

**RANCANG BANGUN SISTEM INFORMASI
MANAJEMEN BENGKEL MODUL KEPEGAWAIAN,
INVENTORY DAN ACCOUNTING BERBASIS WEB**

Diajukan guna memenuhi sebagian persyaratan dalam rangka menyelesaikan
Pendidikan Sarjana Strata Satu (S1) Program Studi Teknologi Informasi



**I Putu Adi Merta Pratama
NIM: 1705551072**

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS UDAYANA
2020**

KATA PENGANTAR

Puji syukur saya panjatkan kehadapan Tuhan Yang Maha Esa karena berkat rahmat dan anugerahnya saya dapat menyelesaikan penyusunan Skripsi yang berjudul “Sistem Informasi Manajemen Bengkel Modul Kepegawaian, Inventory dan Accounting”. Terselesaiannya penyusunan skripsi ini tentunya tidak terlepas dari bantuan dan dukungan berbagai pihak, baik secara moral maupun materiil yang tidak ternilai harganya. Untuk itu, pada kesempatan ini perkenankanlah kiranya saya menghaturkan rasa terimakasih kepada:

1. Bapak Dr. A.A. Kompiang Oka Sudana, S.Kom., MT., selaku dosen pembimbing I dan Ibuk Ni Kadek Dwi Rusjayanti, S.T., M.T., selaku dosen pembimbing II yang telah banyak memberikan masukan dan bimbingan selama penyusunan tugas akhir ini.
2. Ibuk Ni Kadek Dwi Rusjayanti, S.T., M.T., selaku dosen pembimbing akademik, yang telah memberikan bimbingan selama menempuh pendidikan di Jurusan Teknologi Informasi Fakultas Teknik Universitas Udayana.
3. Orang tua, keluarga, serta sahabat-sahabat yang tidak dapat disebutkan satu persatu yang telah banyak memberikan support dan dukungan moral kepada penulis selama pengerjaan skripsi ini
4. Pihak-pihak lain yang tidak dapat disebutkan satu persatu yang telah membantu dalam penulisan skripsi ini.

Penulis menyadari atas ketidak sempurnaan skripsi ini, oleh karena itu penulis sangat mengharapkan saran berupa kritikan-kritikan ataupun pendapat lainnya sebagai bahan pertimbangan dan koreksi kedepannya. Akhir kata penulis mengucapkan terimakasih

Denpasar, 02 November 2020

Penulis

ABSTRAK

Bengkel dengan sistem konvensional yang masih berjalan saat ini mengakibatkan lambatnya suatu proses bisnis manajemen bengkel dan proses pengolahan data menjadi kurang cepat serta kurang akurat dalam penanganannya. Sistem konvensional yang dimaksudkan dalam hal ini adalah sistem yang masih diolah secara manual dengan tenaga manusia seperti proses *input* data yang masih dilakukan secara tertulis. Penelitian ini membuat sebuah sistem informasi yang berguna untuk mempermudahkan proses bisnis dan manajemen pada suatu bengkel khususnya pada bagian kepegawaian, inventory dan accounting. Penelitian ini menggunakan metode waterfall dalam perancangan sistem informasi manajemen bengkel, dimana tahapannya dimulai dari *requirement analysis, architectural design, coding, testing, implementasi* dan *maintenance*. Fitur dari modul kepegawaian meliputi penerimaan pegawai, pendataan pegawai, pendataan riwayat pegawai, penjadwalan kerja, absensi, penggajian dan pelaporan ke pihak manajemen. Fitur dari modul *inventory* meliputi proses master data *inventory* bengkel, proses manajemen *inventory* dan pelaporan. Serta, fitur dari modul *accounting* terdiri dari master data, *account receivable, account payable, purchase requisition form*, jurnal harian dan pelaporan. Sistem Informasi ini akan terhubung ke modul-modul lain seperti *front office, point-of-sale, service* serta terhubung pada *marketplace* bengkel.

Kata Kunci: Sistem Informasi, Bengkel, Kepegawaian, *Inventory, Accounting, Web*

DAFTAR ISI

HALAMAN SAMPUL.....	1
KATA PENGANTAR.....	i
ABSTRAK.....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR	v
DAFTAR TABEL.....	vi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Manfaat Penelitian	4
1.6 Sistematika Penulisan.....	4
BAB II TINJAUAN PUSTAKA	6
2.1 State of The Art Sistem Informasi Manajemen Bengkel.....	6
2.2 Konsep Bengkel.....	10
2.3 Konsep Dasar Sistem.....	11
2.4 Konsep Sistem Informasi.....	13
2.5 Tujuan dan Manfaat Sistem Informasi.....	14
2.6 Sistem Informasi Manajemen.....	15
2.7 Sistem Informasi Manajemen Bengkel.....	16
2.8 Sistem Manajemen Kepegawaian (HRD)	19
2.9 Sistem Manajemen Inventory	20
2.9.1 Proses Bagian Gudang (Store).....	21
2.9.2 Proses Manajemen Aset.....	21
2.10 Sistem Informasi Akutansi dan Keuangan	22
2.10.1 Proses Account Payable	23
2.10.2 Proses Account Receivable	24
2.10.3 Laporan Keuangan.....	25
2.11 System Development Life Cycle (SDLC).....	25
2.12.1 Requirements Analysis	27
2.12.2 Architectural Design.....	27
2.12.3 Coding	27
2.12.4 Integrasi dan Testing.....	27
2.12.5 Training dan Implementasi.....	28
2.12.6 Operasi dan Maintenance	28
2.13 Perangkat Pemodelan Sistem Informasi.....	29
2.13.1 Diagram Konteks.....	29
2.13.2 Data Flow Diagram.....	29
2.13.3 Entity Relationship Diagram	31
2.13.4 Standard Operational Procedure	33
2.14 MySQL.....	33

2.14.1	Data Definition Language (DDL).....	34
2.14.2	Data Manipulation Language (DML).....	35
2.14.1	Tipe Data MYSQL	36
2.15	Laravel	39
2.16	Single Sign On.....	39
BAB III METODOLOGI PENELITIAN.....		41
3.1	Waktu dan Tempat Penelitian.....	41
3.2	Sumber Data.....	41
3.3	Metodologi	41
3.3.1	Requirement Analysis.....	42
3.3.2	Merancang Pemodelan Sistem.....	42
3.3.3	Implementasi Kode Program.....	42
3.3.4	Integrasi dan Testing.....	42
3.3.5	Operasi dan Maintenance	43
3.4	Alat Penelitian	43
3.4.1	Perangkat Keras (Hardware)	43
3.4.2	Perangkat Lunak (Software).....	44
3.5	Perancangan Sistem.....	44
3.5.1	Gambaran Umum	44
3.5.2	Data Flow Diagram (DFD).....	45
3.5.3	Perancangan Antar Muka.....	59
BAB IV IMPLEMENTASI DAN PENGUJIAN.....		62
4.1	Alat Pengujian	62
4.1.1	Perangkat Keras.....	62
4.1.2	Perangkat Lunak.....	62
4.2	Antarmuka Sistem.....	63
4.4.1	Antarmuka Modul Inventory & Purchasing.....	63
4.4.2	Antarmuka Modul Kepegawaian	74
4.3	Pengujian Sistem.....	77
DAFTAR PUSTAKA.....		79

DAFTAR GAMBAR

Gambar 2. 1 Blok Sistem Informasi.....	13
Gambar 2. 2 Metode Waterfall.....	26
Gambar 2. 3 Lambang Penggambaran Data Flow Diagram	30
Gambar 2. 4 Simbol ERD	31
Gambar 2. 5 Sistem Single Sign On	40
Gambar 3. 1 Gambaran Umum Sistem	45
Gambar 3. 2 Diagram Kontens Sistem.....	46
Gambar 3. 3 DFD Level 0 Sistem.....	48
Gambar 3. 4 Diagram Konteks Sub-sistem <i>Inventory</i>	50
Gambar 3. 5 DFD Level 0 Sub-Sistem <i>Inventory</i>	51
Gambar 3. 6 DFD Level 1 Store Request.....	53
Gambar 3. 7 DFD Level 1 Delivery Request.....	54
Gambar 3. 8 DFD Level 1 Stock Control.....	55
Gambar 3. 9 DFD Level 1 Purchase Request	56
Gambar 3. 10 DFD Level 1 Purchase Order.....	58
Gambar 3. 11 Antar muka dashboard <i>inventory</i>	59
Gambar 3. 12 Antar muka master data <i>inventory</i>	60
Gambar 4. 1 Antarmuka Dashboard Inventory & Purchasing.....	63
Gambar 4. 2 Antarmuka Manajemen Master Data Sparepart	64
Gambar 4. 3 Antarmuka Manajemen Master Data Merk Sparepart	65
Gambar 4. 4 Antarmuka Manajemen Master Data Jenis Sparepart.....	65
Gambar 4. 5 Antarmuka Manajemen Master Data Supplier.....	66
Gambar 4. 6 Antarmuka Manajemen Master Data Jenis Supplier.....	67
Gambar 4. 7 Antarmuka Manajemen Master Data Kontrak Kerja Supplier.....	67
Gambar 4. 8 Antarmuka Manajemen Purchasing Order.....	69
Gambar 4. 9 Antarmuka Manajemen Receiving.....	70
Gambar 4. 10 Antarmuka Manajemen Retur Pembelian	71
Gambar 4. 11 Antarmuka Manajemen Kelola Stock Sparepart	72
Gambar 4. 12 Antarmuka Manajemen Approval Purchase Order.....	73
Gambar 4. 13 Antarmuka Manajemen Approval Stock Opname.....	73
Gambar 4. 14 Antarmuka Manajemen Aset Bengkel.....	74
Gambar 4. 15 Antarmuka Dashboard Pegawai.....	75
Gambar 4. 16 Antarmuka Manajemen Master Data pegawai	76
Gambar 4. 17 Antarmuka Manajemen Master Data Unit Kerja.....	76
Gambar 4. 18 Antarmuka Manajemen Master Data Jabatan.....	77

DAFTAR TABEL

Tabel 2. 1 Contoh Perintah DDL.....	34
Tabel 2. 2 Contoh Perintah DML	35
Tabel 2. 3 Tipe Data Numerik.....	36
Tabel 2. 4 Tipe Data Tanggal dan Waktu.....	37
Tabel 2. 5 Tipe Data String	38

BAB I

PENDAHULUAN

Bab ini berisi latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan yang akan dilakukan pada penelitian ini.

1.1 Latar Belakang

Manajemen bengkel merupakan suatu proses perencanaan, pengorganisasian, kepemimpinan dan pengendalian suatu tempat dalam merawat dan memperbaiki mesin kendaraan. Manajemen bengkel yang baik harus didukung oleh administrasi yang tertib, administrasi harus mencatat semua sumber daya yang menjadi aset bengkel. Bengkel dengan sistem konvensional yang masih berjalan saat ini mengakibatkan lambatnya suatu proses bisnis manajemen bengkel dan proses pengolahan data menjadi kurang cepat serta kurang akurat dalam penanganannya. Sistem konvensional yang dimaksudkan dalam hal ini adalah sistem yang masih diolah secara manual dengan tenaga manusia seperti proses *input* data yang masih dilakukan secara tertulis. Melihat permasalahan tersebut perlu diciptakan suatu rancangan sistem informasi yang dapat memudahkan manajemen pada suatu bengkel.

Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* merupakan sistem yang dibuat untuk memudahkan proses bisnis dan manajemen pada suatu bengkel. Sistem Informasi Manajemen terintegrasi pada bengkel dapat memenuhi kebutuhan data dan informasi secara *real time* dan memudahkan pengambilan keputusan oleh manajemen di semua tingkatan. Modul Kepegawaian berguna mengelola data kepegawaian yang meliputi penerimaan, pendataan, penilaian dan penggajian atau *payroll* pegawai pada suatu bengkel. Seluruh data pegawai dapat diakses dari Modul kepegawaian tanpa perlu meng-*input*-kan kembali data pegawai yang dibutuhkan ke *database* sistem. Proses Penggajian yang membutuhkan data dan informasi dari data pegawai seperti

perhitungan tunjangan makan sesuai jumlah absensi bulanan atau perhitungan jumlah potongan berdasarkan pelanggaran kinerja pegawai dapat juga dilakukan.

Modul *inventory* dibutuhkan dalam manajemen bengkel untuk memudahkan mengelola persediaan barang khususnya *spare part* kendaraan dan mencakup pembelian barang dari *supplier* sehingga dapat memenuhi permintaan dari pelanggan semaksimal mungkin. Modul *inventory* terintegrasi dengan modul lainnya seperti modul *front office* dapat menyediakan barang yang tersedia dan melaporkan jumlah barang tersedia serta membuat laporan *inventory* yang akan diterima oleh modul *accounting*. Modul *accounting* berguna mencatat semua aspek keuangan yang timbul dari kegiatan-kegiatan yang terdapat pada suatu bengkel. Modul *accounting* mencakup *invoice*, *inventory control*, *point-of-sales*, sampai dengan laporan laba rugi. Proses pencatatan dan pelaporan kegiatan dengan aspek keuangan dari modul kepegawaian, modul *inventory* dan modul *front office* akan di proses ke dalam modul *accounting*.

Sistem Informasi Manajemen Bengkel terintegrasi dengan Sistem Informasi Bengkel Modul *Front Office* yang dapat memudahkan pelayanan terhadap pelanggan bengkel. Sistem informasi manajemen ini dibangun dengan bahasa pemrograman PHP dan *database MySQL* yang telah didukung dengan *web responsive design* dengan harapan dapat menyajikan *user experience* dan *user interface* yang menarik sehingga dapat memudahkan dalam memanajemen suatu bengkel.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, dapat dirumuskan latar belakang permasalahan yang dapat dibahas dalam penelitian ini yaitu sebagai berikut.

1. Bagaimana perancangan Modul Kepegawaian, *Inventory* dan *Accounting* pada Sistem Informasi Manajemen Bengkel.
2. Bagaimana implementasi dari perancangan Modul Kepegawaian, *Inventory* dan *Accounting* pada Sistem Informasi Manajemen Bengkel.
3. Bagaimana pengujian Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*.

1.3 Tujuan Penelitian

Tujuan penelitian mengenai perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* antara lain sebagai berikut.

1. Mengetahui perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*.
2. Mengetahui implementasi Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*.
3. Mengetahui pengujian Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*.

1.4 Batasan Masalah

Ruang lingkup dan batasan masalah dalam penelitian yang dilakukan adalah sebagai berikut:

1. Modul kepegawaian yang dibahas dalam perancangan Sistem Informasi Manajemen Bengkel ini meliputi penerimaan pegawai, pendataan pegawai, pendataan riwayat pegawai, penjadwalan kerja, absensi, penggajian dan pelaporan ke pihak manajemen.
2. Modul *inventory* yang dibahas dalam perancangan Sistem Informasi Manajemen Bengkel ini meliputi proses *master data inventory* bengkel, proses manajemen *inventory* dan pelaporan.
3. Modul *accounting* yang dibahas dalam perancangan Sistem Informasi Manajemen Bengkel ini meliputi master data, *account receivable*, *account payable*, *purchase requisition form*, jurnal harian dan pelaporan.
4. Sistem informasi ini dirancang agar dapat terintegrasi setiap modulnya serta dengan modul *front office*, *service*, *point of sale* dan *marketplace* bengkel.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dari penelitian mengenai perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* antara lain sebagai berikut:

1. Bahan masukan dan studi kepustakaan bagi peneliti lain yang akan melakukan penelitian yang sama dengan bidang yang dibahas dalam penelitian ini.
2. Pengembangan ilmu yang diperoleh penulis pada Jurusan Teknologi Informasi Fakultas Teknik Universitas Udayana khususnya dalam bidang perancangan sistem informasi.
3. Terciptanya Sistem Informasi Manajemen Bengkel yang dapat membantu proses bisnis yang saat ini masih dilakukan dengan cara konvensional di suatu bengkel.

1.6 Sistematika Penulisan

Penulisan laporan ini terbagi dalam lima bab agar lebih terstruktur serta mudah untuk dipahami. Tiap bab dapat diuraikan sebagai berikut:

BAB I : PENDAHULUAN

Bab ini terdiri dari latar belakang penulisan, perumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan

BAB II : TINJAUAN PUSTAKA

Teori-teori pendukung pembahasan dari permasalahan yang ada diuraikan dalam bab ini. Teori pendukung yang disertakan antara lain berhubungan dengan sistem informasi, kepegawaian, *inventory* dan *accounting*.

BAB III: METODE PERANCANGAN SISTEM

Bab ini berisi metode dan perancangan sistem, meliputi tempat dan waktu penelitian, sumber data, jenis data, metode pengumpulan data, metode pengolahan data, analisis data, dan implementasi.

BAB IV: HASIL DAN PEMBAHASAN

Bab ini berisi pembahasan dan rancangan Sistem Informasi Modul Kepegawaian, *Inventory* dan *Accounting* yang telah dibuat.

BAB V : PENUTUP

Bab ini berisi simpulan yang dapat diambil dari hasil penelitian dipaparkan dan saran dari penulis.

BAB II

TINJAUAN PUSTAKA

Bab II berisi tinjauan pustaka mengenai bahasan tentang teori dasar yang berkaitan dengan Sistem Informasi Manajemen Kepegawaian, *Inventory* dan *Accounting*.

2.1 State of The Art Sistem Informasi Manajemen Bengkel

Penelitian terdahulu mengenai sistem informasi bengkel diantaranya yaitu dari Fakultasi Teknik dan Ilmu Komputer, Universitas Teknorat Indonesia dalam penelitian berjudul “Perancangan Sistem Informasi Manajemen Bengkel Berbasis *Web* (Studi Kasus: Bengkel Anugrah)” yang membantu suatu bengkel studi kasus Bengkel Anugrah yang bergerak dalam bidang jasa service kendaraan dan penjualan sparepart kendaraan roda empat. Setiap transaksi masih dilakukan secara manual yaitu masih dilakukan dengan ditulis menggunakan buku penjualan, dan belum terdapat laporan penjualan per periode. Sering terjadi kehabisan stok barang dikarnakan tidak adanya informasi jika stok barang habis. Dan pelanggan harus datang langsung untuk melakukan service dan mendapatkan informasi barang yang dijual apakah barang masih tersedia atau tidak. Dari permasalahan tersebut peneliti membantu membuat suatu sistem informasi servis motor pada bengkel dengan tujuan membantu proses bisnis bengkel yang memiliki masalah terhadap manajemen yang kurang efektif. Tahapan penelitian ini yaitu Tahap *Communication* (berkomunikasi dengan user dengan melakukan wawancara), Tahap *Planning* (analisis kebutuhan sistem dan studi kelayakan sistem), Tahap *Modelling* (merancang sistem menggunakan *Unified Modelling Languange*, membuat sistem menggunakan aplikasi dreamwever dan MySQL), Tahap *Construction* (Melakukan pengkodean sistem), Tahap *Delivery & Feedback* atau melakukan penyebaran kuisioner kepada responden apakah sistem layak diterapkan atau tidak. (Meri Audrilia, Arief Budiman, 2020).

Rancang Bangun Sistem Informasi Inventory Barang Berbasis Web Studi Kasus di PT. Infinetworks Global Jakarta merancang bangun sistem *inventory*

untuk mempermudah dan mempercepat kinerja perusahaan dalam proses transaksi usaha sehari-hari utamanya dalam PT Infinetworks Global jakarta yang bergereak dalam bidang teknologi informasi. Dalam penelitian tersebut menggunakan metode perancangan sistem *dokumen activity diagram, use case diagram* dan *class diagram* dimana perancangan sistem tersebut sudah lengkap dan jelas dijelaskan. Hasil penelitian berupa sistem informasi yang memiliki beberapa fitur seperti halaman *login admin*, halaman utama sistem *inventory*, laporan data pemakaian barang dan laporan pengembalian barang. Dengan rancang bangun sistem informasi tersebut pekerjaan dalam mencatat dan mengolah data seluruh barang dapat dilakukan semakin mudah dan resiko terjadinya kehilangan data semakin kecil. (Agus Heryanto, 2014).

Sistem Informasi Inventori Gudang Untuk Mengontrol Persediaan Barang Pada Gudang Studi Kasus: PT. Alaisys Sidoarjo dengan latar permasalahan dari PT Alaisys Sidoarjo pada pengolahan data gudang yang masih dilakukan dengan manual seperti pencatatan informasi penjualan dan persediaan barang dengan menggunakan bon nota buku. Penelitian ini membantu dalam perancangan sistem informasi *inventory* khususnya pada studi kasus PT Alaisys Sidoarjo yang bergerak di bidang usaha perdangangan, perindustrian dan jasa. Aplikasi ini dibangun dengan bahasa pemrograman PHP dan database MySQL yang telah didukung *web responsive design*. Tahapan dari perancangan sistem informasi ini mencakup *flowchart, Contex Diagram, DFD Level 0, DFD Level 1, DFD Level 2, Implementasi dan Testing* dengan melakukan analisa kebutuhan sistem dari observasi, wawancara, kuisioner dan identifikasi dokumen. Hasil dari penelitian ini yakni penambahan stok gudang yang otomatis dan praktis, akurasi data stok barang gudang yang masing kurang, informasi stok barang yang diperoleh pegawai yang tepat dan relevan. Dalam Pengujian sistem menggunakan ISO 25010 dengan aspek *functionality, efficiency, dan realibility*. (Hendra Agusvianto, 2017).

Sistem Informasi Kepegawaian Pada Kantor Dewan Teknologi Informasi dan Komunikasi Provinsi Papua Berbasis *Web* penelitian ini bertujuan dalam pengembangan sistem informasi yang dapat membantu mengelola data kepegawaian yang masih dilakukan secara manual yakni sekretaris DETIK Provinsi

Papua mengelola data pegawai, membuat arsip data diri pegawai. Pada penelitian ini merancang aplikasi berbasis *web* dengan menggunakan bahasa pemrograman PHP dan MySQL dalam pengelelolaan *database* dan didukung oleh *web responsive design*. Tahapan dari penelitian ini yakni pengambilan data berupa observasi, interview pengumpulan data yang dilakukan dengan sesi diskusi dengan sekretaris DETIK Provinsi Papua dan pengumpulan data dengan studi pustaka, tahapan selanjutnya yakni analisis sistem berjalan dan perancangan sistem. Metode perancangan yang digunakan yakni perancangan *diagram konteks* dan *Entity Relationship Diagram* yang sudah dijelaskan dengan lengkap. Adapun hasil dari penelitian ini yakni aplikasi informasi pegawai yang terdiri dari menu utama, informasi pegawai, *login admin*, tambah pegawai, tabel pegawai, cetak pegawai. Sistem informasi yang dirancang ini menyediakan informasi data pegawai dan honorer yang mudah digunakan dalam pengelolaan data pegawai dan honorer, berupa penambahan data, pengubahan data, penghapusan data, pencarian data dan laporan. (Mursalim Tonggiroh, Nur Immamul Hakim, 2017).

Restfull Web Service untuk Integrasi Sistem Akademik dan Perpustakaan Universitas Perjuangan penelitian ini bertujuan untuk melakukan integrasi antara sistem yang sudah ada dengan latar masalah terjadinya duplikasi data contohnya pada peminjaman buku yang dilakukan pada sistem informasi perpustakaan yang belum terintegrasi dengan sistem akademik pusat. Penggunaan *Web service* yang mampu mengatasi permasalahan *interoperability* dan mengintegrasikan sistem yang berbeda, dalam penelitian ini menggunakan *web service* yang memiliki kelebihan yaitu dapat melakukan pertukaran data lintas *platform* dan memiliki bahasa independent dengan arsitektur *web service* yakni *simple object access protocol* (SOAP) dan *Representational State Transfer* (REST). Metodologi yang dilakukan dalam penelitian ini mencakup observasi pengumpulan data dengan perolehan sistem informasi akademik menggunakan bahasa pemrograman PHP dengan *framework laravel* dan *database* menggunakan MySQL dengan *web server Apache*, selanjutnya analisis kebutuhan sistem, desain dan coding dengan penjelasan arsitektur sistem. Hasil yang didapat dari penelitian ini berupa implementasi dan pengujian *REST Services* pada sistem dan penerapan teknologi

web service dengan menggunakan arsitektur *REST* pada sistem informasi akademik dan sistem informasi perpustakaan mampu mengintegrasikan kedua sistem tersebut. Sehingga proses *input* dan verifikasi data hanya dilakukan satu kali, hal tersebut mengatasi terjadinya duplikasi data dan mengurangi perkerjaan *input* data. (Randi Rizal, Alam Rahmatulloh, 2019).

Pembangunan Sistem Informasi Pengolahan Data Pegawai dan Penggajian Pada Unit Pelaksana Teknis Taman Kanak Kanak dan Sekolah Dasar Kecamatan Pringku menjelaskan bahwa pada penelitian ini diperlukannya sistem informasi yang dapat mengolah data pegawai dan penggajian secara efektif dan efisien. Pada sistem informasi ini dibangun dengan bahasa pemrograman PHP dan *database* MySQL dan didukung oleh desain *web* yang *responsive*. Metodologi dalam penelitian ini yakni analisis sistem, *Data flow map* yang sedang berjalan, kerangka pemikiran, spesifikasi kebutuhan, *Data flow diagram*, *Entity Relation Diagram* dan persiapan *database*, ketika tahapan tersebut sudah rampung penelitian dilanjutkan ke dalam implementasi sistem. Hasil dari penelitian ini yakni sistem informasi pengolahan data pegawai dan penggajian, dalam pembuatan laporan baik laporan data sekolah dan laporan data pegawai maupun laporan data gaji pegawai menjadi lebih mudah dan akurat. (Titin Purnamasari, 2013).

Rancang Bangun Sistem Informasi Akuntansi Laporan Laba Rugi Berbasis *Web* Pada PT. United Tractors Pontianak yang bergerak dalam bidang penjualan alat-alat berat permasalahan yang terjadi sistem yang berjalan pada perusahaan tersebut masih menggunakan tulis tangan atau dilakukan secara manual, oleh sebab itu penelitian ini berguna dalam merancang sistem informasi yang dapat membantu melakukan pengolahan data penjualan lebih mudah, cepat dan akurat. Sistem informasi pada penelitian ini menggunakan bahasa pemrograman PHP dengan *framework* laravel dan *database* menggunakan MySQL dan didukung oleh desain *web responsive* dengan penggunaan *framework* bootstrap. Perancangan sistem berupa *logical record structure*, normalisasi, relasi, dan spesifikasi basis data. Metode penelitian menggunakan *Unified Modeling Language* (UML) yang terdiri dari *use case diagram* dan *activity diagram*, *sequence diagram* serta *deployment diagram*. Perancangan yang dilakukan dalam penelitian ini dijelaskan

dengan lengkap dan mudah dipahami. Adapun hasil dari penelitian ini berupa sistem informasi atau aplikasi laba rugi yang dibuat dua pengguna yang berhak mengakses aplikasi diantaranya *admin* dan Kepala operasi cabang, Pembuatan sistem informasi akuntansi ini mendukung proses pengolahan keuangan yang diharapkan dapat membantu memberikan kemudahan khususnya untuk bagian proses laporan laba rugi dengan baik dan akurat. (Nurmalasari, Anna, Riska Arissusandi, 2019).

2.2 Konsep Bengkel

Menurut (Jogiyanto, 2008,15) Bengkel motor adalah tempat yang melakukan perbaikan maupun perubahan motor agar dapat kembali berjalan dengan baik sesuai dengan keinginan pemilik atau bentuk asli dari sepeda motor tersebut. Sepeda motor yang diperbaiki dapat menggunakan bahan (spare parts) baru atau bahan yang ada dengan melakukan penyesuaian agar sepeda motor dapat berjalan dengan baik.

Menurut (Iqbal, 2004) pengklasifikasian bengkel secara umum dapat dibagi diantaranya sebagai berikut: Jenis bengkel dibagi menjadi 2:

- 1) Bengkel Umum Bengkel Umum adalah bengkel umum kendaraan bermotor yang berfungsi untuk membetulkan, memperbaiki, merubah dan merawat kendaraan bermotor agar tetap memenuhi persyaratan teknis dan layak jalan.
- 2) Bengkel Resmi Bengkel Resmi yaitu bengkel Dealer yang hanya melayani perawatan (service) untuk merek motor tertentu sesuai dengan rekomendasi dari pembuat kendaraan bermotor (pabrikan).

Pada bengkel umum kita bisa menemui bengkel yang juga sekaligus dapat memodifikasi kendaraan bermotor yang tidak hanya khusus untuk memperbaiki motor saja. Salah satu bengkel umum yang juga dapat memodifikasi motor secara khusus sesuai kapasitas dan perlengkapan dari bengkel ini sendiri yaitu bengkel Zome. Bengkel Zome adalah bengkel umum yang dapat menerima service motor

sekaligus juga dapat meminta untuk memodifikasi motor kita sesuai keinginan dan sesuai kapasitas dari bengkel Zome.

2.3 Konsep Dasar Sistem

(Jogyanto, 2005) mengemukakan bahwa terdapat dua kelompok pendekatan di dalam mendefinisikan sistem, yaitu menekankan pada prosedur dan menekankan pada komponen atau elemennya. Definisi sistem untuk penekanan pada prosedur, maka sistem dapat diartikan sebagai suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. Definisi sistem yang menekankan pada komponen yaitu sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu.

Kedua pendekatan yang paling mudah dilakukan adalah pendekatan sistem yang menekankan pada elemen atau komponennya karena pendekatan ini akan mempermudah dalam mempelajari suatu sistem untuk tujuan analisis dan perancangan suatu sistem dibandingkan dengan pendekatan sistem yang menekankan pada prosedur.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yang tidak dapat dipisahkan antara yang satu dengan yang lainnya. Karakteristik sistem menurut (H.M. JOGIYANTO, 2001) dalam buku yang berjudul Pengenalan Komputer, yaitu:

- 1) Kemampuan Komponen Sistem (*Components*)

Suatu sistem terdiri dari beberapa atau sejumlah komponen yang saling berhubungan atau berinteraksi antara komponen yang satu dengan komponen yang lainnya atau bekerja sama untuk membentuk satu kesatuan. Dalam komponen sistem ini atau elemen-elemen sistem ini dapat berupa suatu subsistem yang lebih kecil.

- 2) Batasan Sistem (*Boundary*)

Batasan sistem merupakan garis besar abstraksi yang memisahkan antara sistem dan lingkungannya. Batas sistem ini juga menunjukkan ruang lingkup (scope) dari sistem tersebut.

3) Lingkungan Luar Sistem (*Environtment*)

Lingkungan luar dari suatu sistem adalah apa saja yang ada di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4) Penghubung Sistem (*Interface*)

Penghubung sistem merupakan suatu media antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumbersumber daya mengalir dari satu subsistem ke subsistem yang lainnya.

5) Masukan Sistem (*Input*)

Masukan atau *input* merupakan energi yang dimasukkan kedalam sistem. Masukan sistem ini bervariasi bisa berupa energi manusia, data model, bahan baku, layanan atau lainnya.

6) Keluaran sistem (*Output*)

Keluaran sistem merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada suara sistem. Definisi lain adalah keluaran sistem (*output*) merupakan hasil dari proses yang merupakan tujuan dari keberadaan sistem.

7) Pengolahan Sistem (*Process*)

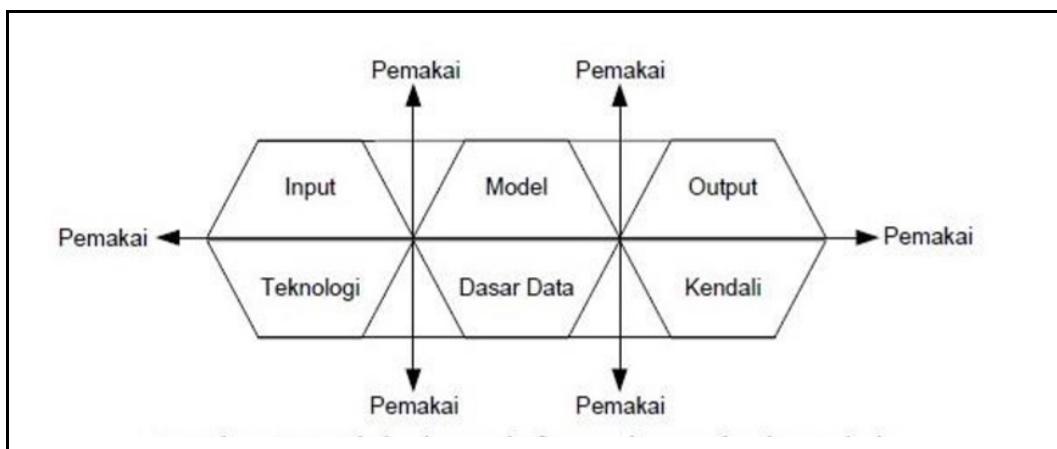
Pengolah atau proses merupakan perubahan dari masukan (*input*) menjadi keluaran (*output*). Proses ini mungkin dilakukan oleh mesin, orang (*user*) atau komputer (*computer*).

8) Sasaran Sistem (*Objective*)

Sasaran dari suatu sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil jika mengenai sasaran atau tujuannya. Karena tujuan ini merupakan target atau sasaran akhir yang akan dicapai oleh suatu sistem.

2.4 Konsep Sistem Informasi

Menurut Robert A. Leitch dan K. Roscoe Davis (1983) sistem informasi merupakan suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat menajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan untuk proses pengambilan keputusan. Jadi dapat ditarik kesimpulan bahwa sistem informasi adalah suatu sistem yang terintegrasi dengan berbagai elemen pendukungnya untuk menyediakan suatu informasi dari data-data yang ada bagi penggunanya. Sistem informasi memiliki beberapa komponen-komponen yang disebut dengan istilah blok bangunan (*building block*) yaitu sebagai berikut.



Gambar 2.1 Blok Sistem Informasi

(Sumber: Jogyanto HM.2005.Analisis dan Desain Sistem Informasi. Andi.Yogyakarta)

Gambar 2.1 merupakan blok sistem informasi. Komponen *Input* mewakili data yang masuk kedalam sistem informasi. *Input* disini termasuk metode dan media untuk menangkap data yang akan dimasukkan yang dapat berupa dokumen-dokumen dasar. Komponen Model 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.

Komponen *Output* yaitu hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem. Komponen Teknologi merupakan “*toolbox*” 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.

Komponen Basis Data merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*). Komponen Kendali yaitu terdapat beberapa hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidakefisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.5 Tujuan dan Manfaat Sistem Informasi

Sistem informasi memiliki beberapa tujuan dan manfaat untuk perusahaan atau organisasi. Tujuan dari sistem informasi memiliki sedikit perbedaan dengan manfaat sistem informasi itu sendiri yang dijabarkan sebagai berikut.

- 1) Menyediakan informasi yang dipergunakan didalam perhitungan harga pokok jasa, produk, dan tujuan lain yang diinginkan manajemen.
- 2) Menyediakan informasi yang dipergunakan dalam perencanaan, pengendalian, pengevaluasian, dan perbaikan berkelanjutan.
- 3) Menyediakan informasi untuk pengambilan keputusan.

Ketiga tujuan tersebut menunjukkan bahwa manajer dan pengguna lainnya perlu memiliki akses ke informasi akuntansi manajemen dan mengetahui bagaimana cara menggunakannya. Informasi akuntansi manajemen dapat

membantu mereka mengidentifikasi suatu masalah, menyelesaikan masalah. Sistem informasi memiliki manfaat diberbagai tingkatan organisasi dan perusahaan, diantaranya adalah sebagai berikut.

- 1) Mengolah transaksi, mengurangi biaya, dan menghasilkan pendapatan.
- 2) Mengolah berbagai laporan penting dan transaksi
- 3) Mempertahankan persediaan pada tingkat paling rendah agar konsisten dalam menjaga produk mereka.

2.6 Sistem Informasi Manajemen

Sistem informasi adalah sistem buatan manusia yang terdiri dari seperangkat komponen yang saling berinteraksi dalam suatu organisasi guna menyajikan informasi. Robert A. Leitch dan K. Roscoe Davis (1983) mendefinisikan sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Manajemen adalah seni dari usaha pengaturan sumber daya manusia untuk mencapai tujuan organisasi atau perusahaan. Manajemen erat kaitannya dengan pimpinan, yaitu orang yang mengorganisir sumber daya manusia dan sumber daya lainnya dalam perusahaan untuk bekerja secara sistematis demi tercapainya suatu tujuan. Murdick (1993) mendefinisikan manajemen adalah terdiri dari proses atau kegiatan yang menjelaskan apa yang dilakukan menggerakkan pada operasi organisasi mereka, yaitu merencanakan, mengorganisasikan, memprakarsai dan mengendalikan operasi.

Para ahli mendefinisikan sistem informasi manajemen dalam berbagai cara, sebagai berikut:

- 1) (Jogyanto, 2005) Sistem informasi manajemen (*management information system* atau yang sering dikenal dengan singkatannya MIS) merupakan penerapan sistem informasi di dalam organisasi untuk mendukung informasi-informasi yang dibutuhkan oleh semua tingkatan manajemen.

- 2) Scott (1986), suatu SIM adalah kumpulan dari interaksi-interaksi sistem-sistem informasi yang menyediakan informasi baik untuk kebutuhan manajerial maupun kebutuhan operasi.
- 3) Davis (1974), SIM adalah sistem manusia/mesin yang menyediakan informasi untuk mendukung operasi manajemen dan fungsi pengambilan keputusan dari suatu organisasi.

Definisi dan pengertian sistem informasi manajemen menurut para ahli disimpulkan sebagai seperangkat sistem informasi yang saling berhubungan dalam menghasilkan dan menyampaikan informasi yang berguna bagi seluruh tingkatan manajemen dalam pengambilan keputusan. Hasil keputusan yang diambil dan ditetapkan pihak manajemen nantinya akan dilaksanakan oleh organisasi atau perusahaan dalam bekerja.

2.7 Sistem Informasi Manajemen Bengkel

Sistem Infromasi Manajemen Bengkel adalah suatu sistem informasi yang dibuat untuk mengendalikan bisnis secara internal pada suatu Bengkel. Masih terdapat banyak bengkel konvensional yang melakukan proses bisnisnya secara manual. Melihat permasalahan tersebut perlu diciptakan suatu rancangan sistem informasi yang dapat dikembangkan untuk mendapatkan informasi yang cepat dan saling terintegrasi yang akan menunjang proses pengambilan keputusan pada suatu bengkel.

Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* merupakan sistem yang dibuat untuk memudahkan proses bisnis dan manajemen pada suatu bengkel. Sistem Informasi Manajemen terintegrasi pada bengkel dapat memenuhi kebutuhan data dan informasi secara *real time* dan memudahkan pengambilan keputusan oleh manajemen di semua tingkatan. Sistem Informasi Manajemen Bengkel terintegrasi dengan Sistem Informasi Bengkel Modul *Front Office* yang dapat memudahkan pelayanan terhadap pelanggan bengkel.

Modul Sistem Informasi Manajemen Bengkel terdiri dari modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan umum masing-masing modul.

1) Modul Kepegawaian

Modul Kepegawaian berguna mengelola data kepegawaian yang meliputi penerimaan, pendataan, penilaian dan penggajian atau *payroll* pegawai pada suatu bengkel. Seluruh data pegawai dapat diakses dari Modul Kepegawaian tanpa perlu meng-*input*-kan kembali data pegawai yang dibutuhkan ke database sistem. Proses Penggajian yang membutuhkan data dan informasi dari data pegawai seperti perhitungan tunjangan makan sesuai jumlah absensi bulanan atau perhitungan jumlah potongan berdasarkan pelanggaran kinerja pegawai dapat juga dilakukan.

2) Modul *Inventory*

Modul *Inventory* dibutuhkan dalam manajemen bengkel untuk memudahkan mengelola persediaan barang khususnya spare part kendaraan dan mencakup pembelian barang dari *supplier* sehingga dapat memenuhi permintaan dari pelanggan semaksimal mungkin. Modul *inventory* terintegrasi dengan modul lainnya seperti modul *front office* dapat menyediakan barang yang tersedia dan melaporkan jumlah barang tersedia dan membuat laporan *inventory* yang akan diterima oleh Modul Akutansi.

3) Modul *Accounting*

Modul *Accounting* berguna mencatat semua aspek keuangan yang timbul dari kegiatan-kegiatan yang terdapat pada suatu bengkel. Modul *Accounting* mencakup *invoice*, *inventory control*, *point-of-sales*, sampai laporan-laporan seperti neraca, laba rugi dan buku besar. Proses pencatatan dan pelaporan kegiatan dengan aspek keuangan dari Modul Kepegawaian, Modul *Inventory* dan Modul *Front Office* akan di proses ke dalam Modul *Accounting*.

Penerapan Sistem Informasi Manajemen Bengkel yang terkompeterisasi dan telah didukung dengan sumber daya manusia yang baik akan sangat membantu dalam peningkatan pelayanan dan beberapa manfaat lain sebagai berikut:

1) Efisiensi

Sistem Informasi Manajemen Bengkel membantu dalam proses pengolahan data yang menjadi lebih cepat dan akurasi data yang meningkat. Waktu yang dibutukan untuk melakukan pekerjaan administrasi akan berkurang jauh sehingga karyawan dapat lebih fokus pada pekerjaan utamanya.

2) Kemudahan

Manfaat yang paling terasa ketika Sistem Informasi Manajemen Bengkel diterapkan adalah memudahkan pekerjaan administrasi. Sistem yang masih berjalan manual dalam pengerajan laporannya memakan waktu hingga 1 bulan, sedangkan dengan Sistem Informasi Manajemen Bengkel hanya memakan waktu satu sampai dua hari saja untuk membuat laporan. User hanya perlu menekan tombol print out dan memakan tempat yang tidak luas untuk menyimpan dokumen. Manajemen Kepegawaian yang mudah diakses dan master data yang mudah di pantau serta sistem inventory yang memudahkan dalam melakukan manajemen inventory atau barang di suatu bengkel.

3) Dokumentasi yang *Auditable* dan *Accountable*

Sistem Informasi Manajemen Bengkel sebaiknya dirancang menganut kebijakan data terpusat, artinya data yang digunakan oleh seluruh rumah sakit berada di bawah satu kendali.

4) Mendukung koordinasi antar bagian

Data yang digunakan di setiap modul akan terintegrasi dan memudahkan koordinasi antar bagian modul yang ada. Sistem Informasi Manajemen Bengkel juga dapat melakukan koordinasi dengan bagian Modul *Front Office*.

5) Membantu dalam pengambilan keputusan

Penggunaan sistem Konvensional di suatu bengkel seringkali membuat pengambilan keputusan yang tidak tepat. Hal tersebut dapat terjadi karena

pihak manajer mengambil keputusan berdasarkan informasi yang sudah tidak relevan. Sistem Informasi Manajemen Bengkel dapat mengatasi permasalahan ini dan membantu dalam pengambilan keputusan dengan menyediakan informasi yang relevan dan bersifat real time.

2.8 Sistem Manajemen Kepegawaian (HRD)

Hariandja (2007), tujuan Manajemen Sumber Daya Manusia adalah meningkatkan dukungan sumber daya manusia dalam usaha meningkatkan efektivitas organisasi dalam rangka mencapai tujuan. Secara lebih operasional (dalam arti yang dapat diamati/ diukur) Tujuan Manajemen Sumber Daya Manusia adalah untuk meningkatkan produktivitas karyawan, mengurangi tingkat absensi, mengurangi tingkat perputaran kerja, atau meningkatkan loyalitas para karyawan pada organisasi.

Kegiatan atau aktivitas MSDM secara umum dapat dikategorikan menjadi empat, yaitu:

- 1) Persiapan dan pengadaan
- 2) Pengembangan dan penilaian
- 3) Pengkompensasian dan perlindungan
- 4) Hubungan-hubungan karyawan.

Gary Dessler (2003), mendefinisikan Manajemen Sumber Daya Manusia (MSDM) adalah kebijakan dan praktik menentukan aspek: “manusia” atau sumber daya manusia dalam posisi manajemen, termasuk merekrut, menyaring, melatih, memberi penghargaan dan penilaian.

Perkembangan teknologi informasi pada masa sekarang ini memungkinkan fungsifungsi organisasi dapat bekerja dengan lebih efisien dan efektif dan menghasilkan *output* berupa informasi yang akurat, relevan, dan tepat waktu yang sangat dibutuhkan oleh manajemen perusahaan dalam mengambil putusan perusahaan. Penggabungan Sistem Informasi dengan Manajemen Sumber Daya Manusia (MSDM) akan menjadi sebuah perangkat kerja organisasi yang disebut sebagai Sistem Informasi Sumber Daya Manusia (SISDM) atau *Human*

Resources Information System (HRIS), yang bekerja menghasilkan informasi Sumber Daya Manusia (SDM) di perusahaan, membantu Departemen Sumber Daya Manusia atau HRD (Human Resources Department) dalam melaksanakan tugasnya dan membantu manajemen perusahaan dalam mengambil putusan atau kebijakan perusahaan dalam mencapai tujuan perusahaan.

Raymond Mc.Leod (2001), sistem informasi sumber daya manusia adalah suatu sistem untuk mengumpulkan dan memelihara data yang menjelaskan sumber daya manusia, mengubah data tersebut menjadi informasi dan melaporkan informasi tersebut kepada pemakai. Model Sistem Informasi Sumber Daya Manusia (SISDM) terdiri sub-sistem *input*, database dan subsistem *output*. Sub sistem *input* merupakan kombinasi standar dari pengolahan data, penelitian dan intelijen. Database merupakan tempat simpanan data atau informasi mengenai sumber daya manusia dan lingkungan organisasi perusahaan. Sub-sistem *output* merupakan arus data atau informasi sumber daya manusia yang dihasilkan dari Sistem Informasi Sumber Daya Manusia (SISDM).

2.9 Sistem Manajem Inventory

Inventory meliputi semua barang yang dimiliki perusahaan pada saat tertentu, dengan dijual kembali atau diasumsikan dalam siklus operasi normal perusahaan sebagai barang yang dimiliki untuk dijual atau diasumsikan dimasa yang akan datang. Semua barang yang berwujud dapat disebut *inventory*, tergantung dari sifat dan jenis usaha perusahaan (Raharjo, Budi, 2011).

Inventory perlu dilaksanakan dengan baik untuk mengetahui secara pasti persediaan dari sisa barang-barang dagangan yang terjual, dan untuk menjamin lancarnya arus lintas barang maka perlu diadakan pencatatan terhadap segala penerimaan barang yang berasal dari supplier, barang yang dipesan oleh pelanggan, barang yang terjual, dan penyesuaian penyesuaian (*adjustment*) terhadap barang. Pencatatan data tersebut agar dapat diketahui dengan mudah barang yang banyak tertimbun (*over stock*), barang yang harus dipesan kembali kepada supplier karena persediannya sudah minim.

2.9.1 Proses Bagian Gudang (Store)

Transaksi Pemesanan Barang ke Gudang atau yang disebut dengan *Store Request* (SR) dibuat oleh Unit Pengguna. Data transaksi pemesanan barang tersebut kemudian diperiksa oleh Kepala Bagian Gudang, bila telah sesuai maka Kepala Gudang akan memberikan persetujuan (*approval*) terhadap transaksi pemesanan barang tersebut. *Store Request* (SR) yang telah disetujui kemudian diberikan kepada *Staff* Gudang sehingga dapat melakukan transaksi *delivery* atau transaksi pengeluaran barang dari gudang. *Staff* Gudang kemudian melihat barang di gudang, bila jumlah barang yang dipesan tidak mencukupi atau tidak ada stok maka akan dilakukan transaksi permintaan barang ke Bagian Pembelian atau *Purchase Request* (PR).

Staff Gudang dapat melakukan Transaksi *Retur* atau pengembalian barang ke *Supplier* jika terjadi masalah pada kondisi barang yang baru datang dari Bagian Penerimaan. *Staff* Gudang kemudian memeriksa barang, jika ditemukan barang rusak maka akan dilakukan Transaksi *Spoil*. *Stock Opname* dilakukan pada akhir bulan, tujuannya adalah menyesuaikan atau menyamakan stok barang pada sistem dengan stok fisik barang (Eka:2007).

2.9.2 Proses Manajemen Aset

Proses Manajemen Aset dilakukan dengan melakukan analisis terhadap data aset yang ada, kemudian melakukan inspeksi atas penugasan yang yang telah diberikan serta melakukan monitoring dan evaluasi tentang sumber daya dalam penggunaan aset. Proses Manajemen Aset dibagi menjadi beberapa subproses yaitu Proses Perencanaan Penyediaan Aset, Proses Pengadaan atau Penyediaan Aset dan Proses Perawatan Aset Fisik.

2.9.2.1 Proses Perencanaan Aset

Proses Perencanaan Aset adalah proses yang membahas mengenai proses dan langkah-langkah dalam melakukan permintaan aset ke Departemen *Inventory*. Proses Perencanaan Aset biasanya dilakukan oleh setiap Unit Departemen dengan melakukan pengecekan kebutuhan aset.

Setiap tahunnya dilakukan Rapat Anggaran Tahunan (RAT) yang membahas mengenai perencanaan anggaran yang akan dikeluarkan setiap tahunnya. Perencanaan aset kemudian akan dibahas di dan akan ditentukan apakah perencanaan tersebut disetujui atau di tunda ke tahun anggaran selanjutnya.

2.9.2.2 Proses Pengadaan Aset

Proses Pengadaan Aset adalah tindak lanjut dari Proses Perencanaan Aset jika telah disetujui kemudian akan diproses oleh Unit *Inventory*. Unit *Inventory* kemudian membuat Draft Penyediaan Aset yang dikirim ke Bagian AP untuk mendapat persetujuan pembelian aset dari Vendor.

Invoice pembelian aset ke *Supplies* yang telah disetujui kemudian dikirim ke Vendor yang sudah menjadi rekanan dari pihak bengkel. Pihak Vendor memberikan konfirmasi jika sudah mengirim aset yang telah diminta.

2.9.2.3 Proses Perawatan dan Pemeliharaan Aset

Proses Perawatan atau Pemeliharaan Aset adalah proses perawatan yang dilakukan terhadap aset fisik yang dimiliki suatu bengkel. Proses Perawatan dilakukan berkala misalnya dilakukan setiap 3 bulan sekali atau 6 bulan sekali.

Perawatan aset diperlukan untuk menjaga kondisi aset yang dimiliki suatu bengkel agar dapat digunakan dengan layak. Perawatan aset fisik di suatu bengkel seperti Perawatan alat-alat *service* sangat dibutuhkan untuk menjaga kondisi alat pada saat akan digunakan.

2.10 Sistem Informasi Akuntansi dan Keuangan

Sistem Manajemen Akuntansi adalah mendorong seoptimal mungkin agar sistem tersebut dapat menghasilkan berbagai informasi akuntansi yang terstruktur yaitu tepat waktu, relevan, dan dapat dipercaya. Unsur-unsur yang terdapat dalam suatu sistem akuntansi saling berkaitan satu sama lain, sehingga dapat dilakukan pengolahan data mulai dari awal transaksi sampai dengan pelaporan yang dapat dijadikan sebagai informasi akuntansi.

Menurut Bodnar dan Hopwood (2004: 1) Sistem informasi akuntansi (SIA) adalah kumpulan sumber daya, seperti manusia dan peralatan yang diatur untuk mengubah data keuangan dan data lainnya menjadi informasi.

Romney dan Steinbart (2004: 473) menyatakan bahwa Sistem informasi akuntansi (SIA) adalah sumber daya manusia dan modal dalam organisasi yang bertanggung jawab untuk persiapan informasi keuangan dan informasi yang diperoleh dari mengumpulkan dan memproses berbagai transaksi perusahaan. Pihak Bengkel harus siap setiap saat dengan sarana, prasarana tenaga maupun dana yang dibutuhkan untuk mendukung pelayanan tersebut. Dengan perubahan sistem keuangan Bengkel serta sistem keuangan Pemerintah secara keseluruhan diharapkan dana yang dikelola oleh Bengkel akan menjadi lebih besar dan terus meningkat.

Sistem *Cash Basis* atau disebut juga Kas Stelsel, hanya dicatat penerimaan dan pengeluaran uang, penerimaan akan dicatat jika telah diterima uang demikian juga untuk pengeluaran. Sedangkan sistem *Accrual Basis* transaksi dan peristiwa diakui pada saat kejadian, bukan pada saat hak diterima atau dibayar, dan dicatat serta dilaporkan pada periode yang bersangkutan. Dengan kata lain, penghasilan diakui pada saat penyerahan barang/jasa, bukan pada saat kas diterima, dan biaya diakui pada saat terjadinya bukan pada saat kas dibayarkan. Dengan metode akrual, harta diakui pada saat diperoleh kepemilikan.

2.10.1 Proses Account Payable

Kebutuhan akan barang dan keterbatasan *Cash Flow* yang dimiliki oleh perusahaan adalah dua masalah yang dialami oleh pelaksana *Account Payable*. Satu sisi perusahaan dituntut untuk terus berproduksi memenuhi kebutuhan pelanggan, namun di sisi yang lain perusahaan tidak selalu memiliki kecukupan dana untuk membayar semua aktivitas pembeliannya. Akibatnya sering salah satu dari masalah tersebut dikorbankan. Namun mengorbankan salah dari keduanya ini bukan tanpa resiko. Menunda pembelian barang akan berdampak pada penurunan *sales*, atau bahkan hilangnya keberlanjutan order. Tapi jika menunda pembayaran, *supplier*

akan menolak untuk mengirimkan barang berikutnya. Akibatnya perusahaan akan gagal berproduksi, dan gagal pula untuk *delivery* barang dan jasa.

Strategi yang tepat agar masalah *payable* ini dapat diatasi. Hal ini menunjukkan bahwa fungsi *Account Payable* saat ini tidak lagi sebagai fungsi yang sekedar melakukan kegiatan administrasi. Namun, lebih jauh menjadi fungsi yang strategis mendukung perusahaan dalam mencapai *profit* yang tinggi dan keberlangsungan aktivitasnya (*going concern*). Peningkatan fungsi tentu saja mengandung konsekuensi, bahwa karyawan bagian *Account payable* haruslah personil yang handal, memahami bisnis proses, serta dapat memberikan masukan dan informasi yang berguna dalam pengelolaan *cash flow* perusahaan dan pengambilan keputusan oleh manajemen. Serta terlibat aktif dalam aktivitas penurunan biaya.

2.10.2 Proses Account Receivable

Proses *Account Receivable* mempunyai tugas utama dalam membangun manajemen dalam bisnis proses bengkel yakni dengan mengurusi penagihan konsumen yang berhutang pada seseorang, suatu perusahaan, atau suatu organisasi untuk barang dan layanan yang telah diberikan pada konsumen tersebut. Pada sebagian besar entitas bisnis, hal ini biasanya dilakukan dengan membuat tagihan dan mengirimkan tagihan tersebut kepada konsumen yang akan dibayar dalam suatu tenggat waktu yang disebut termin kredit atau pembayaran.

Kelonggaran dalam menarik piutang akan berakibat kemungkinan kesulitan *cash flow* dan timbul *opportunity cost* yang mesti dibayar dan dapat berimbas pada harga pokok produksi yang tinggi dan kalah dalam persaingan. Perusahaan mengharapkan seluruh piutang dagangnya dapat diterima sesuai dengan jatuh tempo kredit, dan ini bisa terlaksana apabila ada kesungguhan dalam penagihan dan kontrol yang baik dari masing-masing personal yang ditugasi secara *integrated* antara *Sales*, F&A, Operasional, Kepala Cabang, Kepala Divisi, atau *Project*.

Piutang juga disajikan dalam keuangan negara sejak penerapan sistem akuntansi berbasis akrual pada pelaporan keuangan negara berdasarkan Undang

Undang Nomor 17 Tahun 2003 tentang Keuangan Negara. Definisi Piutang Negara diketahui terdapat dalam ketentuan peraturan perundang-undangan melalui Undang-Undang Nomor 49 Prp. Tahun 1960 tentang Panitia Urusan Piutang Negara yang hingga saat ini (November 2012 masih berlaku). Selanjutnya, setelah Republik Indonesia menganut otonomi daerah, Piutang Negara memperoleh definisi yang berbeda dalam Undang-Undang Nomor 1 Tahun 2004 tentang Perbendaharaan Negara dengan perspektif pelaksanaan otonomi daerah.

2.10.3 Laporan Keuangan

Laporan keuangan adalah hasil akhir dari suatu proses pencatatan, yang merupakan ringkasan dari transaksi-transaksi keuangan yang terjadi selama tahun buku yang bersangkutan. Laporan keuangan dibagi dua, yaitu:

1) Laporan *Intern*

Laporan *Intern* adalah Laporan yang dibuat dan ditujukan untuk pihak-pihak *intern* atau manajemen bengkel, dimana dalam penyajiannya dibedakan dalam:

a) Laporan Bulanan

Laporan Bulanan adalah laporan yang harus dibuat setiap satu bulan satu kali

b) Laporan Tahunan

Laporan Tahunan adalah laporan yang harus dibuat setiap satu tahun satu kali. Laporan ini antara lain: Neraca, Laporan Pendapatan dan Biaya Perincian, dan Laporan Keuangan

2) Laporan *Eksternal*

Laporan eksternal adalah laporan yang ditujukan kepada pihak-pihak diluar bengkel. Contohnya: Pemerintah.

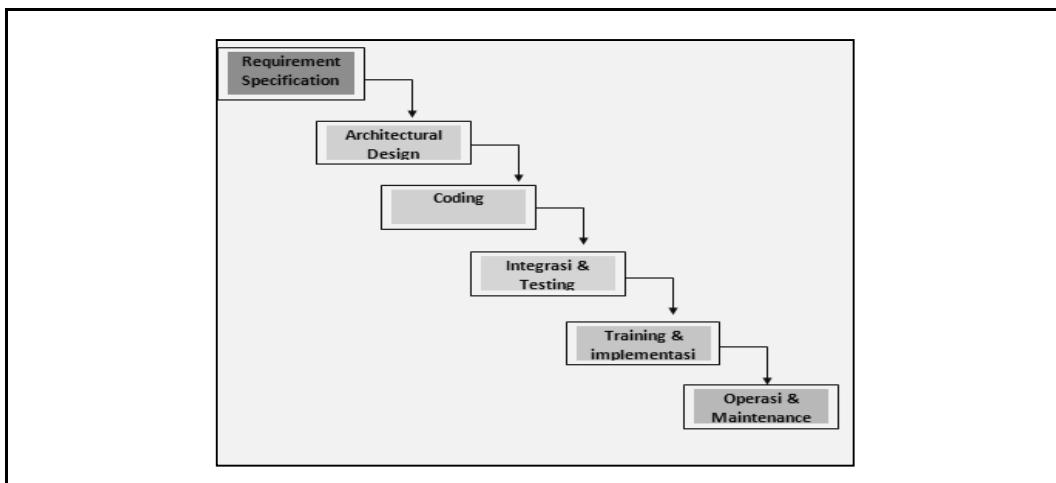
2.11 System Development Life Cycle (SDLC)

Hal dasar dan penting dalam rekayasa perangkat lunak adalah daur hidup perangkat lunak (software development life cycle), yang mendeksripsikan aktivitas

yang terjadi mulai dari pembentukan konsep awal suatu sistem hingga tahap implementasi sistem dan pemeliharaannya. Isu interaksi manusia dan komputer yang menyangkut daya guna sistem interaktif relevan dengan seluruh aktivitas pada SDLC. *Software engineering* untuk sistem interaktif bukan semata-mata menambahkan sebuah tahapan pada SDLC, namun lebih pada melibatkan teknik yang berada sepanjang SDLC itu

Menurut (Jogyanto, 2005) dalam bukunya Analisis & Desain, menjelaskan bahwa metodologi pengembangan sistem adalah metode prosedur, konsep-konsep pekerjaan, aturan akan digunakan untuk mengembangkan suatu sistem informasi. Pengembangan sistem didefinisikan sebagai sistem informasi berbasis komputer untuk menyelesaikan persoalan (*problem*) organisasi atau memanfaatkan kesempatan yang timbul.

Model air terjun (*waterfall*) bias juga disebut siklus hidup perangkat lunak. Mengambil kegiatan dasar seperti spesifikasi, pengembangan, validasi, dan evolusi dan merepresentasikannya sebagai fase seperti spesifikasi persyaratan, perancangan perangkat lunak, implementasi, pengujian dan seterusnya.



Gambar 2.2 Metode Waterfall

(Sumber: Pedoman Analisis dan Desain Sistem Informasi Amikom)

Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam Model Waterfall menurut (Jogyanto, 2005):

2.12.1 Requirements Analysis

Requirement Specification disebut juga sebagai tahap spesifikasi kebutuhan user, dimana Desainer sistem mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan mana yang harus dipenuhi oleh program yg akan dibangun. Tahap ini, Desainer Sistem harus berkomunikasi dengan *Client*. Desainer Sistem atau Sistem Analis harus melakukan pemeriksaan terhadap kebijakan dan prosedur pengolahan data dan sistem informasi yang berlaku saat ini atau disebut dengan istilah *present system*.

2.12.2 Architectural Design

Tahapan *Architectural Design* merupakan tahapan dimana sistem analis berkosentrasi pada bagaimana sistem dibangun, dengan memperhatikan langkah-langkah berikut:

- 1) Mendefinisikan tujuan sistem, tidak hanya berdasarkan informasi dari *user*, tetapi juga berupa analisa dari abstraksi dan karakteristik keseluruhan kebutuhan informasi sistem.
- 2) Membangun sebuah model konseptual, berupa gambaran sistem secara keseluruhan yang menggambarkan satuan fungsional sebagai unit sistem.
- 3) Menerapkan kendala-kendala organisasi.
- 4) Mendefinisikan aktivitas pemrosesan data.
- 5) Menyiapkan proposal sistem desain.

2.12.3 Coding

Desain program diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang telah ditentukan. Setiap komponen diuji untuk memverifikasi apakah sudah berjalan dengan benar setelah Proses *Coding* selesai.

2.12.4 Integrasi dan Testing

Integrasi dan testing dilakukan dengan mengoperasikan program dengan memproses data sehingga kesalahan dapat diketahui seawal mungkin. Pengujian dilakukan dengan teliti, mula-mula per-unit sampai berbagai unit secara

komprehensif, kemudian dilakukan pengujian tes penerimaan dengan *client* untuk memastikan sistem yang dibuat memenuhi kebutuhan mereka.

2.12.5 Training dan Implementasi

Tujuan sistem yang baru adalah untuk mengganti prosedur-prosedur lama, maka pelatihan kepada *user* yang akan menggunakan sistem merupakan hal penting. Tahapan setelah pelatihan selesai dilakukan konversi (peralihan) dari sistem lama ke sistem yang baru, mungkin perlu menulis program khusus untuk menukar *file* yang ada menjadi *file* yang baru atau membuat *file* dari catatan manual, beberapa cara konversi ke sistem yang baru adalah sebagai berikut:

- 1) Konversi langsung yaitu sistem yang lama secara sekaligus diganti dengan sistem yang baru.
- 2) Konversi pararel dengan cara sistem baru dan lama dijalankan secara bersamaan untuk beberapa waktu, sehingga jika sistem baru mengalami gangguan sistem lama dapat mengkompensasi.
- 3) Konversi bertahap adalah peralihan ke sistem yang baru dilakukan bagian per bagian.

2.12.6 Operasi dan Maintenance

Tahapan yang dilakukan setelah pemasangan dan organisasi disesuaikan dengan perubahan-perubahan yang ditimbulkan oleh sistem baru, maka tahap operasional dimulai. Tahap ini perlu dilakukan pemeliharaan terhadap sistem serta peningkatan mutu sistem agar sesuai dengan kebutuhan organisasi, sehingga perlu adanya perubahan dan peningkatan terhadap sistem, tidak masuk akal untuk mengatakan bahwa sebuah sistem informasi berbasis komputer telah selesai, sistem tersebut akan terus berkembang selama daur hidupnya. Maintenance melibatkan koreksi terhadap kesalahan atau error yang ditemui pada sistem setelah dikeluarkan dan segera dilakukan perbaikan terhadap sistem. Pemeliharaan sistem merupakan aktivitas untuk mengadaptasikan sistem dengan tantangan-tantangan baru. Sistem yang terancang baik pada umumnya cukup fleksibel dan terbuka pada perubahan-perubahan kecil yang sesuai dengan perkembangan kebutuhan organisasi.

Perubahan besar dilakukan jika sistem sudah tidak efisien lagi, sehingga dalam hal ini diperlukan daur baru pengembangan sistem informasi.

2.13 Perangkat Pemodelan Sistem Informasi

Suatu sistem dibuat pemodelan untuk memudahkan pemahaman terhadap sistem dalam waktu singkat. Pemodelan sistem dapat dilakukan dengan menggunakan *Data Flow Diagram* (DFD) atau Diagram Alir Data dan konseptual sistem dengan *Entity Relational Diagram* (ERD).

2.13.1 Diagram Konteks

Diagram konteks adalah sebuah diagram sederhana yang melukiskan hubungan antara *entity* luar, masukan dan keluaran dari sistem. Diagram konteks direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem. Diagram Konteks adalah kasus khusus DFD yang berfungsi memetakan model lingkungan, yang direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem.

2.13.2 Data Flow Diagram

Perancangan Diagram Alir Data (DAD) atau dalam Bahasa Inggris disebut *Data Flow Diagram* (DFD) adalah suatu diagram yang menggunakan notasi untuk menggambarkan alir dari data yang penggunaanya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. DFD juga bisa dikatakan sebagai suatu model logika data atau proses yang dibuat untuk menggambarkan asal data dan tujuan data yang keluar dari sistem, tempat data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenalkan pada data tersebut. DFD ini merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi yang dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh *profesional system* kepada pemakai maupun pembuat program.

DFD diperlukan dalam merancang software aplikasi. DFD digunakan untuk menjelaskan *software* aplikasi sebagai jaringan kerja antar proses yang

berhubungan satu sama lain. Diagram ini menunjukkan bagaimana aliran data dari suatu proses ke proses lain atau ke tempat penyimpanan data. DFD dibuat secara bertingkat, dimana suatu proses akan dijelaskan secara rinci pada DFD tingkat yang lebih tinggi (Jogyanto, 2005).

DFD dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mangalir, misalnya lewat telepon, surat, dan sebagainya atau lingkungan fisik dimana data tersebut disimpan, misalnya microfiche, harddisk, tape, diskette, dan lain sebagainya. DFD merupakan alat yang cukup populer sekarang ini, karena dapat menggambarkan alir data di dalam sistem dengan terstruktur dan jelas. DFD juga merupakan dokumentasi dari sistem yang lebih baik.

DFD adalah suatu gambaran grafis dari suatu sistem yang menggunakan sejumlah bentuk-bentuk simbol untuk menggambarkan bagaimana data mengalir melalui proses yang saling berkaitan.

Berikut ini merupakan lambang yang digunakan untuk membuat sebuah Diagram Alir Data (*Data Flow Diagram*).

Gane/Sarson	Yourdon/De Marco	Keterangan
Entitas Eksternal	Entitas Eksternal	Entitas eksternal, dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem
Proses	Proses	Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
Aliran Data ↔ ↔	Aliran Data ↔ ↔	Aliran data dengan arah khusus dari sumber ke tujuan
Data Store	Data Store	Penyimpanan data atau tempat data direfer oleh proses.

Gambar 2.3 Lambang Penggambaran Data Flow Dia gram

(Sumber: Kendall & Kendall 2006. Analisis dan Perancangan Sistem)

Gambar 2.3 menunjukkan lambang yang digunakan dalam penggambaran *data flow diagram*. Terdapat beberapa lambang dalam *data flow diagram* diantaranya yaitu untuk Entitas Eksternal, Proses, Aliran Data, dan *Data Store*.

2.13.3 Entity Relationship Diagram

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan symbol. Menurut Brady dan Loonam (2010), *Entity Relationship Diagram* (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya dilakukan oleh *System Analyst* dalam tahap analisis persyaratan proyek pengembangan sistem. ERD dengan detail pendukung merupakan model data yang digunakan sebagai spesifikasi untuk *database*.

Notasi	Keterangan
	Entitas , adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	Relasi , menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
	Atribut , berfungsi mendeskripsikan karakter entitas (atribut yg berfungsi sebagai key diberi garis bawah)
	Garis , sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

Gambar 2.4 Simbol ERD

(Sumber: Jogyanto HM.2005. Analisis dan Desain Sistem Informasi. Andi. Yogyakarta)

Notasi-notasi simbolik yang digunakan dalam pembuatan *Entity Relationship Diagram* dijelaskan lebih terperinci sebagai berikut.

- 1) Entitas adalah segala sesuatu yang dapat digambarkan oleh data. Entitas juga dapat diartikan sebagai individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain. Ada dua macam entitas yaitu entitas kuat dan entitas lemah. Entitas kuat merupakan entitas yang tidak memiliki ketergantungan dengan entitas lainnya. Contohnya entitas anggota. Sedangkan entitas lemah merupakan entitas yang kemunculannya tergantung pada keberadaan entitas lain dalam suatu relasi.
- 2) Atribut merupakan pendeskripsi karakteristik dari entitas. Atribut digambarkan dalam bentuk lingkaran atau elips. Atribut yang menjadi kunci entitas atau key diberi garis bawah.
- 3) Relasi atau Hubungan menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.
- 4) Penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atribut dinyatakan dalam bentuk garis.

Penggambaran ERD terbagi menjadi tiga jenis, yaitu relasi *one to one*, relasi *one to many*, dan relasi *many to many*, penjelasannya sebagai berikut.

- 1) Relasi satu ke satu (*one-to-one relation*) merupakan hubungan satu entitas A dihubungkan dengan maksimum satu entitas B saja. Peleburan ini didasarkan pada penyertaan atribut ke salah satu tabel yang entitasnya punya derajat minimum paling besar.
- 2) Relasi satu ke banyak atau banyak ke satu (*one-to-many relation or many to-one relation*) merupakan hubungan satu entitas A sebagai 1 (*one*) dihubungkan dengan banyak entitas B sebagai N (*many*), peleburan ini atribut key dari himpunan entitas yang berderajat 1 akan menjadi atribut tambahan bagi himpunan entitas yang berderajat N.
- 3) Relasi banyak ke banyak (*many-to-many relation*) merupakan hubungan satu entitas A sebagai 1 (*one*) dihubungkan dengan banyak entitas B sebagai N (*many*), dan satu entitas B sebagai 1 (*one*) dihubungkan dengan banyak entitas A sebagai N (*many*), peleburan kedua entitas ini akan menjadi sebuah tabel baru dimana *key-key* dari kedua entitas sebagai atributnya

2.13.4 Standard Operational Procedure

Standard Operational Procedure (SOP) adalah dokumen tertulis yang memuat prosedur kerja secara rinci, tahap demi tahap dan sistematis. SOP memuat serangkaian instruksi secara tertulis tentang kegiatan rutin atau berulang-ulang yang dilakukan oleh sebuah organisasi. SOP juga dilengkapi dengan referensi, lampiran, formulir, diagram dan alur kerja. SOP sering juga disebut sebagai manual SOP yang digunakan sebagai pedoman untuk mengarahkan dan mengevaluasi suatu pekerjaan.

Fungsi dari *Standard Operating Procedure* (SOP) adalah untuk menjelaskan secara detail mengenai proses kerja yang berlangsung secara rutin yang harus diikuti dalam suatu perusahaan atau organisasi. Tujuan dari *Standard Operating Procedure* (SOP) adalah sebagai berikut:

- 1) Menjaga konsistensi dan tingkat kinerja petugas atau tim dalam organisasi atau unit.
- 2) Mengetahui dengan jelas peran dan fungsi tiap-tiap posisi dalam organisasi.
- 3) Memperjelas alur tugas, wewenang dan tanggung jawab dari petugas terkait.
- 4) Melindungi organisasi dan staf dari malpraktek atau kesalahan administrasi lainnya.
- 5) Menghindari kegagalan/kesalahan, keraguan, duplikasi dan inefisiensi.

2.14 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. Menurut Kustiyahningsih, “MySQL adalah sebuah basis data yang mengandung satu atau beberapa jumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel”.

MySQL adalah database *server open source* yang cukup popular keberadaanya. Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh praktisi untuk membangun suatu project.

Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh MySQL, memungkinkan bermacam-macam aplikasi Komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data MySQL. Mengoperasikan MySQL tidak hanya memerlukan pengetahuan tentang data yang akan diinputkan ke dalam suatu basis data, tidak luput juga penentuan tipe data yang digunakan akan mempengaruhi efisiensi dan efektivitas dari basis data itu sendiri.

2.14.1 Data Definition Language (DDL)

Struktur atau skema basis data yang menggambarkan atau mewakili desain basis data secara keseluruhan dispesifikasikan dengan bahasa khusus yaitu DDL. Penggunaan DDL dapat membuat tabel (*create table*) baru, indeks, mengubah tabel, menentukan struktur penyimpanan tabel dan lainnya. Hasil dari kompilasi perintah DDL, adalah kumpulan tabel yang disimpan dalam *file* khusus yang disebut kamus data (*data dictionary*). Kamus data merupakan suatu metadata (*superdata*), yaitu data yang mendeskripsikan data sesungguhnya. Contoh perintah DDL dengan *Foxpro* adalah *create matakuliah*, *modify report*, *modify structure*, dan lainnya, sedangkan perintah DDL dengan My-SQL Server 2000. Contohnya adalah *create new database Penjadwalan_mengajar_dosen*.

Tabel 2.1 Contoh Perintah DDL

Perintah	Fungsi
CREATE TABLE	Membuat tabel baru
ALTER TABLE	Menambah satu atau lebih kolom pada tabel yang baru dibuat.
DROP TABLE	Menghapus suatu tabel
CREATE INDEKS	Membuat indeks
DROP INDEKS	Menghapus tabel yang sudah terindeks

CREATE VIEW	Memanipulasi data
DROP VIEW	Menghapus <i>file view</i>

(Sumber: Konsep dan Tuntunan Praktis Basis Data: Abdul Kadir)

2.14.2 Data Manipulation Language (DML)

Bentuk bahasa basis data untuk melakukan menipulasi dan pengambilan data pada suatu basis data. Manipulasi data pada database dapat berupa:

- 1) Penyisipan atau penambahan data pada *file* atau *table* dalam suatu basis data.
- 2) Penghapusan data pada *file* atau *table* dalam suatu basis data.
- 3) Pengubahan data pada *file* atau *table* dalam suatu basis data.
- 4) Penelusuran data pada *file* atau *table* dalam suatu basis data.

Proses *level* fisik dalam DML harus mendefinisikan algoritma yang memungkinkan pengaksesan yang efisien terhadap data. Level yang lebih tinggi yang lebih dipentingkan bukan untuk efisiensi akses, tetapi efisiensi interaksi pemakai dengan sistem.

Tabel 2.2 Contoh Perintah DML

Perintah	Fungsi
SELECT	Menentukan data atau informasi yang akan keluar dari tabel-tabel
UPDATE	Mengubah isi record pada suatu tabel
DELETE	Menghapus sebuah <i>field</i>
INSERT	Menyisipkan suatu <i>record</i>
COMMIT	Menuliskan perubahan ke dalam <i>Disk</i>

ROLLBACK	Membatalkan perintah setelah perintah COMMIT terakhir
----------	---

(Sumber: Konsep dan Tuntunan Praktis Basis Data: Abdul Kadir)

2.14.1 Tipe Data MySQL

MySQL (*My Structured Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data SQL yang mampu *accept* dan *sending* data dengan cepat serta dapat digunakan *multiuser*. MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL). MySQL juga tersedia dalam versi komersial. MySQL adalah sebuah *database* yang dapat digunakan sebagai *client* maupun *server*. Tipe data dalam MySQL dapat disajikan pada Tabel 2.1.

Tabel 2.3 Tipe Data Numerik

Tipe Data	Keterangan
BIT	Menyatakan tipe bit. Satu bit dapat berupa bilangan 0 atau 1. <ul style="list-style-type: none"> • BIT identik dengan BIT(1), yang berarti 1 buah bit • BIT(2) berarti terdapat 2 bit Angka maksimal di dalam kurung yaitu 64, yang berarti 64 bit
TINYINT[(M)]	Menyatakan data bilangan bulat yang berkisar antara -128 sampai 127. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan berkisar dari 0 sampai 127
BOOL atau BOOLEAN, atau TINYINT(1)	Nilai benar dan salah. Nilai 0 dianggap salah dan nilai selain nol dianggap benar
SMALLINT[(M)]	Data bilangan bulat yang berkisar antara -32768 sampai 32767. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai 65535
MEDIUMINT[(M)]	Data bilangan bulat yang berkisar antara -8388608 sampai 8388607. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan berkisar dari 0 sampai dengan 16777215
INT[(M)] atau INTEGER[(M)]	Data bilangan bulat yang berkisar antara -2147683648 sampai 2147683647. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai 4294967295
BIGINT[(M)]	Data bilangan bulat yang berkisar antara -9223372036854775808 sampai 9223372036854775807. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai 18446744073709551615

FLOAT[(M),(D)]	Tipe bilangan pencacah antara -3.402823466E+38 sampai dengan 1.175494351E-38, 0, dan -1.175494351E-38 sampai 3.402823466E+38 <ul style="list-style-type: none"> • M, jumlah digit desimal • D, jumlah digit desimal di belakang titik desimal • Jika kata UNSIGNED disertakan dibelakangnya, nilai negatif tidak diperkenankan
DOUBLE[(M),(D)] atau DOUBLE PRECISION[(M),(D)]	Tipe bilangan pencacah antara -1.7976931348623157E+308 sampai dengan -2.2250738585072014E-308, 0, dan -2.2250738585072014E-308 sampai 3.402823466E+38 <ul style="list-style-type: none"> • M, jumlah digit desimal • D, jumlah digit desimal di belakang titik desimal • Jika kata UNSIGNED disertakan nilai negatif tidak diperkenankan
FLOAT(p)	Tipe bilangan pecahan. Dalam hal ini, p adalah jumlah presisi dalam bit. <ul style="list-style-type: none"> • Jika p berada antara 0 sampai dengan 24, tipe data identik FLOAT tanpa nilai M ataupun D • Jika p berada antara 25 sampai dengan 53, tipe data identik DOUBLE tanpa nilai M ataupun D
DEC[(M,D)] atau DECIMAL[(M,D)]	Bilangan dengan sejumlah digit dibelakang titik desimal <ul style="list-style-type: none"> • M, jumlah digit desimal • D, jumlah digit dibelakang titik desimal • Nila i terbesar untuk M adalah 65 • Jika D tidak disertakan, dianggap D bernilai 0 (yang berarti tidak ada bagian pecahan) • Cocok digunakan jika terdapat data yang butuh kakurasan tinggi, misalnya uang

Tabel 2.1 menjelaskan mengenai tipe data numerik yang terdapat pada MYSQL. Tipe data numerik terdiri dari BIT, TINYINT, BOOLEAN, SMALLINT, MEDIUMINT, INTEGER, BIGINT, FLOAT, DOUBLE, DECIMAL.

Tabel 2.4 Tipe Data Tanggal dan Waktu

Tipe Data	Keterangan
DATE	Tipe untuk data tanggal. Tanggal yang didukung berkisar antara ‘1000-01-01’ sampai dengan ‘999-12-31’. Perlu diketahui, notasi tanggal bisa ditulis dengan format tahun-bulan-tanggal
TIME	Tipe untuk data waktu yang berformat: ‘jam:menit:detik’
DATETIME	Tipe data kombinasi tanggal dan waktu. Format ‘tahun-bulan-tanggal jam:menit:detik’
TIMESTAMP	Tipe kolom yang nilainya akan diisi secara otomatis dengan tanggal dan jam saat ada operasi INSERT atau UPDATE

YEAR[(2 4)]	Tipe untuk tahun. Bila angka tidak disebutkan, tahun dianggap berformat 4 digit. Angka 2 berarti tahun dalam format 2 digit. <ul style="list-style-type: none"> Format 4 digit, tahun dapat berkisar 1901 sampai dengan 2155. Bisa juga 000 Format 2 digit, tahun dapat berkisar 70 sampai dengan 69. 70 berarti 1970 dan 69 berarti 2069 Nilai untuk YEAR(4) dapat diisi berupa string ataupun bilangan
-------------	---

Tabel 2.2 merupakan tabel tipe data tanggal dan waktu yang terdapat pada MYSQL. Tipe data tanggal waktu terdiri dari DATE, TIME, DATETIME, TIMESTAMP, YEAR.

Tabel 2.5 Tipe Data String

Tipe Data	Keterangan
CHAR(M)	String dengan panjang tetap yaitu M karakter. String yang dimasukkan panjang kurang dari M maka sisanya diisi dengan spasi. Nilai terbesar untuk M adalah 255
CHAR	Sinonim dengan CHAR(M)
VARCHAR(M)	String dengan panjang variabel. M merupakan jumlah maksimum karakter yang bisa disimpan. Nilai terbesar untuk M yaitu 65535
BINARY(M)	Serupa dengan CHAR(M), tetapi disimpan dalam bentuk biner
VARBINARY(M)	Serupa dengan VARCHAR(M), tetapi disimpan dalam bentuk biner
TINYBLOB	BLOB dengan ukuran paling kecil
BLOB[(M)]	BLOB dengan ukuran lebih besar daripada TINYBLOB
MEDIUMBLOB	BLOB berukuran medium
LONGBLOB	BLOB berukuran paling besar
TINYTEXT	String text dengan ukuran paling kecil
TEXT[(M)]	String text dengan ukuran lebih besar daripada TINYBLOB
MEDIUMTEXT	String text berukuran medium
LONGTEXT	String text berukuran paling besar
ENUM('nilai1','nilai2',...)	Tipe kolom yang kemungkinan nilai kolom berupa salah satu string yang disebutkan pada ENUM. Jumlah elemen ENUM dapat mencapai 65535. Nilai untuk kolom dapat juga diisi dengan nomor indeks. Daftar berikut menunjukkan nilai dan indeksnya: <ul style="list-style-type: none"> NULL (indeks NULL) '' (indeks 0) 'nilai1' (indeks 1) 'nilai2' (indeks 2), dan seterusnya.
SET('nilai1','nilai2',...)	Satu objek string yang bisa berisi nol, satu, atau beberapa nilai. Sebuah kolom bertipe SET maksimum berisi 64 elemen yang berbeda.

Tabel 2.3 merupakan tipe data *string* yang terdapat pada MYSQL. Tipe data string terdiri dari CHAR(M), CHAR, VARCHAR, BINARY, VARBINARY, TINYBLOB, BLOB, MEDIUMBLOB, LONGBLOB, TINYTEXT, TEXT, MEDIUMTEXT, LONGTEXT, ENUM, SET.

2.15 Laravel

Laravel Framework merupakan sebuah kerangka kerja pemrograman yang berbasis *open source* yang dipakai oleh banyak developer dari seluruh dunia. Menurut Mulyadi, kemudahan penggunaan dan dokumentasi yang lengkap menjadi salah satu faktor mengapa Laravel menjadi primadona dalam beberapa tahun terakhir. Laravel mengikuti pola arsitektur Model-View Controller (MVC). MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti manipulasi data, *controller*, dan *user interface*. Keuntungan pengembangan aplikasi ini adalah dalam proses *maintenance* dan *scalability* yang lebih mudah.

- 1) *Model*, *Model* mewakili struktur data. Biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.
- 2) *View*, *View* adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman *web*.
- 3) *Controller*, *Controller* merupakan bagian yang menjembatani model dan *view*.

2.16 Single Sign On

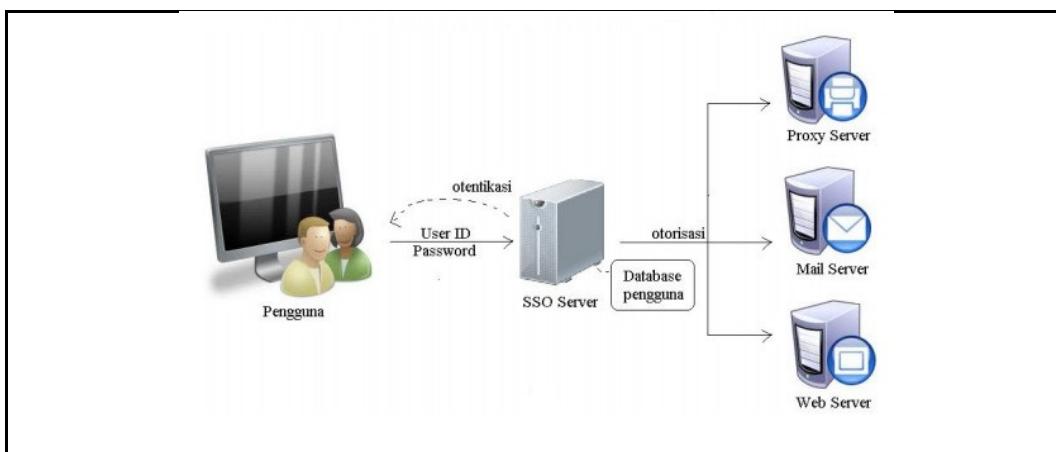
Single Sign On (Priyo Puji Nugroho, 2012) adalah sebuah mekanisme yang membuat *user* hanya perlu mengingat satu *username* dan *password* yang autentik untuk membuka beberapa layanan sekaligus. Dari dua pengertian tersebut yang dapat saya ambil bahwa *single sign on system* merupakan suatu mekanisme *autentifikasi* untuk dapat mengakses keseluruhan sumber daya seperti beberapa situs atau layanan lainnya hanya dengan satu kali *login* saja.

Sistem Single Sign On menghindari *login* ganda dengan cara mengidentifikasi subjek secara ketat dan memperkenankan informasi otentikasi untuk digunakan dalam sistem atau kelompok sistem yang terpercaya. Sistem SSO dapat meningkatkan kegunaan jaringan secara keseluruhan dan pada saat yang sama dapat memusatkan pengelolaan dari parameter sistem yang relevan.

Pengguna layanan lebih menyukai sistem *Single Sign On* daripada sistem *sign on* biasa, namun pengelola layanan jaringan memiliki banyak tugas tambahan

yang harus dilakukan, seperti perlunya perhatian ekstra untuk menjamin bukti-bukti otentikasi agar tidak tersebar dan tidak disadap pihak lain ketika melintasi jaringan.

Sistem ini tidak memerlukan interaksi yang manual, sehingga memungkinkan pengguna melakukan proses sekali *login* untuk mengakses seluruh layanan aplikasi tanpa berulang kali menginputkan password-nya. Teknologi ini sangat diminati dalam jaringan yang sangat besar dan bersifat heterogen, di mana sistem operasi serta aplikasi yang digunakan berasal dari banyak vendor, dan pengguna diminta untuk mengisi informasi dirinya ke dalam setiap *multi-platform* yang hendak diakses.



Gambar 2.5 Sistem Single Sign On

(Sumber: K.D Lewis (2009).Jurnal Web Single Sign-On Authentication using SAML," International Journal of Computer Science Issues Vol. 02)

BAB III

METODOLOGI PENELITIAN

Bab III berisi pembahasan mengenai gambaran sistem. Gambaran sistem tersebut berupa metodologi penelitian, waktu dan tempat dilakukan penelitian, metode pengumpulan data, gambaran umum dan perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*.

3.1 Waktu dan Tempat Penelitian

Tempat pelaksanaan penelitian ini dilakukan di Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Udayana, Kampus Bukit Jimbaran. Waktu pelaksanaan penelitian ini dimulai pada bulan September 2020 sampai Juni 2021.

3.2 Sumber Data

Sumber data dalam penelitian tugas akhir ini diperoleh dari data yang berasal dari studi literatur. Studi literatur yang digunakan ini diperoleh dari penelitian yang sudah dilakukan terdahulu, seperti jurnal ilmiah, buku, *paper* publikasi maupun *e-book*.

3.3 Metodologi

Metodologi penelitian merupakan aspek dasar dalam penggerjaan suatu sistem. Metodologi penelitian berisi tahapan atau gambaran dari pembuatan tugas akhir. Metodologi yang digunakan dalam perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* adalah model pendekatan SDLC (System Development Life Cycle). Model pendekatan SDLC yang digunakan yaitu model *waterfall*. Model *waterfall* merupakan proses permodelan yang memiliki sifat linear dan berurutan atau sekuensial, dalam pembentukannya model *waterfall* terbagi menjadi beberapa tahapan dimana setiap tahapannya nanti akan memiliki *output* dan juga memerlukan *input*. *Input* yang diperlukan dari suatu tahapan nanti merupakan *output* dari tahapan sebelumnya, sehingga model ini berjalan secara linear dan

beurutan, tidak dapat langsung melompati suatu tahapan dalam permodelannya jika tahap tersebut belum terselesaikan.

3.3.1 Requirement Analysis

Requirement Specification disebut juga sebagai tahap spesifikasi kebutuhan user, dimana Desainer sistem mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan mana yang harus dipenuhi oleh program yg akan dibangun. Tahap ini, Desainer Sistem harus berkomunikasi dengan *Client*. Desainer Sistem atau Sistem Analis harus melakukan pemeriksaan terhadap kebijakan dan prosedur pengolahan data dan sistem informasi yang berlaku saat ini atau disebut dengan istilah *present system*.

3.3.2 Merancang Pemodelan Sistem

Pada tahap ini akan dirancang struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengkodean. Permodelan *Use Case* dan juga alur program akan mentranslasikan kebutuhan perangkat lunak agar bisa diimplementasikan. Perancangan ini dibutuhkan agar nantinya aplikasi yang berjalan memiliki alur dan dasar yang jelas.

3.3.3 Implementasi Kode Program

Tahapan selanjutnya yaitu pengimplementasian kode program dimana tahapan ini merupakan salah satu tahapan utama dari pembuatan dan pengembangan sistem. Desain program diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang telah ditentukan. Setiap komponen diuji untuk memverifikasi apakah sudah berjalan dengan benar setelah Proses *Coding* selesai.

3.3.4 Integrasi dan Testing

Pada tahap ini dilakukan pengujian masing masing fitur dan fungsi yang ada di dalam aplikasi untuk mengetahui apakah dapat bekerja dengan semestinya. Penulis melakukan testing apakah fitur-fitur aplikasi yang dibuat sudah berjalan dengan

semestinya atau tidak. Pengujian yang penulis lakukan dengan menggunakan metode pendekatan *blackbox* testing, uji coba *blackbox* digunakan untuk mendemonstrasikan fungsi *software* yang dioperasikan

3.3.5 Operasi dan Maintenance

Tahapan yang dilakukan setelah pemasangan dan organisasi disesuaikan dengan perubahan-perubahan yang ditimbulkan oleh sistem baru, maka tahap operasional dimulai. Tahap ini perlu dilakukan pemeliharaan terhadap sistem serta peningkatan mutu sistem agar sesuai dengan kebutuhan organisasi, sehingga perlu adanya perubahan dan peningkatan terhadap sistem, tidak masuk akal untuk mengatakan bahwa sebuah sistem informasi berbasis komputer telah selesai, sistem tersebut akan terus berkembang selama daur hidupnya *Maintenance* melibatkan koreksi terhadap kesalahan atau *error* yang ditemui pada sistem setelah dikeluarkan dan segera dilakukan perbaikan terhadap sistem.

Pemeliharaan sistem merupakan aktivitas untuk mengadaptasikan sistem dengan tantangan-tantangan baru. Sistem yang terancang baik pada umumnya cukup fleksibel dan terbuka pada perubahan-perubahan kecil yang sesuai dengan perkembangan kebutuhan organisasi. Perubahan besar dilakukan jika sistem sudah tidak efisien lagi, sehingga dalam hal ini diperlukan daur baru pengembangan sistem informasi.

3.4 Alat Penelitian

Alat-alat penelitian yang digunakan dalam perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* adalah sebagai berikut.

3.4.1 Perangkat Keras (Hardware)

Perangkat keras yang digunakan dalam perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* adalah laptop ASUS ROG dengan CPU Core i7-7700HQ kecepatan proses skisaran 2.8 GHz - 3.2GHz

disertai dukungan GPU NVIDIA GeForce GTX 1050, with 2GB/4GB GDDR5 VRAM, RAM sebesar 8096 MB, HDD sebesar 1000 GB atau 1TB dengan didukung SSD 128 GB V-gen, 64-bit Operating System Windows 10, x64-based processor.

3.4.2 Perangkat Lunak (Software)

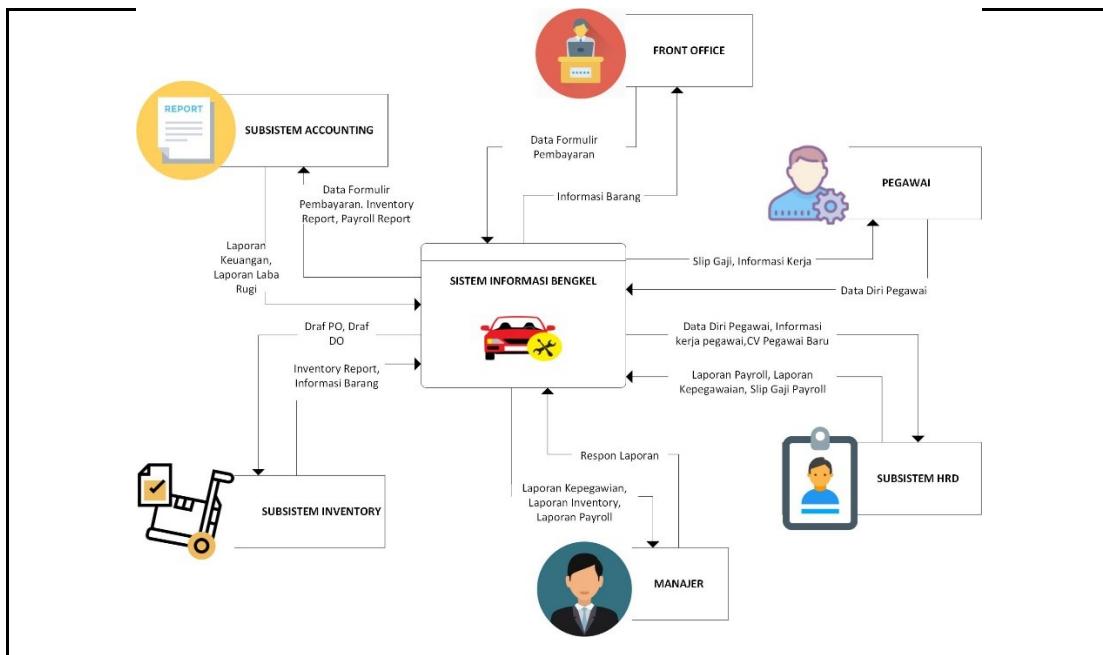
Perangkat lunak yang digunakan dalam perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* adalah sistem operasi Windows 10 pro, SQLyog Ultimate –MySQL GUI v12.09 (64-bit), Visual Studio Code, XAMPP Control Panel, Google Chrome dan Postman.

3.5 Perancangan Sistem

Perancangan Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* terdiri dari gambaran umum sistem, *flowchart*, perancangan basis data, dan *design interface* sistem. Tujuan dari perancangan sistem adalah untuk mengkonsepkan terlebih dahulu sistem yang akan dibuat. Hal ini dilakukan guna meminimalisir kekurangan dan kelemahan dalam proses pembuatan sistem. Perancangan sistem juga diharapkan dapat mempermudah kinerja penulis dalam merancang dan membangun sistem informasi.

3.5.1 Gambaran Umum

Gambaran umum sistem merupakan merupakan suatu sketsa mengenai alur proses yang terdapat dalam aplikasi secara umum. Gambaran umum diperlukan agar dapat memberikan suatu bayangan terhadap sistem yang akan dibangun nantinya. Gambaran umum dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* dapat dilihat pada gambar berikut.



Gambar 3.1 Gambaran Umum Sistem

Gambar 3.1 merupakan gambaran umum dari Sistem Informasi Manajemen Bengkel Modul Kepergawaan, *Inventory* dan *Accounting*. Sistem informasi manajemen bengkel merupakan sistem yang ditujukan untuk memudahkan dalam melakukan manajemen. Khususnya dalam manajemen terhadap sistem kepegawaan, *Accounting* dan *inventory* suatu bengkel yang terintegrasi dengan sistem *front office*. Subsistem pada Sistem Informasi Manajemen Bengkel ini terdiri dari

- 1) Sub Sistem *Human Resources Departemen* (HRD), Melakukan manajemen kepegawaian dan *Payroll* sistem atau sistem penggajian pegawai.
- 2) Sub Sistem *Accounting*, Pembuatan laporan dan pembukuan secara berkala dan analisis laba rugi.
- 3) Sub Sistem *Inventory*, Melakukan manajemen *inventory* sparepart kendaraan.

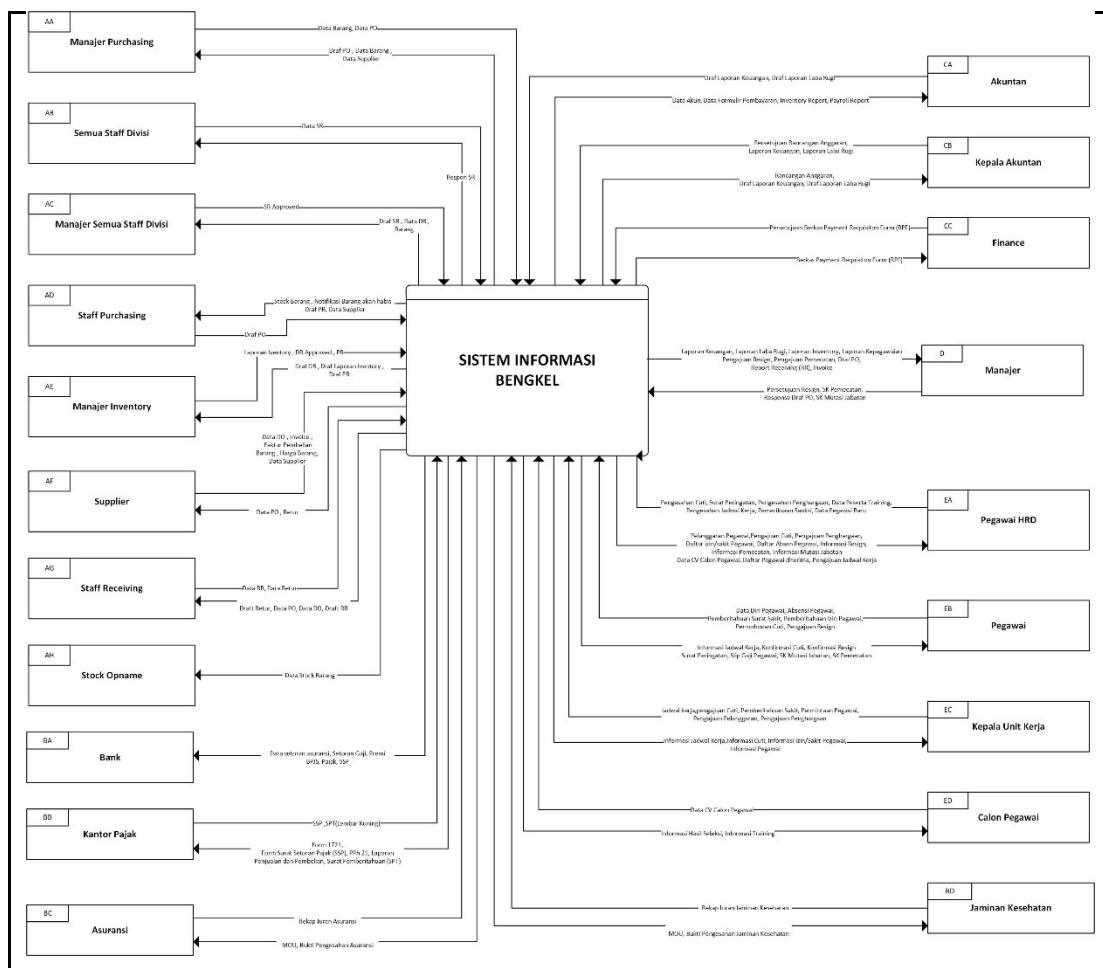
3.5.2 Data Flow Diagram (DFD)

Data flow diagram atau diagram data ini merupakan penjabaran sistem dari *diagram konteks*, diturunkan menjadi beberapa proses yang terjadi antara entitas yang

terlihat dengan sistem. *Data flow diagram* juga memperjelas dokumen-dokumen yang mengalir dalam sistem.

3.5.2.1 Diagram Konteks Sistem

Diagram konteks dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* memiliki 14 buah entitas didalam sistem dan 6 entitas diluar sistem. *Diagram konteks* pada Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* adalah sebagai berikut.



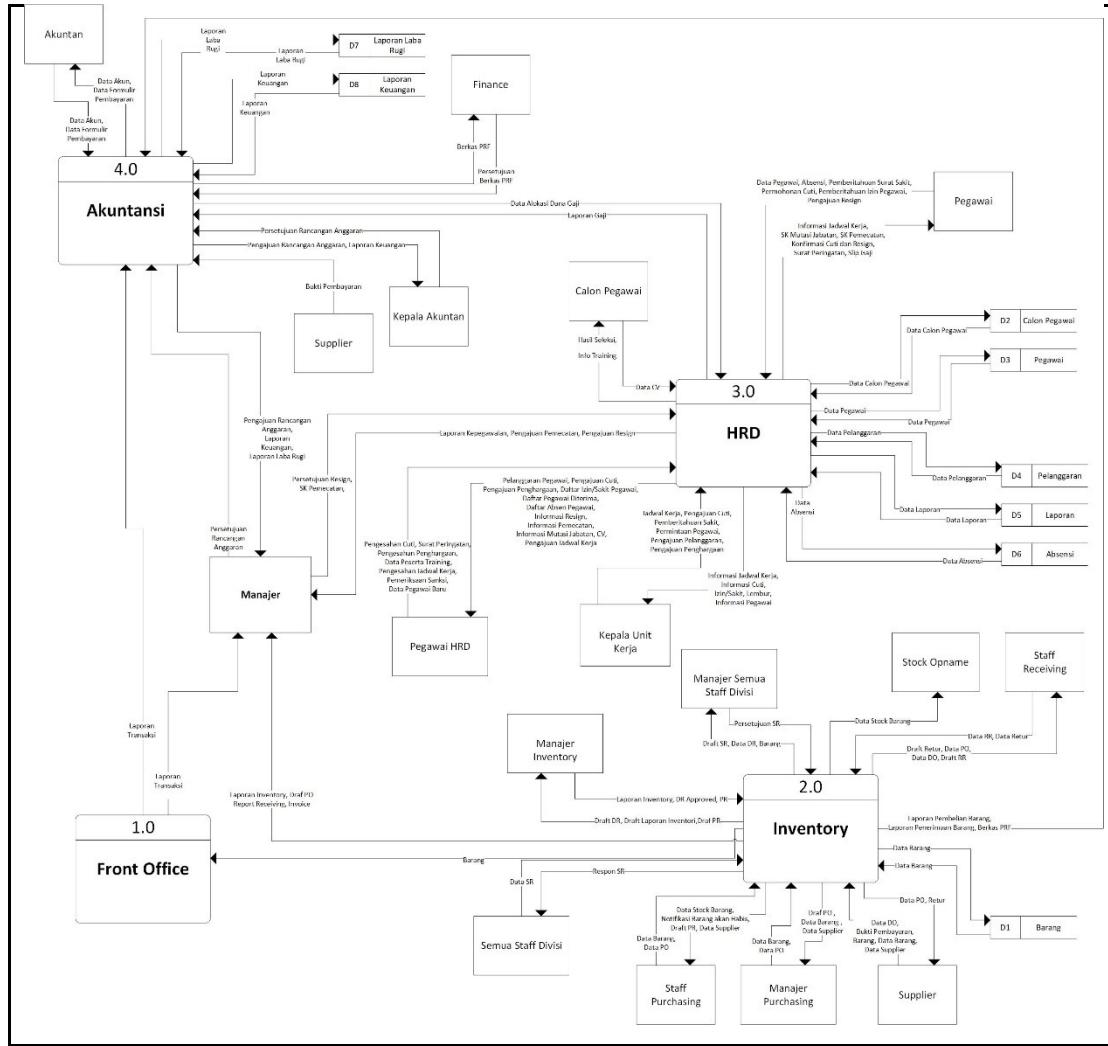
Gambar 3.2 Diagram Konteks Sistem

Gambar 3.2 merupakan diagram konteks dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Entitas yang terlibat dalam

sistem yakni terdiri dari 14 entitas yang berasal dari dalam sistem diantaranya manajer *purchasing*, staff *purchasing*, manajer *inventory*, staff *receiving*, stok *opname*, manajer semua staff divisi, semua staff divisi, akuntan, kepala akuntan, *finance*, manajer, pegawai hrd, pegawai dan kepala unit kerja. Sedangkan untuk entitas diluar sistem terdiri dari 6 entitas diantaranya calon pegawai, *supplier*, bank, kantor pajak, asuransi dan jaminan kesehatan.

3.5.2.2 DFD Level 0 Sistem

DFD Level 0 menggambarkan bagaimana sistem berinteraksi dengan *external* entitas. Pada diagram konteks akan diberikan nomor untuk setiap proses yang berjalan, umumnya mulai dari angka 0 untuk start awal. Berikut merupakan DFD Level 0 dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*.



Gambar 3.3 DFD Level 0 Sistem

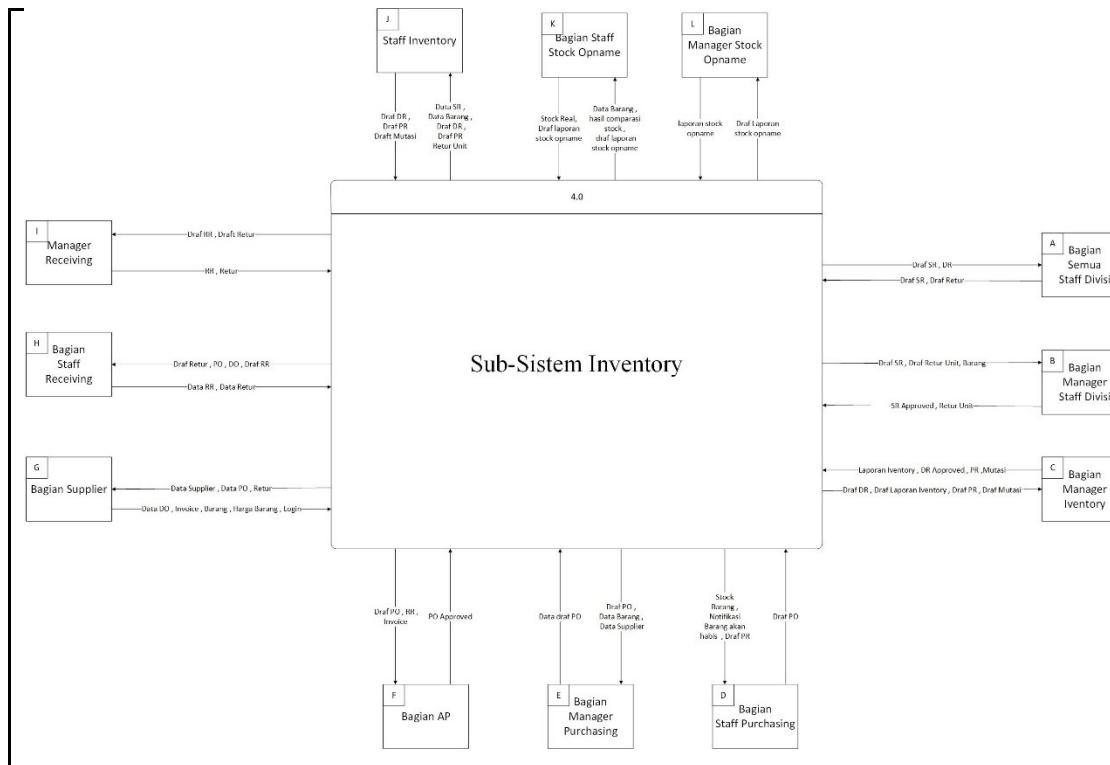
Gambar 3.3 merupakan DFD Level 0 dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. DFD Level 0 menjelaskan setiap entitas dalam maupun entitas luar yang terlibat dan saling berkaitan dan menunjang proses bisnis dari manajemen bengkel. DFD Level 0 terdiri dari 4 modul proses yakni

- 1) *front office*, yang berguna menerima data dari ketiga modul lainnya yakni *inventory*, *accounting* dan *HRD* proses yang dilakukan seperti mengirim laporan transaksi yang akan diproses pada modul *Accounting* untuk dilakukan

- analisis laba rugi dan proses laporan keuangan serta selanjutnya dikirim kepada manajer.
- 2) *Inventory*, merupakan modul dengan proses manajemen *inventory* sparepart pada Sistem Informasi Manajemen Bengkel ini dimana proses yang dilakukan oleh entitas sudah dijelaskan pada gambar. Entitas *inventory* terdiri dari 7 entitas dari dalam sistem yakni manajer *inventory*, stok *opname*, staff *receiving*, manajer *purchasing*, manajer semua staff divisi, semua staff divisi dan staff *purchasing* sedangkan entitas dari luar sistem yakni *supplier* yang bertugas untuk memenuhi *supply* sparepart dari bengkel.
 - 3) *Human Resources Departemen*, merupakan modul 3 dengan proses manajemen kepegawaian pada Sistem Informasi Manajemen Bengkel dimana setiap proses dilakukan oleh setiap entitas yang terhubung ke modul. Entitas HRD terdiri dari 4 entitas dari dalam sistem yakni manajer, kepala HRD, pegawai dan kepala unit kerja.
 - 4) *Accounting*, merupakan modul 4 dengan proses melakukan manajemen keuangan pada Sistem Informasi Manajemen Bengkel dimana setiap proses dilakukan oleh setiap entitas yang ada. Entitas *Accounting* terdiri dari 3 entitas dari dalam sistem yakni akuntan, kepala akuntan dan *finance*.

3.5.2.3 Diagram Konteks Sub-Sistem *Inventory*

Diagram konteks selanjutnya yakni pada sub-sistem *inventory* yang terdapat pada Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan diagram konteks sub-sistem *inventory*.

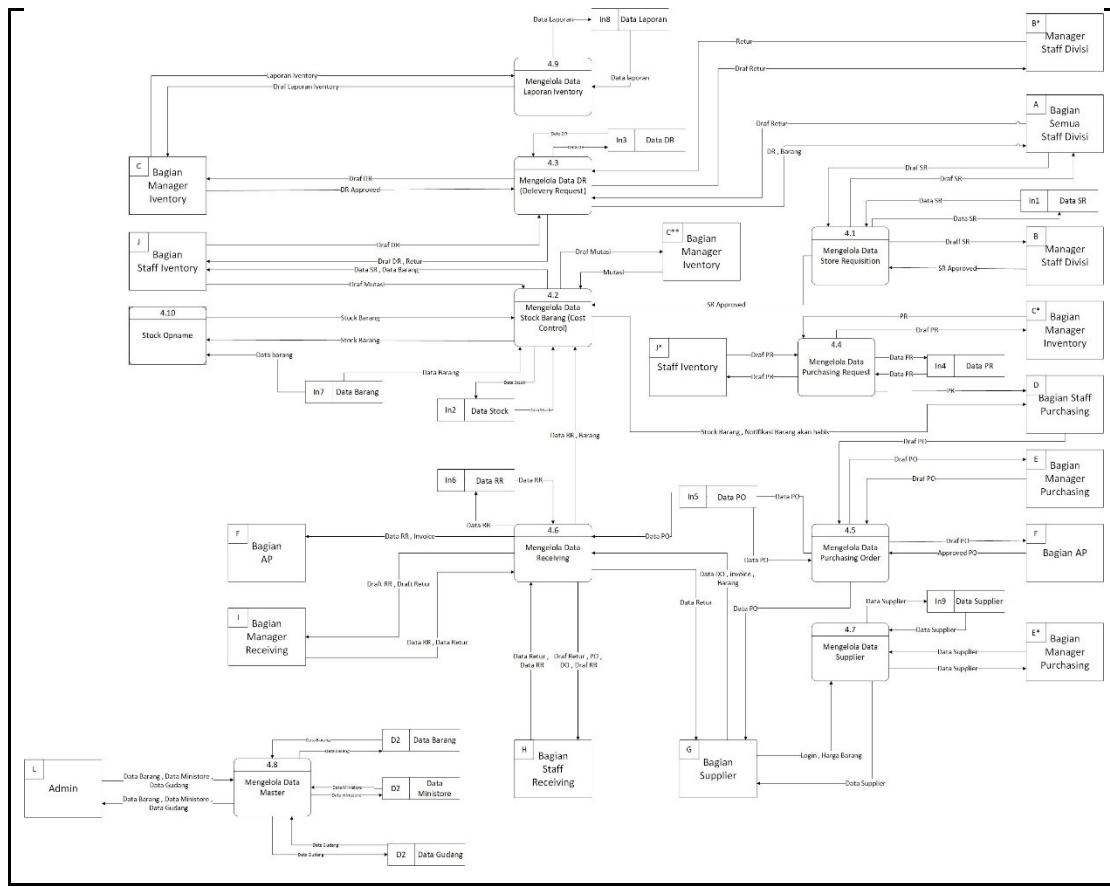


Gambar 3.4 Diagram Konteks Sub-sistem *Inventory*

Gambar 3.4 merupakan diagram konteks Sub-sistem *Inventory* pada Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Diagram Konteks Sub-sistem *Inventory* menjelaskan entitas yang terlibat pada sub-sistem *inventory* dimana pada sub-sistem tersebut terdiri dari 12 entitas yang masing-masing entitas memiliki tugas dan proses yang berbeda dalam menunjang proses manajemen *inventory* pada bengkel.

3.5.2.4 DFD Level 0 Sub-Sistem *Inventory*

DFD Level 0 selanjutnya yakni pada sub-sistem *inventory* yang terdapat pada Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan DFD Level 0 dari sub-sistem *inventory*.



Gambar 3.5 DFD Level 0 Sub-Sistem *Inventory*

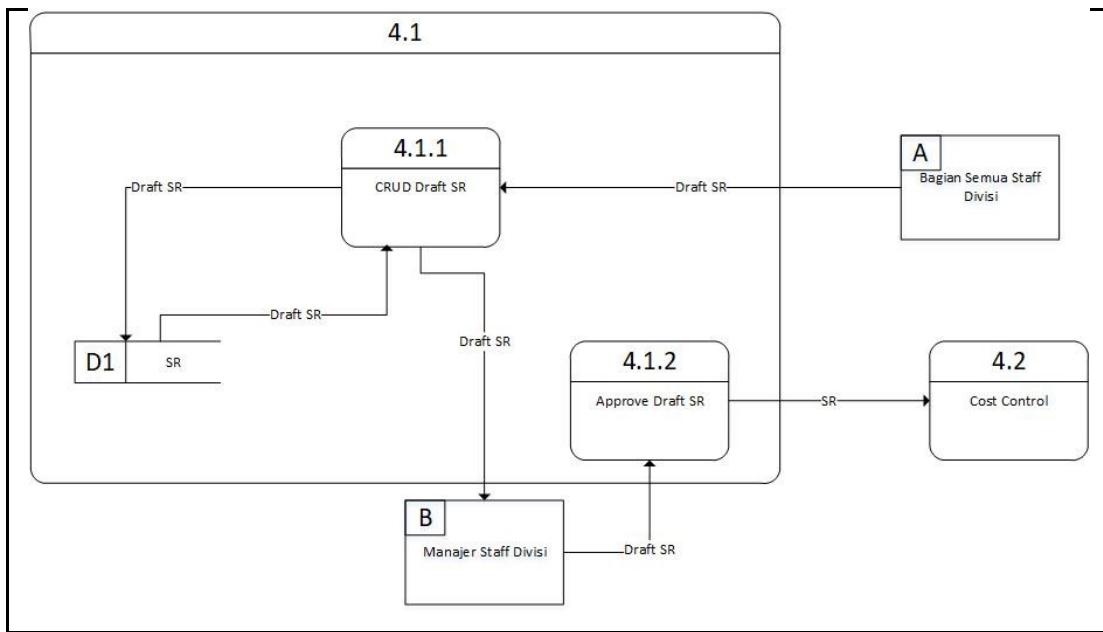
Gambar 3.5 merupakan DFD level 0 sub-sistem *inventory* dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. DFD level 0 sub-sistem tersebut menjelaskan proses dan entitas yang terlibat dalam setiap proses yang ada. Setiap proses pada sub-sistem *inventory* antara lain.

- 1) Mengelola data *store request*, proses ini dilakukan oleh entitas seluruh staff divisi yang memerlukan barang dan manager staff divisi yang melakukan *approved SR*.
- 2) Mengelola data stok barang, proses ini dilakukan oleh entitas bagian staff *inventory* dan bagian stok *opname* untuk melakukan pengecekan data stok.
- 3) Mengelola data *delivery request*, proses ini dilakukan setelah *store request* sudah di *approved* dan data barang sudah di cek.

- 4) Mengelola data *purchasing request*, proses ini dilakukan oleh entitas staff *inventory* dimana akan di *approve* oleh bagian manajer *inventory* jika sudah sesuai.
- 5) Mengelola data *purchasing order*, proses ini terjadi setelah *purchasing request* dilakukan dan sudah di *approve* setelah itu bagian ini akan terhubung dengan *supplier* dalam melakukan pembelian barang.
- 6) Mengelola data *supplier*, proses ini bertujuan untuk kelola data *supplier* dimana data *supplier* akan masuk ke dalam sistem dan bagian manager *purchasing* mendapatkan data *supplier*.
- 7) Mengelola data *receiving*, proses ini dijalankan setelah *purchasing order* dilakukan dimana entitas yang terlibat yakni bagian staff *receiving* dan di *approve* oleh bagian manager *receiving*.
- 8) Mengelola data master, proses ini dilakukan oleh entitas *administrator* dimana data yang terdapat pada proses ini data barang, data *ministore* dan data gudang.
- 9) Mengelola data laporan *inventory*, proses ini merupakan proses terakhir dari sub-sistem *inventory* yang berguna dalam melakukan pelaporan manajemen *inventory* kepada manajer dan dilaporkan oleh entitas bagian manajer *inventory*.

3.5.2.5 DFD Level 1 Store Request

DFD Level 1 selanjutnya yakni *store request* yang terdapat pada Sub-Sistem *Inventory* Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan DFD Level 1 dari *store request* sub-sistem *inventory*.

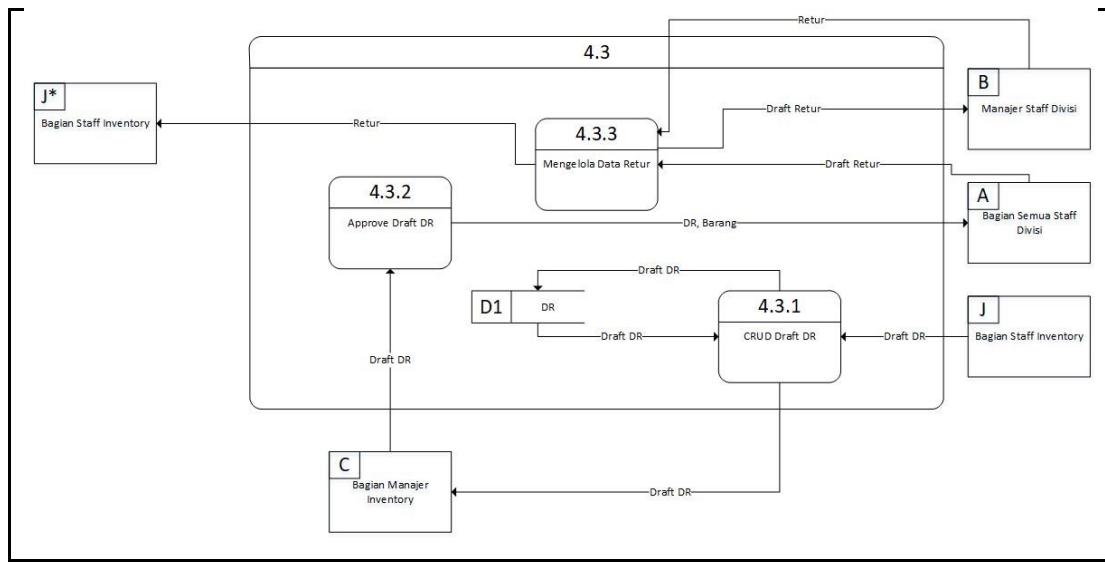


Gambar 3.6 DFD Level 1 Store Request

Gambar 3.6 merupakan DFD Level 1 *Store Request* yang terdapat pada sub-sistem *inventory* Sistem Informasi Manajemen Bengkel. DFD level 1 *store request* menjelaskan bagaimana bagian semua staff divisi melakukan request barang. Proses dari *store request* ini antara lain *CRUD Draft SR* dan *Approve Draft SR* oleh manajer staff divisi yang dihasilkan data 1 yakni data SR atau *store request*.

3.5.2.6 DFD Level 1 Delivery Request

DFD Level 1 selanjutnya yakni *delivery request* yang terdapat pada Sub-Sistem *Inventory* Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan DFD Level 1 dari *delivery request* sub-sistem *inventory*.



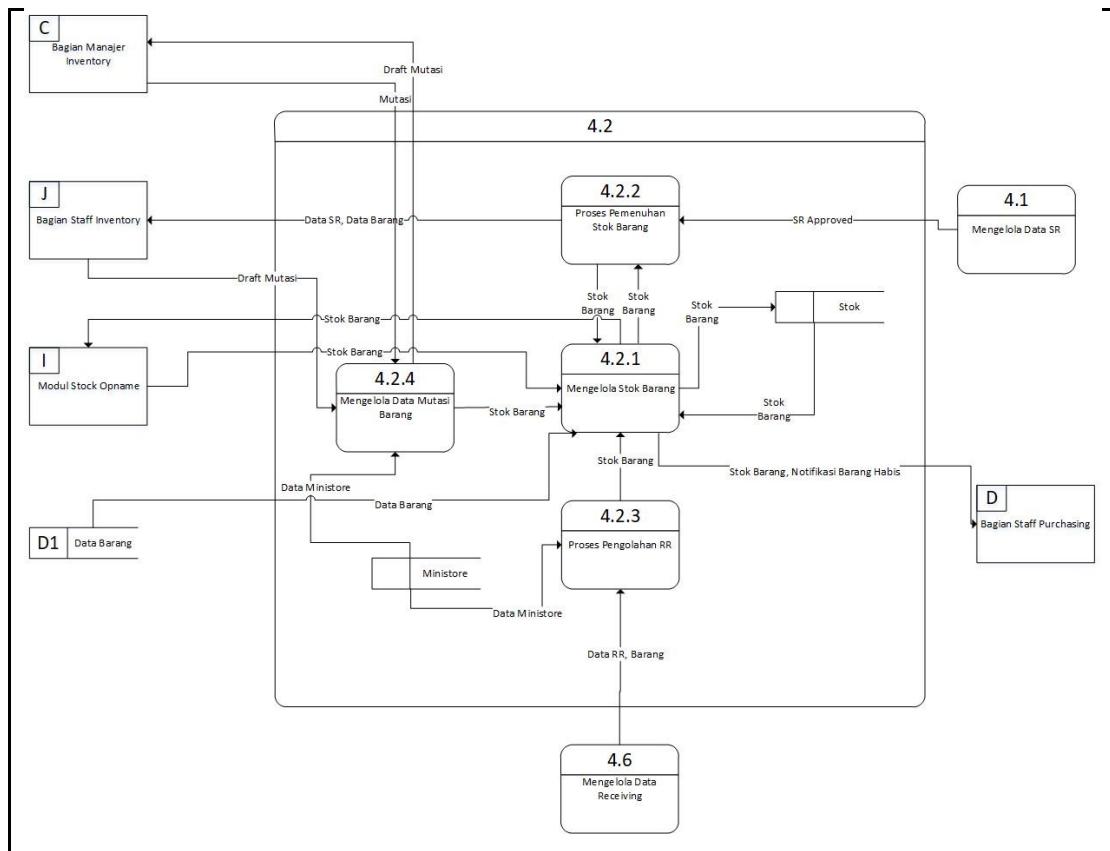
Gambar 3.7 DFD Level 1 Delivery Request

Gambar 3.7 merupakan DFD Level 1 dari *Delivery request* yang terdapat pada sub-sistem *inventory* Sistem Informasi Manajemen Bengkel. DFD Level 1 *delivery request* menjelaskan proses setelah store request dilakukan yakni mengirim barang yang di *request*. Proses dari *delivery request* antara lain

- 1) CRUD draft DR yang dilakukan oleh bagian staff *inventory*.
- 2) Proses *approve* draft DR yang dilakukan oleh bagian manajer *inventory* yang selanjutnya DR yang berupa barang akan dikirim ke divisi terkait.
- 3) Proses terakhir yakni mengelola data *retur* pengembalian kepada bagian staff *inventory* jika terdapat kesalahan pengiriman saat DR dilakukan.

3.5.2.7 DFD Level 1 Stock Control

DFD Level 1 selanjutnya yakni stok *control* yang terdapat pada Sub-Sistem *Inventory* Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan DFD Level 1 dari stok *control* sub-sistem *inventory*.



Gambar 3.8 DFD Level 1 Stock Control

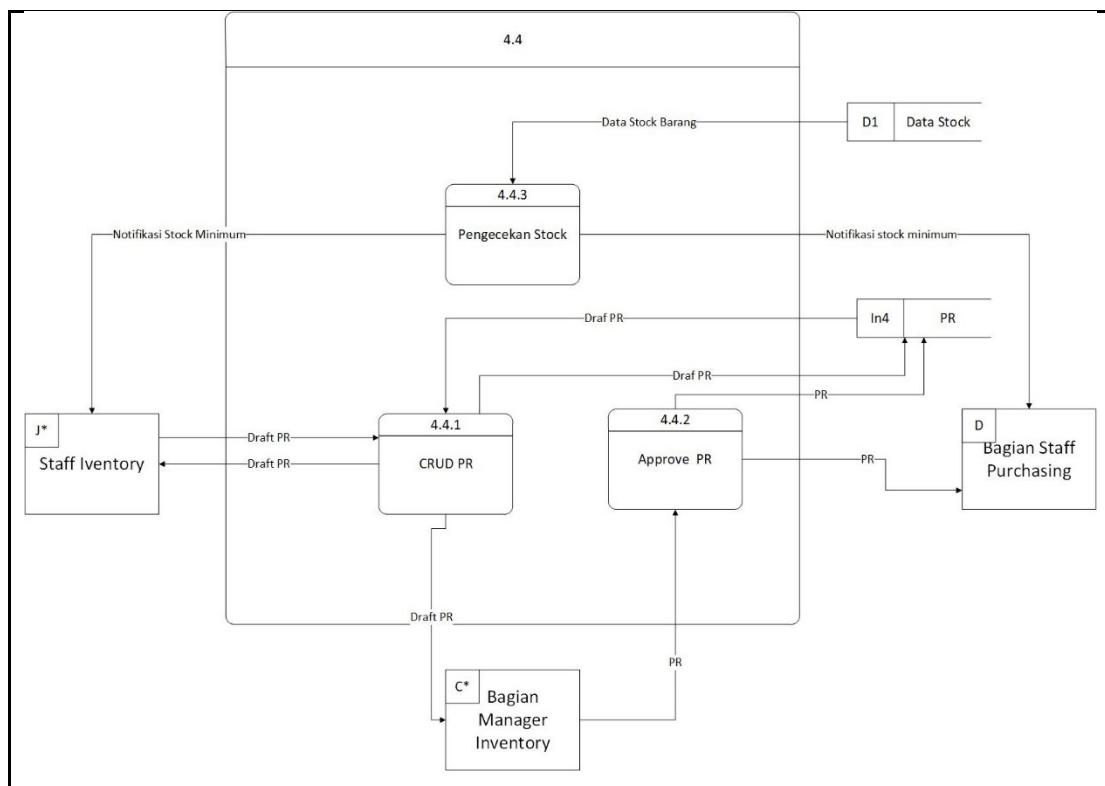
Gambar 3.8 merupakan DFD Level 1 dari stok *control* yang terdapat pada Sub-Sistem *Inventory* Sistem Informasi Manajemen Bengkel. Stok *control* berguna untuk melakukan controlling stok minimal, mutasi dan pemenuhan stok barang. Proses dari stok *control* yakni antara lain

- 1) Mengelola stok *barang*, proses ini dilakukan oleh bagian stok *opname* dan jika stok *barang* habis maka bagian staff *purchasing* akan melakukan pembelian ke *supplier* terkait.
- 2) Proses pemenuhan stok *barang*, proses ini dilakukan oleh bagian bagian staff *inventory* dimana setelah dilakukannya *store request* dan sudah di *approve*.
- 3) Proses pengolahan RR, proses ini dilakukan setelah pembelian *barang* terjadi dan *receiving* berjalan.

- 4) Mengelola data mutasi barang, proses ini dilakukan oleh bagian staff *inventory* dan di *approve* oleh bagian manajer *inventory* yang selanjutnya data stok barang akan dikirim kembali ke proses mengelola stok barang.

3.5.2.8 DFD Level 1 Purchase Request

DFD Level 1 selanjutnya yakni *purchase request* yang terdapat pada Sub-Sistem *Inventory* Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan DFD Level 1 dari *purchase request* sub-sistem *inventory*.



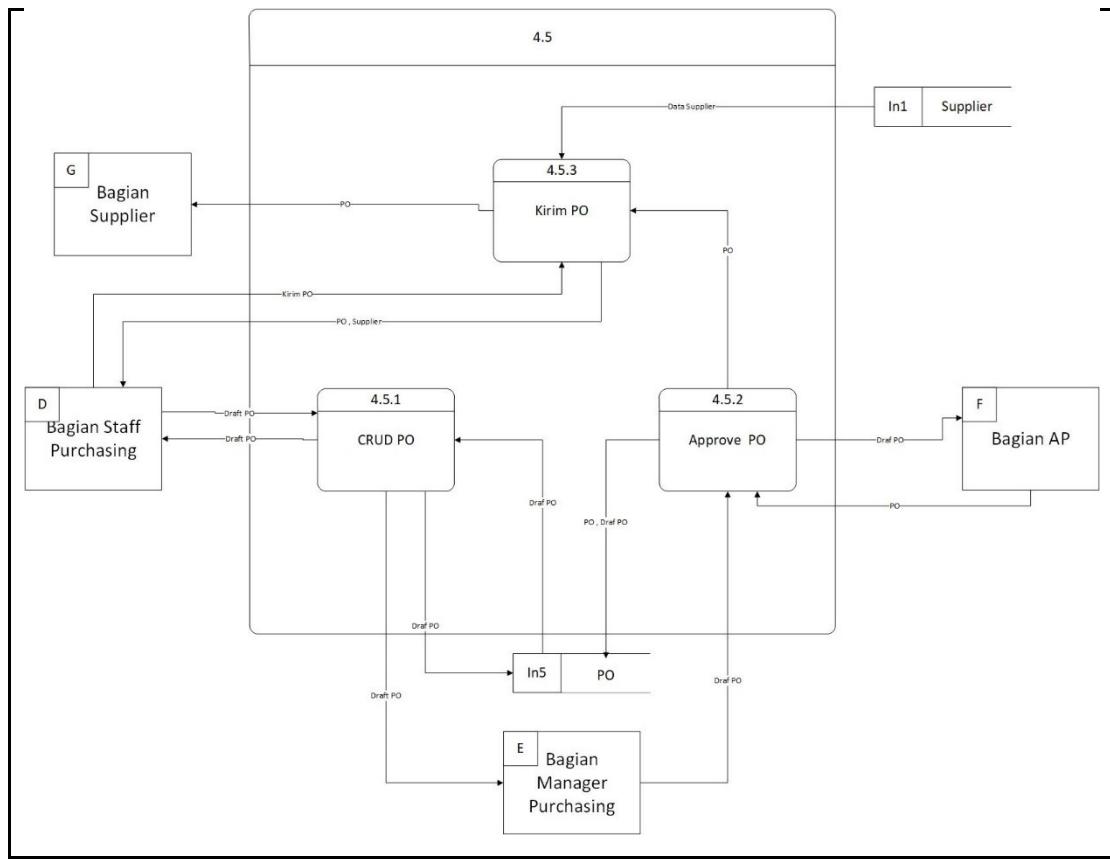
Gambar 3.9 DFD Level 1 Purchase Request

Gambar 3.9 merupakan DFD level 1 *purchase request* yang terdapat pada sub-sistem *inventory* Sistem Informasi Manajemen Bengkel. DFD Level 1 ini menjelaskan *request pembelian* dilakukan, dimana proses dari *purchase request* antara lain yakni

- 1) CRUD *purchase request*, proses ini dilakukan oleh bagian staff *inventory* yang dikirim dalam bentuk draf PR.
- 2) Approve purchase request, proses ini dilakukan oleh bagian manager *inventory* dimana manager melakukan approval terhadap draf PR yang dikirim oleh staff *inventory*. Selanjutnya, data PR yang sudah di *approve* akan dikirim ke bagian staff *purchasing* untuk dilakukan pembelian.
- 3) Pengecekan stok, proses ini merupakan proses terakhir dari *purchase request* yang dimana proses ini akan mengirimkan notifikasi stok minimum ke bagian staff *purchasing* dan staff *inventory*.

3.5.2.9 DFD Level 1 Purchase Order

DFD Level 1 selanjutnya yakni *purchase order* yang terdapat pada Sub-Sistem *Inventory* Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Berikut merupakan pemaparan DFD Level 1 dari *purchase order* sub-sistem *inventory*.



Gambar 3.10 DFD Level 1 Purchase Order

Gambar 3.10 merupakan DFD Level 1 *purchase order* yang terdapat pada sub-sistem dari *inventory* Sistem Informasi Manajemen Bengkel. DFD Level 1 *purchase order* menjelaskan tahapan dan proses dari order atau pembelian barang. Proses dari *purchase order* antara lain yakni

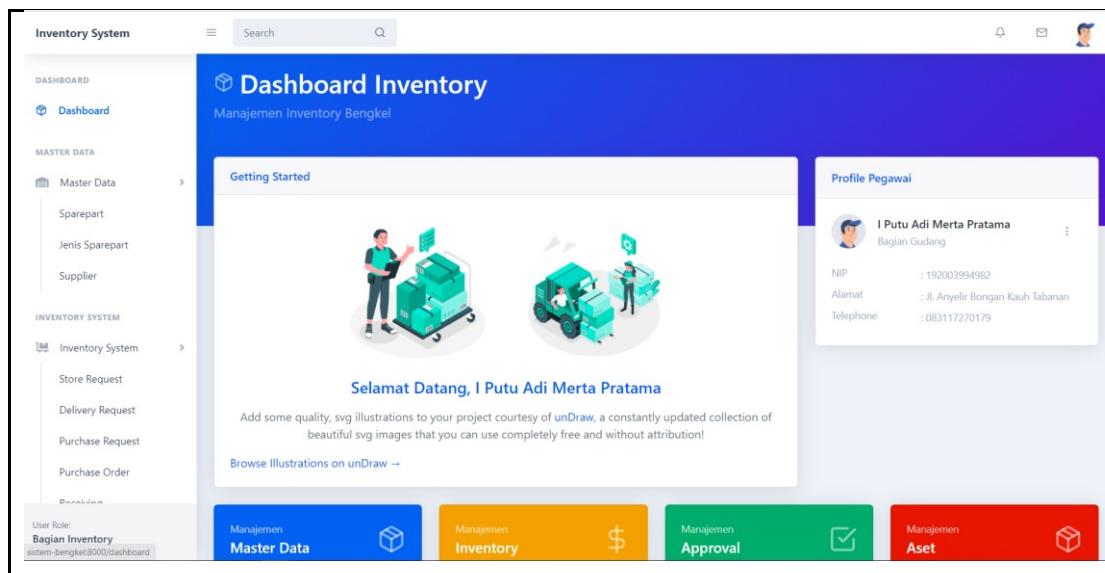
- 1) *CRUD purchase order*, proses ini dilakukan oleh bagian staff *purchasing* yang mengirimkan draft PO kepada bagian manager *purchasing*.
- 2) *Approve purchase order*, proses ini dilakukan setelah draft PO di *approve* oleh bagian manager *purchasing* dan bagian AP dan didapatkan data PO.
- 3) *Kirim purchase order*, proses ini dilakukan setelah data PO didapatkan dan bagian *supplier* mengirim barang.

3.5.3 Perancangan Antar Muka

Perancangan antar muka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun. Sehingga akan mempermudah dalam mengimplementasikan aplikasi serta akan memudahkan pembuatan aplikasi yang *user friendly*. Rancangan antar muka Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* adalah sebagai berikut.

3.5.3.1 Antar Muka Dashboard Modul *Inventory*

Perancangan antar muka *dashboard* modul *inventory* ini memberikan gambaran rancangan dari *dashboard* agar memudahkan pembuatan aplikasi pada modul *inventory*. Berikut merupakan rancangan antara muka *dashboard* modul *inventory*.



Gambar 3.11 Antar muka dashboard *inventory*

Gambar 3.11 merupakan rancangan antar muka *dashboard* dari modul *inventory* Sistem Informasi Manajemen Bengkel. Pada rancangan tersebut terdapat profil pegawai yang sedang melakukan *login* dan *card* yang berfungsi untuk navigasi dari setiap manajemen yang terdapat pada sub-sistem *inventory*.

3.5.3.2 Antar Muka Master Data Modul *Inventory*

Perancangan antar muka master data modul *inventory* ini memberikan gambaran rancangan antar muka dari master data agar memudahkan pembuatan aplikasi pada modul *inventory*. Berikut merupakan rancangan antara muka master data modul *inventory*.

The image consists of two vertically stacked screenshots of a web-based inventory management system. Both screenshots show the same overall layout with a sidebar on the left and a main content area on the right.

Sidebar (Left):

- DASHBOARD:** Includes a 'Dashboard' link.
- MASTER DATA:**
 - Sparepart:** This item is highlighted in blue, indicating it is the active module.
 - Jenis Sparepart
 - Supplier
- INVENTORY SYSTEM:**
 - Inventory System
 - Store Request
 - Delivery Request
 - Purchase Request
 - Purchase Order
- User Role: Bagian Inventory

Main Content Area (Right):

The title bar says 'Master Data Sparepart'. Below it, there are two tabs: 'Sparepart' (selected) and 'Tambah Sparepart' (Add Sparepart).

Screenshot 1 (Top): Shows a table titled 'List Sparepart' with columns: Kode, Sparepart, Jenis Sparepart, Stock, Merk, Satuan, Status, and Actions. The table contains 8 rows of sample data.

Kode	Sparepart	Jenis Sparepart	Stock	Merk	Satuan	Status	Actions
Airi Satou	Accountant	Tokyo	33	2008/11/28	\$162,700	Full-time	[Edit] [Delete]
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09	\$1,200,000	Full-time	[Edit] [Delete]
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12	\$86,000	Part-time	[Edit] [Delete]
Bradley Greer	Software Engineer	London	41	2012/10/13	\$132,000	Freelance	[Edit] [Delete]
Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07	\$206,850	Full-time	[Edit] [Delete]
Brielle Williamson	Integration Specialist	New York	61	2012/12/02	\$372,000	Full-time	[Edit] [Delete]
Bruno Nash	Software Engineer	London	38	2011/05/03	\$163,500	Contract	[Edit] [Delete]
Caesar Vance	Pre-Sales Support	New York	21	2011/12/12	\$106,450	Part-time	[Edit] [Delete]

Screenshot 2 (Bottom): Shows the 'Tambah Sparepart' (Add Sparepart) form. It has two tabs: 'Jenis Sparepart' and 'Tambah Sparepart'.

The 'Jenis Sparepart' tab shows a dropdown menu with 'Default select'.

The 'Tambah Sparepart' tab contains the following fields:

- Kode Sparepart (auto-filled as 00800123)
- Nama Sparepart (text input: Masukan Nama Sparepart)
- Jumlah Stock Min (text input: Masukan Jumlah Stock Sparepart)
- Merk (text input: Masukan Merk Sparepart)
- Satuan (dropdown: Default select)
- Status (dropdown: Default select)

A large blue 'Tambah' (Add) button is located at the bottom of the form.

Gambar 3.12 Antar muka master data *inventory*

Gambar 3.12 merupakan rancangan antar muka master data dari modul *inventory* Sistem Informasi Manajemen Bengkel. Pada rancangan tersebut terdapat rancangan dari tabel kelola master data dari sparepart kendaraan dan formulir tambah sparepart kendaraan untuk menambahkan sparepart kendaraan.

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Bab IV berisi pembahasan mengenai hasil dan analisis dari pengujian dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* yang telah dirancang sebelumnya. Hasil tersebut adalah hasil pengujian aplikasi dan analisa dari aplikasi yang telah diterapkan kembangkan.

4.1 Alat Pengujian

Alat pengujian adalah media yang digunakan untuk menguji Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting*. Alat pengujian aplikasi terbagi menjadi dua bagian yaitu perangkat keras dan perangkat lunak. Sub-bab berikut akan menjelaskan mengenai kebutuhan implementasi perangkat lunak dan perangkat keras yang dibutuhkan untuk menjalankan aplikasi.

4.1.1 Perangkat Keras

Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* telah dibuat diuji dan dijalankan pada laptop dengan spesifikasi yang mencukupi untuk menjalankan aplikasi tersebut. Perangkat keras yang dipergunakan pada saat pengujian sistem adalah Laptop Asus ROG GL553VD core i7 Nvidia Ram 8GB HDD 1TB SSD 128GB.

4.1.2 Perangkat Lunak

Kebutuhan implementasi perangkat lunak dibutuhkan untuk mengimplementasikan aplikasi yang sudah dibuat. Perangkat lunak yang digunakan untuk pengujian Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* adalah perangkat keras dengan spesifikasi perangkat lunak sistem operasi Windows 10 pro serta *browser* Chrome.

4.2 Antarmuka Sistem

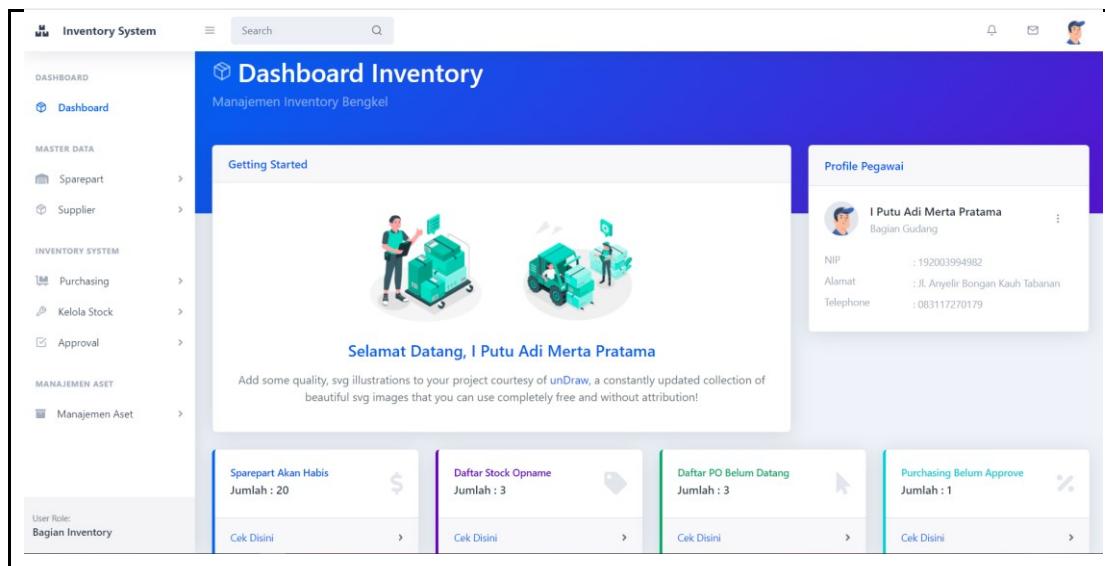
Tampilan yang dibuat pada bagian ini desesuaikan dengan rancangan yang telah dilakukan pada bab III. Berikut adalah tampilan antar muka dari Sistem Informasi Manajemen Bengkel Modul Kepegawaian, *Inventory* dan *Accounting* yang telah dikembangkan.

4.4.1 Antarmuka Modul Inventory & Purchasing

Antarmuka pada Sistem Informasi Manajemen Bengkel modul *Inventory* dan *purchasing* meliputi *dashboard inventory* dan *purchasing*, proses mengelola *stock* sparepart, manajemen supplier, manajemen master data, manajamen *purchasing* order, manajemen *receiving*, retur pembelian, *Approval system* dan manajemen aset. Berikut merupakan pemaparan dari antarmuka modul *inventory* dan *purchasing*.

4.4.1.1 Antarmuka Dashboard Inventory dan Purchasing

Dashboard merupakan halaman utama ketika pertama kali mengakses dari modul *inventory* dan *purchasing*. Berikut merupakan antarmuka dari *dashboard inventory* dan *purchasing*.



Gambar 4.1 Antarmuka Dashshboard Inventory & Purchasing

Gambar 4.1 merupakan antarmuka dari dashboard *inventory* dan *purchasing* yang telah diuji. Pada *dashboard* terdapat beberapa fitur diantaranya menampilkan jumlah sparepart yang akan habis, menampilkan daftar *sotck opname*, menampilkan daftar *purchasing order* yang belum datang dan data *approval* dari *purchasing order*. Pada *dashboard* juga akan menampilkan *profile* dari pegawai yang sedang mengakses.

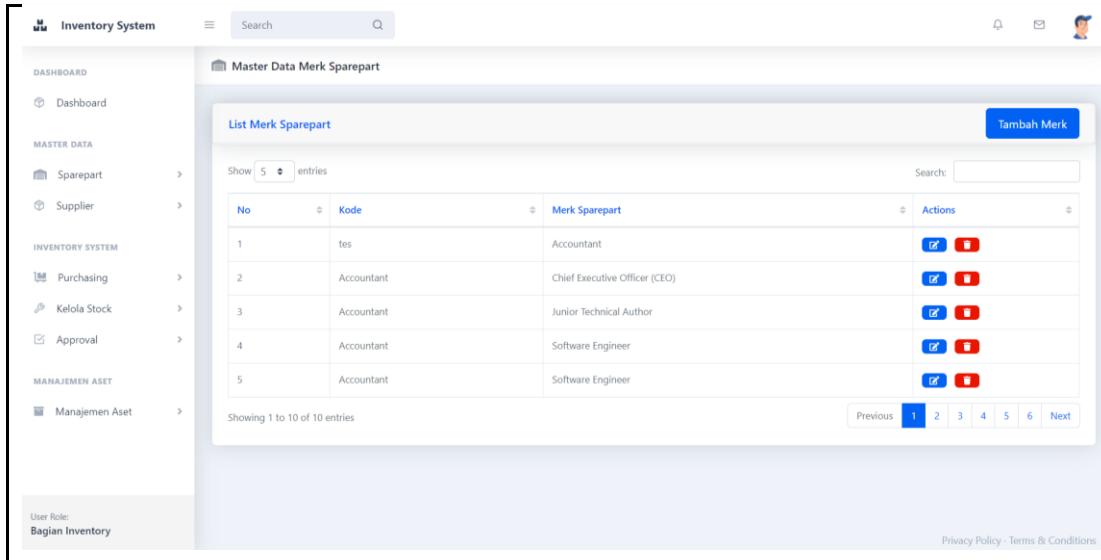
4.4.1.2 Antarmuka Manajemen Master Data Sparepart

Halaman antarmuka manajemen master data sparepart difungsikan untuk melakukan manajemen terhadap *stock* sparepart, merk sparepart dan jenis sparepart yang terdapat pada bengkel. Berikut merupakan antarmuka dari halaman manajemen master data sparepart.

No	Kode	Sparepart	Jenis Sparepart	Stock	Merk	Satuan	Actions
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]
1	SA00918	Ban Dalam	Ban	221	Tokyo	bah	[Edit] [Delete]

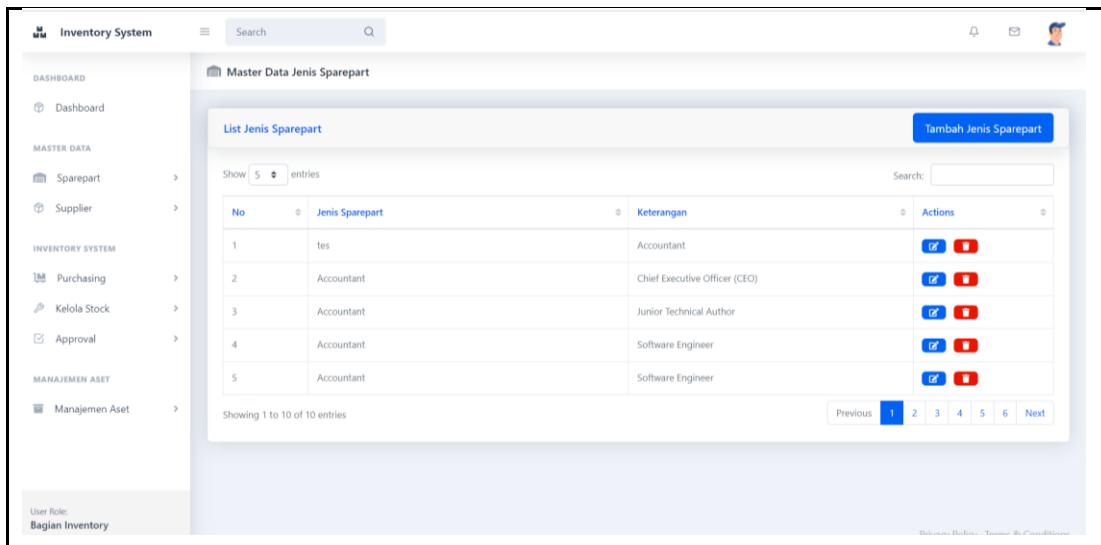
Gambar 4.2 Antarmuka Manajemen Master Data Sparepart

Gambar 4.2 merupakan halaman antarmuka dari manajemen data sparepart, terlihat pada gambar antarmuka ini berfungsi untuk mengelola data sparepart yang terdapat pada bengkel. Fitur manajemen data sparepart diantaranya kelola data sparepart, tambah data sparepart, edit data sparepart dan hapus data sparepart.



Gambar 4.3 Antarmuka Manajemen Master Data Merk Sparepart.

Gambar 4.3 merupakan antarmuka dari manajemen master data merk sparepart. Halaman ini berfungsi sebagai kelola data merk sparepart, seluruh merk yang terdapat pada sparepart akan dikelola pada halaman ini. Fitur yang terdapat pada halaman master data merk sparepart antara lain tambah merk sparepart, edit merk sparepart dan hapus merk sparepart.



Gambar 4.4 Antarmuka Manajemen Master Data Jenis Sparepart

Gambar 4.4 merupakan halaman antarmuka dari manajemen master data jenis sparepart. Halaman ini berfungsi memudahkan pengguna untuk melakukan kelola sparepart dan akan disesuaikan dengan jenis sparepart. Fitur yang terdapat pada halaman manajemen master data jenis sparepart yakni tambah jenis sparepart, edit jenis sparepart dan hapus jenis sparepart.

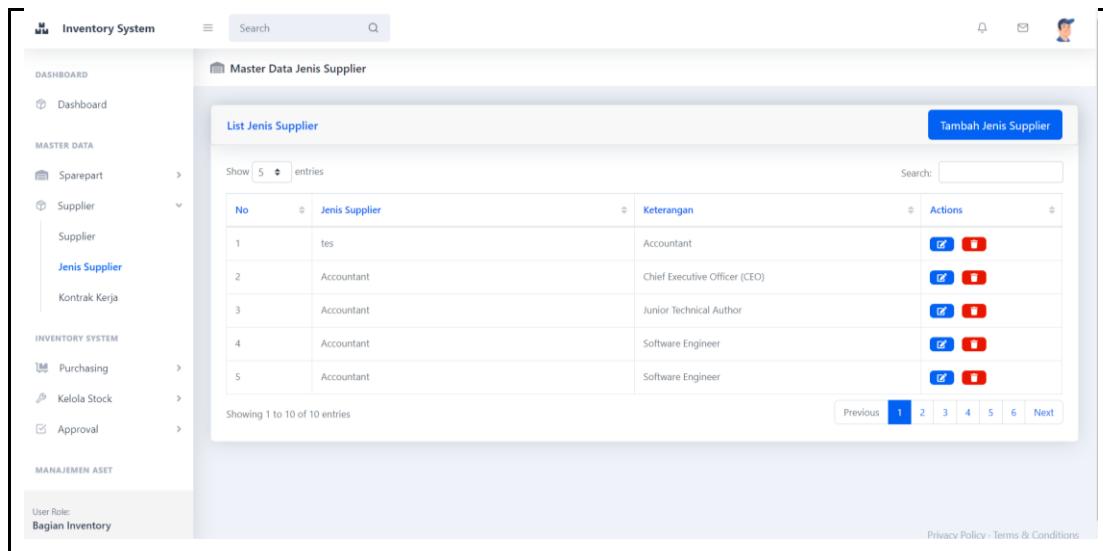
4.4.1.3 Antarmuka Manajemen Master Data Supplier

Halaman antarmuka manajemen master data supplier difungsikan untuk melakukan manajemen terhadap supplier yang dikontrak oleh bengkel. Master data yang dikelola antara lain master data supplier, master data jenis supplier dan master data kontrak kerja supplier. Berikut merupakan antarmuka dari manajemen master data supplier.

No	Kode	Nama Supplier	no HP	Tanggal Kontrak	Jenis Supplier	Status	Actions
1	SUP00192	Accountant	33	2008/11/28	\$162,700	Kontrak Aktif	[Edit, Delete]
2	SUP00192	Chief Executive Officer (CEO)	47	2009/10/09	\$1,200,000	Kontrak Aktif	[Edit, Delete]
3	SUP00192	Junior Technical Author	66	2009/01/12	\$86,000	Kontrak Aktif	[Edit, Delete]
4	SUP00192	Software Engineer	41	2012/10/13	\$132,000	Kontrak Aktif	[Edit, Delete]
5	SUP00192	Software Engineer	28	2011/06/07	\$206,850	Kontrak Habis	[Edit, Delete]

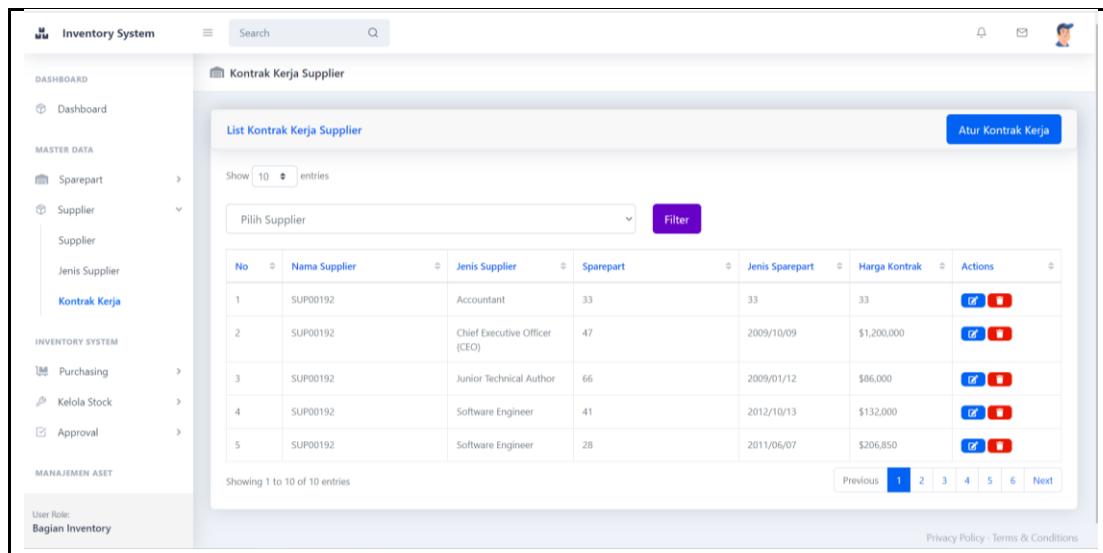
Gambar 4.5 Antarmuka Manajemen Master Data Supplier

Gambar 4.5 merupakan halaman antarmuka dari manajemen master data supplier. Manajemen master data supplier terdapat fitur kelola data supplier, tambah data supplier, edit data supplier dan hapus data supplier.



Gambar 4.6 Antarmuka Manajemen Master Data Jenis Supplier

Gambar 4.6 merupakan halaman antarmuka dari manajemen master data jenis supplier. Fitur yang terdapat pada proses manajemen master data jenis supplier antara lain tambah data jenis supplier, edit data jenis supplier dan hapus data jenis supplier.



Gambar 4.7 Antarmuka Manajemen Master Data Kontrak Kerja Supplier

Gambar 4.7 merupakan halaman antarmuka dari manajemen master data kontrak kerja supplier dimana pada halaman ini pengguna dapat melakukan

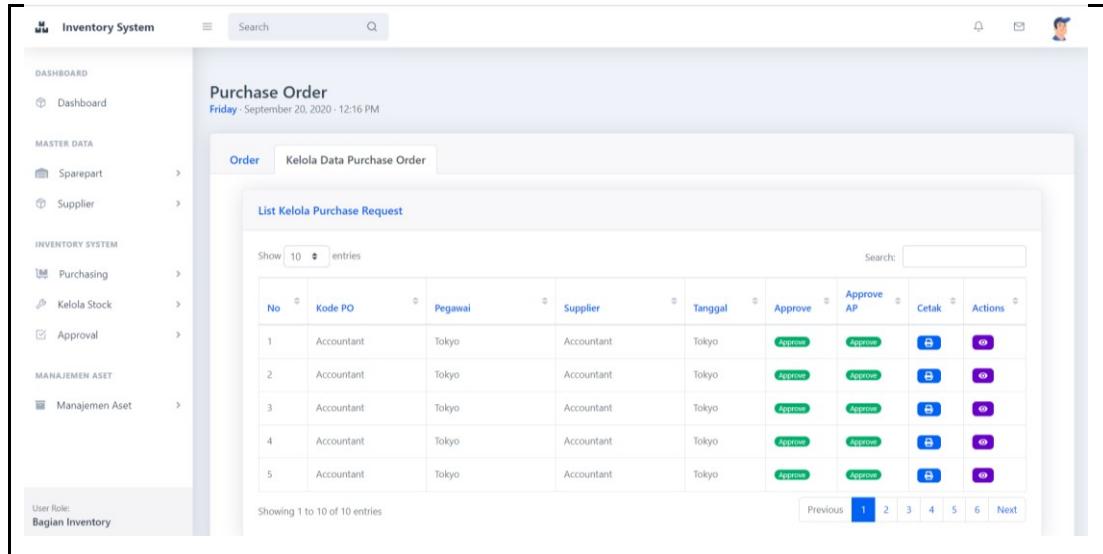
manajemen harga kontrak sparepart setiap suppliernya. Fitur yang terdapat pada manajemen master data kontrak kerja supplier antara lain tambah kontrak kerja, edit kontrak kerja dan hapus kontrak kerja selain itu pengguna dimudahkan dengan dapat melakukan filter setiap supplier.

4.4.1.4 Antarmuka Manajemen Purchasing Order

Halaman antarmuka manajemen *purchasing* order difungsikan untuk melakukan manajemen terhadap setiap pembelian sparepart ke supplier dari bengkel. Manajemen *purchasing* order terdapat 3 menu yakni *purchasing* order, *receiving* dan retur pembelian. Berikut merupakan halaman antarmuka dari manajemen *purchasing* order.

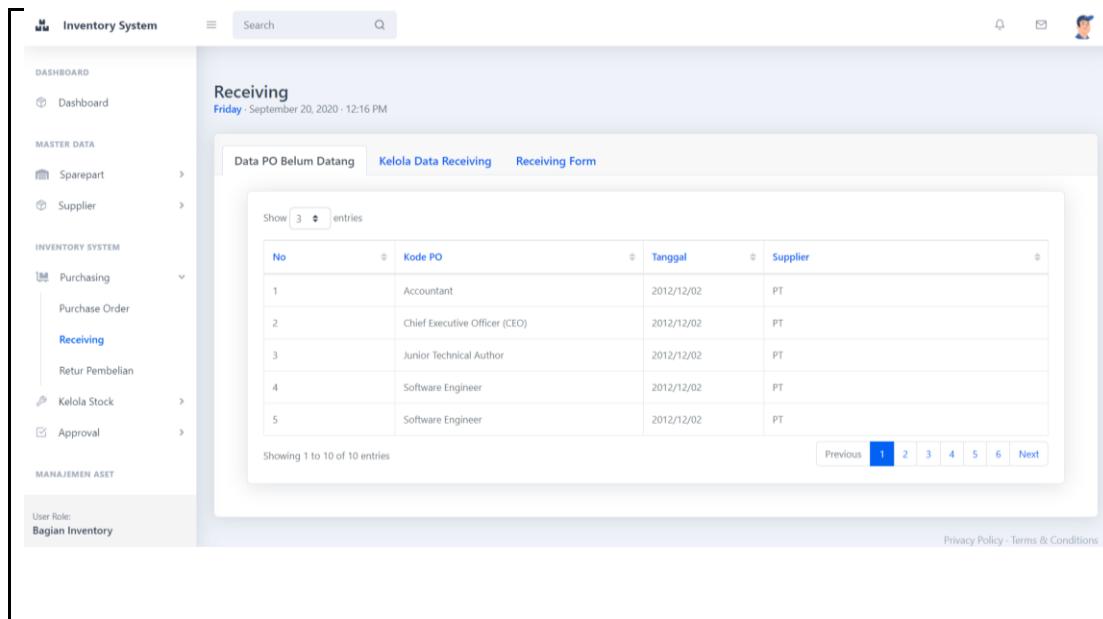
The screenshot shows the 'Purchase Order' section of the Inventory System. On the left, there's a sidebar with navigation links for Dashboard, Master Data (Sparepart, Supplier), Inventory System (Purchasing, Kelola Stock, Approval), and Asset Management (Manajemen Aset). The main area has a header 'Purchase Order' with the date 'Friday - September 20, 2020 - 12:16 PM'. Below it is a 'Purchase Order Form' containing fields for 'Kode PO' (Cari Kode Store Request), 'Pegawai' (I Putu Adi Merta Pratama), 'Akun ID' (Cari Akun ID), 'Pilih Supplier' (PT. Adi Jaya), and two 'Approve' buttons ('Belum Approve'). To the right, there are two tabs: 'Sparepart yang dipesan' (selected) and 'Konfirmasi Detail PO'. The 'Konfirmasi Purchase Request' section displays a table of purchase details:

No	Sparepart	Merk	Jumlah Pesanan	Satuan	Harga Total	Actions
1	Ban Dalam	Tokyo	55	buah	Harga	Hapus
2	Ban Luar	London	20	buah	Harga	Hapus
2	Ban Luar	London	1000	buah	Harga	Hapus
2	Ban Luar	London	100	buah	Harga	Hapus



Gambar 4.8 Antarmuka Manajemen Purchasing Order

Gambar 4.8 merupakan halaman antarmuka dari manajemen *purchasing* order. Fitur yang terdapat pada manajemen *purchasing* order antara lain melakukan *purchasing* dengan *form* atau tambah order dan kelola data *purchase* order. Pada *purchasing* order sparepart yang akan dipesan ke supplier akan tambahkan pada *form* dan dikonfirmasi melalui sub-menu konfirmasi pembelian.



The figure consists of two vertically stacked screenshots of a web-based inventory management system. Both screenshots have a header bar with the title 'Inventory System', a search bar, and a user profile icon.

Left Sidebar (Common to both screenshots):

- DASHBOARD
- MASTER DATA
 - Sparepart
 - Supplier
- INVENTORY SYSTEM
 - Purchasing
 - Purchase Order
 - Receiving** (highlighted in blue)
 - Retur Pembelian
 - Kelola Stock
 - Approval
- MANAJEMEN ASET

User Role: Bagian Inventory

Top Screenshot (Receiving List):

Receiving

Friday - September 20, 2020 - 12:16 PM

Data PO Belum Datang Kelola Data Receiving Receiving Form

List Kelola Data Receiving

No	Kode Receive	Pegawai	Kode PO	Kode DO	Perusahaan	Tanggal	Actions
1	RE112010	I Putu Adi Merta Pratama	PO102012	DO120102	PT. Adi Jaya	29/08/2020	
2	RE112010	I Putu Adi Merta Pratama	PO102012	DO120102	PT. Adi Jaya	29/08/2020	
3	RE112010	I Putu Adi Merta Pratama	PO102012	DO120102	PT. Adi Jaya	29/08/2020	
4	RE112010	I Putu Adi Merta Pratama	PO102012	DO120102	PT. Adi Jaya	29/08/2020	
5	RE112010	I Putu Adi Merta Pratama	PO102012	DO120102	PT. Adi Jaya	29/08/2020	

Show 10 entries Search:

Showing 1 to 10 of 10 entries Previous **1** 2 3 4 5 6 Next Privacy Policy · Terms & Conditions

User Role: Bagian Inventory

Bottom Screenshot (Receiving Form):

Receiving Form

Kode PO (Kode Otomatis akan terisi) Akun ID Pilih Supplier

Kode PO Cari Akun ID Q PT. Adi Jaya

Pegawai (Nama otomatis akan terisi pada form) Approve Approve AP

I Putu Adi Merta Pratama Belum Approve Belum Approve

Keterangan Masukan Keterangan

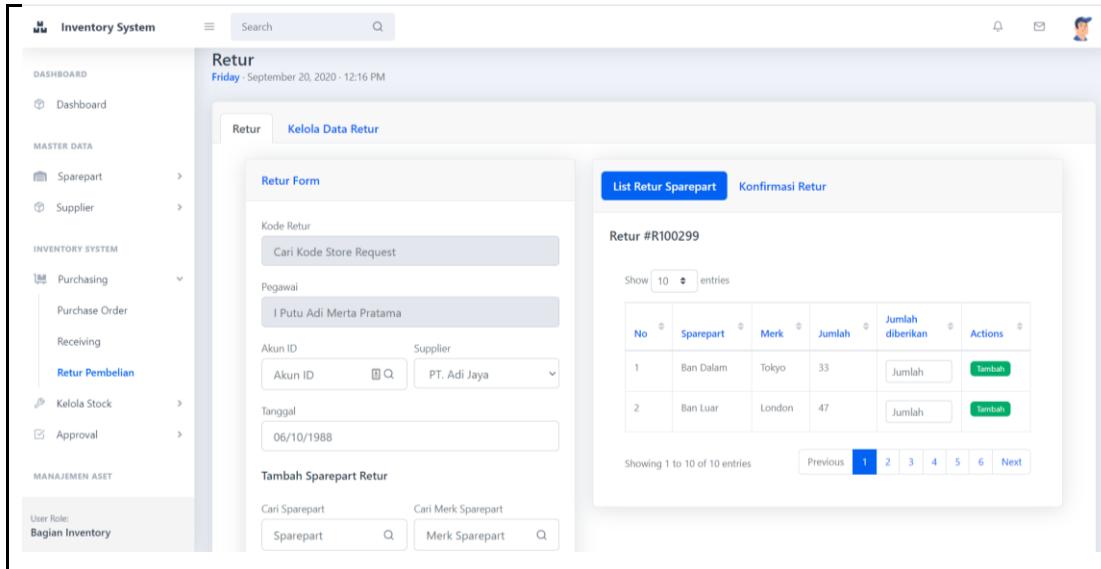
Tanggal PR Cari Kode PR No PR

06/10/1988 Q

Buttons: Simpan (blue), Batal (red)

Gambar 4.9 Antarmuka Manajemen Receiving

Gambar 4.9 merupakan halaman antarmuka dari manajemen *receiving*. Manajemen *receiving* ditunjukkan untuk mengelola sparepart yang didatang dari pesanan oleh supplier. Fitur yang terdapat pada manajemen *receiving* yakni data *purchasing* order yang belum datang, kelola data *receiving* dan *receiving* formulir.

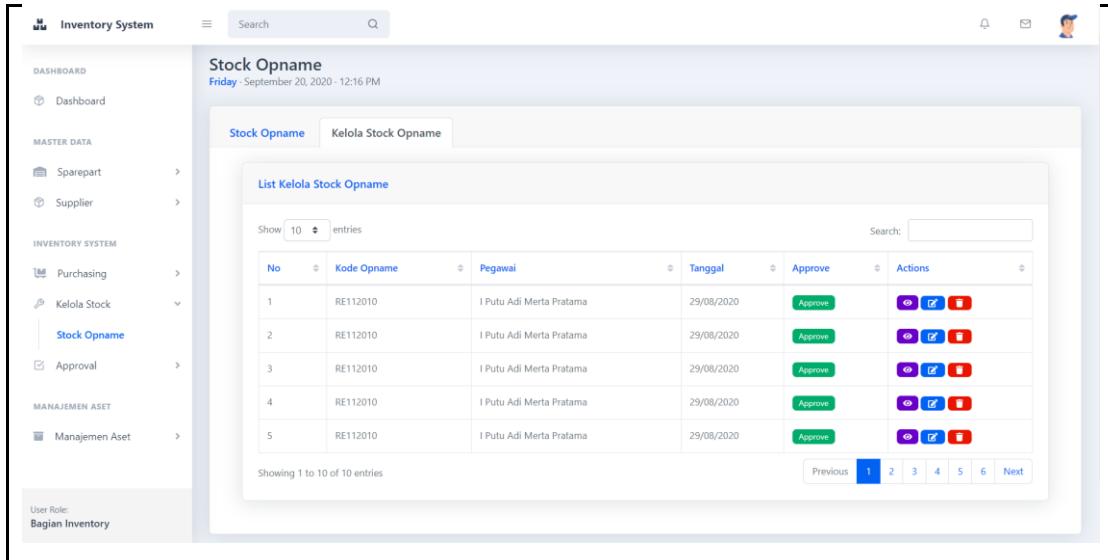


Gambar 4.10 Antarmuka Manajemen Retur Pembelian

Gambar 4.10 merupakan antarmuka dari manajemen retur pembelian, dimana retur pembelian difungsikan untuk pengembalian barang rusak saat barang datang. Fitur yang teradapat pada manajemen retur pembelian yakni tambah retur pembelian, konfirmasi retur pembelian dan kelola data retur pembelian.

4.4.1.5 Antarmuka Manajemen Kelola Stock Sparepart

Halaman antarmuka manajemen kelola *stock* sparepart difungsikan untuk melakukan manajemen terhadap pengelolaan *stock* sparepart *real* dan *stock* sparepart pada sistem. Berikut merupakan halaman antarmuka dari manajemen kelola *stock* sparepart.

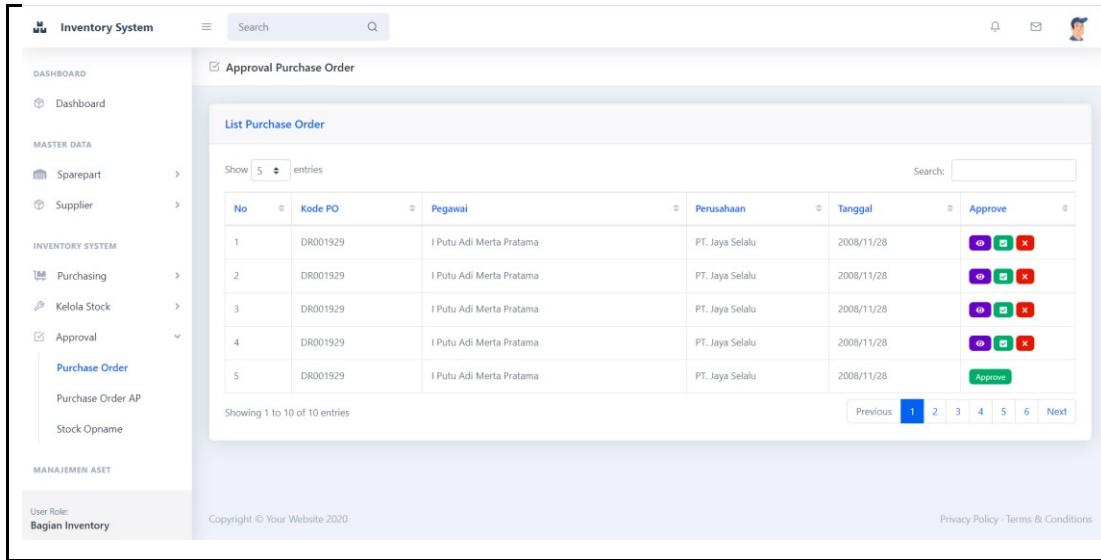


Gambar 4.11 Antarmuka Manajemen Kelola Stock Sparepart

Gambar 4.11 merupakan halaman antarmuka dari manajemen kelola *stock* sparepart dimana kelola *stock* sparepart dilakukan dengan pengecekan secara *real* dan pengecekan pada sistem. Fitur yang terdapat pada kelola *stock* sparepart yakni kelola *stock* opname dan formulir *stock* opname.

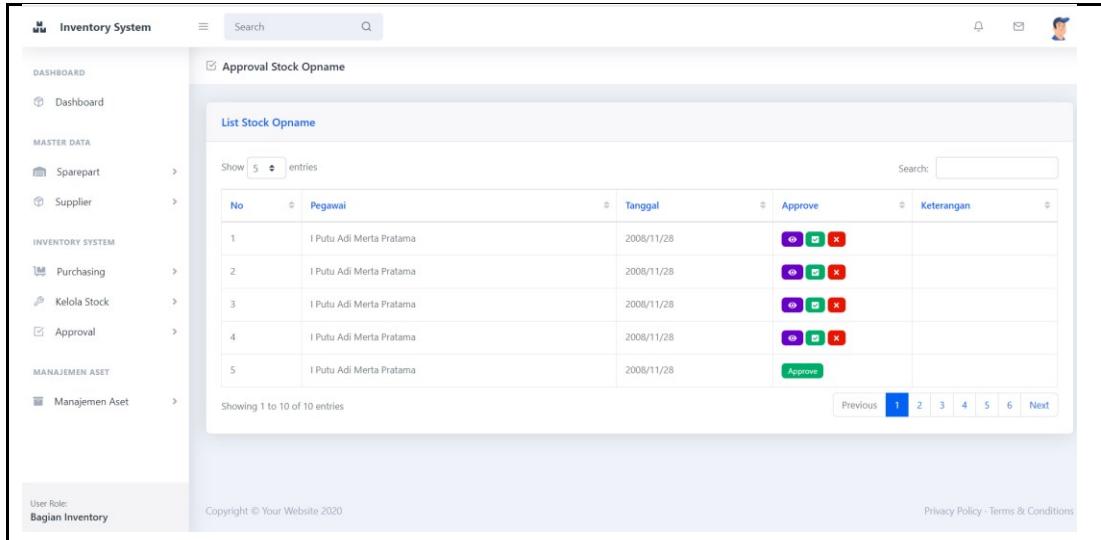
4.4.1.6 Antarmuka Manajemen Approval System

Halaman antarmuka manajemen *approval system* difungsikan untuk *role* manager Bengkel melakukan *approval* terhadap pembelian sparepart dan kelola *stock* sparepart. Berikut merupakan halaman antarmuka dari manajemen *approval system*.



Gambar 4.12 Antarmuka Manajemen Approval Purchase Order

Gambar 4.12 merupakan halaman antarmuka dari manajemen *approval purchasing* order, dimana pada halaman ini hanya dapat diakses oleh *role* manager bengkel. Fitur yang terdapat pada manajemen *approval* yakni kelola list *purchasing* order, detail *purchasing* order, *approval purchasing* order dan not *approval* atau tidak disetujui.



Gambar 4.13 Antarmuka Manajemen Approval Stock Opname

Gambar 4.13 merupakan halaman antarmuka dari manajemen *approval stock opname* atau kelola *stock sparepart*, setelah bagian *inventory* mengelola *stock real* dan sistem selanjutnya manajer bengkel akan melakukan *approval* dari kegiatan tersebut. Fitur yang terdapat pada manajemen *approval stock opname* yakni lihat detail kelola *stock*, *approval* kelola *stock* dan not *approval* atau tidak disetujui.

4.4.1.7 Antarmuka Manajemen Aset Bengkel

Halaman antarmuka manajemen aset bengkel difungsikan untuk melakukan kelola aset yang terdapat pada bengkel yang berupa aset berwujud dan tidak berwujud. Berikut merupakan halaman antarmuka dari manajemen aset bengkel.

No	Jenis Aset	Kode	Nama Aset	Merk Aset	Harga	Tanggal Masuk	Status	Actions
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	
1	SA00918	Ban Dalam	Ban	221	221	Tokyo	Aktif	

Gambar 4.14 Antarmuka Manajemen Aset Bengkel

Gambar 4.14 merupakan halaman antarmuka dari manajemen aset bengkel. Fitur yang terdapat pada manajemen aset bengkel antara lain tambah aset, lihat detail aset, edit aset, hapus aset dan cetak aset.

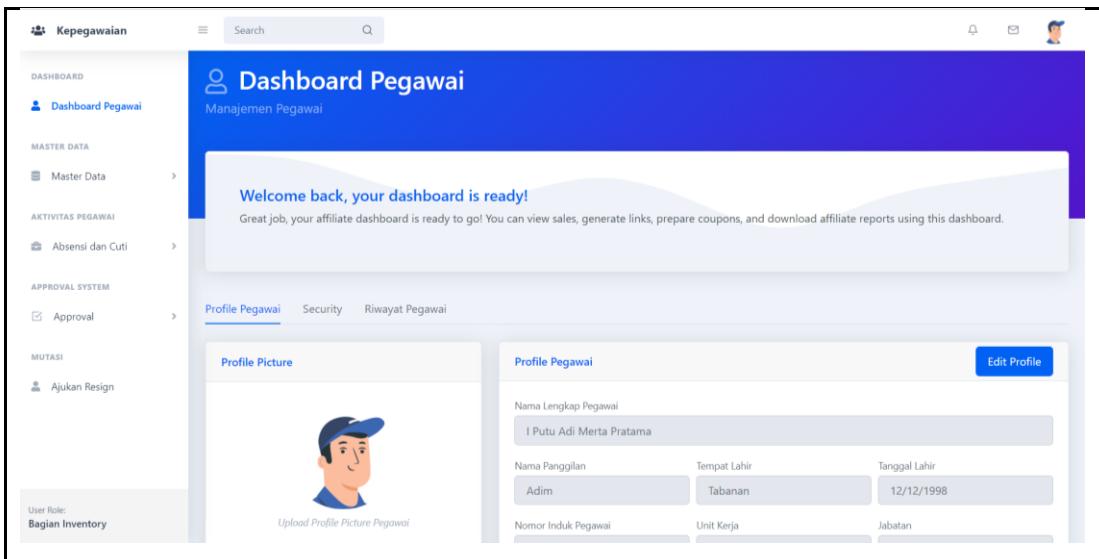
4.4.2 Antarmuka Modul Kepegawaian

Antarmuka pada Sistem Informasi Manajemen Bengkel modul Kepegawaian meliputi *dashboard* pegawai, manajemen master data, manajemen absensi dan cuti,

approval system dan pengajuan resign. Berikut merupakan pemaparan dari antarmuka modul kepegawaian.

4.4.2.1 Antarmuka Dashboard Pegawai

Dashboard merupakan halaman utama ketika pertama kali mengakses dari modul kepegawaian yang berisi informasi pegawai. Berikut merupakan antarmuka dari *dashboard* kepegawaian.



Gambar 4.15 Antarmuka Dashboard Pegawai

Gambar 4.15 merupakan halaman antarmuka dari *dashboard* pegawai, dimana pada *dashboard* pegawai akan menampilkan *profile* pegawai yang sedang mengakses sistem. Fitur yang terdapat pada *dashboard* pegawai yakni *profile* pegawai, *security* yang digunakan untuk mengatur *username* dan *password* serta riwayat pegawai yang terdiri dari riwayat cuti.

4.4.2.2 Antarmuka Manajemen Master Data Kepegawaian

Halaman antarmuka manajemen master data difungsikan untuk melakukan manajemen terhadap data pegawai, data jabatan, data unit kerja dan data jenis cuti yang

terdapat pada bengkel. Berikut merupakan antarmuka dari halaman manajemen master data kepegawaian.

No	NIP	Nama Pegawai	Unit Kerja	Jabatan	Jenis Kelamin	No Telephone	Status	Actions
1	1990299182	I Putu Adi Merta Pratama	Mechanic	Pegawai	Laki-Laki	081246602400	Aktif	[Edit] [Delete]
2	1990299182	I Putu Adi Merta Pratama	Mechanic	Pegawai	Laki-Laki	081246602400	Aktif	[Edit] [Delete]
3	1990299182	I Putu Adi Merta Pratama	Mechanic	Pegawai	Laki-Laki	081246602400	Aktif	[Edit] [Delete]
4	1990299182	I Putu Adi Merta Pratama	Mechanic	Pegawai	Laki-Laki	081246602400	Aktif	[Edit] [Delete]
5	1990299182	I Putu Adi Merta Pratama	Mechanic	Pegawai	Laki-Laki	081246602400	Aktif	[Edit] [Delete]
6	1990299182	I Putu Adi Merta Pratama	Mechanic	Pegawai	Laki-Laki	081246602400	Aktif	[Edit] [Delete]

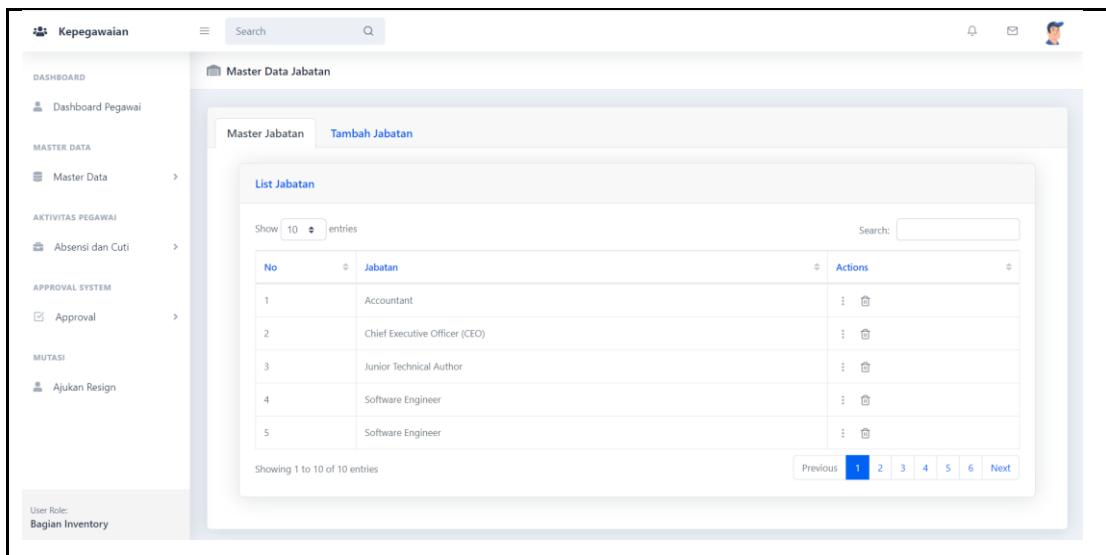
Gambar 4.16 Antarmuka Manajemen Master Data pegawai

Gambar 4.16 merupakan antarmuka dari manajemen master data pegawai, dimana pada manajemen ini dapat dilakukan oleh *user role* HRD untuk menambahkan data pegawai baru atau edit data pegawai.

No	Unit Kerja	Actions
1	Accountant	[Edit] [Delete]
2	Chief Executive Officer (CEO)	[Edit] [Delete]
3	Junior Technical Author	[Edit] [Delete]
4	Software Engineer	[Edit] [Delete]
5	Software Engineer	[Edit] [Delete]

Gambar 4.17 Antarmuka Manajemen Master Data Unit Kerja

Gambar 4.17 merupakan halaman antarmuka manajemen master data unit kerja. Unit kerja yang dimaksud yakni struktur kerja seperti *accounting*, hrd dll. Fitur yang terdapat pada manajemen master data unit kerja yakni tambah data unit kerja, edit data unit kerja dan hapus data unit kerja.



Gambar 4.18 Antarmuka Manajemen Master Data Jabatan

Gambar 4.18 merupakan halaman antarmuka dari manajemen master data jabatan. Fitur yang terdapat pada manajemen master data jabatan yakni tambah data jabatan, edit data jabatan dan hapus data jabatan.

4.3 Pengujian Sistem

Tahap ini merupakan tahap untuk melakukan pengujian sistem. Pengujian sistem ini bertujuan untuk menemukan kesalahan terjadi apabila terjadi kesalahan input pada data. Teknik pengujian yang dilakukan adalah Teknik *black-box* testing. *Black box* testing dilakukan pengujian yang didasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi, dan kesesuaian alur fungsi dengan bisnis proses yang diinginkan oleh customer. *Black-box* testing ini lebih menguji ke tampilan luar (*interface*) dari suatu aplikasi agar mudah digunakan oleh pengguna.

DAFTAR PUSTAKA

- Danang N. (2016) "RANCANG BANGUN APLIKASI LOCATION BASED SERVICE UNTUK PENCARIAN RUTE TERDEKAT BENGKEL RESMI HONDA PADA WILAYAH SURABAYA BERBASIS ANDROID."
- Lasmaya.M (2016) " PENGARUH SISTEM INFORMASI SDM, KOMPETENSI DAN DISIPLIN KERJA TERHADAP KINERJA KARYAWAN."
- Jogyanto, H. (2005). Analisis dan Desain Sistem Informasi.
- Kendall, K. (2006). Analisis dan Perancangan Sistem.
- Kurniawan, H. (2015). "Perancangan Sistem Informasi Bengkel Mobil Berbasis Web" Nasional Sistem & Informatika 636-641.
- Welim, Y. Y. (2015). "PENGEMBANGAN SISTEM INFORMASI SERVICE KENDARAAN PADA BENGKEL KFMP" SIMETRIS 6: 17-26.
- Meri Audrilia, A. B. (2020, March). Perancangan Sistem Informasi Manajemen Bengkel Berbasis Web (Studi Kasus: Bengkel Anugrah). JURNAL MADANI: Ilmu Pengetahuan, Teknologi, dan Humaniora, Vol. 3, No. 1, 1-12.
- Nurmalasari, R.A. (2019, September). Rancang Bangun Sistem Informasi Akuntansi Laporan Laba Rugi Berbasis Web pada PT. United Tractors Pontianak. Jurnal Sains dan Manajemen, Vol. 7, No. 2.
- Randi Rizal, A.R. (2019, March). RESTful Web Service untuk Integrasi Sistem Akademik dan Perpustakaan Universitas Perjuangan. Jurnal Ilmiah Informatika (JIF), Vol. 7, No. 01.
- Mursalim Tonggiroh, N.I (2017, Februari). Sistem Informasi Kepegawaian pada Kantor Dewan Teknologi Informasi dan Komunikasi Provinsi Papua Berbasis Web. Jurnal Ilmiah Teknik dan Informatika, Vol. 2, No. 1.
- Titin Purnamasari (2013). Pembangunan Sistem informasi Pengolahan Data Pegawai dan Penggajian pada Unit Pelaksana Teknis Taman Kanak-Kanak dan Sekolah Dasar Kecamatan Pringkuku. Jurnal Speed: Sentra Penelitian Engineering dan Edukasi, Vol. 5, No. 2.

Hendra Agusvianto (2017). Sistem Informasi Inventori Gudang untuk Mengontrol Persediaan Barang pada Gudang Studi Kasus: PT. Alaisys Sidoarjo. Jurnal JIEET: Journal Information Engineering and Educational Technology, Vol. 01, No. 01.

Agus Heryanto, H.F (2014, September). Rancang Bangun Sistem Infromasi Inventory Barang Berbasis Web Studi Kasus di PT. Infinetworks Global Jakarta. Jurnal Sisfotek Global, Vol. 4, No. 02.