013 2. NPWP No: 02.182.710.0-006.000 3. TDP: 102614611073 4. AHU: 12765.AH.02. Tahun 2012

4.3 Visi dan Misi Perusahaan

Visi merupakan pandangan jauh tentang suatu perusahaan ataupun lembaga dan lain-lain yang ingin dicapai pada masa yang akan datang. Visi juga dapat diartikan sebagai tujuan perusahaan atau lembaga dan apa yang harus dilakukan untuk mencapai tujuannya tersebut pada masa depan.

Misi merupakan suatu pernyataan tentang apa yang harus dikerjakan oleh perusahaan atau lembaga dalam usaha mewujudkan visi tersebut. Misi perusahaan diartikan sebagai tujuan dan alasan mengapa perusahaan atau lembaga itu dibuat. Misi juga akan memberikan arah sekaligus batasan-batasan proses pencapaian tujuan.

PT Nusa Indah Jaya Utama merupakan perusahaan industri yang melayani jasa *stamping* untuk pembuatan komponen-komponen otomotif roda dua maupun roda empat. PT Nusa Indah Jaya Utama mempunyai moto *the future start from*

here, are you ready for participation. PT Nusa Indah Jaya Utama mempunyai Visi dan Misi Sebagai berikut:

Visi:

Menjadi perusahaan manufaktur yang menyediakan jasa *stamping* komponen-komponen otomotif baik roda dua maupun roda empat.

Misi:

Membangun perusahaan yang mampu menciptakan lapangan kerja berkualitas bagi sebanyak mungkin rakyat Indonesia dengan mengusung nilai-nilai berikut:

1. Pengembangan kompetensi karyawan secara berkelanjutan.
2. Mengupayakan pertumbuhan finansial, intelektual dan citra perusahaan yang konsisten serta melakukan investasi kembali ke dalam bisnis yang dijalankan.
3. Mempertahankan standar kode etik yang tinggi dalam aktivitas bisnis.

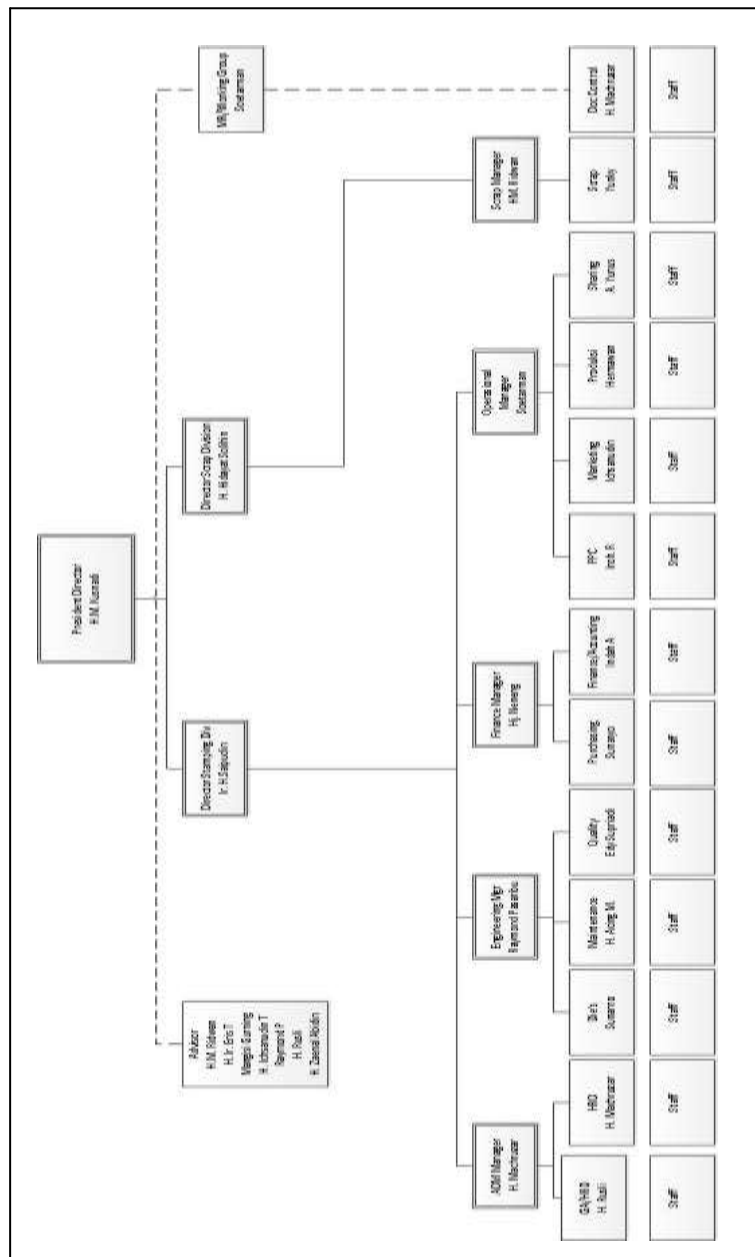
4.4 Struktur Organisasi Perusahaan

Struktur Organisasi adalah suatu susunan dan hubungan antara tiap bagian baik secara posisi maupun tugas yang ada pada perusahaan dalam menjalin kegiatan operasional untuk mencapai tujuan. Dengan adanya struktur organisasi maka akan memberikan informasi kepada seluruh manusia yang menjadi anggotanya untuk mengetahui kegiatan atau pekerjaan yang harus ia kerjakan, berkonsultasi atau bertanggung jawab kepada siapa, sehingga proses kerjasama menuju pencapaian tujuan organisasi dapat terwujud sesuai dengan perencanaan yang telah ditetapkan sebelumnya.

Struktur organisasi sebagai suatu mekanisme formal di mana sebuah organisasi dikelola. Struktur ini mengandung unsur-unsur spesialisasi kerja, standarisasi, koordinasi, sentralisasi/desentralisasi dalam pembuatan keputusan dan ukuran satuan kerja.

Struktur Organisasi pada PT Nusa Indah Jaya Utama memakai struktur organisasi lini/garis. Dengan alasan dalam struktur organisasi PT Nusa Indah Jaya Utama adanya perintah terpusat pada atasan langsung dan koordinasi dengan bawahannya langsung sehingga perintah jelas dan mudah di laksanakan

1. Mempunyai wewenang dan tanggung jawab secara langsung secara vertikal yang dikaitkan dengan jabatan dan tugas tiap tingkatan.
2. Adanya kesatuan komando atau perintah.
3. Hanya cocok untuk organisasi yang kecil atau sederhana.



Gambar IV.2 Struktur Organisasi PT Nusa Indah Jaya Utama
(Sumber: PT Nusa Indah Jaya Utama, 2017)

4.5 *Job Description*

Job description adalah sebuah kumpulan informasi jabatan dan disusun secara sistematis yang dapat mengidentifikasi dan menguraikan suatu jabatan atau posisi tertentu. *Job description* membuat status setiap jabatan menjadi jelas akan fungsi dan perannya, hasilnya, serta tanggung jawabnya. Dari struktur organisasi diatas, maka *job description* dari masing-masing bagian yang ada di PT Nusa Jaya Indah Utama adalah sebagai berikut:

1. *President Director*, mempunyai tugas dan tanggung jawab sebagai berikut:
 - a. Bertanggung jawab terhadap semua aktivitas yang berlangsung di perusahaan.
 - b. Memimpin dan mengendalikan aktivitas perusahaan.
 - c. Berkoordinasi dengan semua kepala divisi untuk menentukan target produksi.
 - d. Bertanggung jawab terhadap kemajuan sumber daya manusia yang ada di perusahaan.
2. *Director*, mempunyai tugas dan tanggung jawab sebagai berikut:
 - a. Bertanggung jawab kepada *president director* dalam hal penanganan Sistem Manajemen Mutu serta Manajemen Lingkungan.
 - b. Menentukan dan menetapkan efisiensi perusahaan.
 - c. Bertanggung jawab terhadap kemajuan sumber daya manusia kepada *President Director*.
3. Bagian *Administration*, tanggung jawab dan wewenangnya adalah:
 - a. Bertanggung jawab atas fungsi operasional manajemen sumber daya manusia (lingkup pengadaan, pelatihan dan pemeliharaan).
 - b. Bertanggung jawab atas perbuatan dan penyempurnaan kebijakan dan prosedur perusahaan.
 - c. Berwenang memberi sanksi kepada karyawan yang melanggar sistem mutu.
 - d. Bertanggung jawab atas fungsi sistem penggajian.
 - e. Bertanggung jawab mengelola fungsi perizinan.
 - f. Bertanggung jawab mengelola fungsi umum lainnya
 - g. Bertanggung jawab mengelola fungsi rumah tangga

- h. Bertanggung jawab fungsi transportasi
 - i. Bertanggung jawab mengelola fungsi pemeliharaan kebersihan pabrik / kantor.
4. Bagian *Finance*, tugasnya adalah:
- a. Mendukung dan melakukan koordinasi operasional dengan semua fungsi / departemen untuk mencapai target manual *business plan* yang sudah ditetapkan.
 - b. Mengkoordinir dan bertanggung jawab dalam pembuatan manual *business plan* dan *financial budget*
 - c. Bertanggung jawab melakukan pembelian *spare part*, bahan baku, dan kebutuhan operasional perusahaan lainnya seperti sarung tangan, kain majun, seragam dan lain-lain.
 - d. Bertanggung jawab melakukan pembelian barang dengan mutu yang tepat.
 - e. Bertanggung jawab mencari *supplier* dengan melakukan penawaran harga yang tepat.
5. Bagian Operasional, bertanggung jawab terhadap:
- a. Tersedianya *material* produksi sesuai dengan persyaratan yang telah ditentukan
 - b. Mencatat material dan komponen jadi yang masuk dan keluar
 - c. Perencanaan program produksi dan pengendaliannya dengan memperhatikan stok dan jadwal pengiriman yang telah ditetapkan
 - d. Pembuatan program produksi, seperti *core*, *hand mould*, *shoot blast* dan *finishing*.
 - e. Bertanggung jawab terhadap target hasil produksi serta kualitas produk yang dihasilkan.
 - f. Berkoordinasi dengan bagian *marketing* dalam hal penentuan jenis produksi.
 - g. Bertanggung jawab terhadap pemakaian - pemakaian bahan kimia serta efisiensi pada bagian produksi. Memastikan kebutuhan pelanggan telah tercapai
 - h. Menjamin kepuasan pelanggan

- i. Berkewajiban menentukan harga jual produk atas dasar *cost estimate* dari *finance*.
 - j. Mencari mitra kerja
6. Bagian *Engineering*, bertanggung jawab terhadap:
- a. Pengembangan sistem *maintenance* dan *engineering* untuk menjamin mesin dan peralatan produksi dapat dioperasikan sesuai dengan jadwal dan *standard* mutu yang telah ditetapkan
 - b. Pengawasan kegiatan *maintenance* pada mesin dan peralatan produksi untuk mencegah dan meminimalisasi timbulnya kerusakan pada saat mesin dan peralatan beroperasi.
 - c. Mutu produk yang akan dikirim ke pelanggan
 - d. Penghentian proses produksi, jika ditemukan ketidaksesuaian pada proses.
 - e. Keakurasian alat ukur / tes yang digunakan
 - f. Mutu barang yang masuk
 - g. Penentuan kualifikasi personel dibagiannya.
7. Bagian Scrap, bertanggung jawab terhadap:
- a. Pengumpulan *scrap* atau sampah material hasil produksi
 - b. Penyimpanan *scrap* di gudang *scrap*
 - c. Penjualan *scrap*

4.6 Mesin-mesin Produksi Perusahaan

PT Nusa Indah Jaya Utama menggunakan mesin *press* sebagai mesin utama yang digunakan untuk tetap menjadi *supplier* terpercaya beberapa perusahaan di Indonesia. Proses *stamping* tidak akan berjalan jika tidak terdapat mesin *press* ini sendiri. PT Nusa Indah Jaya Utama memiliki 3 jenis mesin *press*, yaitu dari jenis *portable press* dengan kekuatan 25 Ton hingga 80 Ton, untuk jenis *medium press* berkekuatan 110 hingga 160 Ton, sedangkan untuk jenis *high press* berkekuatan 200 hingga 315 Ton. Mesin-mesin *press* digunakan untuk proses membuat produk antara lain seperti Proses *Blank* (memotong material), Proses *Drawing* (pembentukan material), Proses *Triming* (pemotongan material setelah dibentuk), Proses *Flange* (membentuk sisi-sisi produk), Proses *Piercing* (proses

pembolongan produk). Selain mesin *press* terdapat beberapa mesin yang digunakan pendukung produksi perusahaan seperti untuk proses *repairing* produk.

Dalam proses pembelian *spare part*. *Spare part* yang dibeli oleh divisi *purchasing* digunakan untuk memenuhi kebutuhan mesin-mesin ini. Berikut daftar mesin-mesin yang digunakan PT Nusa Indah Jaya Utama dalam proses produksinya:

Tabel IV.1 Data Mesin PT Nusa Indah Jaya Utama

No	Nama Mesin	Spesifikasi	Unit
1	<i>Press Portable</i>	25 Ton	2
2	<i>Press Portable</i>	35 Ton	2
3	<i>Press Portable</i>	40 Ton	1
4	<i>Press Medium</i>	110 Ton	1
5	<i>Press Medium</i>	160 Ton	5
6	<i>Press Medium</i>	200 Ton	2
7	<i>Press Medium</i>	250 Ton	1
8	<i>Press Medium</i>	315 Ton	2
9	<i>Spot Machine</i>	35 KVA	2
10	<i>Grinding Machine</i>	Bad 500 x 750	1
11	<i>Drilling Maching</i>	Cap 12 mm	2
12	<i>Drilling Maching</i>	Cap 13 mm	1
13	<i>Drilling Maching</i>	Cap 16 mm	4
14	<i>Tapping Machine</i>	Cap 12 mm	1
15	<i>Saeing Machine Pipe</i>		1
16	<i>Compressor Swan</i>	<i>Big Capacity</i>	1
17	<i>Compressor Tab</i>	<i>Small Capacity</i>	2

(Sumber: PT Nusa Indah Jaya Utama, 2017)

4.7 *Spare Part*

Untuk menjaga kelangsungan produksi perusahaan PT Nusa Jaya Indah Utama membutuhkan perawatan mesin secara berkala untuk menghindari terjadinya kerusakan mesin, namun jika kerusakan mesin produksi telah terjadi

diharapkan proses perbaikan tidak membutuhkan waktu lama, untuk itu divisi *purchasing* diharapkan dapat memenuhi kebutuhan *spare part* tepat waktu, agar proses produksi terus berjalan. Selain itu divisi *purchasing* juga harus dapat mengetahui umur pemakaian *spare part* pada sebuah mesin untuk dapat menyediakan *spare part* tersebut di gudang perusahaan agar waktu penggantian *spare part* tidak terlalu lama. Beberapa *spare part* yang dibutuhkan oleh PT Nusa Jaya Indah Utama untuk mesin-mesin produksinya antara lain:

Tabel IV.2 Data *Spare part* PT Nusa Jaya Indah Utama

No	Nama	Fungsi	Gambar
1	<i>Seal</i>	Menjaga kebocoran pelumas (lubrikasi). Menjaga kotoran dan material lain masuk ke sistem. Memberikan batasan cairan supaya tidak tercampur.	
2	<i>Solenoid</i>	Digunakan untuk menggerakkan tabung cylinder.	

Tabel IV.2 Data *Spare part* PT Nusa Jaya Indah Utama (lanjutan)

No	Nama	Fungsi	Gambar
3	Oli	Digunakan sebagai bahan pelumas agar mesin berjalan bebas gangguan. Dan berfungsi sebagai pendingin dan penyekat.	
4	V Belt	Meneruskan putaran atau menghubungkan bagian-bagian mesin	
5	Auto Drain	Membuang kandungan air yang terdapat pada saluran pipa udara pada mesin <i>press</i>	
6	Contactor	Menghubungkan atau memutuskan aliran listrik pada mesin agar tidak terjadi korsleting	

(Sumber: PT Nusa Indah Jaya Utama, 2017)

2. Nota permintaan pembelian yang berisi permintaan pembelian yang dibuat oleh divisi *maintenance* berdasarkan hasil laporan analisis kerusakan mesin yang ditujukan untuk divisi *purchasing* sebagai acuan divisi *purchasing* dalam melakukan pembelian. Dokumen permintaan pembelian yang dimiliki PT Nusa Indah Jaya Utama memiliki kekeliruan penamaan, dalam nota permintaan

pembelian tertulis judul bukti pengeluaran kas yang merupakan bukti jika kas dari perusahaan telah keluar. Contoh dari dokumen ini adalah sebagai berikut



Gambar IV.4 Nota Permintaan Pembelian
(Sumber: PT Nusa Indah Jaya Utama, 2017)

Keterangan:

Rp. :Keterangan harga barang atau jumlah uang yang dibutuhkan dengan angka.

Dengan huruf :Keterangan harga barang atau jumlah uang yang dibutuhkan dengan huruf.

Keperluan :Keterangan barang yang dibutuhkan.

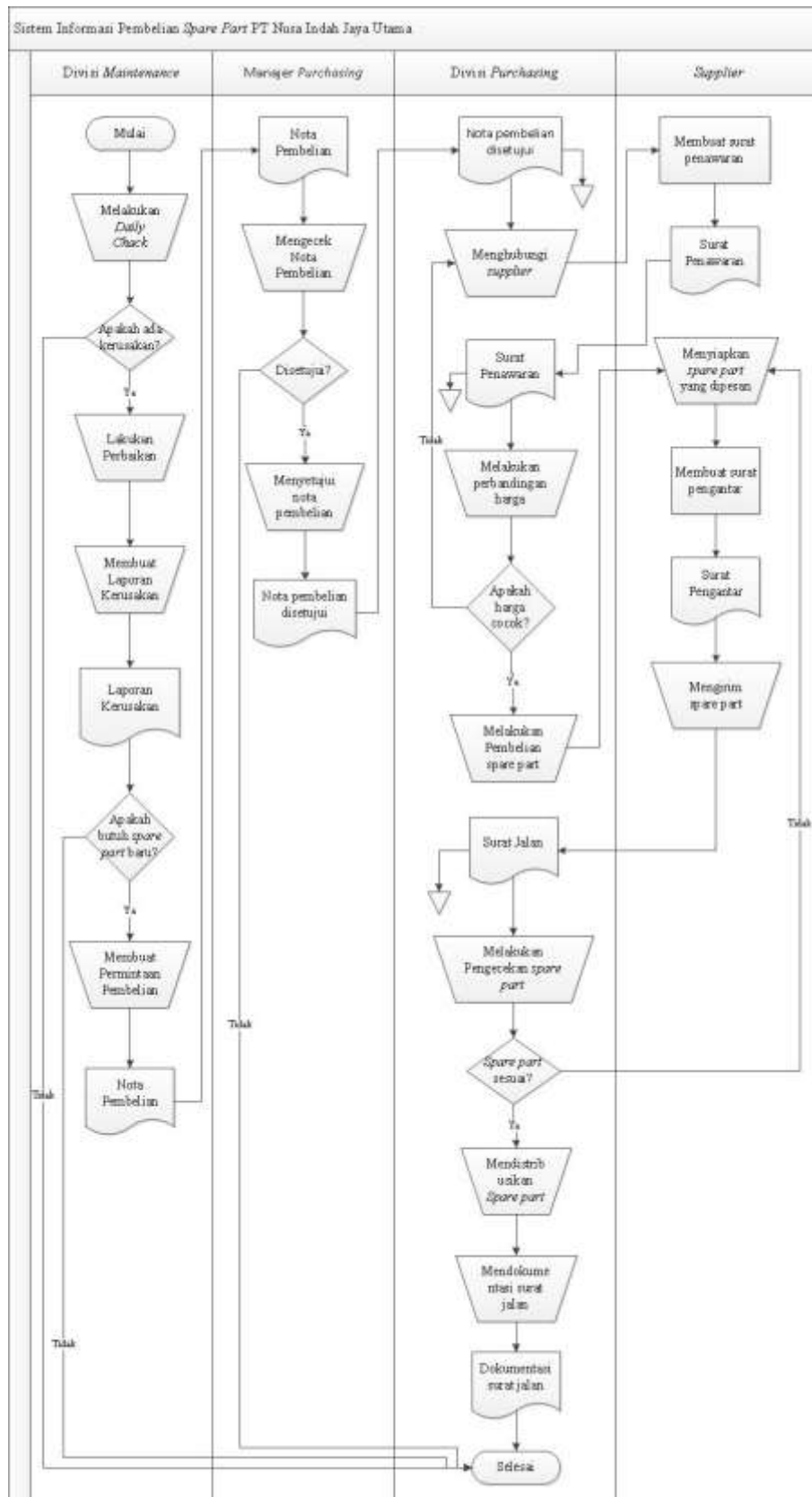
3. Surat penawaran dari *supplier* yang berisi penawaran barang yang mungkin dibutuhkan oleh perusahaan, surat penawaran dibuat berdasarkan permintaan yang dilakukan divisi *purchasing* kepada *supplier* bersangkutan melalui telepon. Contoh dari surat penawaran adalah sebagai berikut:

4.9 Prosedur Pembelian *Spare Part*

Dalam proses pembelian *spare part* mesin *stamping* terdapat beberapa tahapan yang dilakukan divisi-divisi terkait, prosedur sistem informasi pembelian *spare part* mesin *stamping* di PT Nusa Indah Jaya Utama adalah sebagai berikut:

1. Divisi *maintenance* melakukan *daily check*, jika ada kerusakan lalu diperbaiki jika memungkinkan namun jika mesin yang rusak membutuhkan *spare part* baru, divisi *maintenance* akan membuat hasil analisis kerusakan dan juga solusi yang direkomendasikan kepada manajer *engineering* beserta *spare part* yang dibutuhkan untuk melakukan perbaikan mesin *stamping*.
2. Untuk sebuah pengadaan *spare part* yang dibutuhkan, divisi *maintenance* harus membuat permintaan pembelian *spare part* berupa nota pembelian yang ditujukan kepada divisi *purchasing*, dimana pembelian telah diketahui oleh manajer *engineering*.
3. Setelah nota diterima oleh divisi *purchasing* selanjutnya manajer *purchasing* akan melakukan pengecekan terhadap nota pembelian dan melakukan penyetujuan pembelian dengan menandatangani nota pembelian.
4. Setelah harga disetujui divisi *purchasing* akan melakukan pembelian *spare part* melalui kontak telepon kepada *supplier*.
5. Setelah pembelian dilakukan *spare part* akan dikirimkan beserta surat jalan untuk dilakukan pengecekan apakah barang yang dikirim sesuai dengan barang yang diminta.
6. Jika *spare part* yang dikirim telah sesuai maka akan dilakukan pendistribusian *spare part* ke divisi *maintenance*, namun jika *spare part* yang dikirimkan *supplier* tidak sesuai dengan pesanan, maka *spare part* akan dikembalikan kepada *supplier* untuk ditukar dengan *spare part* yang sesuai dengan pesanan perusahaan.

Berikut merupakan aliran data dari sistem informasi pembelian *spare part* PT Nusa Indah Jaya Utama pada Gambar IV.6:



Gambar IV.7 Aliran Data Sistem Informasi Pembelian Spare Part
(Sumber: Hasil Analisis, 2018)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Pengguna

Berdasarkan hasil pengolahan data dan analisis sistem pada divisi *purchasing* PT Nusa Indah Jaya Utama maka dibutuhkan sebuah perancangan aplikasi sistem informasi pembelian *spare part* mesin *stamping*. Tabel V.1 berikut merupakan *system requirement* untuk aplikasi sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.1 *System Requirement*

No	Masalah	Kebutuhan User	Solusi	Kebutuhan sistem	
				Kebutuhan <i>Functional</i> Sistem	Kebutuhan <i>non-functional</i>
1	Tidak terdapat media pengolahan data yang terhubung dengan basis data yang dapat mengintegrasikan informasi antar bagian.	Sistem dengan database terkomputerisasi seperti MySQL yang dapat memberikan layanan pengolahan data yang dapat saling terhubung antar bagian	Merancang dan membangun suatu sistem yang membantu pengguna dalam melakukan pengelolaan data	Sistem dapat memberikan layanan pengelolaan data master yaitu berupa data <i>spare part</i> dan data <i>supplier</i> .	Data master hanya dapat dikelola oleh staff <i>purchasing</i> .

Tabel V.1 *System Requirement* (lanjutan)

No	Masalah	Kebutuhan User	Solusi	Kebutuhan Sistem	
				Kebutuhan <i>Functional</i> Sistem	Kebutuhan <i>non-functional</i>
				Sistem dapat memberikan layanan pengelolaan data transaksi berupa transaksi pembelian <i>spare part</i> mesin <i>stamping</i> dan <i>purchase order</i>	Data transaksi yang disediakan dalam sistem informasi pembelian <i>spare part</i> mesin <i>stamping</i> yaitu permintaan pembelian yang dilakukan oleh staff <i>maintenance</i> . selanjutnya data <i>purchase order</i> , dan data surat jalan. Dikelola oleh staff <i>purchasing</i>
2	Belum terdapat fasilitas yang menyediakan pengolahan data permintaan pembelian dan <i>purchase order</i> yang baik	Pengelolaan data permintaan pembelian memiliki detail yang baik, dan pengelolaan <i>purchase order</i> .	Membuat pengelolaan data permintaan pembelian terkomputerisasi dan menyediakan form permintaan yang detail dan membuat <i>purchase order</i> terkomputerisasi	Sistem menyediakan fitur pengelolaan permintaan pembelian.	Data yang dikelola adalah nama permintaan <i>sparepart</i> dan jumlah permintaan, tanpa harga yang diajukan. Permintaan pembelian hanya dapat diakses oleh staff <i>maintenance</i> .

Tabel V.1 *System Requirement* (lanjutan)

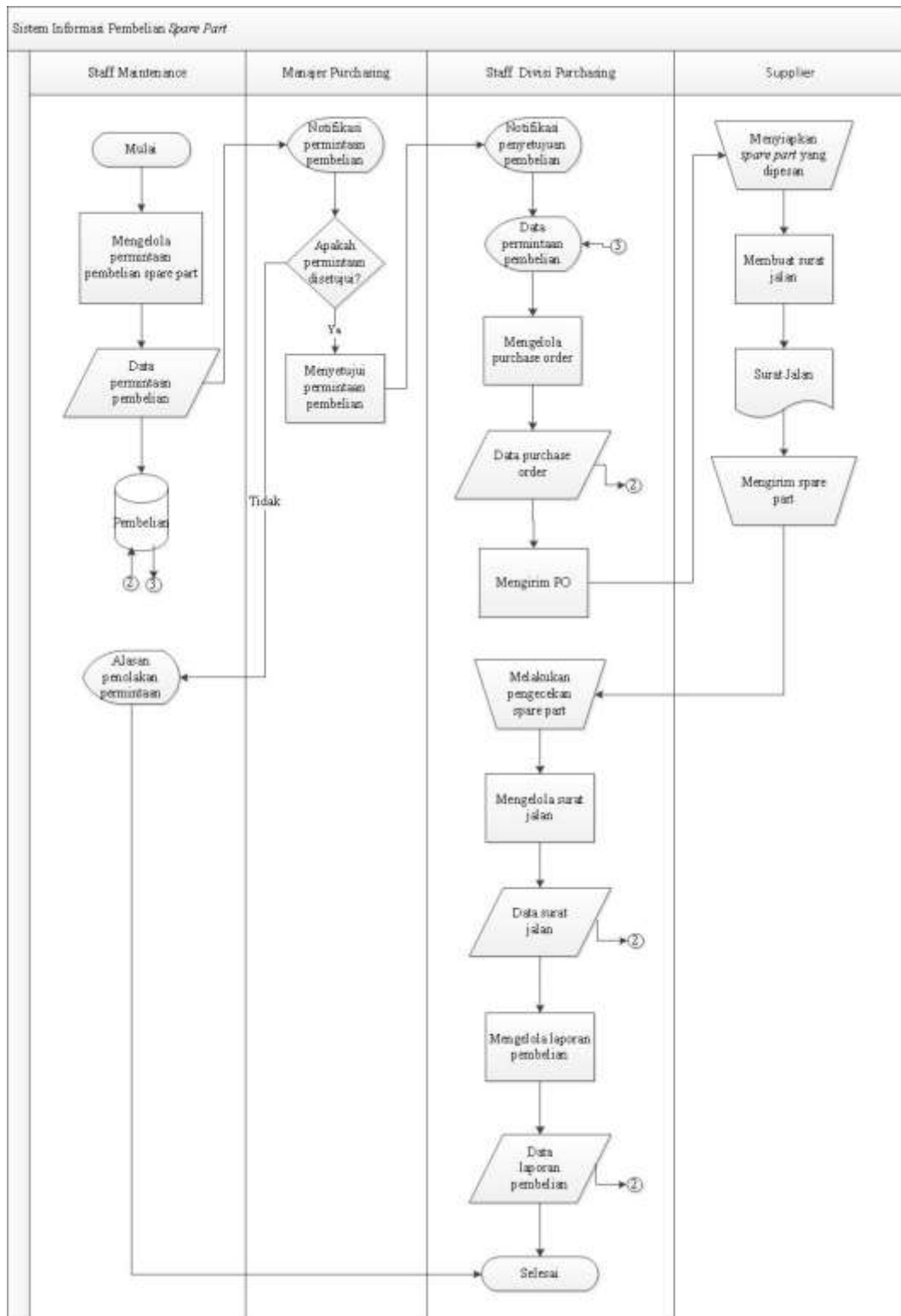
No	Masalah	Kebutuhan User	Solusi	Kebutuhan Sistem	
				Kebutuhan <i>Functional</i> Sistem	Kebutuhan <i>non-functional</i>
				Sistem menyediakan fitur pengelolaan <i>purchase order</i> .	PO dibuat berdasarkan permintaan detail dengan tanpa harga <i>sparepart</i> . <i>Fitur pengelolaan purchase order</i> hanya dapat dikelola oleh staff <i>purchasing</i>
3	Belum terdapat proses validasi yang bersifat <i>mobile</i> ataupun terkomputerisasi	Proses validasi dilakukan secara otomatis melalui komputer atau <i>handphone</i>	Membuat proses validasi yang terkomputerisasi serta terhubung dengan database melalui aplikasi <i>mobile</i> .	Sistem menyediakan fitur validasi yang dapat digunakan dimana saja.	Menyediakan proses validasi yang hanya dapat dilakukan oleh manajer <i>purchasing</i> dengan aplikasi ataupun menggunakan aplikasi pesan Telegram.

(Sumber: Hasil Analisis, 2018)

5.2 Prosedur Usulan Sistem Informasi Pembelian *Spare part* Mesin *Stamping*

Prosedur sistem informasi pembelian *spare part* mesin *stamping* yang diusulkan sebagai bentuk perbaikan sistem informasi pembelian di PT Nusa Indah Jaya Utama melalui tahapan-tahapan sebagai berikut:

1. Staff divisi *maintenance* membuat *purchase request* yang telah disediakan oleh sistem berisi detail *spare part* yang dibutuhkan untuk melakukan perbaikan, selanjutnya data akan tersimpan di *database*.
2. Setelah *purchase request* tersimpan maka sistem akan memberikan notifikasi didalam aplikasi web dan notifikasi melalui aplikasi pesan Telegram dengan akun @nijubot kepada manajer *purchasing* yang menginformasikan bahwa terdapat permintaan pembelian baru.
3. Jika pembelian yang diminta disetujui oleh manajer *purchasing* selanjutnya akan terdapat notifikasi kepada staff divisi *purchasing* untuk membuat *purchase order* sesuai dengan *purchase request* yang telah distujui oleh manajer.
4. Setelah *purchase order* diterima *supplier* akan menyiapkan barang yang dipesan untuk dikirimkan.
5. Setelah barang dikirimkan perusahaan akan mendapatkan surat jalan yang berisi detail barang.
6. Data dari surat jalan akan diinput kedalam sistem dan tersimpan didalam *database*.
7. Setelah *input* surat jalan selanjutnya adalah mengupdate data stok *spare part* yang dimiliki perusahaan.
8. Setiap pembelian yang masuk kedalam *database* dapat diolah berdasarkan bulan untuk dijadikan sebuah laporan pembelian setiap bulan, yang akan diserahkan kepada pemimpin perusahaan.



Gambar V.1 *Flowmap* usulan
(Sumber: Hasil Analisis, 2018)

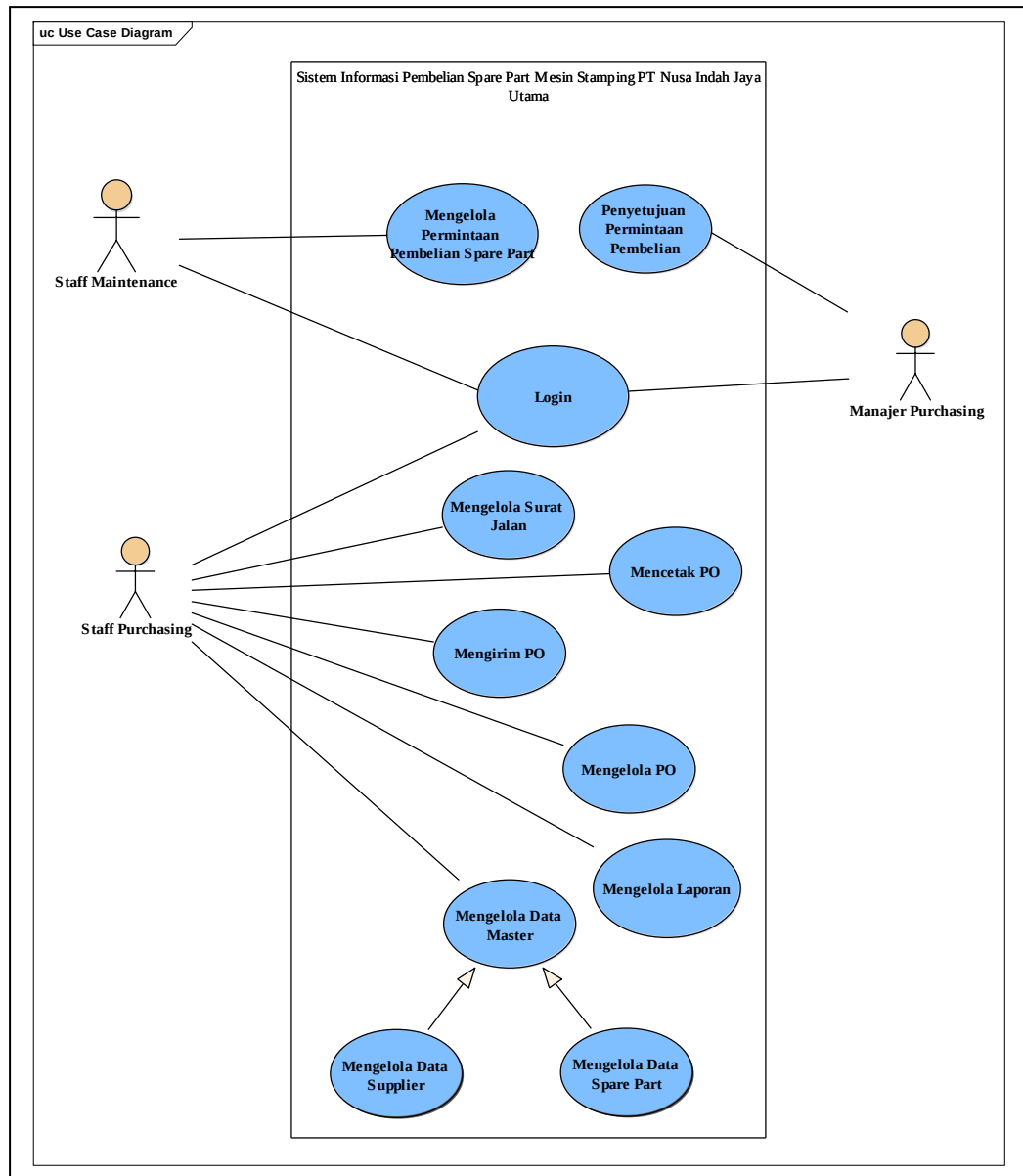
5.3 Analisis Sistem Usulan

Analisis proses sistem informasi pembelian *spare part* mesin *stamping* di PT Nusa Indah Jaya Utama menggunakan pemodelan sistem *Unified Modelling Language* (UML), berikut akan dimodelkan analisis menggunakan beberapa model yaitu *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*, *deployment diagram*. Tahapan analisis ini akan memberikan gambaran mengenai aliran informasi dan data pada sistem informasi pembelian *spare part* yang akan dibangun.

5.3.1 *Use case diagram*

Diagram *use case* merupakan sebuah diagram yang menyajikan sebuah penggambaran sistem dengan menggunakan diagram ini akan menggambarkan interaksi antara aktor dengan sistem, diagram ini menunjukkan apa yang dilakukan oleh sistem yang dibangun, berikut diagram *use case* sistem informasi pembelian *spare part*.

Penjelasan *use case diagram* sistem informasi pembelian *spare part* mesin *stamping* yang menjadi usulan sistem pada PT Nusa Indah Jaya Utama digambarkan pada Gambar V.2 sebagai berikut:



Gambar V.2 Use Case Diagram Usulan
(Sumber: Hasil Analisis, 2018)

1. Definisi Aktor

Pendefinisian aktor pada *use case diagram* berdasarkan usulan sistem informasi pembelian *spare part* mesin *stamping* dapat dilihat pada Tabel V.2 sebagai berikut:

Tabel V.2 Definisi Aktor

No	Aktor	Definisi
1	Staff Maintenance	Bertugas melakukan perbaikan mesin <i>stamping</i> dan

		membuat permintaan pembelian yang ditujukan kepada divisi <i>purchasing</i> .
2	Staff <i>Purchasing</i>	Bertugas melakukan pembelian <i>spare part</i> , mengelola data master pembelian, dan melakukan analisis terhadap harga yang ditawarkan.
3	Manajer <i>Purchasing</i>	Melakukan analisis terhadap permintaan pembelian yang diajukan, dan melakukan penyetujuan/penolakan terhadap permintaan pembelian.

(Sumber: Hasil Analisis, 2018)

2. Definisi *Use case*

Pendefinisian *use case* pada *use case diagram* berdasarkan usulan sistem informasi pembelian *spare part* mesin *stamping* dapat dilihat pada Tabel V.3 sebagai berikut:

Tabel V.3 Definisi *Use Case*

No	Nama <i>Use case</i>	Definisi
1	<i>Login</i>	Proses melakukan <i>login</i> pada aplikasi dengan memasukkan <i>username</i> dan <i>password</i>
2	Mengelola data master	Proses generalisasi yang meliputi dua proses pengelolaan data master yaitu data <i>supplier</i> , dan data <i>spare part</i>
3	Mengelola data <i>supplier</i>	Proses mengelola data <i>supplier</i> yang terdiri dari tambah, ubah, dan hapus pada data <i>supplier</i>
4	Mengelola data <i>spare part</i>	Proses mengelola data <i>supplier</i> yang terdiri dari tambah, ubah, dan hapus pada data <i>spare part</i>
5	Mengelola <i>PO</i>	Proses mengelola <i>purchase order</i> yang terdisir dari tambah <i>spare part</i>
6	Mengelola surat jalan	Proses pengelolaan data surat jalan berdasarkan <i>PO</i> keluar
7	Mencetak <i>PO</i>	Proses mencetak <i>PO</i>

Tabel V.3 Definisi *Use Case* (lanjutan)

No	Nama <i>Use case</i>	Definisi
8	Penyetujuan permintaan pembelian	Proses manajer <i>purchasing</i> meyetujui atau menolak permintaan pembelian <i>spare part</i> yang diajukan
9	Mengelola permintaan pembelian <i>spare part</i>	Proses membuat sebuah permintaan pembelian yang diikuti dengan kerusakan yang terjadi pada mesin <i>stamping</i>
10	Mengirim <i>PO</i>	Proses pengiriman <i>PO</i> kepada suplier melalui email

(Sumber: Hasil Analisis, 2018)

3. Skenario *Use case*

Skenario jalannya masing-masing *use case* pada usulan sistem informasi pembelian *spare part* mesin *stamping* akan dijelaskan sebagai berikut:

a. *Use case login*

Berikut adalah skenario *use case login* yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.4 Deskripsi *Use Case Login*

Nama <i>Use case</i>	<i>Login</i>
Aktor	Staff <i>maintenance</i> , staff <i>purchasing</i> , manajer <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan proses pengguna masuk ke sistem pembelian <i>spare part</i> mesin <i>stamping</i> .
<i>Normal flow events</i>	1.User membuka aplikasi. 2.Sistem menampilkan <i>form login</i>. 3.User mengisi <i>username</i> beserta <i>password</i> di kolom yang disediakan pada <i>form login</i>. 4.Sistem akan melakukan validasi terhadap <i>username</i> dan <i>password</i> yang dimasukkan oleh user. 5.Jika <i>username</i> dan <i>password</i> valid maka akan muncul halaman utama aplikasi. 6.Jika <i>username</i> dan <i>password</i> tidak valid maka akan muncul notifikasi kesalahan <i>username/password</i>.

(Sumber: Hasil Analisis, 2018)

b. *Use case* mengelola data master

Berikut adalah skenario *use case* mengelola data master yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.5 Deskripsi *Use case Mengelola Data Master*

Nama <i>Use case</i>	Mengelola data master
Aktor	Staff <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan pengelolaan data master perusahaan yaitu data master <i>supplier</i> dan data master <i>spare part</i>
<i>Normal flow events</i>	1.Divisi <i>purchasing</i> membuka menu data master 2.Sistem menampilkan sub data master yang terdiri dari dua sub menu yaitu data master <i>supplier</i> dan data master <i>spare</i>

	<i>part</i> 3.User dapat tambah, ubah, cari dan hapus data master <i>supplier</i> dan <i>spare part</i> ke/dari <i>database</i>. 4.Sistem akan memperbarui <i>database</i> 5.Sistem akan menampilkan data
--	---

(Sumber: Hasil Analisis, 2018)

c. *Use case* mengelola permintaan pembelian

Berikut adalah skenario *use case* permintaan pembelian yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.6 Deskripsi *Use case* Mengelola Permintaan Pembelian

Nama <i>Use case</i>	Membuat permintaan pembelian
Aktor	Staff <i>maintenance</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan proses penginputan data untuk pembuatan permintaan pembelian
<i>Normal flow events</i>	1.Staff divisi <i>maintenance</i> membuka aplikasi pembelian <i>spare part</i> mesin <i>stamping</i> lalu melakukan <i>login</i> kedalam sistem. 2. Setelah melakukan <i>login</i> staff divisi <i>maintenance</i> memilih menu permintaan pembelian.

Tabel V.6 Deskripsi *Use case* Mengelola Permintaan Pembelian (lanjutan)

<i>Normal flow events</i>	3.Selanjutnya sistem menampilkan permintaan yang telah masuk. 4.Selanjutnya user memilih buat permntaan. 5.Sistem menampilkan <i>form</i> permintaan pembelian <i>spare part</i>. 6.Staff mengisi <i>form</i> permintaan pembelian secara detail lalu klik tombol simpan dan sistem menyimpan data detail. 7.Staff memilih tombol selesai. 8.Lalu muncul <i>form</i> konfirmasi. 9.Setelah <i>form</i> konfirmasi diisi selanjutnya klik selesai. 10.Selanjutnya data permintaan pembelian akan tersimpan di <i>database</i>.
---------------------------	---

(Sumber: Hasil Analisis, 2018)

d. *Use case* penyetujuan permintaan pembelian

Berikut adalah skenario *use case* membuat permintaan pembelian yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.7 Deskripsi *Use case* Penyetujuan Permintaan Pembelian

Nama <i>Use case</i>	Menyetujui permintaan pembelian
Aktor	Manajer <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan proses penyetujuan sebuah permintaan pembelian
<i>Normal flow events</i>	1.Manajer <i>purchasing</i> mendapatkan notifikasi permintaan pembelian yang telah diajukan. 2.Manajer <i>purchasing</i> memilih menu validasi. 3.Manajer <i>purchasing</i> memilih sub menu notifikasi permintaan pembelian. 4.Manajer <i>purchasing</i> memilih detail permintaan. 5.Manajer <i>purchasing</i> melakukan analisis permintaan yang diajukan 6.Manajer <i>purchasing</i> menyetujui permintaan pembelian

(Sumber: Hasil Analisis, 2018)

e. *Use case* mengelola PO

Berikut adalah skenario *use case* mengelola *purchase order* yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.8 Deskripsi *Use case* Mengelola PO

Nama <i>Use case</i>	Membuat <i>purchase order</i>
Aktor	Staff <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan proses penginputan data untuk pembuatan <i>purchase order</i>
<i>Normal flow events</i>	1.Staff divisi <i>purchasing</i> membuka menu pembelian. 2.Staff divisi <i>purchasing</i> memilih sub menu buat PO. 3.Sistem menampilkan permintaan yang telah divalidasi 4.Staff divisi <i>purchasing</i> memilih permintaan yang akan dimasukkan kedalam PO dengan memilih tombol buat PO. 5.Sistem menampilkan <i>form</i> buat PO.

	6.Staff mengisi <i>form</i> buat PO dan klik simpan. 7.Sistem menyimpan detail PO. 8.Staff memilih tombol hapus untuk menghapus detail PO dan dapat memilih selesai. 9.Sistem menampilkan <i>form</i> konfirmasi PO. 10.Staff mengisi <i>form</i> konfirmasi PO lalu klik simpan. 11.Sistem menyimpan PO yang telah <i>diinput</i> .
--	---

(Sumber: Hasil Analisis, 2018)

f. *Use case* mencetak PO

Berikut adalah skenario *use case* mencetak *purchase order* yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.9 Deskripsi *Use case* Mencetak PO

Nama <i>Use case</i>	Mencetak PO
Aktor	Staff <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan proses cetak PO yang dilakukan oleh user
<i>Normal flow events</i>	1.Staff divisi <i>purchasing</i> membuka menu pembelian. 2.Staff divisi <i>purchasing</i> membuka menu cetak 3.Staff melakukan pemilihan data <i>purchase order</i> yang akan dicetak. 4.Sistem menampilkan data <i>purchase order</i> yang dipilih berformat PDF. 5.Staff melakukan pengiriman <i>purchase order</i> .

(Sumber: Hasil Analisis, 2018)

g. *Use case* mengirim PO

Berikut adalah skenario *use case* mengirim PO yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.10 Deskripsi *Use case* Mengirim PO

Nama <i>Use case</i>	Mencetak PO
Aktor	Staff <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan proses mengirim PO yang dilakukan oleh user
<i>Normal flow events</i>	1.Staff divisi <i>purchasing</i> membuka menu pembelian. 2.Staff divisi <i>purchasing</i> membuka menu kirim PO 3.Staff memasukkan email <i>supplier</i> , pesan, dan file PO 4.PO dikirim melalui email oleh sistem.

(Sumber: Hasil Analisis, 2018)

h. *Use case* mengelola surat jalan

Berikut adalah skenario *use case* mengirimkan *purchase order* yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.11 Deskripsi *Use case* Mengelola Surat Jalan

Nama <i>Use case</i>	Input surat jalan
Aktor	Staff <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan proses penginputan data untuk mengelola surat jalan
<i>Normal flow events</i>	1.Staff divisi <i>purchasing</i> membuka menu pembelian. 2.Staff divisi <i>purchasing</i> memilih sub menu surat jalan. 3.Sistem menampilkan surat jalan yang telah masuk. 4.Selanjutnya user memilih <i>input</i> surat jalan. 5.User mengisi <i>form</i> surat jalan, lalu klik simpan. 6.Data detail tersimpan didalam <i>database</i> . 7.User klik selesai. 8.Sistem memunculkan <i>form</i> konfirmasi. 9.User memasukkan data konfirmasi. 10.Selanjutnya data tersimpan di <i>database</i>

(Sumber: Hasil Analisis, 2018)

i. *Use case* mengelola laporan

Berikut adalah skenario *use case* mencetak *purchase order* yang terdapat dalam sistem informasi pembelian *spare part* mesin *stamping*:

Tabel V.12 Deskripsi *Use case* Mencetak *Purchase Order*

Nama <i>Use case</i>	Mengirimkan <i>purchase order</i>
Aktor	Staff <i>purchasing</i>
Deskripsi <i>use case</i>	<i>Use case</i> ini menggambarkan mengelola laporan
<i>Normal flow events</i>	1.Staff divisi <i>purchasing</i> membuka menu pembelian. 2.Staff divisi <i>purchasing</i> membuka menu cetak 3.Staff melakukan pemilihan jarak tanggal yang diinginkan. 4.Sistem menampilkan data laporan pembelian sesuai pilihan.

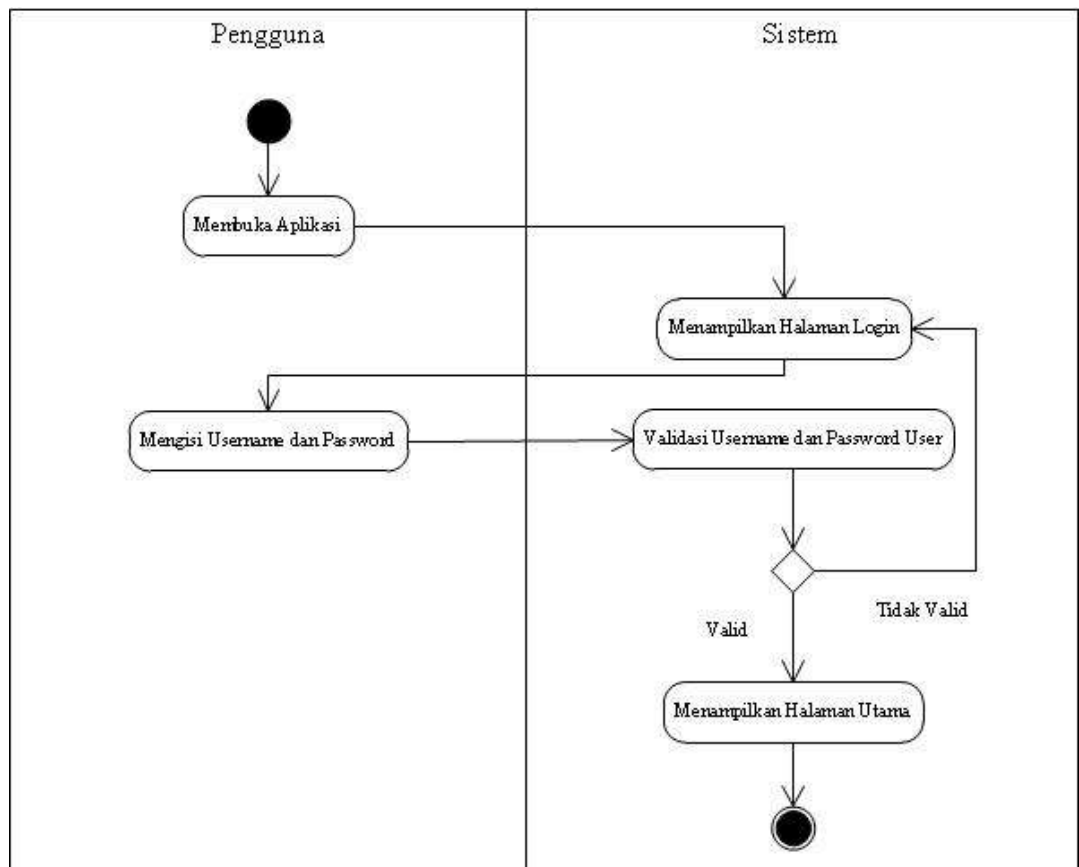
(Sumber: Hasil Analisis, 2018)

5.3.2 *Activity diagram*

Activity diagram merupakan sebuah diagram yang menggambarkan urutan aktifitas proses bisnis pada suatu sistem, sebagai bantuan dalam proses memahami proses bisnis yang berjalan pada sistem informasi pembelian *spare part* maka akan ditampilkan *activity diagram* sebagai berikut:

1. *Activity diagram login*

Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu staff divisi *maintenance*, staff divisi *purchasing*, dan manajer *purchasing* untuk mendapatkan akses kedalam sistem informasi pembelian *spare part*. Berikut merupakan *activity diagram login* pada Gambar V.3:

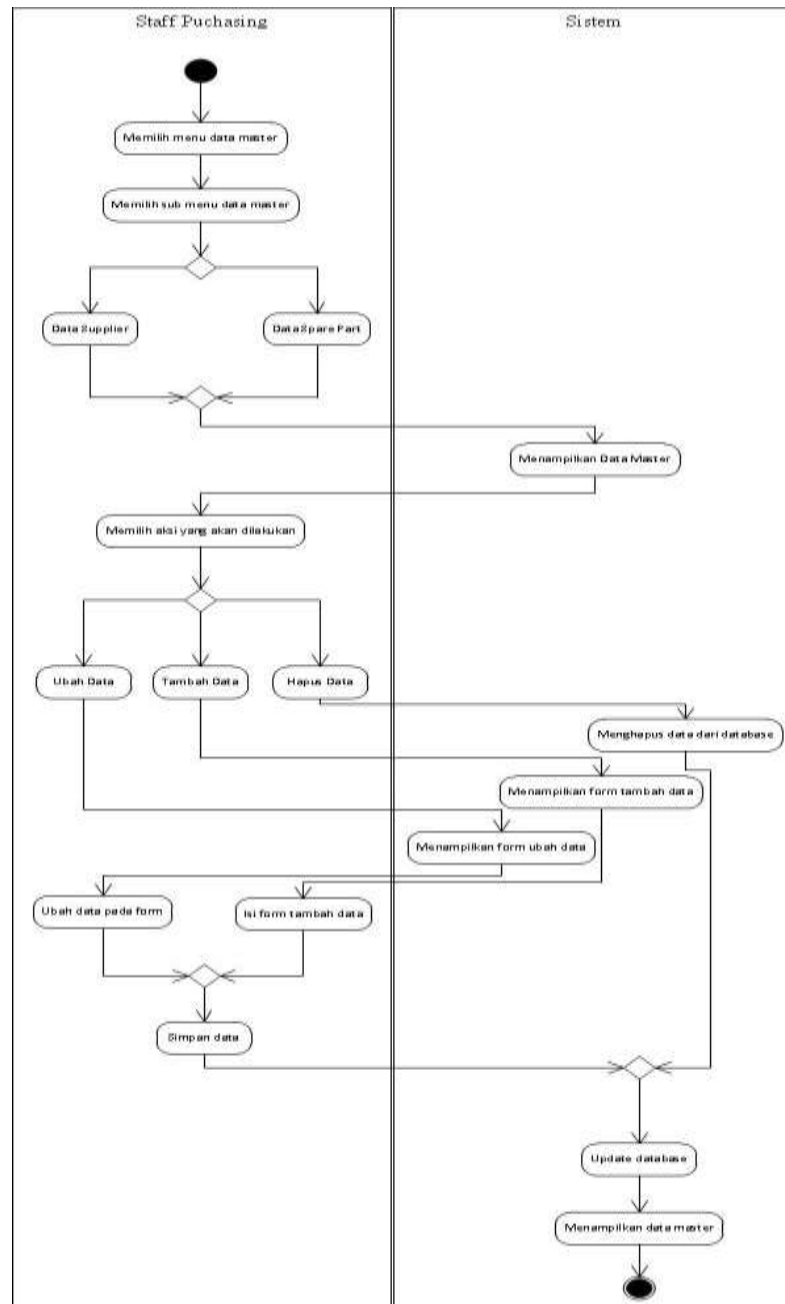


Gambar V.3 Activity Diagram Login
(Sumber: Hasil Analisis, 2018)

2. Activity diagram mengelola data master

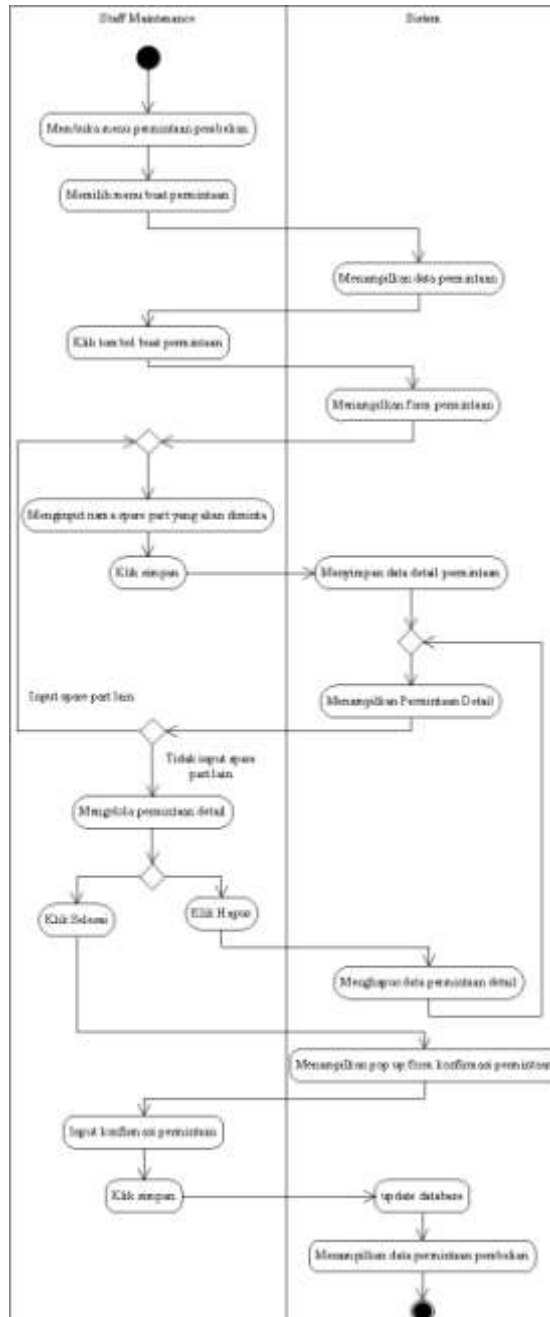
Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu staff divisi *purchasing* untuk mengelola data master mulai dari menambah data, mengubah data, sampai hapus data, terdapat dua data master dalam sistem usulan yaitu data master *spare part* dan data master *supplier*. Gambar V.4 menggambarkan diagram tersebut sebagai berikut:

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



3. *Activity diagram* mengelola permintaan pembelian *spare part*.

Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu staff divisi *maintenance* untuk mengelola permintaan pembelian *spare part* mesin *stamping* yang nantinya ditujukan kepada manajer *purchasing*. Berikut merupakan *activity diagram* membuat permintaan pembelian *spare part* pada gambar V.5:

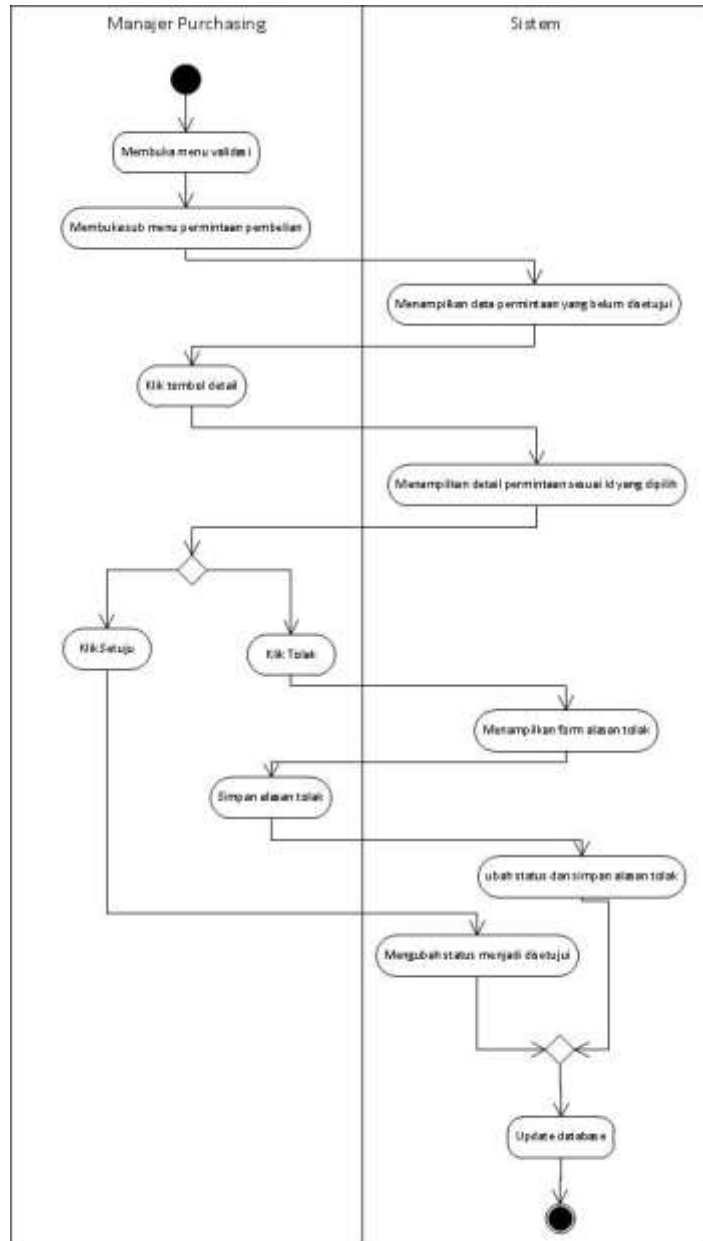


Gambar V.5 Activity Diagram Permintaan Pembelian
(Sumber: Hasil Analisis, 2018)

4. Activity diagram melakukan penyetujuan permintaan pembelian

Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu Manajer *Purchasing* untuk melakukan persetujuan atas permintaan pembelian yang

dilakukan oleh divisi *maintenance*, berikut merupakan *activity diagram* penyetujuan permintaan pembelian pada Gambar V.6:

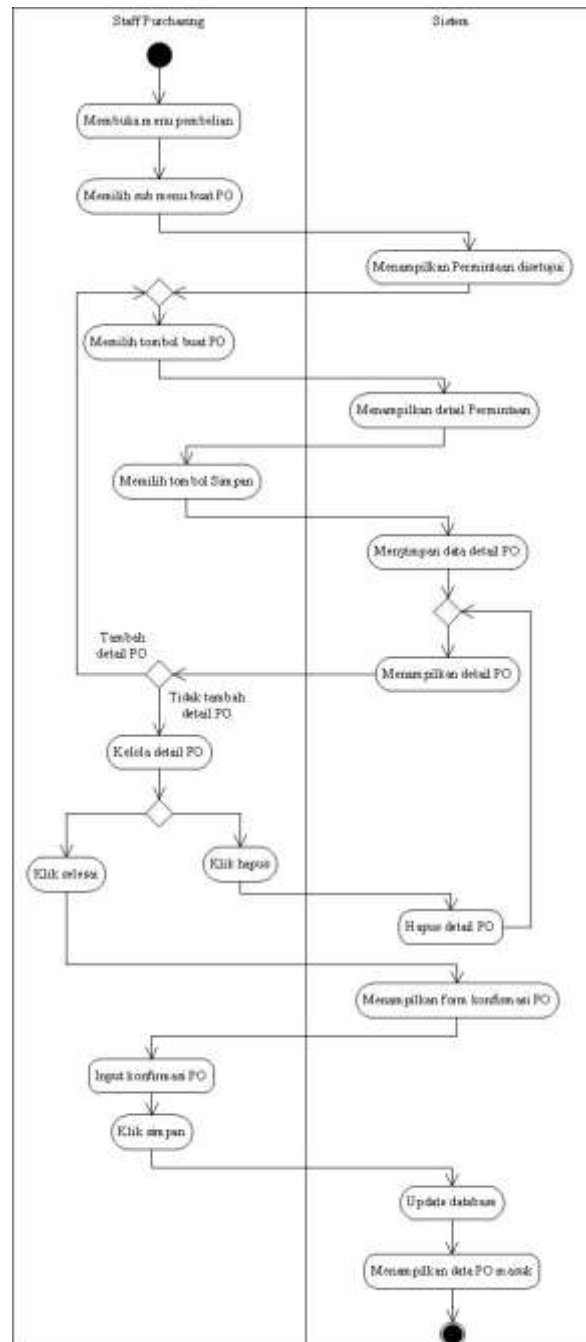


Gambar V.6 *Activity Diagram* Penyetujuan Permintaan Pembelian
(Sumber: Hasil Analisis, 2018)

5. *Activity diagram* mengelola PO

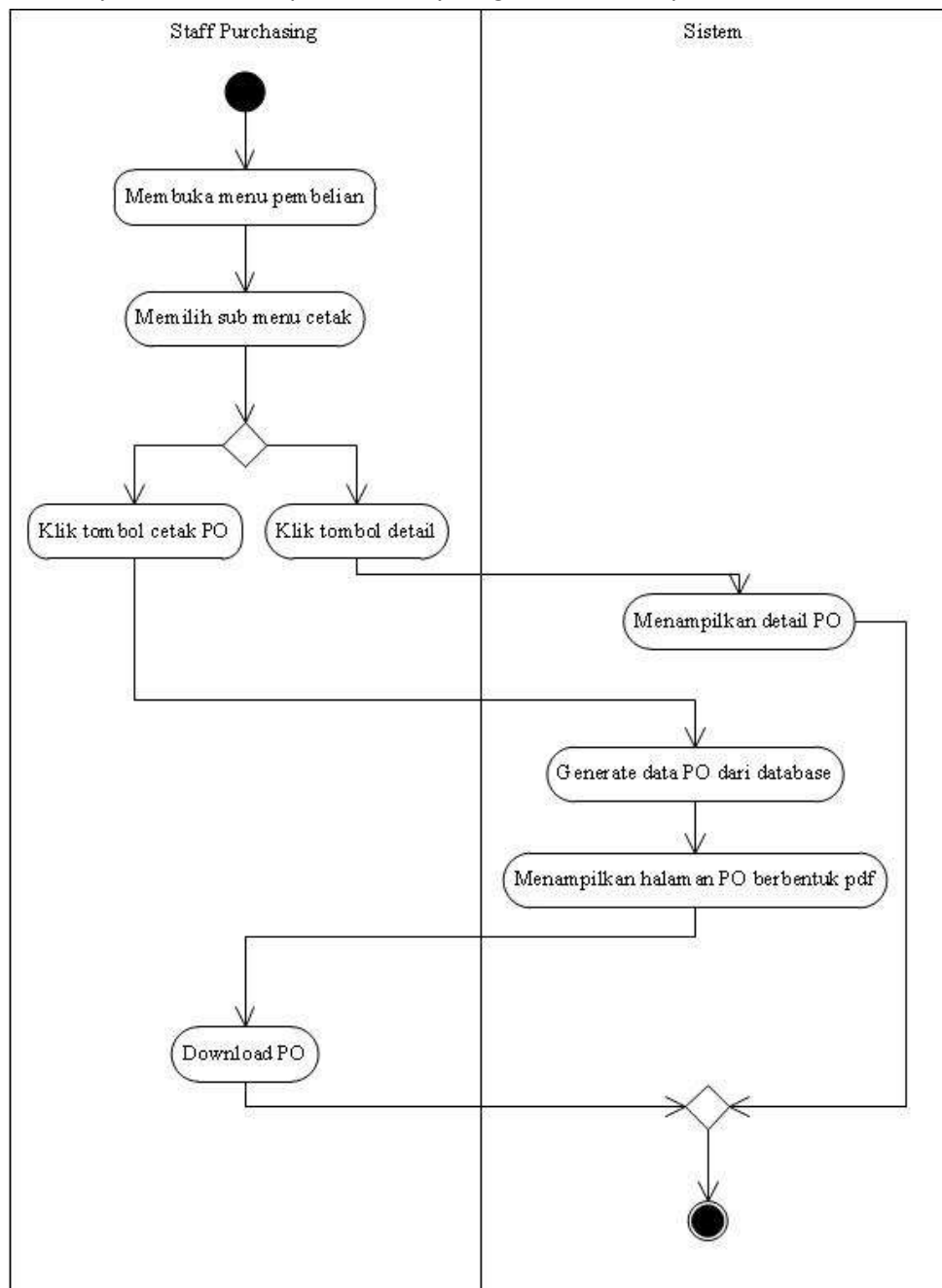
Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu Staff Divisi *Purchasing* untuk melakukan pembuatan *purchase order* atas permintaan pembelian yang telah disetujui oleh manajer *purchasing*, berikut merupakan *activity diagram* membuat PO pada Gambar V.7:

Gambar V.7 Activity Diagram Membuat PO
(Sumber: Hasil Analisis, 2018)



6. Activity diagram cetak PO

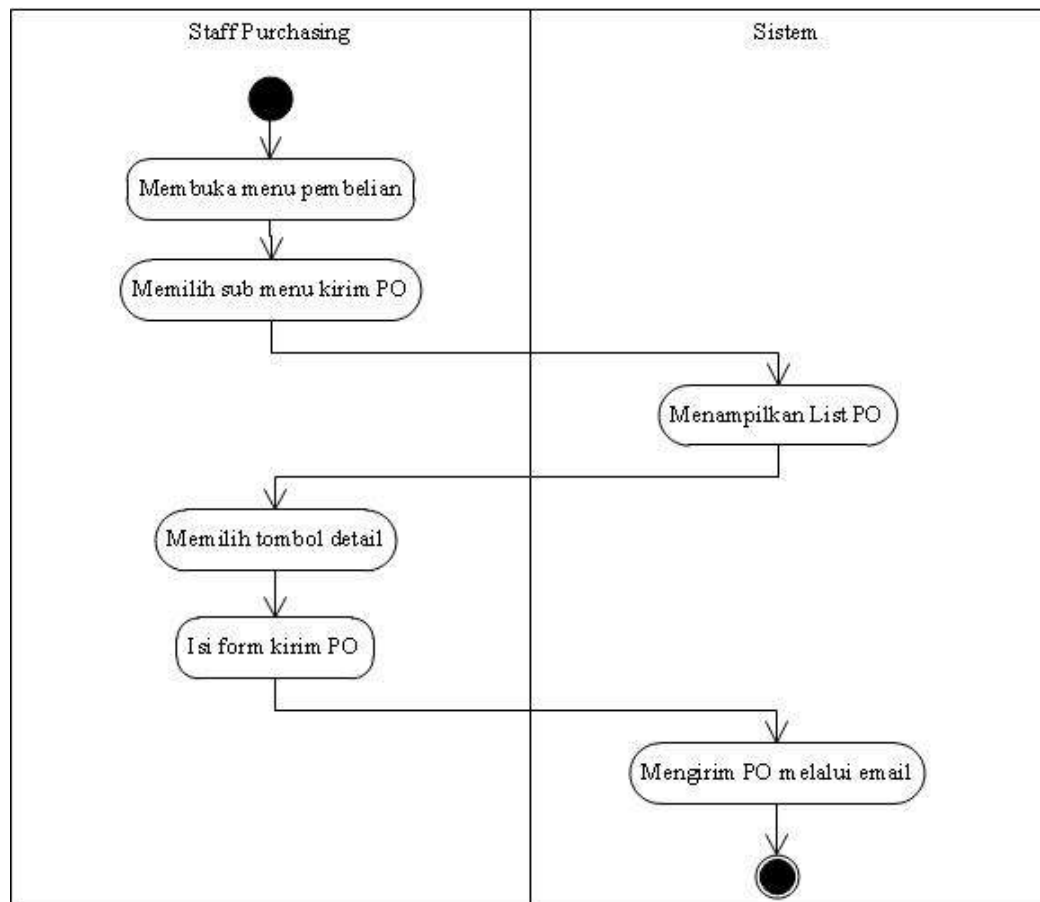
Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu Staff Divisi *Purchasing* untuk melakukan cetak *Purchase Order* yang telah diinput sebelumnya, berikut merupakan *activity diagram* cetak PO pada Gambar V.8:



Gambar V.8 *Activity Diagram* Mencetak PO
(Sumber: Hasil Analisis, 2018)

7. *Activity diagram* mengirim PO

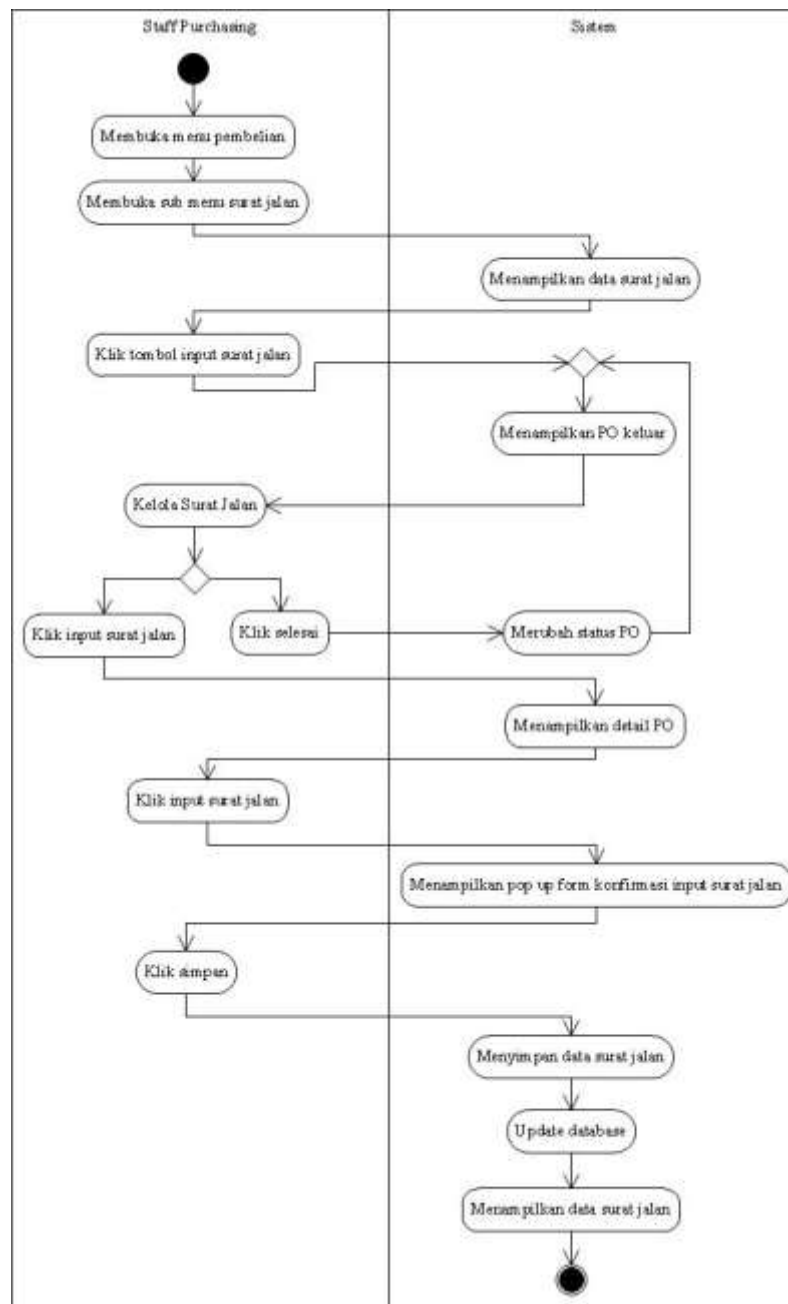
Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu Divisi *Purchasing* untuk mengirimkan PO ke *supplier* yang telah ditentukan melalui email, berikut *activity diagram* mengirim PO pada Gambar V.9:



Gambar V.9 *Activity Diagram* Mengirim PO
(Sumber: Hasil Analisis, 2018)

8. *Activity diagram* kelola surat jalan

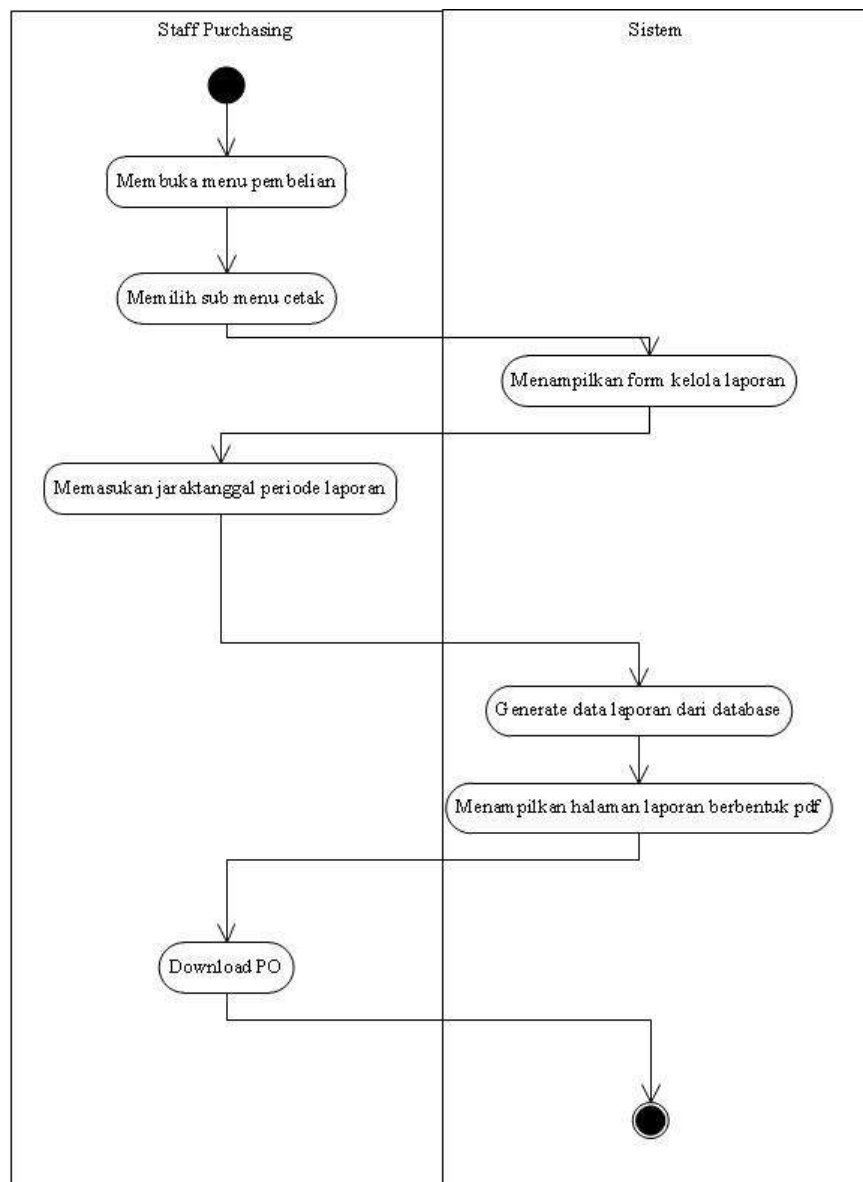
Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu Divisi *Purchasing* untuk melakukan *input* data surat jalan yang diterima perusahaan bersamaan dengan *spare part* yang telah dipesan sebelumnya, berikut merupakan *activity diagram input* surat jalan pada Gambar V.10:



Gambar V.10 Activity Diagram Kelola Surat Jalan
(Sumber: Hasil Analisis, 2018)

9. Activity diagram kelola laporan pembelian

Activity diagram ini menjelaskan alur aktivitas yang dilakukan oleh user yaitu Divisi Purchasing untuk melakukan kelola laporan berdasarkan PO keluar yang telah dikirim, berikut merupakan activity diagram input surat jalan pada Gambar V.11:



Gambar V.11 Activity Diagram Kelola Laporan Pembelian
(Sumber: Hasil Analisis, 2018)

5.3.3 Sequence diagram

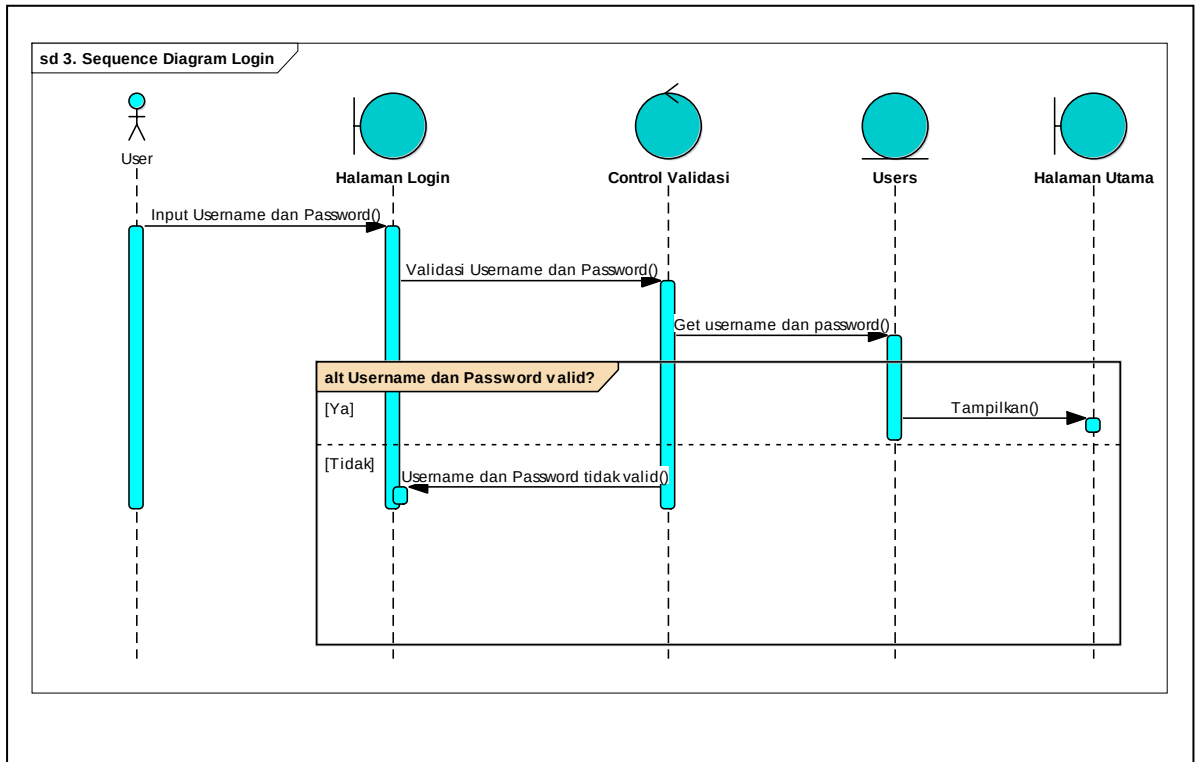
Sequence diagram menggambarkan interaksi yang terjadi antar objek pada sebuah *use case*, dimana urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada diagram. Hubungan yang ada pada gambar di bawah ini adalah proses yang dilakukan oleh sistem ketika melakukan proses yang sesuai dengan suatu objek pada *use case diagram* usulan, berikut adalah *sequence diagram* usulan pada sistem informasi pembelian *spare part* mesin *stamping*:

1. Sequence diagram Login

Sequence diagram login merupakan sebuah diagram *sequence* yang menjelaskan interaksi objek-objek dalam sebuah proses *login*. Proses *login* dalam sistem informasi pembelian *spare part* ini dilakukan oleh user agar dapat mengakses sistem, berikut merupakan *sequence diagram* dari *use case login* pada Gambar V.12 sebagai berikut:

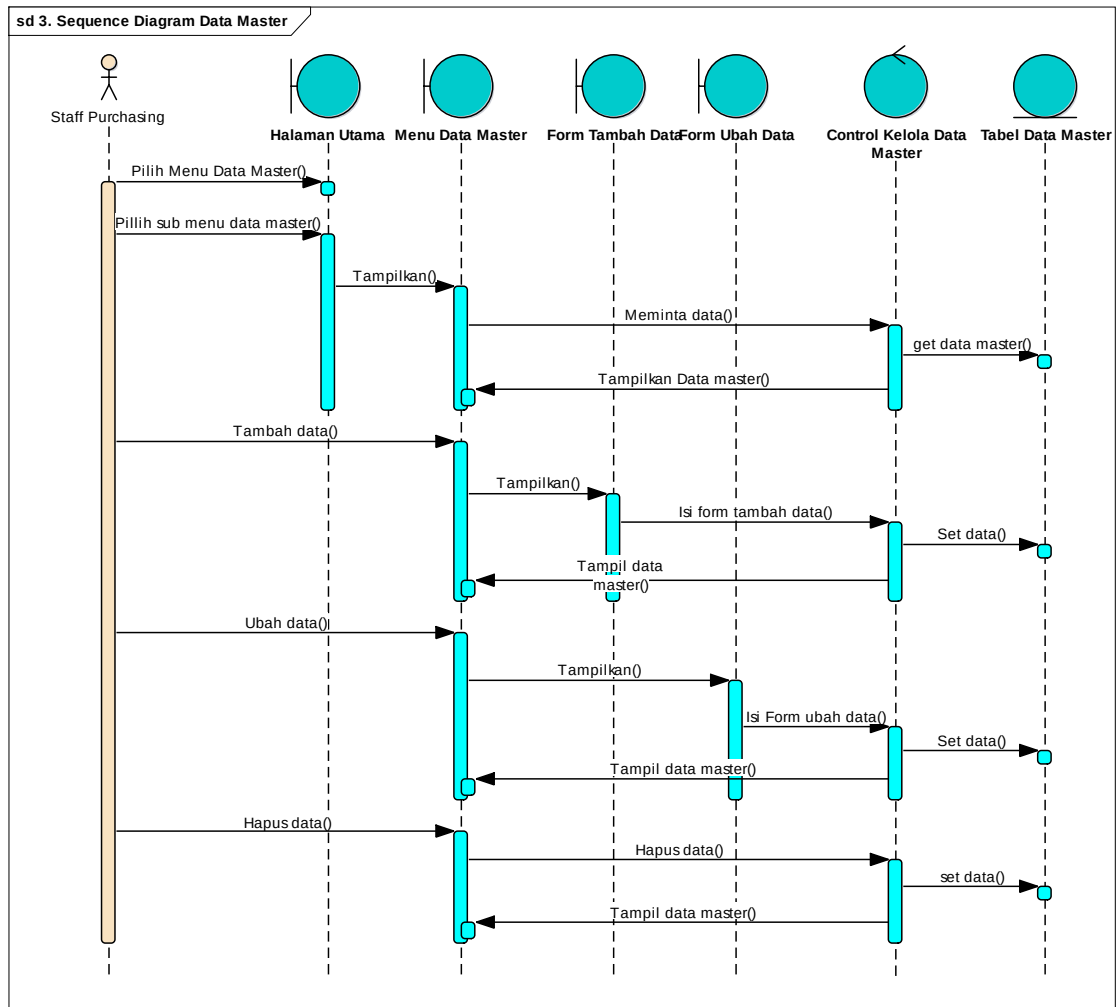
Gambar V.12 *Sequence Diagram Login*

(Sumber: Hasil Analisis, 2018)



2. *Sequence diagram* mengelola data master

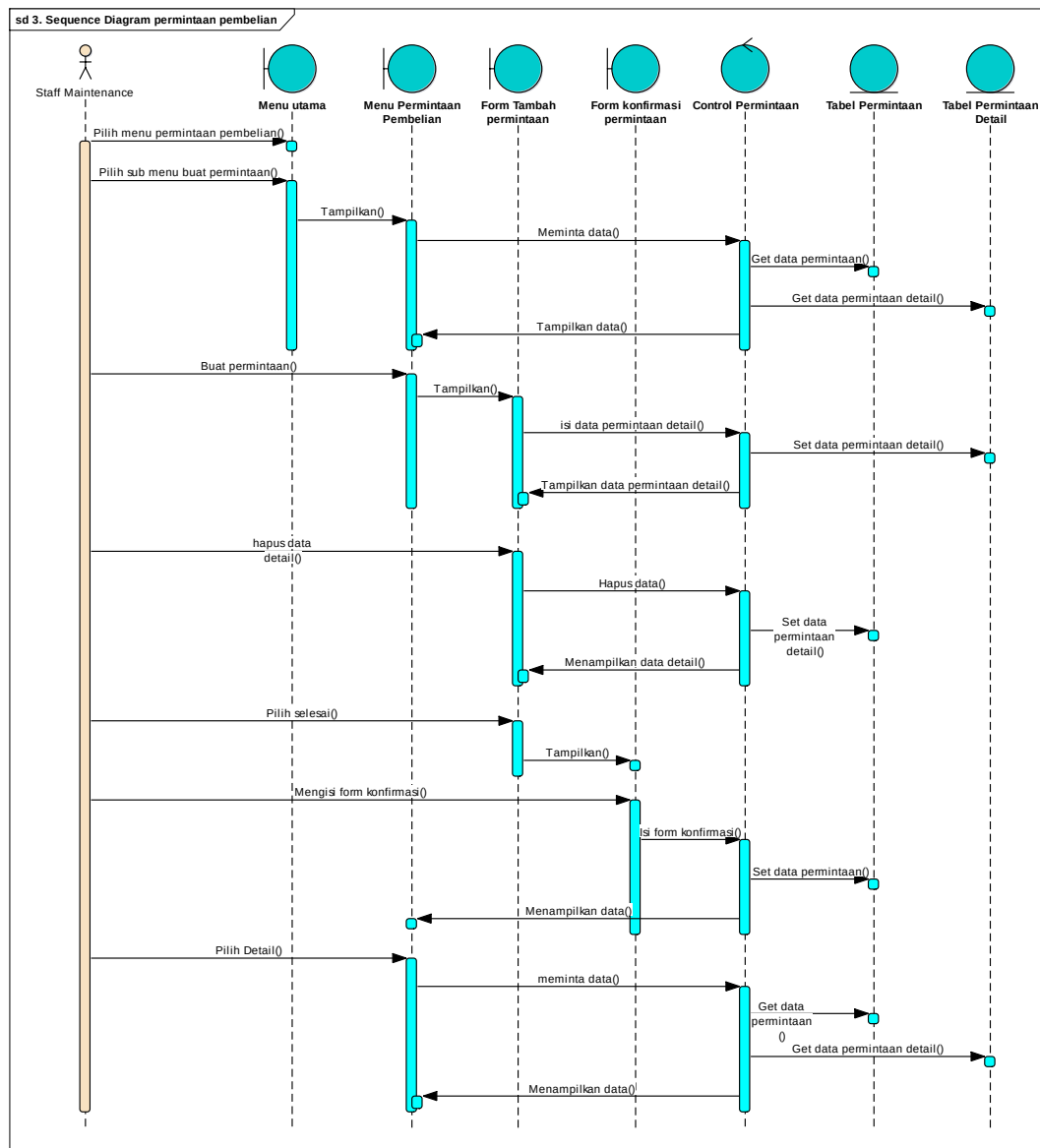
Sequence diagram mengelola data master merupakan sebuah diagram *sequence* yang menjelaskan interaksi objek-objek dalam proses pengelolaan data master yang terdapat di dalam sistem informasi pembelian, terdapat dua data master dalam sistem informasi pembelian *spare part* mesin *stamping* yaitu data master *spare part* dan data master *supplier*, berikut merupakan *sequence diagram* mengelola data master pada Gambar V.13:



Gambar V.13 Sequence Diagram Mengelola Data Master
(Sumber: Hasil Analisis, 2018)

3. Sequence diagram mengelola permintaan pembelian

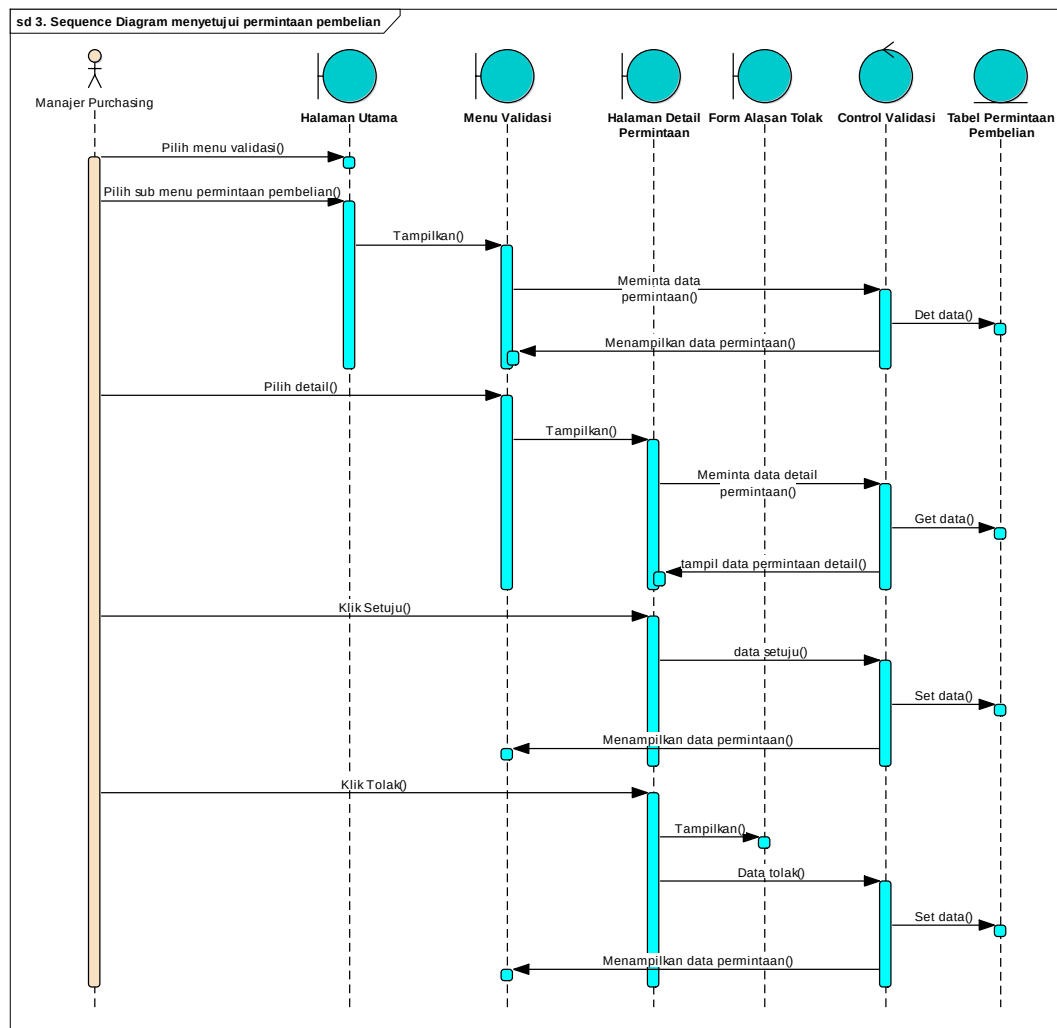
Sequence diagram mengelola permintaan pembelian merupakan sebuah diagram sequence yang menjelaskan interaksi objek-objek dalam proses pembuatan permintaan pembelian yang dilakukan oleh divisi *maintenance* dan ditujukan kepada manajer *purchasing*, berikut merupakan sequence diagram mengelola permintaan pembelian pada Gambar V.14:



Gambar V.14 Sequence Diagram Membuat Permintaan Pembelian
(Sumber: Hasil Analisis, 2018)

4. Sequence diagram menyetujui permintaan pembelian

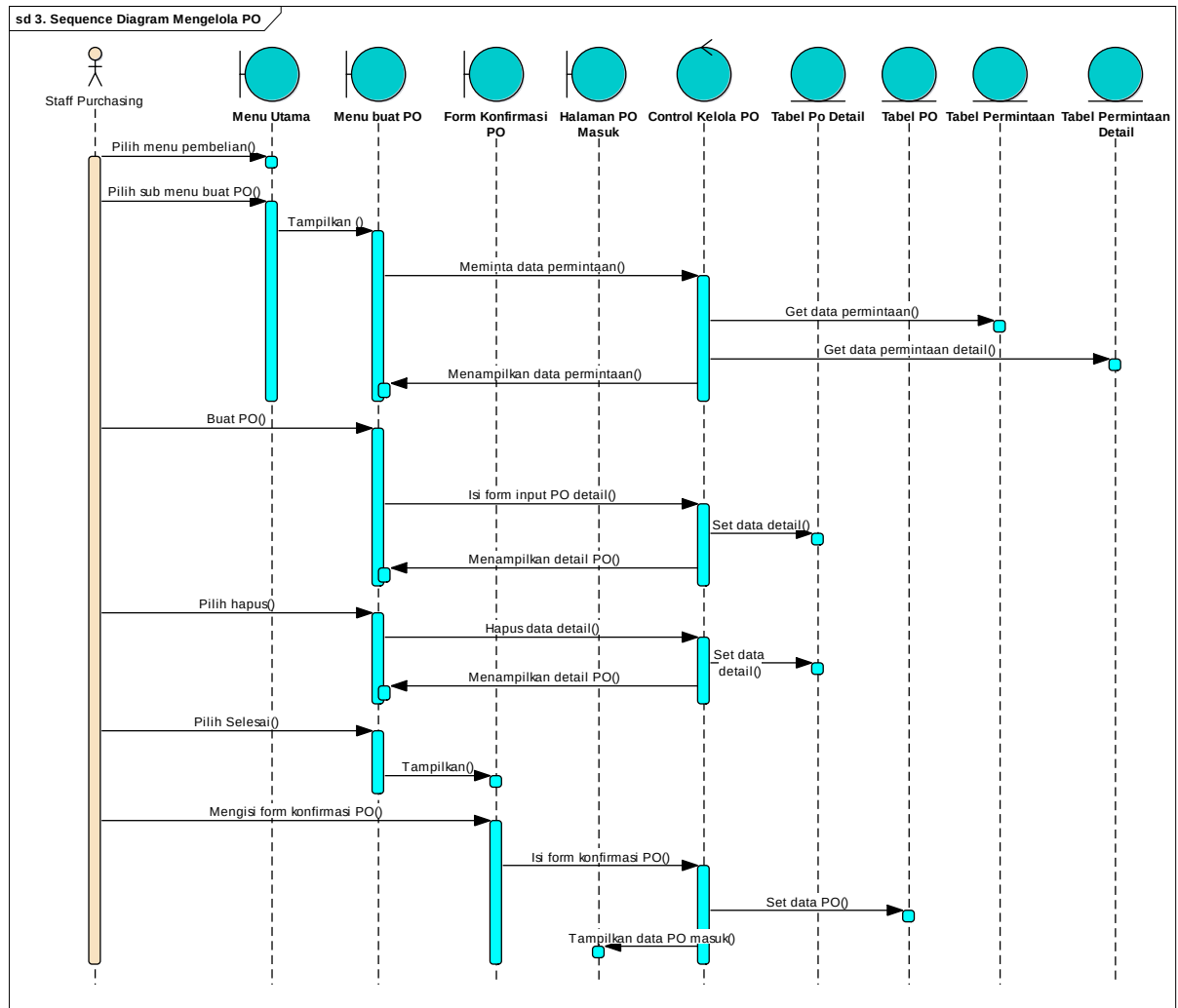
Sequence diagram menyetujui permintaan pembelian merupakan sebuah diagram sequence yang menjelaskan interaksi objek-objek dalam proses validasi atau penyetujuan permintaan pembelian, proses validasi ini dilakukan oleh manajer purchasing, berikut merupakan sequence diagram validasi pada Gambar V.15:



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

5. *Sequence diagram* mengelola PO

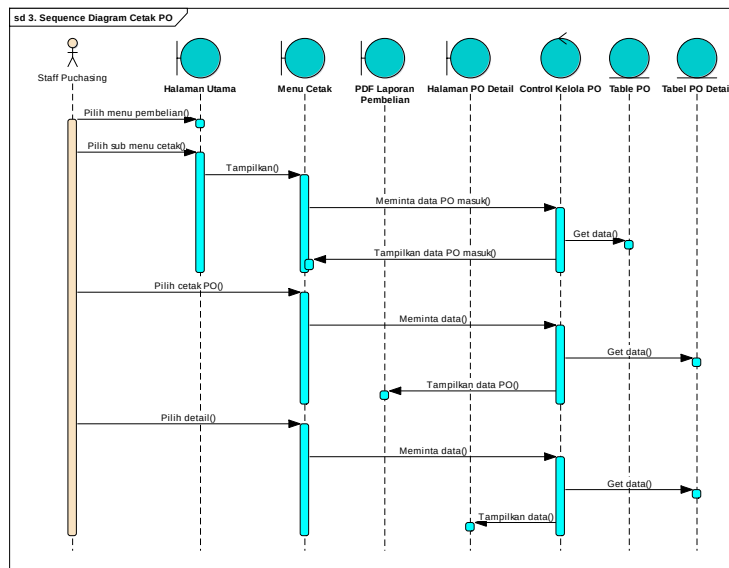
Sequence diagram mengelola PO merupakan sebuah diagram *sequence* yang menjelaskan interaksi objek-objek dalam proses pengelolaan *purchase order* yang dilakukan oleh staff divisi *purchasing*, berikut merupakan *sequence diagram* mengelola PO pada Gambar V.16:



Gambar V.16 *Sequence Diagram* Mengelola PO
(Sumber: Hasil Analisis, 2018)

6. Sequence diagram cetak PO

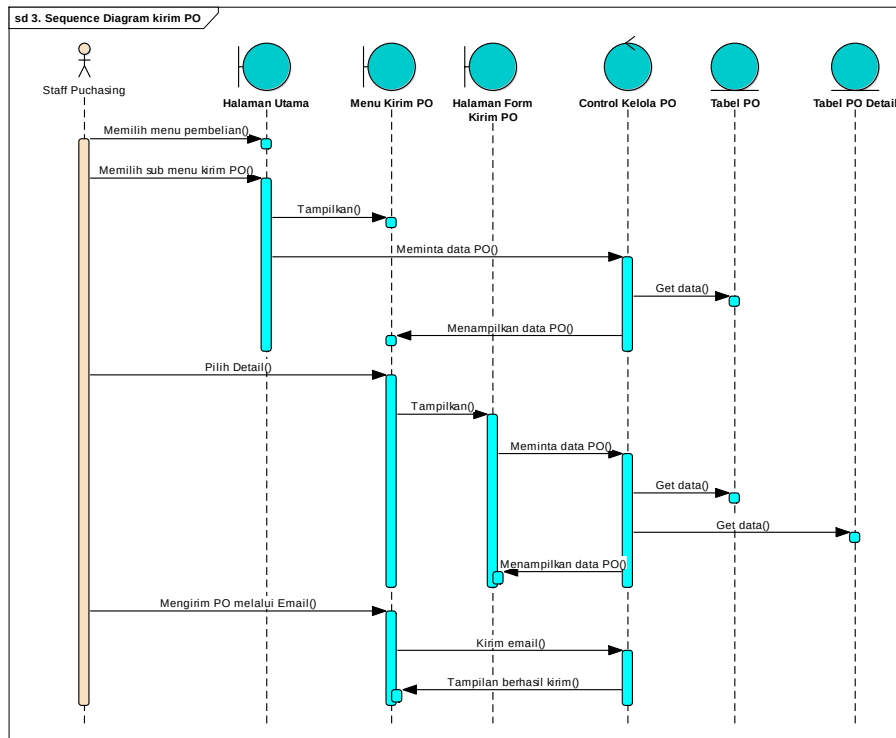
Sequence diagram cetak PO merupakan sebuah diagram *sequence* yang menjelaskan interaksi objek-objek dalam proses mencetak *purchase order* yang dilakukan oleh staff divisi *purchasing* untuk selanjutnya dikirimkan kepada *supplier*, berikut merupakan *sequence diagram* mengelola PO pada Gambar V.17:



Gambar V.17 *Sequence Diagram* Cetak PO
(Sumber: Hasil Analisis, 2018)

7. Sequence diagram mengirim PO

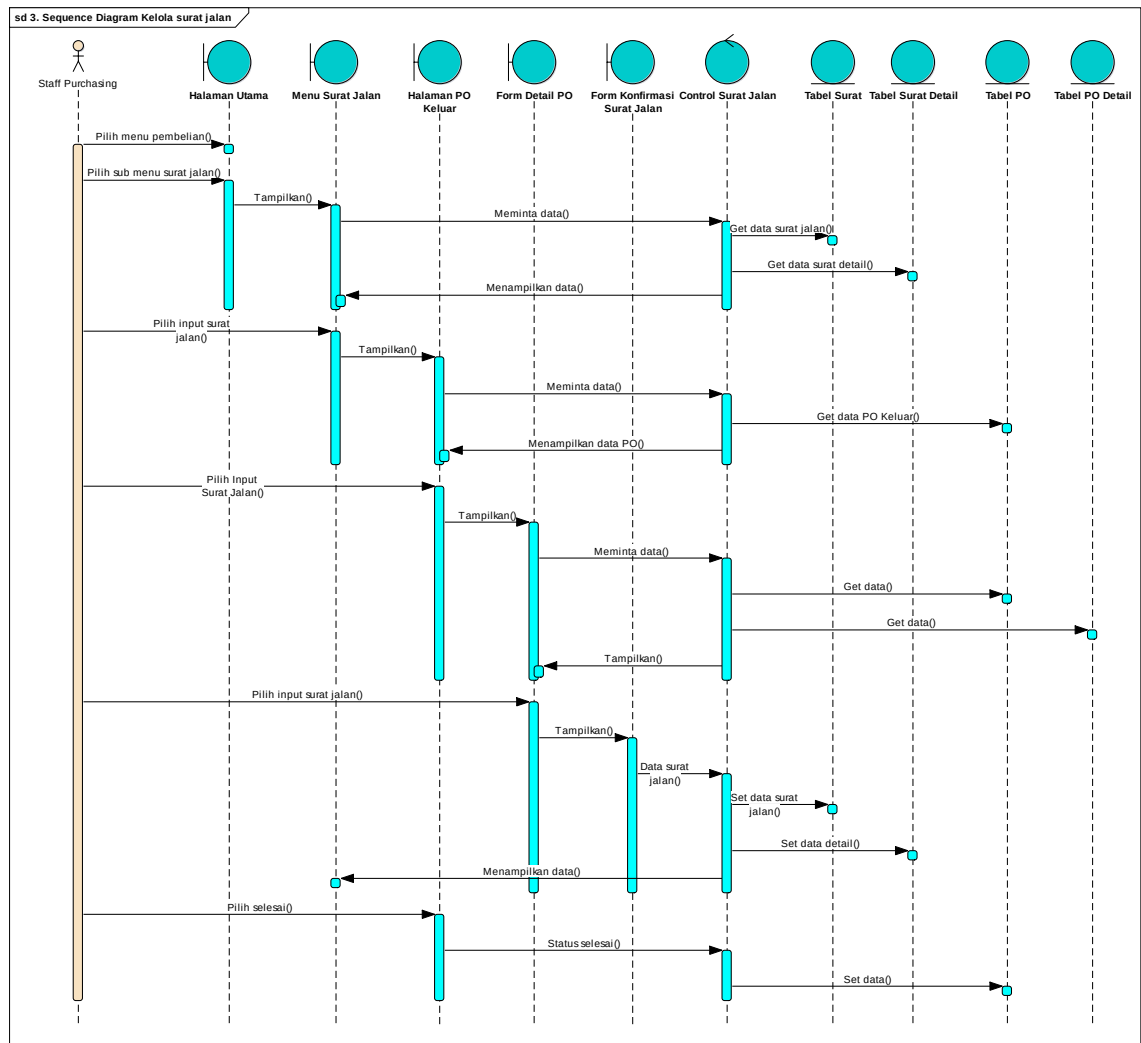
Sequence diagram mengirim PO merupakan sebuah diagram *sequence* menjelaskan interaksi objek-objek dalam proses mengirim PO kepada *supplier*, berikut diagram tersebut pada Gambar V.18:



Gambar V.18 *Sequence Diagram* Mengirim PO
(Sumber: Hasil Analisis, 2018)

8. Sequence diagram kelola surat jalan

Sequence diagram input surat jalan merupakan sebuah diagram *sequence* yang menjelaskan interaksi objek-objek dalam proses *input* surat jalan yang merupakan dokumen yang dikirimkan kepada perusahaan oleh *supplier* sebagai bukti pengiriman, proses ini dilakukan oleh divisi *purchasing*, berikut merupakan *sequence diagram* kelola surat jalan pada Gambar V.19:

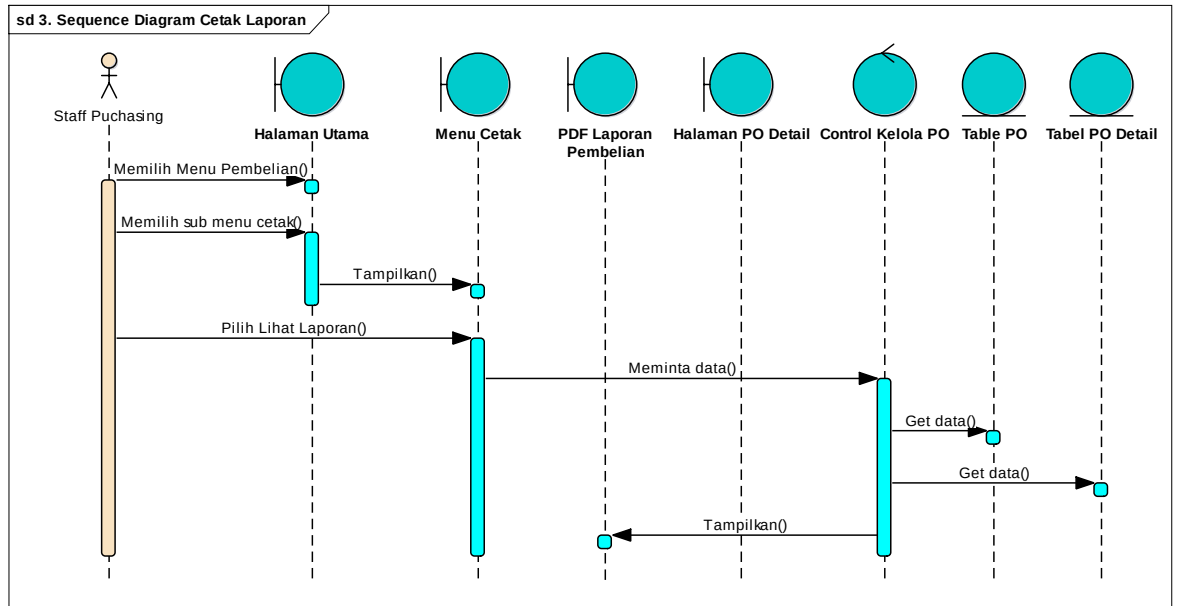


Gambar V.19 *Sequence Diagram* Kelola Surat Jalan
(Sumber: Hasil Analisis, 2018)

9. *Sequence diagram* kelola laporan

Sequence diagram kelola laporan merupakan sebuah diagram *sequence* yang menjelaskan interaksi objek-objek dalam proses kelola laporan permintaan

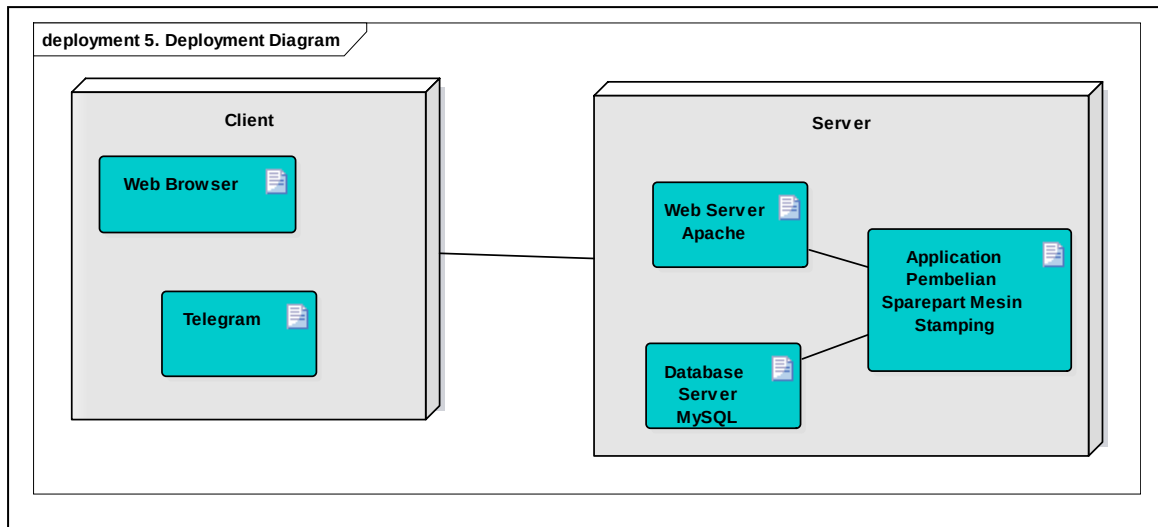
pembelian yang laporan ini berfungsi sebagai informasi dan dasar pengambilan keputusan, berikut merupakan *sequence diagram* kelola laporan pada Gambar V.20:



Gambar V.20 *Sequence Diagram* Kelola Laporan
(Sumber: Hasil Analisis, 2018)

5.3.4 *Class diagram*

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, *class diagram* sistem informasi pembelian *spare part* mesin *stamping* digambarkan pada Gambar V.21 sebagai berikut:



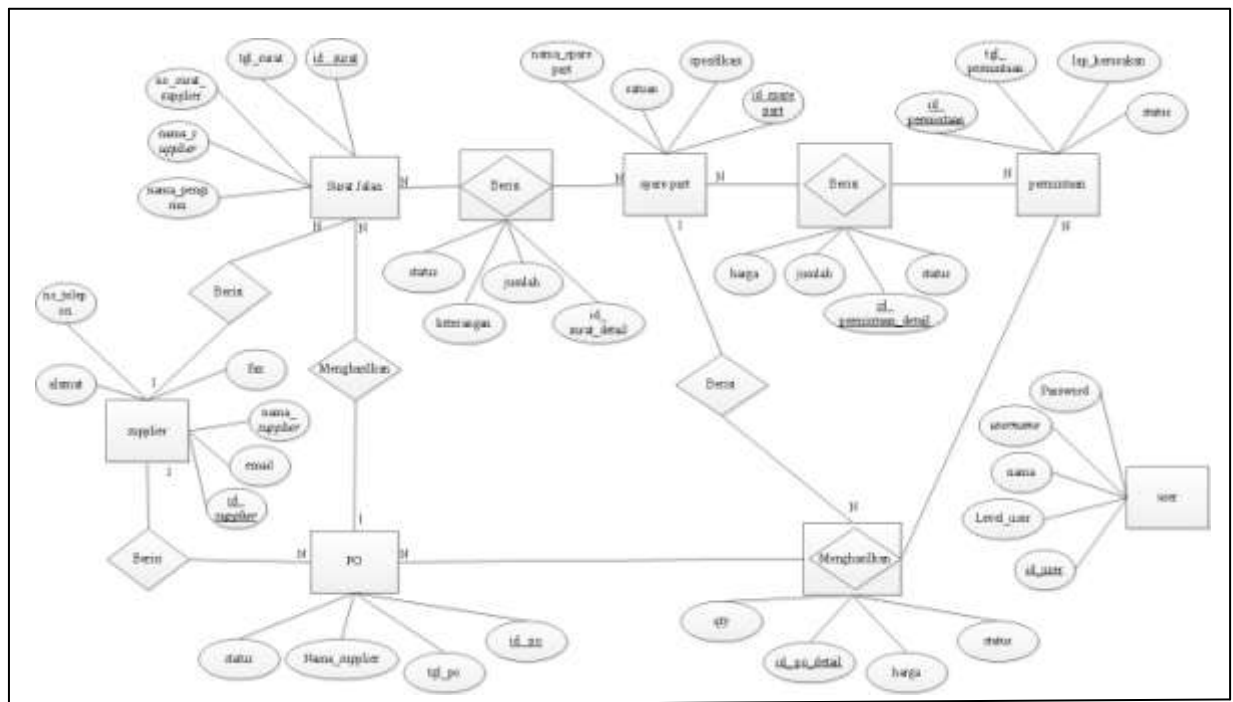
Gambar V.22 *Deployment Diagram*
(Sumber: Hasil Analisis, 2018)

5.4 Pemodelan Data Sistem Usulan

Pemodelan data pada sistem informasi pembelian *spare part* mesin *stamping* di PT Nusa Indah Jaya Utama 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 *spare part* mesin *stamping* pada Gambar V.23:

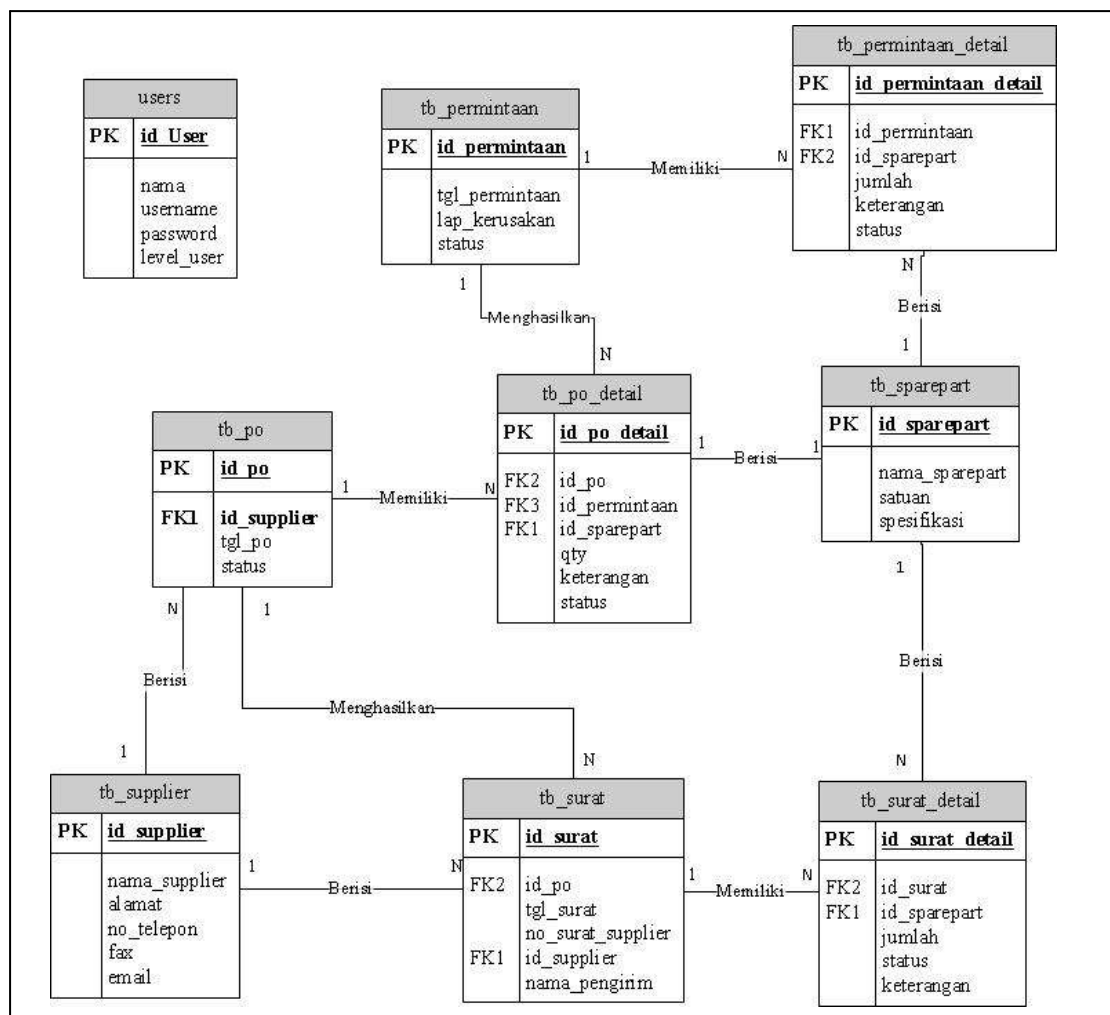


Gambar V.23 Entity Relationship Diagram

(Sumber: Hasil Analisis, 2018)

5.4.2 Conceptual Data Model (CDM)

Conceptual Data Model digunakan untuk menggambarkan atau merepresentasikan seluruh muatan informasi yang dikandung oleh basis data, berikut merupakan *Conceptual Data Model* dari sistem usulan pada Gambar V.24:



Gambar V.24 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 *spare part* mesin *stamping*:

1. Spesifikasi tabel *user*

- Nama tabel : users
- Fungsi : Untuk menyimpan data *user*/pengguna
- Tipe : File data *master*

Tabel V.13 Spesifikasi Tabel User

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id User	<i>id_user</i>	Int	11	<i>Primary key</i>
2	Nama	nama	Varchar	40	
3	<i>Username</i>	<i>username</i>	Varchar	40	
4	<i>Password</i>	<i>password</i>	Varchar	40	
5	Level_user	<i>level_user</i>	Varchar	40	

(Sumber: Hasil Analisis, 2018)

2. Spesifikasi tabel *spare part*

- a. Nama tabel : *tb_spare part*
- b. Fungsi : Untuk menyimpan data *spare part*
- c. Tipe : File data *master*

Tabel V.14 Spesifikasi Tabel *Spare part*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id <i>spare part</i>	<i>id_spare part</i>	Varchar	7	<i>Primary key</i>
2	Nama <i>spare part</i>	<i>nama_spare part</i>	Varchar	35	
3	Satuan	satuan	Varchar	10	
4	Spesifikasi	spesifikasi	Varchar	200	

(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.15 Spesifikasi Tabel *Supplier*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id <i>supplier</i>	<i>id_supplier</i>	Varchar	7	<i>Primary key</i>
2	Nama <i>supplier</i>	<i>nama_supplier</i>	Varchar	30	
3	Alamat	Alamat	Varchar	300	
4	No. telepon	no_telepon	Varchar	13	
5	Fax	Fax	Varchar	13	

6	Email	Email	Varchar	30	
---	-------	-------	---------	----	--

(Sumber: Hasil Analisis, 2018)

4. Spesifikasi tabel permintaan

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

Tabel V.16 Spesifikasi Tabel Permintaan

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id permintaan	id_ permintaan	Varchar	7	<i>Primary key</i>
2	Tangga permintaan	tgl_ permintaan	Date		
3	Laporan kerusakan	lap_ kerusakan	Varchar	300	
4	Status	Status	Varchar	20	
5	Keterangan	keterangan	Varchar	120	

(Sumber: Hasil Analisis, 2018)

5. Spesifikasi tabel permintaan detail

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

Tabel V.17 Spesifikasi Tabel Permintaan Detail

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id permintaan detail	id_ permintaan_detail	Int	11	<i>Primary key</i>
2	Id permintaan	id_ permintaan	Varchar	7	<i>Foreign key</i>
3	Id <i>spare part</i>	id_ <i>spare part</i>	Varchar	7	<i>Foreign key</i>
4	Jumlah	Jumlah	Int	11	
5	Keterangan	keterangan	Varchar	120	

(Sumber: Hasil Analisis, 2018)

6. Spesifikasi tabel surat jalan

- a. Nama tabel : tb_ surat
- b. Fungsi : Untuk menyimpan data surat jalan

- c. Tipe : File data transaksi

Tabel V.18 Spesifikasi Tabel Surat

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id surat	id_surat	Varchar	7	Primary key
2	Id PO	id_po	Varchar	7	Foreign key
3	Tanggal surat	tgl_surat	Date		
4	No surat dari supplier	no_surat_supplier	Varchar	20	
5	Id Supplier	id_supplier	Varchar	7	Foreign key
6	Nama pengirim	nama_pengirim	Varchar	50	

(Sumber: Hasil Analisis, 2018)

7. Spesifikasi tabel surat jalan detail

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

Tabel V.19 Spesifikasi Tabel Surat Detail

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id surat detail	id_surat_detail	Int	11	Primary key
2	Id surat	id_surat	Varchar	7	Foreign key
3	Id spare part	id_spare part	Varchar	7	Foreign key
4	Jumlah	Jumlah	Int	11	
5	Keterangan	keterangan	Varchar	100	

(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.20 Spesifikasi Tabel *Purchase Order*

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id <i>purchase order</i>	id_po	Varchar	7	Primary key
2	Id supplier	id_supplier	Varchar	7	Foreign key
3	Tanggal	tgl_po	Date		

	<i>purchase order</i>				
--	-----------------------	--	--	--	--

(Sumber: Hasil Analisis, 2018)

9. Spesifikasi tabel *purchase order* detail

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

Tabel V.21 Spesifikasi Tabel *Purchase Order* Detail

No.	Nama Elemen	Akronim	Tipe Data	Panjang	Keterangan
1	Id detail <i>purchase order</i>	id_po_detail	Int	11	<i>Primary key</i>
2	Id <i>purchase order</i>	id_po	Varchar	7	<i>Foreign key</i>
3	Id permintaan	id_permintaan	Varchar	7	<i>Foreign key</i>
4	Kuantitas	Qty	Int	11	
5	Keterangan	keterangan	Varchar	120	

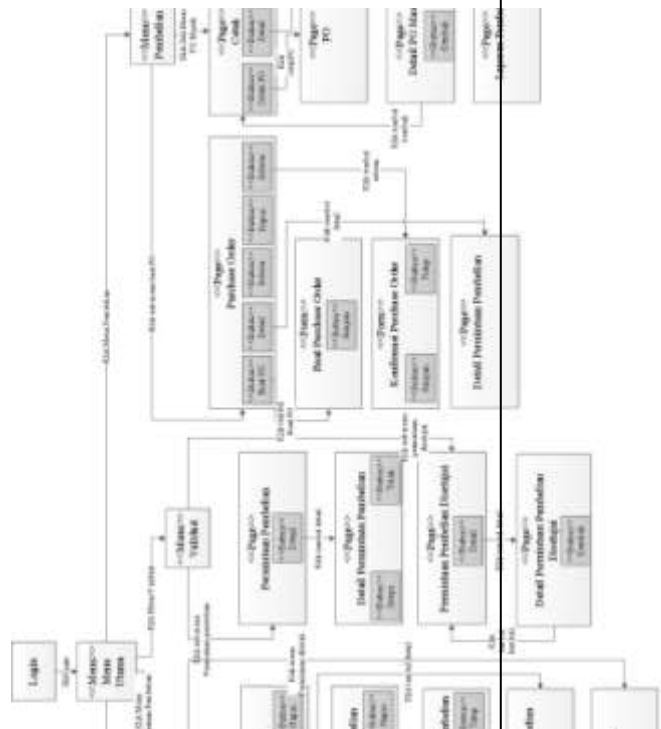
(Sumber: Hasil Analisis, 2018)

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 *spare part* mesin *stamping* pada Gambar V.25:



Gambar V.25 Windows Navigation Diagram

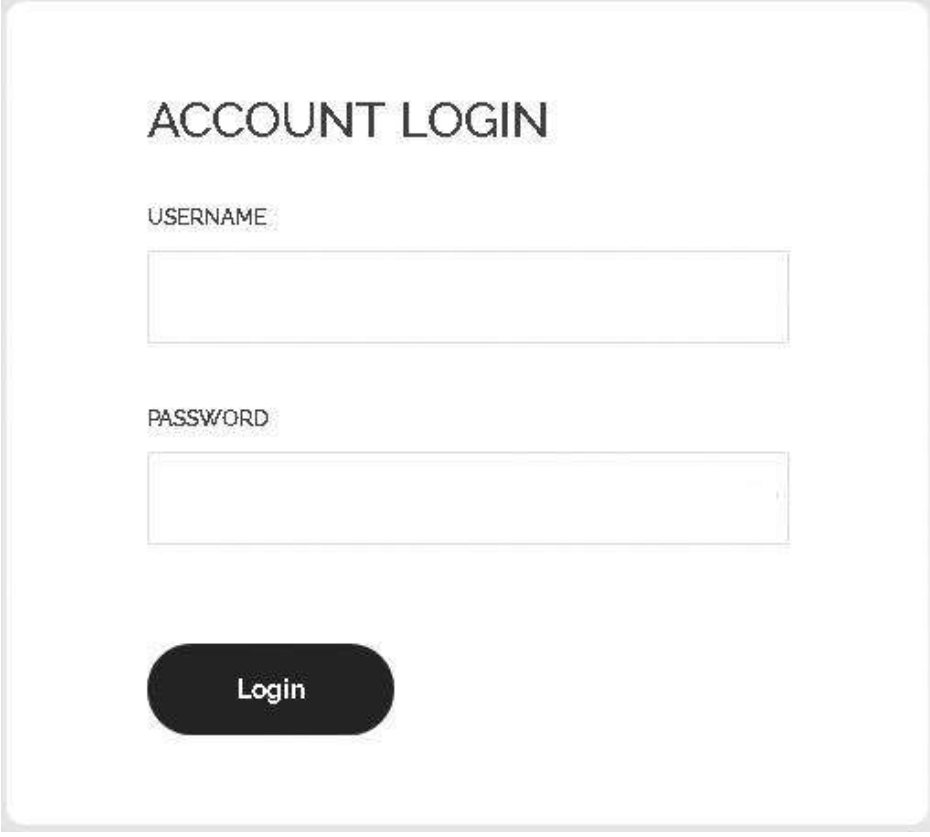
(Sumber: Hasil Analisis, 2018)

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 *spare part* mesin *stamping*:

1. Form Login

Form login merupakan *form* yang digunakan pengguna sebagai jalan masuk ke dalam aplikasi dengan cara memasukkan *username* dan *password* yang terdapat didalam *form login* tersebut, rancangan *form login* dapat dilihat pada Gambar V.26 berikut:

A login form titled "ACCOUNT LOGIN" in a bold, sans-serif font. Below the title, there are two input fields: the first is labeled "USERNAME" and the second is labeled "PASSWORD". Both labels are in a smaller, bold, sans-serif font. The input fields are rectangular with a thin border. Below the input fields, there is a dark, rounded rectangular button with the word "Login" in white, bold, sans-serif font. The entire form is enclosed in a light gray border with rounded corners.

ACCOUNT LOGIN

USERNAME

PASSWORD

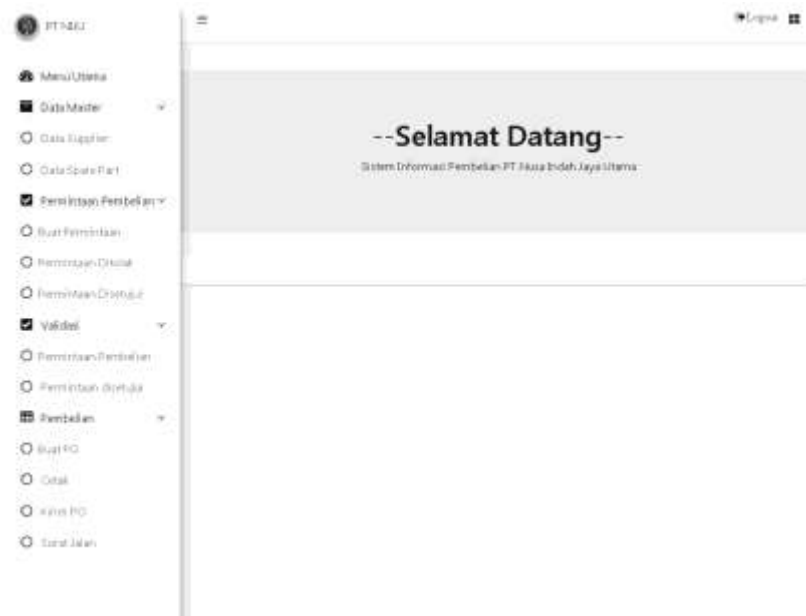
Login

Gambar V.26 *Form Login*

(Sumber: Hasil Analisis, 2018)

2. Tampilan Halaman Utama

Tampilan halaman utama merupakan tampilan yang muncul pertama kali saat user berhasil melakukan *login* ke dalam sistem, tampilan halaman utama dapat dilihat pada Gambar V.27 sebagai berikut:

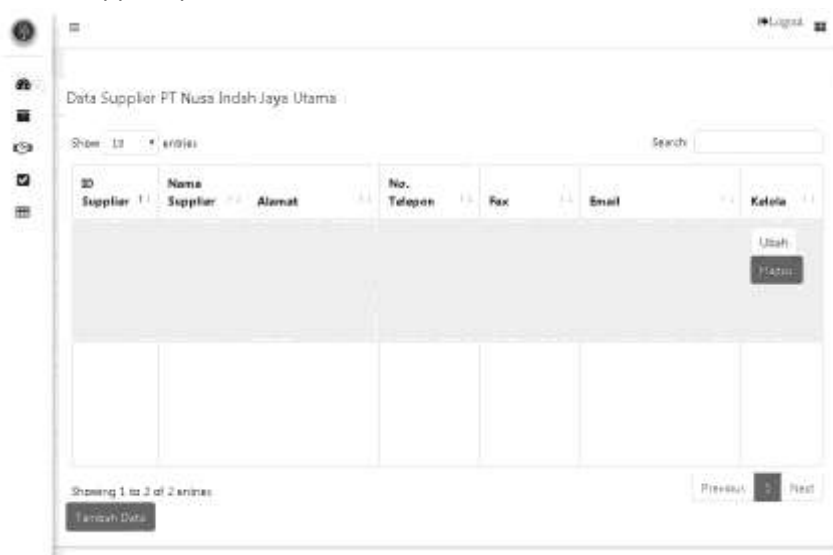


Gambar V.27 Tampilan Halaman Utama

(Sumber: Hasil Analisis, 2018)

3. Tampilan Halaman Data Master *Supplier*

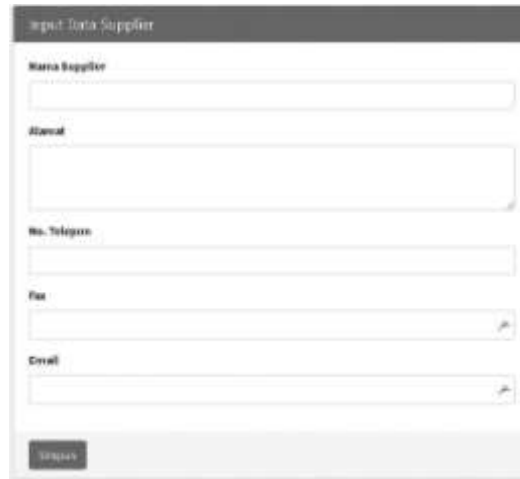
Tampilan halaman data master *supplier* merupakan tampilan yang muncul ketika user memilih sub menu data *supplier*, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data master *supplier*, berikut tampilan halaman data master *supplier* pada Gambar V.28:



Gambar V.28 Tampilan Halaman Data Master *Supplier*

(Sumber: Hasil Analisis, 2018)

Pada tampilan data master *supplier* diatas terdapat tiga tombol yaitu tambah data, ubah, dan hapus jika tombol tambah data atau ubah dipilih maka sistem akan menampilkan sebuah *form* untuk menambah data atau mengubah data *supplier* sebagai berikut pada Gambar V.29:

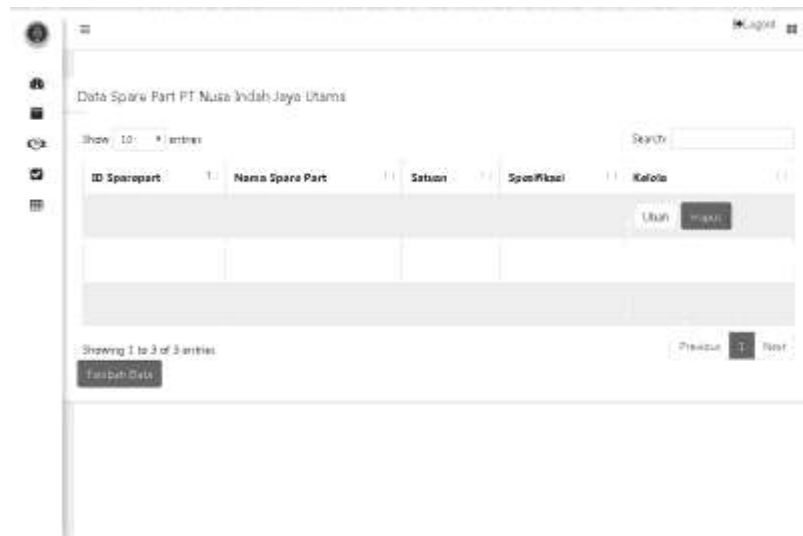


Gambar V.29 Tampilan *Form Input Data Master Supplier*

(Sumber: Hasil Analisis, 2018)

4. Tampilan Halaman Data Master *Spare part*

Tampilan halaman data master *spare part* merupakan tampilan yang muncul ketika user memilih sub menu data *spare part*, halaman ini menyediakan tombol-tombol yang dapat digunakan untuk mengelola data master *spare part*, berikut tampilan halaman data master *spare part* pada Gambar V.30:



Gambar V.30 Tampilan Halaman Data Master *Spare part*

(Sumber: Hasil Analisis, 2018)

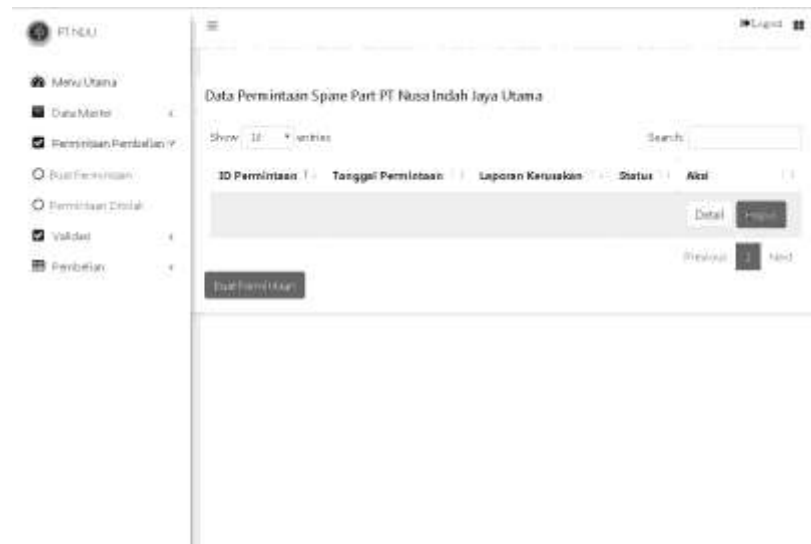
Pada tampilan data master *spare part* diatas terdapat tiga tombol yaitu tambah data, ubah, dan hapus jika tombol tambah data atau ubah dipilih maka sistem akan menampilkan sebuah *form* untuk menambah data atau mengubah data *spare part* sebagai berikut pada Gambar V.31:

Gambar V.31 Tampilan *Form Input Data Master Spare part*

(Sumber: Hasil Analisis, 2018)

5. Halaman Buat Permintaan

Halaman buat permintaan merupakan halaman yang tampil ketika user memilih menu sub menu buat permintaan pada halaman ini akan ditampilkan list permintaan yang telah dilakukan, berikut tampilan halaman buat permintaan:



Gambar V.32 Tampilan Halaman Buat Permintaan

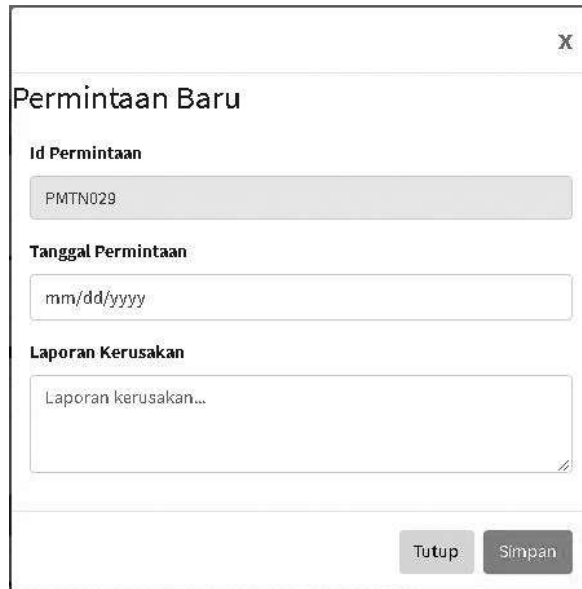
(Sumber: Hasil Analisis, 2018)

Pada halaman tampilan buat permintaan diatas terdapat tiga tombol yaitu tombol hapus untuk menghapus data permintaan, tombol detail untuk melihat detail permintaan, dan tombol buat permintaan untuk membuat permintaan baru, berikut *form* yang akan muncul apabila memilih tombol buat permintaan:

Gambar V.33 Tampilan *Form* Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

Jika user memilih tombol selesai pada *form* permintaan maka akan tampil *form* konfirmasi sebagai berikut pada Gambar V.34:



Permintaan Baru

Id Permintaan
PMTN029

Tanggal Permintaan
mm/dd/yyyy

Laporan Kerusakan
Laporan kerusakan...

Tutup Simpan

Gambar V.34 Tampilan *Form* Konfirmasi Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

Berikut merupakan halaman detail yang muncul jika user memilih tombol detail pada halaman buat permintaan pada Gambar V.35:



Detail Permintaan dengan id permintaan

ID Permintaan

Tanggal Permintaan

Laporan Kerusakan

Status

Spare Part yang diminta

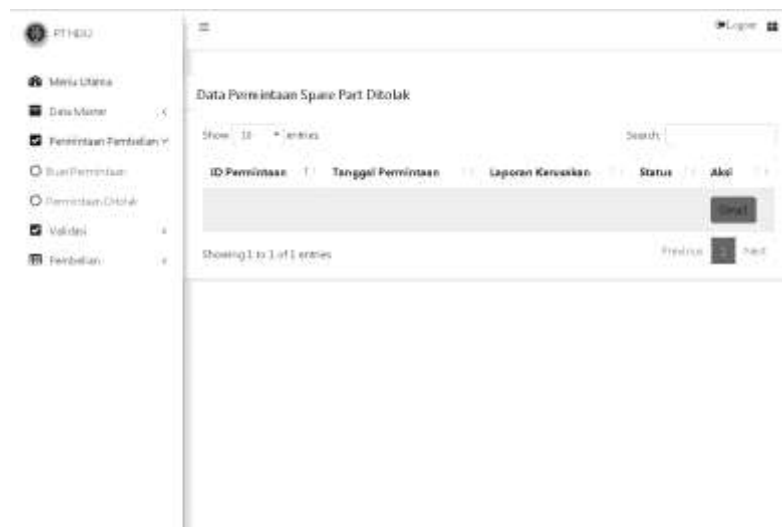
No	Nama Spare Part	Jumlah Permintaan	Harga
----	-----------------	-------------------	-------

Gambar V.35 Tampilan Detail Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

6. Halaman Permintaan Ditolak

Halaman permintaan ditolak merupakan halaman yang tampil ketika user memilih menu sub menu permintaan ditolak pada halaman ini akan ditampilkan list permintaan tidak disetujui oleh manajer, berikut tampilan halaman permintaan ditolak pada Gambar V.36:

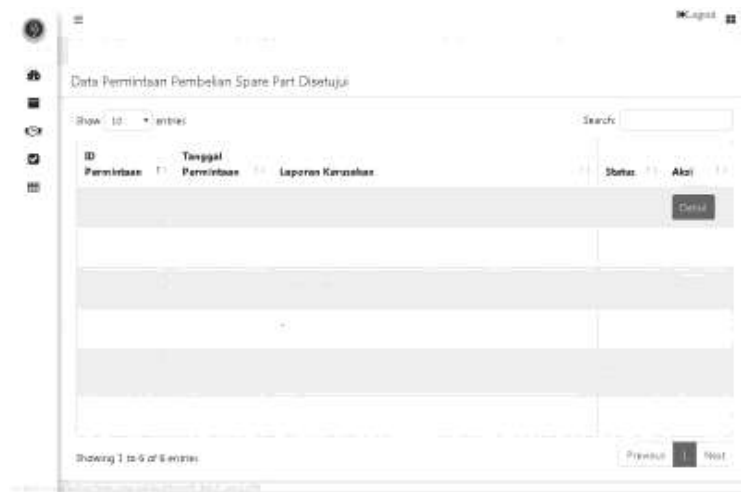


Gambar V.36 Tampilan Halaman Permintaan Ditolak

(Sumber: Hasil Analisis, 2018)

7. Halaman Permintaan Disetujui

Halaman permintaan disetujui merupakan halaman yang tampil ketika user memilih menu permintaan pembelian lalu memilih sub menu permintaan disetujui pada halaman ini akan ditampilkan list permintaan disetujui oleh manajer, berikut tampilan halaman permintaan disetujui pada Gambar V.37:



Gambar V.37 Tampilan Halaman Permintaan Disetujui

(Sumber: Hasil Analisis, 2018)

Jika user memilih tombol detail pada tampilan diatas maka sistem akan menampilkan halaman detail permintaan yang telah disetujui sebagai berikut:

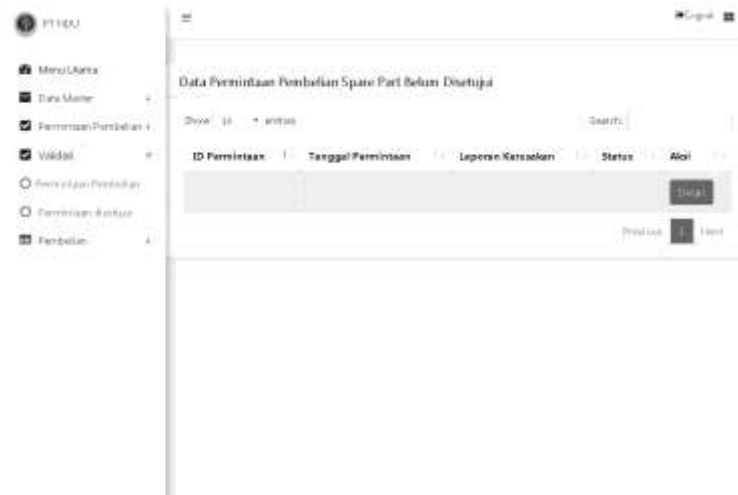


Gambar V.38 Tampilan Halaman Detail Permintaan Disetujui

(Sumber: Hasil Analisis, 2018)

8. Halaman Penyetujuan Permintaan Pembelian

Halaman penyetujuan permintaan pembelian merupakan halam yang ditujukan kepada manajer purchasing untuk melihat permintaan yang diajukan namun belum disetujui, berikut halaman penyetujuan permintaan pembelian :



Gambar V.39 Tampilan Halaman Penyetujuan Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

Pada halaman diatas terdapat tombol detail yang apabila dipilih akan menampilkan halaman detail permintaan dan tombol persetujuan, tampilannya sebagai berikut pada Gambar V.40:

Gambar V.40 Tampilan Halaman Detail Penyetujuan Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

Jika manajer menolak permintaan yang diajukan maka akan muncul sebuah form untuk memasukan alasan tolak sebagai berikut pada Gambar V.41:

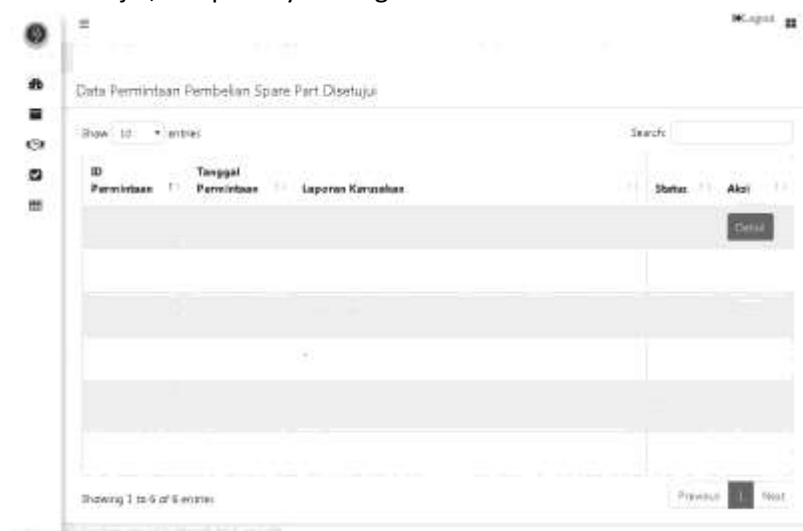


Gambar V.41 Tampilan Form Alasan Tolak Permintaan

(Sumber: Hasil Analisis, 2018)

9. Halaman Permintaan Disetujui

Halaman permintaan disetujui merupakan halaman yang menampilkan permintaan-permintaan yang telah disetujui oleh manajer *purchasing*, halaman ini akan muncul apabila user yaitu manajer purchasing memilih menu validasi lalu memilih menu permintaan disetujui, tampilannya sebagai berikut:



ID Permintaan	Tanggal Permintaan	Laporan Kerusakan	Status	Aksi
				Detail

Gambar V.42 Tampilan Halaman Permintaan Disetujui

(Sumber: Hasil Analisis, 2018)

Jika user memilih tombol detail pada tampilan diatas maka sistem akan menampilkan halaman detail permintaan yang telah disetujui sebagai berikut:

Gambar V.43 Tampilan Halaman Detail Permintaan Disetujui

(Sumber: Hasil Analisis, 2018)

10. Halaman Buat PO

Halaman buat PO merupakan halaman yang muncul ketika user memilih sub menu buat PO, pada halaman ini menampilkan permintaan yang siap untuk dijadikan PO, berikut tampilannya:

Gambar V.44 Tampilan Halaman Buat PO

(Sumber: Hasil Analisis, 2018)

Pada halaman tersebut terdapat beberapa tombol diantaranya tombol buat PO, tombol selesai untuk mengupdate permintaan, dan tombol detail untuk menampilkan halaman detail, dan tombol selesai pada tabel bagian bawah untuk menyelesaikan PO, berikut tampilan *form* buat PO:

The screenshot shows a web form titled "Input Data Purchase Order". It has the following fields and controls:

- Id Permintaan:** A text input field containing the value "PMTN004".
- Spare Part:** A dropdown menu with the placeholder text "--pilih spare part --".
- Jumlah Beli:** A text input field.
- Keterangan:** A text input field.
- Simpan:** A button at the bottom of the form.

Gambar V.45 Tampilan *Form* Buat PO

(Sumber: Hasil Analisis, 2018)

Jika user memilih detail maka akan tampil detail dari permintaan sebagai berikut:

The screenshot shows a web form titled "Detail Permintaan". It displays the following information:

- ID Permintaan:** A text input field.
- Tanggal Permintaan:** A text input field.
- Laporan Kerusakan:** A text input field.
- Status:** A text input field.
- Spare Part yang diminta:** A table with the following structure:

No	Nama Spare Part	Jumlah Permintaan	Harga
- Kembali:** A button at the bottom of the form.

Gambar V.46 Tampilan *Form* PO Detail

(Sumber: Hasil Analisis, 2018)

Jika user memilih tombol selesai pada bagian tabel data pesanan po *spare part* pt nusa indah jaya utama maka sistem akan menampilkan *form* konfirmasi PO sebagai berikut:

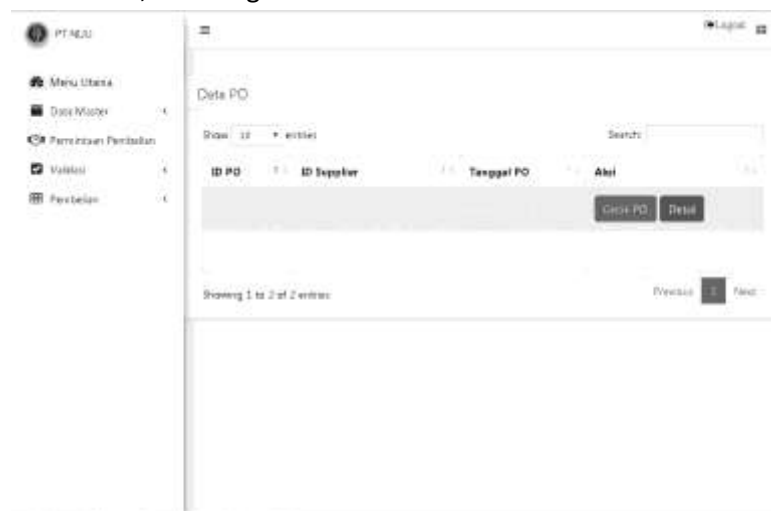


Gambar V.47 Tampilan *Form* Konfirmasi PO

(Sumber: Hasil Analisis, 2018)

11. Halaman Cetak

Halaman cetak menampilkan PO yang telah dibuat oleh divisi purchasing dan siap dikirimkan ke *supplier* yang telah ditentukan terdapat dua tombol yaitu tombol cetak PO dan tombol detail, berikut gambar dari halaman PO masuk:



Gambar V.48 Tampilan Halaman Cetak

(Sumber: Hasil Analisis, 2018)

Jika user memilih tombol detail maka sistem akan menampilkan halaman detail sebelum PO dicetak, berikut tampilan halaman detail PO keluar:

Detail PO

ID Purchase Order

Tanggal Purchase Order

Nama Supplier

Spare Part yang diminta

No	Nama Spare Part	Jumlah Permintaan	Harga

Kembali

Gambar V.49 Tampilan Detail PO Keluar

(Sumber: Hasil Analisis, 2018)

Untuk tombol cetak PO sistem akan *generate* PO dari tabel PO detail dan tabel PO agar menjadi sebuah PO dengan bentuk PDF yang siap dikirimkan kepada *supplier*, berikut PO yang dibuat oleh sistem:

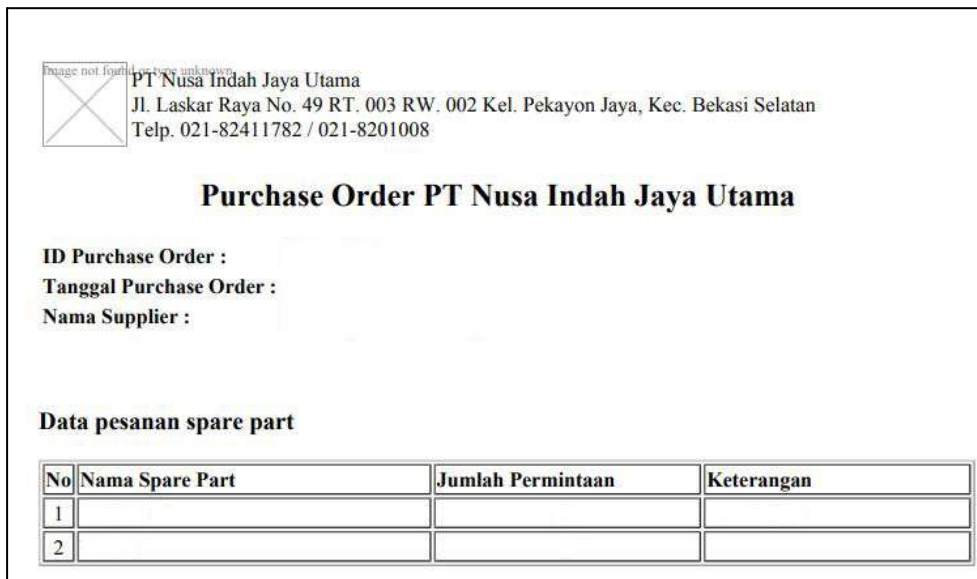


image not found
PT Nusa Indah Jaya Utama
Jl. Laskar Raya No. 49 RT. 003 RW. 002 Kel. Pekayon Jaya, Kec. Bekasi Selatan
Telp. 021-82411782 / 021-8201008

Purchase Order PT Nusa Indah Jaya Utama

ID Purchase Order :
Tanggal Purchase Order :
Nama Supplier :

Data pesanan spare part

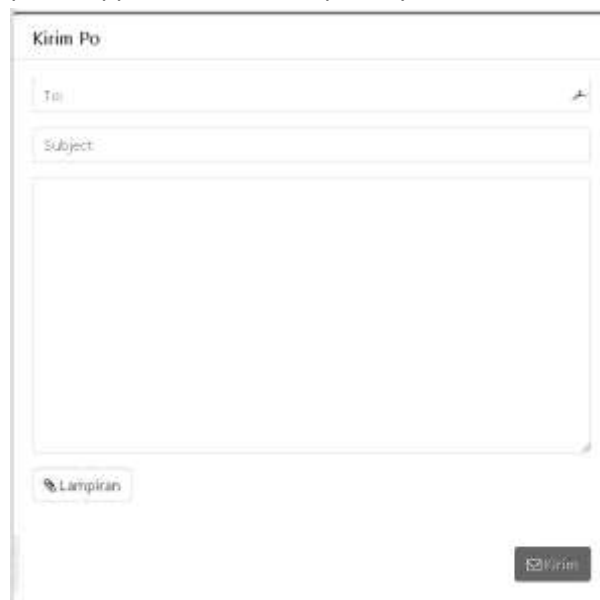
No	Nama Spare Part	Jumlah Permintaan	Keterangan
1			
2			

Gambar V.50 *Purchase Order*

(Sumber: Hasil Analisis, 2018)

12. Halaman Kirim PO

Halaman kirim PO merupakan halaman yang muncul apabila user memilih menu pembelian lalu memilih sub menu kirim PO, halaman ini menampilkan form untuk mengirim PO kepada *supplier*, berikut tampilannya:



Kirim Po

To:

Subject:

Lampiran

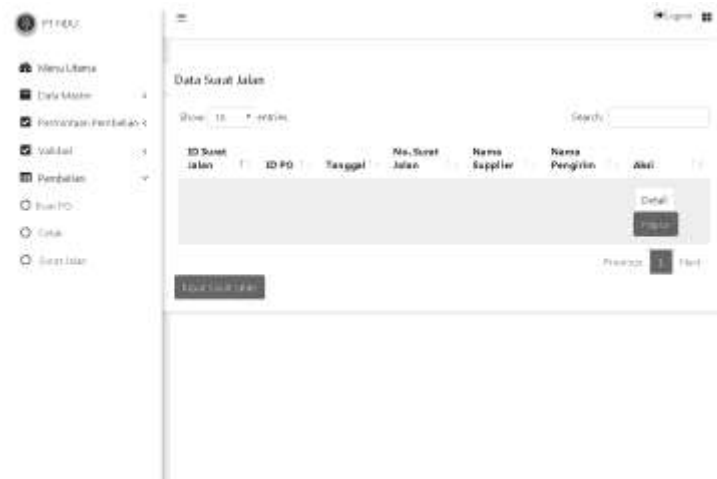
Kirim

Gambar V.51 Halaman Kirim PO

(Sumber: Hasil Analisis, 2018)

13. Halaman Surat Jalan

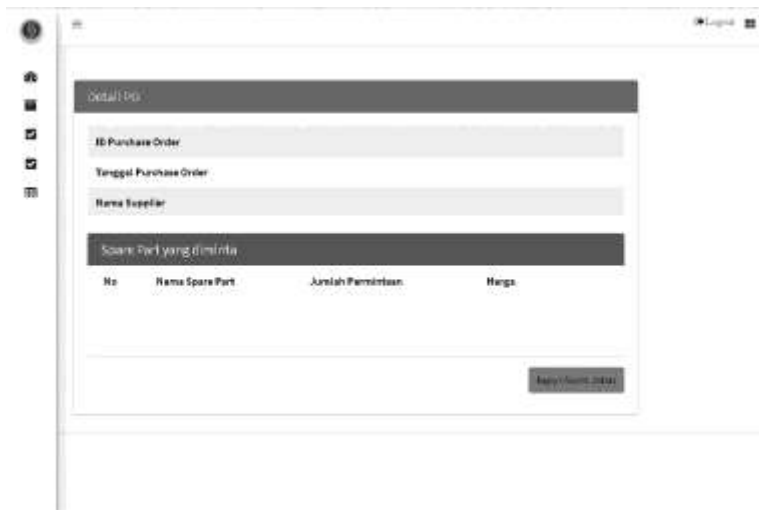
Halaman surat jalan merupakan halaman yang tampil ketika user memilih sub menu surat jalan pada halaman ini akan ditampilkan list surat jalan yang telah diterima perusahaan, berikut tampilan halaman surat jalan:



Gambar V.52 Halaman Surat Jalan

(Sumber: Hasil Analisis, 2018)

Pada halaman tampilan surat diatas terdapat tiga tombol yaitu tombol hapus untuk menghapus data surat jalan, tombol detail untuk melihat detail surat jalan, dan tombol *input* surat jalan untuk meng*input* surat jalan baru, berikut halaman yang akan muncul apabila memilih tombol *input* surat jalan:



Gambar V.53 Halaman *Input* Surat Jalan

(Sumber: Hasil Analisis, 2018)

Jika user memilih tombol *input* surat jalan pada halaman *input* surat jalan maka akan tampil *form* konfirmasi sebagai berikut:

Konfirmasi Surat Jalan

Id Surat Jalan

Id PO

Tanggal Surat

mm/dd/yyyy

No Surat dari Supplier

Id Supplier

Nama Pengirim

Tutup Simpan

Gambar V.54 *Form Konfirmasi Input Surat Jalan*

(Sumber: Hasil Analisis, 2018)

Berikut merupakan halaman detail yang muncul jika user memilih tombol detail pada halaman surat jalan:

Detail Surat Jalan

ID Surat Jalan	SRTJ004
ID PO	PORD001
Tanggal Surat	2018-07-17
No. Surat Supplier	1234
Nama pengirim	Kevin
Nama Supplier	PT Timurray Lestari

Spare Part yang diminta

No	Nama Spare Part	Jumlah	Keterangan
1	Compressor	1	

Gambar V.55 *Form Konfirmasi Detail Surat Jalan*

(Sumber: Hasil Analisis, 2018)

14. Form Cetak Laporan

Form cetak laporan merupakan form yang digunakan untuk mengelola tanggal yang diinginkan sebagai *range* dalam pencetakan laporan, *form* ini terdapat di sub menu cetak, berikut tampilan halaman *form* cetak laporan:

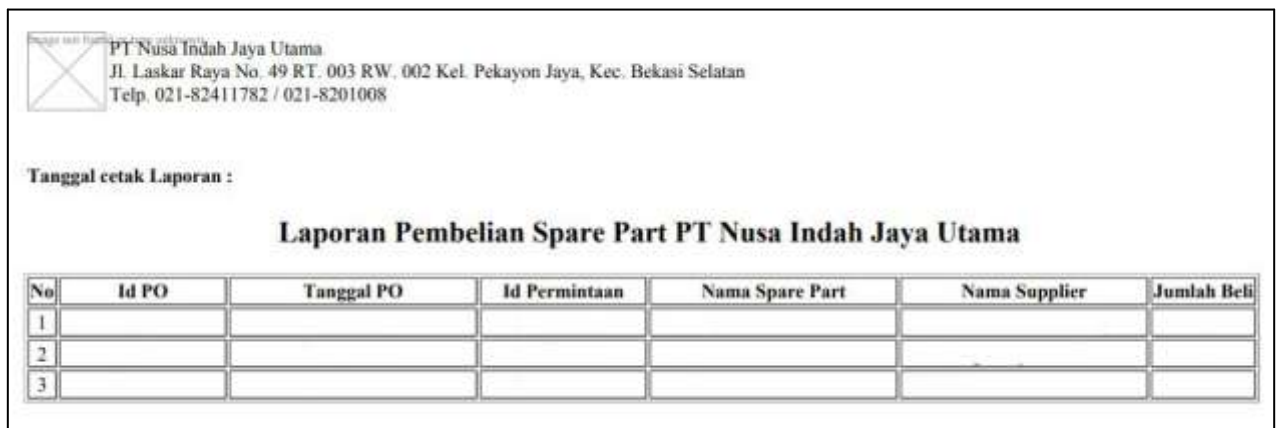


Gambar V.56 Form Cetak Laporan

(Sumber: Hasil Analisis, 2018)

15. Laporan Pembelian

Laporan pembelian merupakan hasil dari rekap data pembelian yang telah dilakukan oleh divisi *prchasing*, laporan pembelian berbentuk PDF yang dapat diunduh oleh staff untuk selanjutnya dikirimkan kepada *supplier*, berikut laporan pembelian *spare part*



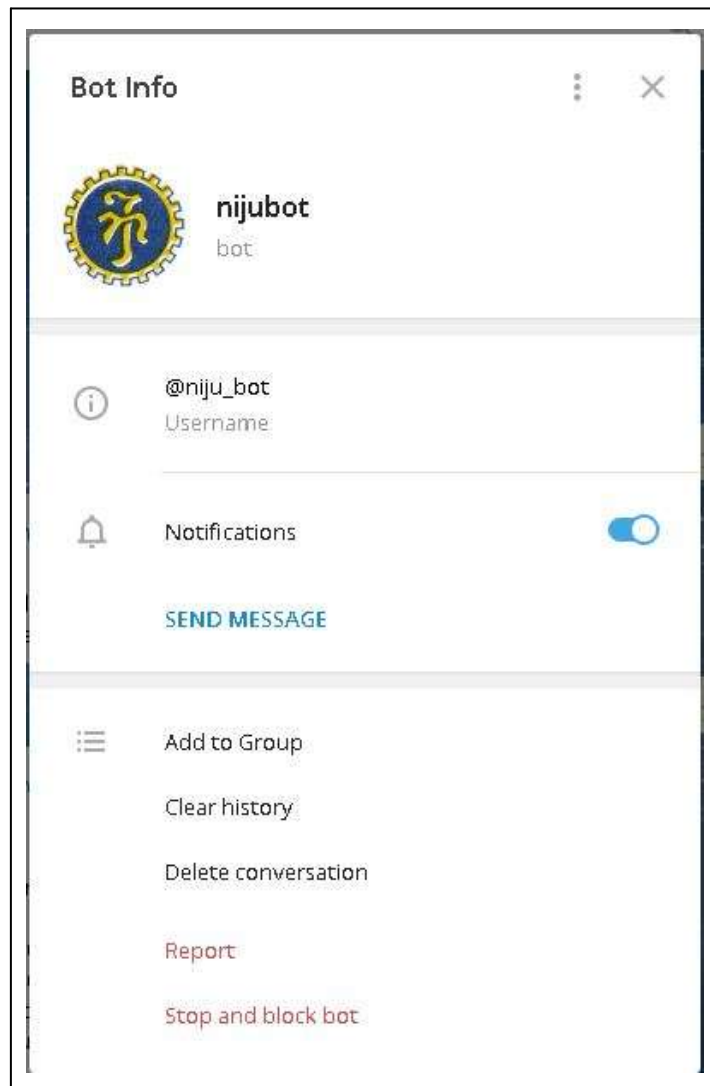
No	Id PO	Tanggal PO	Id Permintaan	Nama Spare Part	Nama Supplier	Jumlah Beli
1						
2						
3						

Gambar V.57 Laporan Pembelian *Spare part*

(Sumber: Hasil Analisis, 2018)

5.5.3 Fitur Bot Telegram

Fitur bot telegram merupakan fitur tambahan yang ada didalam sistem informasi pembelian *spare part* mesin *stamping*. Bot ini bernama @niju_bot Fitur ini dapat digunakan oleh user untuk melakukan pengecekan informasi yang berhubungan dengan sistem informasi pembelian seperti melihat data permintaan yang belum dilakukan proses persetujuan, melihat data permintaan yang telah disetujui, dan melihat data permintaan yang ditolak oleh manajer. Berikut merupakan profil Bot pembelian PT Nusa Indah Jaya Utama



Gambar V.58 Profil Bot

(Sumber: Hasil Analisis, 2018)

Bot Telegram akan menampilkan informasi berdasarkan perintah yang diberikan oleh user kepada Bot melalui aplikasi telegram, berikut merupakan halaman awal ketika berinteraksi dengan bot



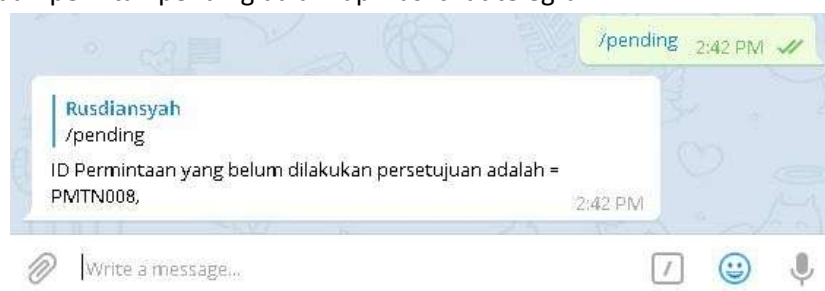
Gambar V.59 Interaksi Awal Bot

(Sumber: Hasil Analisis, 2018)

Perintah /help diatas menjelaskan perintah-perintah lain yang dapat digunakan untuk berinteraksi dengan Bot, berikut penjelasan dari perintah-perintah tersebut:

1. /pending

Perintah ini dapat digunakan oleh manajer *purchasing* untuk melihat data permintaan yang masuk dan belum dilakukan proses persetujuan. Berikut merupakan hasil dari perintah pending dalam aplikasi chat telegram:

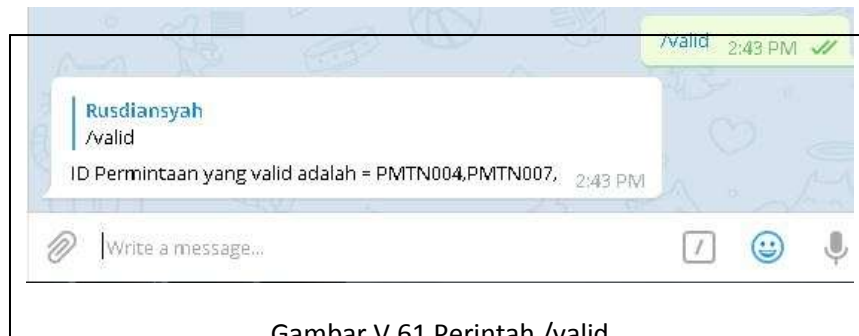


Gambar V.60 Perintah /pending

(Sumber: Hasil Analisis, 2018)

2. /valid

Perintah ini digunakan sebagai perintah untuk melihat data permintaan yang telah disetujui oleh manajer, perintah ini dapat digunakan oleh staff maintenance, berikut tampilan perintah tersebut:

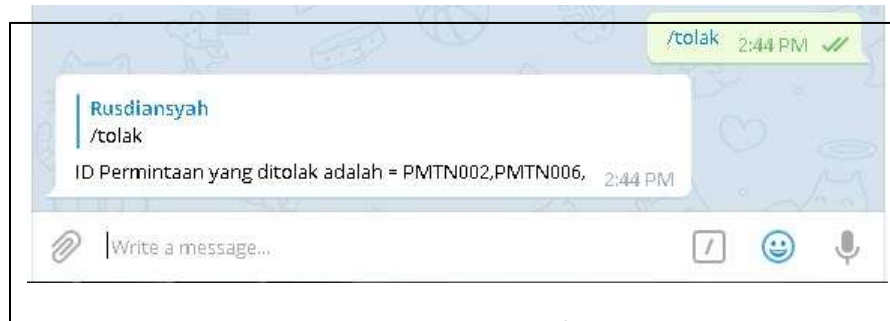


Gambar V.61 Perintah /valid

(Sumber: Hasil Analisis, 2018)

3. /tolak

Perintah ini digunakan untuk menampilkan permintaan yang ditolak oleh manajer purchasing. Berikut tampilannya:

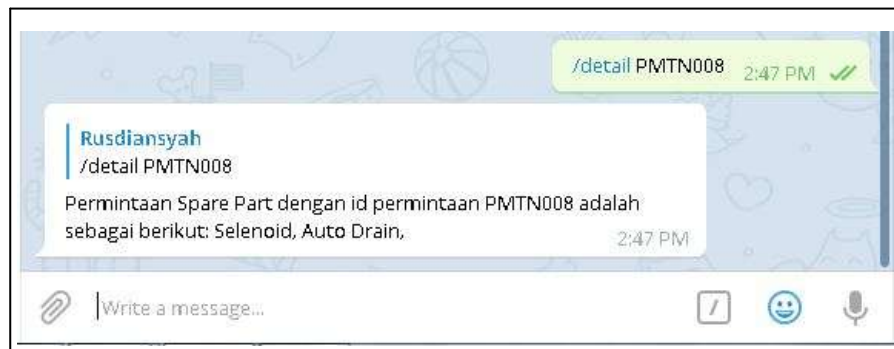


Gambar V.62 Perintah /tolak

(Sumber: Hasil Analisis, 2018)

4. /detail

Perintah ini dapat digunakan untuk melihat list spare part yang diminta berdasarkan id permintaan, perintah ini diikuti dengan id permintaan yang akan dilihat contoh: /detail_rusak PMTN005. Berikut tampilan dalam aplikasi telegram:

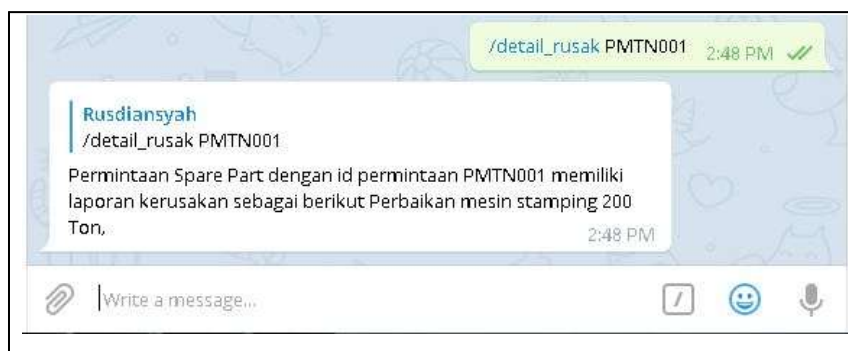


Gambar V.63 Perintah /detail

(Sumber: Hasil Analisis, 2018)

5. /detail_rusak

Perintah ini dapat digunakan oleh manajer purchasing dalam pengecekan terhadap laporan kerusakan yang terjadi pada permintaan pembelian yang diajukan, perintah ini diikuti dengan id permintaan yang akan dilihat contoh: /detail_rusak PMTN005.



Gambar V.64 Perintah /detail_rusak

(Sumber: Hasil Analisis, 2018)

6. /validasi

Perintah ini hanya dapat digunakan oleh manajer purchasing dalam melakukan persetujuan pembelian, perintah ini diikuti dengan id permintaan dan status yang diberikan apakah tolak atau valid, contoh: /validasi[id telegram manajer] PMTN002 Valid.

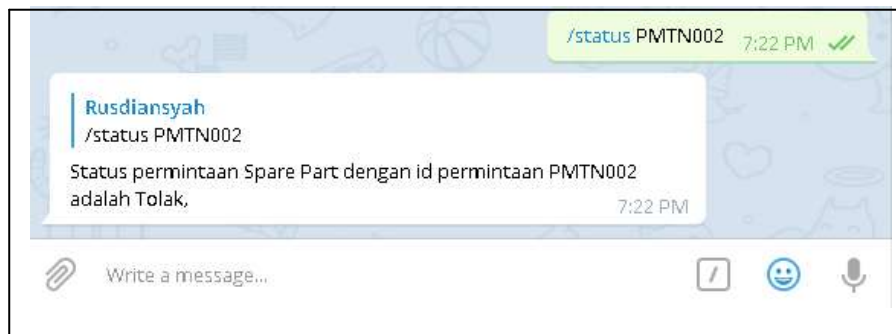


Gambar V.65 Perintah /validasi

(Sumber: Hasil Analisis, 2018)

7. /status

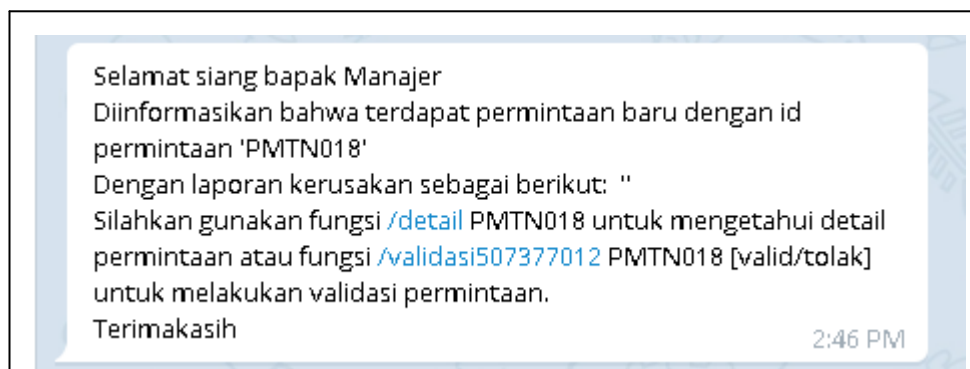
Perintah ini dapat digunakan untuk pengecekan terhadap status permintaan dengan id permintaan yang spesifik, perintah ini diikuti dengan id permintaan yang akan dilihat contoh: /status PMTN002.



Gambar V.66 Perintah /status

(Sumber: Hasil Analisis, 2018)

Selain perintah-perintah diatas telegram juga dapat memberikan informasi kepada manajer *purchasing* jika terdapat sebuah permintaan baru yang dilakukan oleh staff *maintenance* pada aplikasi sistem informasi pembelian *spare part* mesin *stamping*, sehingga dapat membuat proses validasi lebih cepat. Berikut tampilan notifikasi yang muncul pada aplikasi telegram jika permintaan pembelian dilakukan:



Gambar V.67 Notifikasi Permintaan Pembelian

(Sumber: Hasil Analisis, 2018)

5.6 Implementasi Sistem *Software dan Hardware*

Tahap implementasi ini dimaksudkan kepada perusahaan agar dapat menjalankan program yang telah diusulkan di dalam proses produksinya, untuk mendukung jalannya sistem usulan diperlukan suatu spesifikasi perangkat lunak dan perangkat keras agar program dapat berjalan, adapun spesifikasinya adalah sebagai berikut:

1. Processor minimal quad core 64 bit dengan clock rate minimal 1.6 GHz.
2. RAM minimal 8 GB, DDR 3
3. Kapasitas hard disk minimal 250 GB.
4. Sistem operasi (32 bit): Windows 7 atau Windows 8.
5. Browser seperti Google Chrome atau Mozilla Firefox.
6. LAN Card.
7. Keyboard, mouse, layar monitor.

5.7 Functional Testing

Functional testing merupakan pengujian yang dilakukan untuk memeriksa apakah fungsionalitas dari aplikasi yang dibangun sesuai dengan apa yang diharapkan, *functional testing* menguji sistem berdasarkan *use case* yang telah dibuat berikut merupakan pengujian yang dilakukan terhadap aplikasi sistem informasi pembelian *spare part* mesin *stamping*:

1. Functional Test Use Case Login

Berikut merupakan *functional test* dari *use case* login:

Tabel V.22 *Functional Test Use Case Login*

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Validasi Login	Masuk kedalam sistem menggunakan <i>username</i> dan <i>password</i> yang telah diberikan	<i>Username</i> = "purchase" <i>Password</i> = "purchase"	Berhasil masuk	Berhasil masuk	Pass
2	Validasi Login	Masuk kedalam sistem menggunakan	<i>Username</i> = "purchase" <i>Password</i> =	Tidak berhasil masuk	Tidak berhasil masuk	Pass

		<i>username</i> dan <i>password</i> yang tidak sesuai	"maintenance"	kedalam sistem	kedalam sistem	
3	Validasi Login	Tidak mengisi kolom <i>username</i> dan <i>password</i>	<i>Username</i> = " " <i>Password</i> = " "	Menampilkan pesan error kolom harus diisi	Menampilkan pesan error	<i>Pass</i>

(Sumber: Hasil Analisis, 2018)

2. Functional Test Use Case mengelola data *supplier*

Berikut merupakan *functional test* dari *use case* mengelola data *supplier*:

Tabel V.23 *Functional Test Use Case Mengelola Data Supplier*

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Menambah data <i>supplier</i>	Menambah data <i>supplier</i> dengan mengisi semua kolom pada form tambah data	Nama <i>spare part</i> = "Pandawa Jaya Steel" Satuan = "Jakarta" No.telepon = "0219890" Fax = "0219890" Email = "pandawa@gmail.com"	Berhasil tambah data	berhasil tambah data	<i>Pass</i>
2	Menambah data <i>supplier</i>	Menambah data <i>supplier</i> dengan mengosongkan salah satu kolom selain kolom fax dan email	Nama <i>Supplier</i> = "Pandawa Jaya Steel" Alamat = " " No.telepon = "0219890" Fax = "0219890" Email = "pandawa@gmail.com"	Muncul pesan kolom harus terisi	Muncul pesan kolom harus terisi	<i>Pass</i>
3	Mengubah data <i>supplier</i>	Melakukan perubahan pada atribut <i>supplier</i>	Mengganti alamat <i>supplier</i> dari jakarta menjadi bekasi	Data <i>supplier</i> berubah	Data <i>supplier</i> berubah	<i>Pass</i>
4	Menghapus data <i>supplier</i>	Hapus data <i>supplier</i> dengan Id	Klik hapus	Menghapus data dari tabel	Menghapus data dari tabel	<i>pass</i>

	yang dipilih	Supplier="SPPL001"		supplier	supplier	
--	--------------	--------------------	--	----------	----------	--

(Sumber: Hasil Analisis, 2018)

3. *Functional Test Use Case* mengelola data *spare part*

Berikut merupakan *functional test* dari *use case* mengelola data *spare part*:

Tabel V.24 *Functional Test Use Case* Mengelola Data *Spare Part*

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Menambah data <i>spare part</i>	Menambah data <i>spare part</i> dengan mengisi semua kolom pada form tambah data	Nama <i>spare part</i> = "Auto Drain" Satuan = "Unit" Spesifikasi = "Applicable pressure"	Berhasil tambah data	berhasil tambah data	Pass
2	Menambah data <i>spare part</i>	Menambah data <i>spare part</i> dengan mengosongkan salah satu kolom	Nama <i>spare part</i> = "Auto Drain" Satuan = "Unit" Spesifikasi = " "	Muncul pesan kolom harus terisi	Muncul pesan kolom harus terisi	Pass
3	Mengubah data <i>spare part</i>	Melakukan perubahan pada atribut <i>spare part</i>	Mengganti satuan <i>spare part</i> dari unit menjadi pail	Data <i>spare part</i> berubah	Data <i>spare part</i> berubah	Pass
4	Menghapus data <i>spare part</i> yang dipilih berdasar id	Hapus data <i>spare part</i> dengan Id <i>spare part</i> ="SPPA015"	Klik hapus	Menghapus data dari tabel <i>spare part</i>	Menghapus data dari tabel <i>spare part</i>	pass

(Sumber: Hasil Analisis, 2018)

4. *Functional Test Use Case* mengelola permintaan pembelian

Berikut merupakan *functional test* dari *use case* mengelola permintaan pembelian:

Tabel V.25 *Functional Test Use Case* Mengelola Permintaan Pembelian

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Membuat permintaan baru	Membuat permintaan baru dengan mengisi semua kolom pada form	Memasukan data yang sesuai pada kolom nama <i>spare part</i> , jumlah dan keterangan	Berhasil tambah data detail permintaan	Berhasil tambah data detail permintaan	Pass

		permintaan				
2	Membuat permintaan baru	Memilih tombol selesai dan mengisi form konfirmasi permintaan	Memasukan data pada kolom tanggal permintaan dan laporan kerusakan	Berhasil membuat permintaan baru	Berhasil membuat permintaan baru	Pass
3	Menghapus data permintaan detail	Menghapus data permintaan detail dari tabel berdasar id permintaan detail	Memilih hapus pada tabel permintaan detail	Permintaan detail yang dipilih terhapus dari tabel	Permintaan detail yang dipilih terhapus dari tabel	Pass
4	Menghapus data permintaan	Menghapus data permintaan dari tabel berdasar id permintaan	Memilih hapus pada tabel permintaan	Permintaan yang dipilih terhapus dari tabel	Permintaan yang dipilih terhapus dari tabel	pass

(Sumber: Hasil Analisis, 2018)

5. *Functional Test Use Case* penyetujuan permintaan pembelian

Berikut merupakan *functional test* dari *use case* permintaan pembelian:

Tabel V.26 *Functional Test Use Case* Penyetujuan Permintaan Pembelian

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Menyetujui permintaan	Menyetujui permintaan berdasarkan id permintaan	Memilih tombol setuju	Status permintaan berubah menjadi valid	Status permintaan berubah menjadi valid	Pass

Tabel V.26 *Functional Test Use Case* Penyetujuan Permintaan Pembelian (lanjutan)

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
2	Menolak permintaan	Menolak permintaan berdasarkan id permintaan	Memilih tombol tolak	Mengeluarkan form konfirmasi tolak dan merubah status menjadi tolak	Mengeluarkan form konfirmasi tolak dan merubah status menjadi tolak	Pass
3	Menghapus permintaan detail	Menghapus salah satu permintaan detail berdasarkan <i>spare part</i> yang dipesan	Memilih tombol hapus pada tabel detail	Menghapus permintaan detail	Menghapus permintaan detail	pass

(Sumber: Hasil Analisis, 2018)

6. *Functional Test Use Case* mengelola PO

Berikut merupakan *functional test* dari *use case* mengelola PO:

Tabel V.27 *Functional Test Use Case* Mengelola PO

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Membuat PO	Membuat PO berdasarkan id permintaan	Mengisi semua kolom pada form buat PO	Menambah PO detail kedalam tabel	Menambah PO detail kedalam tabel	Pass
2	Membuat PO	Menambahkan PO	Mengisi semua kolom pada form konfirmasi PO	Menambah PO ke tabel	Menambah PO ke tabel	Pass
3	Menghapus PO detail	Menghapus PO detail berdasarkan id PO detail	Memilih tombol hapus pada tabel PO detail	Data PO detail yang dipilih akan terhapus dari tabel	Data PO detail yang dipilih akan terhapus dari tabel	pass

(Sumber: Hasil Analisis, 2018)

7. *Functional Test Use Case* mencetak PO

Berikut merupakan *functional test* dari *use case* mencetak PO:

Tabel V.28 *Functional Test Use Case* Mencetak PO

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Mencetak PO berdasarkan id PO	Melakukan cetak PO dengan tombol cetak pada tabel	Memilih tombol cetak PO pada tabel	Menampilkan PO berformat pdf dengan mengenerate tabel dari database	Menampilkan PO berformat pdf dengan mengenerate tabel dari database	Pass

(Sumber: Hasil Analisis, 2018)

8. *Functional Test Use Case* kirim PO

Berikut merupakan *functional test* dari *use case* kirim PO:

Tabel V.29 *Functional Test Use Case* Kirim PO

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Mengirim PO	Mengirim PO melalui email yang terdapat didalam aplikasi	Memasukan alamat email <i>supplier</i> yang dituju, menuliskan pesan dan memilih PO yang akan dikirim ke <i>supplier</i>	Email berisi PO terkirim ke <i>supplier</i>	Email berisi PO terkirim ke <i>supplier</i>	Pass
2	Mengirim PO	Mengirim email tanpa attachment	Memasukan alamat email <i>supplier</i> yang dituju, menuliskan pesan, tidak menginput attachment PO	Muncul pesan error dan email tidak terkirim	Muncul pesan error dan email tidak terkirim	Pass

(Sumber: Hasil Analisis, 2018)

9. *Functional Test Use Case* mengelola surat jalan

Berikut merupakan *functional test* dari *use case* mengelola surat jalan:

Tabel V.30 *Functional Test Use Case* Mengelola Surat Jalan

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Menginput surat jalan	Menginput surat jalan berdasarkan PO yang telah dikirimkan surat jalan dari <i>supplier</i>	Memilih tombol <i>input</i> surat jalan berdasarkan id PO dan memasukkan data surat jalan pada semua kolom form konfirmasi surat jalan	Menyimpan data surat jalan detail dan data surat jalan	Menyimpan data surat jalan detail dan data surat jalan	<i>Pass</i>
2	Menginput surat jalan	Menginput surat jalan berdasarkan PO yang telah dikirimkan surat jalan dari <i>supplier</i>	Memilih tombol <i>input</i> surat jalan berdasarkan id PO dan tidak mengisi semua kolom pada form konfirmasi surat jalan	Muncul pesan kolom harus diisi	Muncul pesan kolom harus diisi	<i>pass</i>

(Sumber: Hasil Analisis, 2018)

10. *Functional Test Use Case* mengelola laporan pembelian

Berikut merupakan *functional test* dari *use case* mengelola laporan pembelian:

Tabel V.31 *Functional Test Use Case* Mengelola Laporan Pembelian

No	Test Case Name	Test Case	Data Masukan	Hasil Yang Diharapkan	Hasil Yang Terjadi	Status
1	Membuat laporan pembelian	Membuat laporan berdasarkan jarak dua tanggal yaitu awal dan akhir	Memasukan tanggal awal dan tanggal akhir pada form cetak laporan	Mengenerate laporan dari database dengan format pdf	Mengenerate laporan dari database dengan format pdf	<i>Pass</i>
2	Membuat laporan pembelian	Membuat laporan dengan tanggal awal saja	Memasukan tanggal awal dan tidak memasukkan tanggal akhir pada form cetak laporan	Muncul pesan kolom harus diisi	Muncul pesan kolom harus diisi	<i>pass</i>

(Sumber: Hasil Analisis, 2018)

BAB VI

KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan pengumpulan, pengolahan, dan analisis data yang telah dilakukan pada bab-bab sebelumnya pada penelitian ini maka dapat disimpulkan beberapa poin kesimpulan sebagai berikut:

1. Aplikasi sistem informasi pembelian *spare part* mesin *stamping* yang dibuat berdasarkan proses bisnis yang berjalan dan permasalahan yang ditemukan di PT Nusa Indah Jaya Utama ini diharapkan dapat mengintegrasikan informasi antar bagian khususnya divisi *maintenance* dan divisi *purchasing*..
2. Aplikasi sistem informasi pembelian *spare part* mesin *stamping* mempunyai *database* yang terkomputerisasi dan menyediakan fungsi pengelolaan data dan penyimpanan dokumen-dokumen pembelian *spare part* mesin *stamping*. Penggunaan *database* pada aplikasi ini diharapkan dapat menunjang proses pembelian dalam hal kelola data dan informasi.
3. Sistem informasi pembelian *spare part* mesin *stamping* yang dibangun memiliki fasilitas validasi atau penyetujuan yang ditujukan kepada manajer *purchasing* secara terkomputerisasi melalui aplikasi sistem informasi pembelian *spare part* mesin *stamping* dan validasi melalui aplikasi pesan Telegram dengan fasilitas bot yang terhubung langsung ke *database* pembelian sehingga validasi dapat dilakukan dengan lebih *mobile*.

6.2 Saran

Saran yang dapat diberikan penulis untuk pengembangan dan penerapan sistem informasi pembelian *spare part* mesin *stamping* selanjutnya adalah sebagai berikut:

1. Melakukan sosialisasi untuk penggunaan aplikasi secara intensif kepada divisi-divisi terkait agar penggunaan aplikasi ini dapat berjalan dengan baik dan membantu jalannya proses pembelian *spare part* mesin *stamping*.

2. Melakukan pengembangan aplikasi dalam proses pembelian *spare part* mesin *stamping* secara lebih lanjut dengan pengembangan aplikasi untuk fungsi pembayaran kepada *supplier* dan fungsi pemilihan *supplier*, agar divisi *purchasing* dapat menggunakan aplikasi untuk proses pembelian *spare part* mesin *stamping* secara menyeluruh.
3. Melakukan pengembangan aplikasi agar dapat melakukan pengecekan stok *spare part* agar divisi *maintenance* dapat melakukan pengecekan stok *spare part* sebelum melakukan pembelian.
4. Melakukan pengembangan integrasi lebih lanjut antara aplikasi pesan Telegram dengan aplikasi sistem informasi pembelian *spare part* mesin *stamping* agar dapat melakukan pengolahan data menggunakan aplikasi pesan Telegram.

DAFTAR PUSTAKA

- Damiri, J. 2005. *Manajemen Pembelian, Penerimaan, dan Penyimpanan*. Yogyakarta: Graha Ilmu.
- Dennis, Alan., Wixom, Barbara Haley., dan Tegarden, David. 2010. *System Analysis & Design. An Object-Oriented Approach with UML*. USA: Wiley.
- Dennis, Alan., Wixom, Barbara Haley., dan Tegarden, David. 2015. *System Analysis & Design. An Object-Oriented Approach with UML (Fifth Edition)*. USA: Wiley.
- Djahir, Yulia., dan Pratita, Dewi. 2014. *Sistem Informasi Manajemen*. Yogyakarta: Deepublish.
- Fathansyah. 2012. *Basis Data*. Bandung: Informatika.
- Hakim, Lukmanul. 2010. *Membangun Web Berbasis PHP dengan Framework CodeIgniter*. Yogyakarta: Lokomedia.
- Hoffer, Jeffrey A., Prescott Mary B., Mcfadden, Fred R. 2010. *Modern Database Management*. New Jersey: Pearson Prentice Hall.
- Indrajit, Richardus Eko., dan Djokopranoto, Richardus. 2003. *Manajemen Persediaan*. Jakarta: Gramedia Widiasarana Indonesia.
- Indrajit, Richardus Eko., dan Djokopranoto, Richardus. 2005. *Strategi Manajemen Pembelian dan Supply Chain*. Jakarta: Grasindo
- Jogiyanto, H.M. 2005, *Analisa dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*. Yogyakarta: Andi.
- McLeod, Raymond. 2011. *Sistem Informasi Manajemen*. Jakarta: PT Prenhallindo.
- Muslihudin, Muhammad., dan Oktafianto. 2016. *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta: Andi
- Noor, Juliansyah. 2011. *Metodologi Penelitian: Skripsi, Tesis, Disertasi & Karya Ilmiah*. Jakarta: Kencana.

Pahlevi, Said Mirza. 2013. *Tujuh Langkah Praktis Pembangunan Basis Data*. Jakarta: Elex Media Komputindo.

PPM, Penerbit., dan Asosiasi Logistik Indonesia. 2011. *Panduan & Direktori Logistik Indonesia*. Jakarta: Penerbit PPM.

Puspitawati, Lilis., dan Anggadini, Sri Dewi. 2011. *Sistem Informasi Akuntansi*. Yogyakarta: Graha Ilmu.

Richardson, Thodor, dan Thies, Charles N. 2013 *Secure Software Design*. USA: Jones & Bartlett Learning.

Rosa, A.S, dan Shalahuddin, M. 2015. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.

Rusdiana., dan Irfan, Moch. 2014. *Sistem Informasi Manajemen*. Bandung: Pustaka Setia.

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

Simamata, Janner. 2010. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.

Sitorus, Lamhot. 2015. *Algoritma dan Pemrograman*. Yogyakarta: Andi.

Sugiyono. 2013. *Metode Penelitian Kuantitatif, Kualitatif Dan R&D*. Bandung: Alfabet

Sunyoto, Danang. 2014. *Sistem Informasi Manajemen Perspektif Organisasi*. Yogyakarta: CAPS.

Sutabri, Tata. 2004. *Analisa Sistem Informasi*. Yogyakarta: Andi.

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

Situs:

Uml-digrams. Diagram UML. <https://www.uml-diagrams.org/uml-25-diagrams.html> (Tanggal Akses: 06 Januari 2018).

Bot Telegram. Bot Telegram. <https://core.telegram.org/bots> (Tanggal Akses: 05 April 2018).

LAMPIRAN

- *Source code* Permintaan Pembelian

```
function inputminta()
{
    $id_permintaan = $this->input->post
('id_permintaan');
    $tgl_permintaan = $this->input->post
('tgl_permintaan');
    $lap_kerusakan = $this->input->post ('lap_kerusakan');
    $status = $this->input->post ('status');

    $data = array(
        'id_permintaan' => $this->input-
>post('id_permintaan'),
        'tgl_permintaan' => $this->input-
>post('tgl_permintaan'),
        'lap_kerusakan' => $this->input-
>post('lap_kerusakan'),
        'status' => $this->input->post('status')
    );
    $this->db->insert('tb_permintaan', $data);
    $this->db->query("update tb_permintaan_detail set
id_permintaan = '". $id_permintaan.'" where status =
'waiting'");
    $this->db->query("update tb_permintaan_detail set
status = 'pending' where status = 'waiting'");

    //ini bot
    $id_permintaan = $_POST['id_permintaan'];
    $lap_kerusakan = $_POST['lap_kerusakan'];

    $message = "Halo bapak Manajer %0ADIinformasikan bahwa
terdapat permintaan baru dengan id permintaan '$id_permintaan'
%0ADengan laporan kerusakan sebagai berikut: '$lap_kerusakan'
%0ASilahkan gunakan fungsi /detail $id_permintaan untuk
mengetahui detail permintaan atau fungsi /validasi507377012
$id_permintaan [valid/tolak] untuk melakukan validasi
permintaan.%0ATERimakasih";

    $api =
'https://api.telegram.org/bot571896094:AAGrYNWY7FsSwT0XFz0R4b1X
SJJe5-SXBHmw/sendMessage?chat_id=507377012&text="'. $message.'";
```

```

        $ch = curl_init($api);
        curl_setopt($ch, CURLOPT_HEADER, false);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        curl_setopt($ch, CURLOPT_POST, 1);
        curl_setopt($ch, CURLOPT_POSTFIELDS, ($params));
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        $result = curl_exec($ch);
        curl_close($ch);

        var_dump($api);
    }

```

- *Source code Validasi*

```

function setuju($id_permintaan)
{
    $this->db->query("update tb_permintaan set status =
'Valid' where id_permintaan='".$id_permintaan."'");
    $this->db->query("update tb_permintaan_detail set
status = 'Valid' where id_permintaan='".$id_permintaan."'");
}

```

- *Source code Buat PO*

```

function inputdetail()
{
    $id_permintaan = $this->input->post('id_permintaan');
    $id_sparepart = $this->input->post('id_sparepart');
    $qty = $this->input->post('qty');
    $keterangan = $this->input->post('keterangan');
    $status = $this->input->post('status');

    $data = array(
        'id_permintaan' => $this->input-
>post('id_permintaan'),
        'id_sparepart' => $this->input-
>post('id_sparepart'),
        'qty' => $this->input->post('qty'),
        'status' => $this->input->post('status'),
        'keterangan' => $this->input->post('keterangan')
    );

    $this->db->insert('tb_po_detail', $data);
    $this->db->query("update tb_permintaan_detail set
status = 'Po' where id_permintaan = '".$id_permintaan."' AND
id_sparepart='".$id_sparepart."'");
}

```

- **Source code Surat Jalan**

```
function inputminta()
{
    $id_surat = $this->input->post ('id_surat');
    $id_po = $this->input->post ('id_po');
    $tgl_surat = $this->input->post ('tgl_surat');
    $no_surat_supplier = $this->input->post
('no_surat_supplier');
    $id_supplier = $this->input->post ('id_supplier');
    $nama_pengirim = $this->input->post ('nama_pengirim');

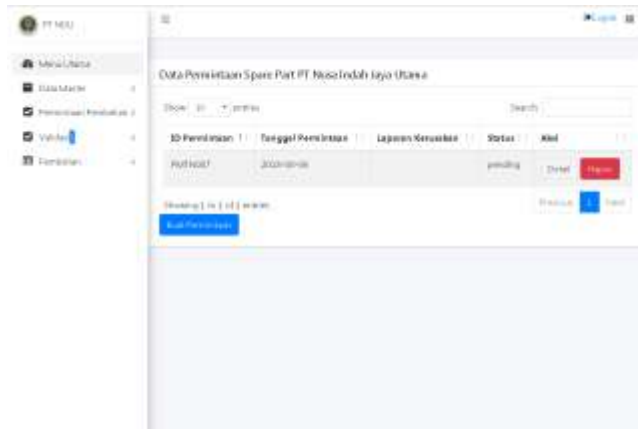
    $data = array(
        'id_surat' => $this->input->post('id_surat'),
        'id_po' => $this->input->post('id_po'),
        'tgl_surat' => $this->input->post('tgl_surat'),
        'no_surat_supplier' => $this->input-
>post('no_surat_supplier'),
        'id_supplier' => $this->input-
>post('id_supplier'),
        'nama_pengirim' => $this->input-
>post('nama_pengirim')
    );

    $this->db->insert('tb_surat', $data);
    $this->db->query ("INSERT INTO tb_surat_detail
(id_sparepart, jumlah, status)
                                SELECT id_sparepart,
qty, status
                                FROM tb_po_detail WHERE
tb_po_detail.id_po='".$id_po.'"");
    $this->db->query("update tb_surat_detail set id_surat
= '".$id_surat.'" where status = 'po'");
    $this->db->query("update tb_po set status = 'po' where
tb_po.id_po='".$id_po.'"");
    $this->db->query("update tb_surat_detail set status =
'masukpo' where status = 'po'");
}
```

- **Source code Validasi Dengan Bot**

```
else if($perintah == "/validasi507377012"){
    $kode = $pecah[1];
    $status = $pecah[2];
    $theset = mysql_query("update tb_permintaan set
status='$status' where id_permintaan='$kode'");
    $theset = mysql_query("update tb_permintaan_detail set
status = 'Valid' where id_permintaan ='".$kode."");
    $isi = "ID Permintaan = ". $kode . " telah
diubah menjadi ". $status' }
```

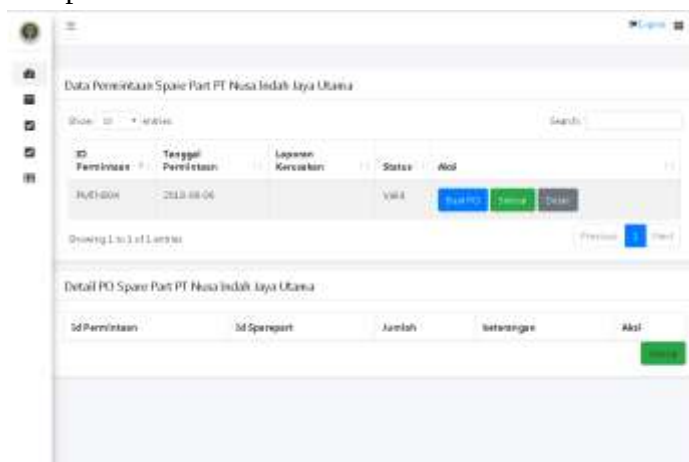
- Lampiran antarmuka data permintaan *spare part*



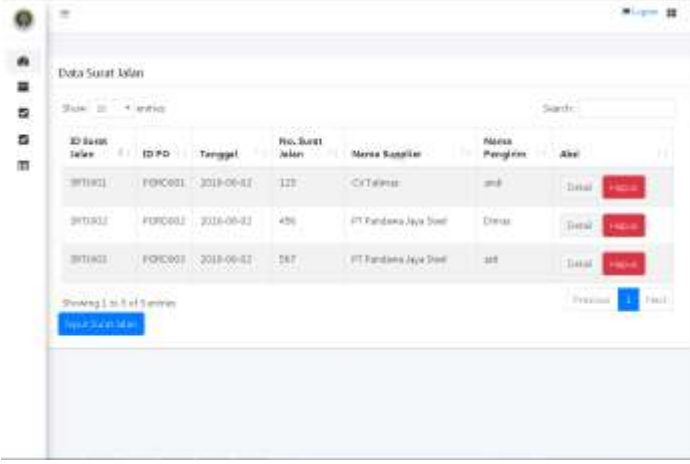
- Lampiran antarmuka validasi



- Lampiran antarmuka buat PO



- Lampiran antarmuka surat jalan



Header: Logout

Section: Data Surat Jalan

Search:

ID Surat Jalan	ID PO	Tanggal	No. Surat Jalan	Nama Supplier	Nama Pengirim	Aksi
SP11901	POHC001	2018-06-01	123	CYTAlinea	amh	Detail Hapus
SP11902	POHC002	2018-06-01	456	PT Pandawa Jaya Steel	Dhruv	Detail Hapus
SP11903	POHC003	2018-06-01	787	PT Pandawa Jaya Steel	adit	Detail Hapus

Showing 1 to 3 of 3 entries

[Previous](#) [Next](#)