

SISTEM INFORMASI *INVENTORY* PADA

CV. GRATIA MOTORINDO

JAKARTA



SKRIPSI

Diajukan untuk memenuhi salah satu syarat kelulusan Program Sarjana

SITI WIYARTI

11145368

Program Studi Sistem Informasi

Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri

Jakarta

2019

PERSEMBAHAN

Dengan mengucap puji syukur kepada Allah S.W.T, skripsi ini kupersembahkan untuk :

1. Bapak Suwirto dan Ibu Mimin Aminah tercinta yang telah membesarkan aku dan selalu membimbing, mendukung, memotivasi, memberi apa yang terbaik bagiku serta selalu mendoakan aku untuk meraih kesuksesanku.
2. Adikku (Willy Elang Wijaya) yang telah mendukung dan memberiku semangat
3. Terima kasih kepada Bapak Agus Junaidi selaku dosen pembimbing dan seluruh karyawan di lingkungan STMIK Nusa Mandiri Jakarta.
4. Terima kasih kepada Bapak Darwin Wijaya selaku direktur utama dan para staff gratia motorindo atas bantuannya selama riset.
5. Prihan Firmanullah dan semua teman-teman kelas 11.AC.25 terima kasih atas dukungan dan bantuannya.

SURAT PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Siti Wiyarti
Nim : 1145268
Perguruan tinggi : STMIK Nusa Mandiri Jakarta

Dengan ini menyatakan bahwa skripsi yang telah saya buat dengan judul **“Sistem entory Pada CV.Gratia Motorindo Jakarta”**, adalah asli (orisinil) atau tidak plagiat (menjiplak) dan belum pernah diterbitkan/dipublikasikan dimanapun dan dalam bentuk apapun.

Demikianlah surat pernyataan ini saya buat denag sebenar-benarnya tanpa ada paksaan dari pihak manapun juga. Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa skripsi yang telah saya buat adalah hasil karya milik seseorang atau badan tertentu, saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari **STMIK Nusa Mandiri Jakarta** dicabut/dibatalkan.

Dibuat di : Jakarta

Pada tanggal : Agustus 2019

Yang menyatakan,

Siti Wiyarti

SURAT PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Siti Wiyarti

Nim : 1145268

Perguruan tinggi : STMIK Nusa Mandiri Jakarta

Dengan ini menyatakan bahwa skripsi yang telah saya buat dengan judul **“Sistem Informasi Inventory Pada CV.Gratia Motorindo Jakarta”**, adalah asli (orisinil) atau tidak plagiat (menjiplak) dan belum pernah diterbitkan/dipublikasikan dimanapun dan dalam bentuk apapun.

Demikianlah surat pernyataan ini saya buat dengan sebenar-benarnya tanpa ada paksaan dari pihak manapun juga. Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa skripsi yang telah saya buat adalah hasil karya milik seseorang atau badan tertentu, saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari **STMIK Nusa Mandiri Jakarta** dicabut/dibatalkan.

Dibuat di : Jakarta

Pada tanggal : 30 Agustus 2019

Yang menyatakan,

Siti Wiyarti

PERSETUJUAN DAN PENGESAHAN SKRIPSI

Skripsi ini diajukan oleh :

Nama : Siti Wiyarti
Nim : 1145368
Program Studi : Sistem Informasi
Jenjang : Strata Satu (S1)
Judul Skripsi : Sistem Informasi Inventory Pada CV.Gratia

Motorindo Jakarta

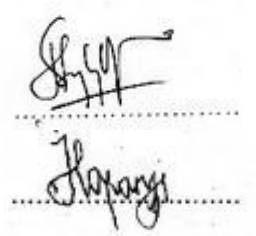
Telah dipertahankan pada priode II-2018 dihadapan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh sarjana ilmu komputer (S.KOM) pada Program Sarjana Program Studi Sistem Informatika di STMIK Nusa Mandiri Jakarta.

Jakarta, 30 Agustus 2019

PEMBIMBING SKRIPSI

Dosem Pembimbing I : Agus Junaidi M.KOM

Dosem Pembimbing II : Popon Handayani M.KOM



DEWAN PENGUJI

Penguji I : Ahmad Ishaq, M.Kom

Penguji II : Ahmad Al Kaafi, M.Kom



PANDUAN PENGGUNAAN HAK CIPTA

Skripsi sarjana yang berjudul **“Sistem Informasi Inventory Pada CV.Gratia Motorindo Jakarta”** adalah hasil karya tulis asli SITI WIYARTI dan bukan hasil terbitan sehingga peredaran karya tulis hanya berlaku dilingkungan akademik saja, serta memiliki hak cipta. Oleh karena itu, dilarang keras untuk menggandakan baik sebagian maupun seluruhnya karya tulis ini, tanpa seizin penulis.

Referensi kepustakaan diperkenankan untuk dicatat tetapi pengutipan atau peringkasan isi tulisan hanya dapat dilakukan dengan seizin penulis dan di sertai ketentuan pengutipan secara ilmiah dengan menyebutkan sumbernya.

Untuk keperluan perizinan pada pemilik dapat menghubungi informasi yang tertera dibawah ini :

Nama : SITI WIYARTI

ALAMAT : Jl,Jelambar Timur Rt.11/09 Jakarta Barat

No. Telp : 087885050519

E-mail : sitiwiyarti_13@yahoo.com



KATA PENGANTAR

Alhamdulillah, dengan mengucapkan puji syukur kehadiran Allah SWT, yang telah melimpahkan rahmat dan karunia-Nya, sehingga pada akhirnya penulis dapat menyelesaikan makalah skripsi ini dengan baik. Dimana Skripsi ini penulis sajikan dalam bentuk makalah yang sederhana. Adapun judul Skripsi, yang penulis ambil sebagai beriku, **“Sistem Informasi Inventory Pada CV.Gratia Motorindo Jakarta”**, Tujuan penulisan Skripsi ini dibuat sebagai salah satu syarat kelulusan program sarjana STMIK Nusa Mandiri Jakarta. Sebagai bahan penulisan diambil berdasarkan hasil penelitian (eksperimen), observasi dan beberapa sumber literatur yang mendukung penulisan ini. Penulisan menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan skripsi ini tidak akan lancer, oleh karena itu pada kesempatan ini, izinkanlah penulis menyampaikan ucapan terima kasih kepada :

1. Ketua STMIK Nusa Mandiri Jakarta.
2. Ketua Program Studi Sistem Informasi STMIK Nusa Mandiri Jakarta.
3. Bapak Agus Junaidi, selaku Dosen Pembimbing I Skripsi.
4. Bapak/ibu dosen Manajemen Informatika STMIK Nusa Mandiri Jakarta yang telah memberikan penulis dengan semua bahan yang diperlukan.
5. Orang tua tercinta yang telah memberikan dukungan moral maupun spiritual.
6. Terima kasih kepada Bapak Darwin Wijaya selaku direktur utama dan para staff gratia motorindo atas bantuannya selama riset.
7. Rekan-rekan mahasiswa kelas 11.7AC.25.

Serta semua pihak yang terlalu banyak untuk disebut satu persatu sehingga terwujudnya penulisan ini. Penulis menyadari bahwa penulisan skripsi ini masih jauh sekali dari sempurna, untuk ini penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan dimasa yang akan datang.

Akhir kata semoga skripsi ini dapat berguna bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya.

Jakarta, Agustus 2019

Penulis



Siti Wiyarti

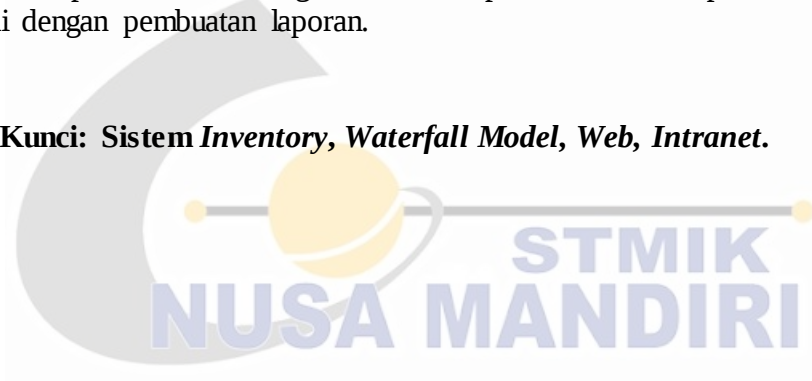
ABSTRAK

Siti Wiyarti (11145368), Sistem Informasi Inventory Pada CV.Gratia Motorindo

Jakarta

Sistem *inventory* merupakan suatu sistem yang digunakan untuk mencatat proses keluar masuknya barang dari tempat persediaan dan semuanya di simpan ke dalam database, sehingga pengguna bisa tahu, mengawasi dan mengontrol persediaan barang dengan mudah. Dari hal pencatatan tersebut, pengguna juga bisa menarik laporan transaksi keluar maupun masuk. Kebanyakan sistem *inventory* di perusahaan masih menggunakan prosedur atau sistem manual sehingga tidak efektif dan efisien dalam pelaksanaannya. Metode perancangan sistem yang digunakan dalam penelitian ini adalah Waterfall Model yang terdiri dari lima tahapan yaitu, Analisa kebutuhan Sistem, Perancangan, Pengkodean, Percobaan dan Pemeliharaan. Hasil dari penelitian ini adalah terciptanya aplikasi sistem informasi *inventory* berbasis web dengan jaringan internal atau *intranet* yang dapat mempermudah proses transaksi pencatatan penerimaan barang masuk dan proses transaksi pembelian dari *supplier*, sampai dengan pembuatan laporan.

Kata Kunci: *Sistem Inventory, Waterfall Model, Web, Intranet.*



ABSTRACT

Siti Wiyarti (11145368), Inventory Information System at CV. Gratia Motorindo Jakarta

Inventory system is a system used to record the process of entry and exit of goods from the inventory and everything is stored in a database, so users can know, monitor and control the supply of goods easily. From this recording, the user can also pull out or incoming transaction reports. Most inventory systems in companies still use manual procedures or systems so that they are not effective and efficient in their implementation. The system design method used in this study is the Waterfall Model which consists of five stages, namely, Analysis of system requirements, Design, Coding, Experiment and Maintenance. The results of this study are the creation of a web-based inventory information system application with an internal network or intranet that can simplify the process of recording the receipt of incoming goods and the process of purchasing transactions from suppliers, up to making reports.

Keywords: Inventory System, Waterfall Model, Web, Intranet.





DAFTAR ISI

LEMBAR JUDUL SKRIPSI.....	i
LEMBAR PERSEMBAHAN	ii
LEMBAR PERNYATAAN KEASLIAN SKRIPSI.....	iii
LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH.....	iv
LEMBAR PERSETUJUAN DAN PENGESAHAN SKRIPSI.....	v
LEMBAR PANDUAN PENGGUNAAN HAK CIPTA.....	vi
Kata Pengantar	vii
Abstrak.....	ix
Daftar Isi.....	xi
Daftar Simbol.....	xiii
Daftar Gambar	xv
Daftar Tabel.....	xvii
Daftar Lampiran	xviii
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Identifikasi Permasalahan.....	2
1.3. Perumusan Masalah.....	2
1.4. Maksud dan Tujuan.....	3
1.5. Metode Penelitian.....	4
1.5.1. Teknik Pengumpulan Data.....	4
A. Observasi.....	4
B. Wawancara.....	4
C. Studi Pustaka	4
D. Model Pengembangan Sistem.....	4
E. Analisa Kebutuhan Sistem.....	4
F. Desain.....	5
G. Code Generation.....	6
H. Testing.....	6
I. Support.....	6
1.6. Ruang Lingkup	6
BAB II LANDASAN TEORI.....	7
2.1. Tinjauan Pustaka	7
2.2. Penelitian Terkait.....	14

BAB III ANALISA SISTEM BERJALAN 15

3.1. Tinjauan Institusi/Perusahaan 16

3.1.1. Sejarah Institusi/Perusahaan 16

3.1.2. Struktur Organisasi dan Fungsi 16

3.2. Prosedur Sistem Berjalan 19

3.3. Spesifikasi Dokumen sistem Berjalan 21

BAB IV RANCANGAN SISTEM DAN PROGRAM USULAN 22

4.1. Analisa Kebutuhan Software 22

4.2. Desain 33

4.2.1. *Database* 33

4.2.2. *Software Architecture* 45

4.2.3. *User Interface* 47

4.3. *Code Generation* 107

4.4. *Testing* 103

4.5. *Support* 109

4.5.1. Publikasi Web * 109

4.5.2. Spesifikasi *Hardware* dan *Software* 109

4.6. Spesifikasi Dokumen Sistem Usulan 110

BAB V PENUTUP 112

5.1. Kesimpulan 112

5.2. Saran 113

DAFTAR PUSTAKA

DAFTAR RIWAYAT HIDUP

LEMBAR KONSULTASI BIMBINGAN

SURAT KETERANGAN RISET

LAMPIRAN

Lampiran A. Dokumen Sistem Berjalan

Lampiran B. Dokumen Sistem Usulan



DAFTAR SIMBOL

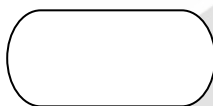
a. Simbol *UML*

sesuaikan dengan symbol UML yang dipakai dalam penulisan skripsi sesuai dengan referensi yang digunakan.

b. Simbol *ERD*

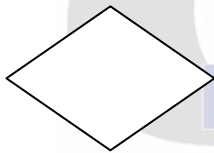
sesuaikan dengan symbol ERD yang dipakai dalam penulisan skripsi sesuai dengan referensi yang digunakan.

c. Simbol *Flowchart*



TERMINAL

Digunakan untuk menggambarkan awal dan akhir dari suatu kegiatan.



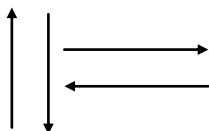
DECISION

Digunakan untuk menggambarkan proses pengujian suatu kondisi yang ada.



PREPARATION

Digunakan untuk menggambarkan persiapan harga awal, dari proses yang akan dilakukan.



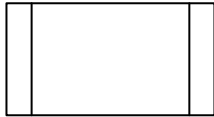
FLOW LINE

Digunakan untuk menggambarkan hubungan proses dari satu proses ke proses lainnya.



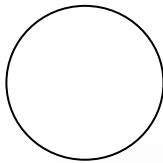
INPUT/OUTPUT

Digunakan untuk menggambarkan proses memasukan data yang berupa pembacaan data dan sekaligus proses keluaran yang berupa pencetakan data.



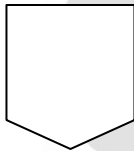
SUBROUTINE

Digunakan untuk menggambarkan proses pemanggilan sub program dari main program (*recursivitas*).



PAGE CONNECTOR

Digunakan untuk menghubungkan alur proses ke dalam satu halaman atau halaman yang sama.



CONNECTOR

Digunakan untuk menghubungkan alur proses dalam halaman yang berbeda atau ke halaman berikutnya.

DAFTAR GAMBAR

	Halaman
Gambar II.1 Model Sistem.....	23
Gambar III.1 Stuktur Orgnisasi.....	32
Gambar III.2 Activity Diagram ProsesPersediaan Barang.....	36
Gambar IV.1 Use Case Menu Login Admin Gudang.....	41
Gambar IV.2 Use Case Diagram Login Admin Staff Inventory	42
Gambar IV.3 Use Case Diagram Login Pimpinan	43
Gambar IV.4 Activity Diagram Login Admin.....	44
Gambar IV.5 Activity Diagram Login Menu Master	45
Gambar IV.6 Activity Diagram Login Staff Inventory	46
Gambar IV.7 Activity Diagram Halaman Login Staff Pimpinan.....	47
Gambar IV.8 Activity Diagram Halaman Login Pimpinan.....	48
Gambar IV.9 Activity Diagram Setelah Login Pimpinan	49
Gambar IV.10 Entity Relationship Diagram.....	50
Gambar IV.11 Logical Record Structure	51
Gambar IV.12 Component Diagram.....	62
Gambar IV.13 Deployment Diagram.....	63
Gambar IV.14 Tampilan Halaman Login.....	63
Gambar IV.15 Tampilan Menu Data Master	64
Gambar IV.16 Tampilan List Merk.....	64
Gambar IV.17 Tampilan List Kategori.....	65
Gambar IV.18 Tampilan Stok Barang.....	65
Gambar IV.19 Tampilan Tambah Produk	66
Gambar IV.20 Tampilan Data Customer	66
Gambar IV.21 Tampilan Tambah Customer	67
Gambar IV.22 Tampilan Halaman Data Supplier	67
Gambar IV.23 Tampilan Tambah Supplier	68
Gambar IV.24 Tampilan Menu Data Pembeli.....	68
Gambar IV.25 Tampilan Menu Data Penjualan.....	69
Gambar IV.26 Tampilan Halaman Return.....	70
Gambar IV.27 Tampilan History Penjualan.....	71
Gambar IV.28 Tampilan Menu Laporan.....	71
Gambar IV.29 Tampilan Menu Halaman Laporan Pembelian.....	72
Gambar IV.30 Tampilan List User	72

DAFTAR TABEL

Halaman

Tabel IV.1 Use Case Diagram Menu Login Admin Gudang.....	41
Tabel IV.2 Use Case Diagram Login Staff Inventory.....	42
Tabel IV.3 Use Case Diagram Menu Login Pimpinan.....	43
Tabel IV.4 Spesifikasi File Tabel Category.....	52
Tabel IV.5 Spesifikasi File Tabel Customer	53
Tabel IV.6 Spesifikasi File Tabel Grup	54
Tabel IV.7 Spesifikasi File Tabel Merk	55
Tabel IV.8 Spesifikasi File Tabel Produk.....	56
Tabel IV.9 Spesifikasi File Tabel Supplier	57
Tabel IV.10 Spesifikasi File Tabel Tabel Toko	58
Tabel IV.11 Spesifikasi File Tabel Tabel.....	59
Tabel IV.12 Spesifikasi File Tabel Transaksi Rell.....	60
Tabel IV.13 Spesifikasi File Tabel User.....	61
Tabel IV.14 Hasil Pengujian Black Box Testing From Login Admin.....	120
Tabel IV.15 Hasil Pengujian Black Box Testing From Tambah Barang.....	121
Tabel IV.16 Hasil Pengujian Black Box Testing From Tambah Pelanggan.....	122
Tabel IV.17 Hasil Pengujian Black Box Testing From Tambah Supplier.....	123
Tabel IV.18 Hasil Pengujian Black Box Testing From Login Staff Inventory...	124
Tabel IV.19 Hasil Pengujian Black Box Testing From Login Pimpinan.....	125
Tabel IV.20 Spesifikasi Hardware dan Software.....	125



DAFTAR LAMPIRAN

1. Lampiran A.1. Dokumen Sistem Berjalan Surat Jalan.....	118
2. Lampiran A.2.Dokumen Sistem Berjalan Keluar Purchase Order.....	119
3. Lampiran B.1. Dokumen Usulan Laporan Pembelian.....	120
4. Lampiran B.2. Dokumen Usulan Laporan Penjualan.....	121



BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Sistem *Inventory* merupakan suatu sistem yang digunakan untuk mencatat proses keluar masuknya barang dari tempat persediaan dan semuanya disimpan kedalam *database*, sehingga pengguna bisa mengetahui, mengawasi dan mengontrol persediaan barang dengan mudah. Dari hal pencatatan tersebut, pengguna juga bisa menarik laporan transaksi keluar maupun masuk barang. Kebanyakan sistem *inventory* diperusahaan masih menggunakan prosedur secara manual sehingga tidak efektif dan efisien dalam pelaksanaannya.

Salah satu perusahaan yang masih menerapkan atau menggunakan sistem *inventory* manual adalah CV.Gratia Motorindo, CV.Gratia Motorindo merupakan perusahaan yang bergerak dalam bidang penjualan *sperpart motor*. Pada CV.Gratia Motorindo sistem *inventory* masih menggunakan sistem secara manual. Dengan tidak adanya sistem yang terkomputerisasi didalam gudang, maka sistem tidak terhubung antara transaksi penjualan dan pembelian barang, sehingga sering terjadi kesalahan dalam pemasukan dan perubahan data.

Inventory merupakan sejumlah barang yang disediakan untuk memenuhi permintaan dari pelanggan. Dalam perusahaan perdagangan pada dasarnya hanya ada satu golongan *inventory* (persediaan), yang mempunyai sifat perputaran yang sama yaitu yang disebut “*Merchandise Inventory*” (persediaan barang dagangan). Persediaan ini merupakan persediaan barang yang selalu dalam perputaran, yang selalu dibeli dan dijual, yang tidak mengalami proses lebih lanjut didalam perusahaan tersebut yang mengakibatkan perubahan bentuk dari barang yang bersangkutan. (Minarni & Susanti, 2014).

Berdasarkan permasalahan diatas untuk dapat mempermudah CV.Gratia motorindo dalam hal kegiatan *inventory* maka diperlukan sistem informasi *inventory*

berbasis *web*, yang merupakan salah satu solusi untuk dapat menyelesaikan masalah tersebut. Dengan adanya aplikasi tersebut dapat membantu semua kegiatan *inventory* sehingga lebih efektif dan efisien.

1.2. Identifikasi Permasalahan

Berdasarkan Latar Belakang dan hasil obeservasi awal yang penulis lakukan di CV.Gratia Motorindo, maka identifikasi masalah yang di dapat adalah sebagai berikut :

1. Pengolahan data barang yang dilakukan baru sebatas manual atau menggunakan *Ms.Excel*.
2. Tidak adanya sistem yang berhubung antara transaksi pembelian dan penjualan, yang menyebabkan terdapat banyaknya *database*, sehingga membutuhkan waktu yang cukup lama pada saat pencarian data pembelian dan penjualan.
3. Pengajuan pembelian barang masih manual.
4. Pembuatan laporan dan stock opname masih manual atau menggunakan *Ms. Excel*, sehingga sering memperlambat dalam pembuatannya.

1.3. Perumusan Masalah

Dari identifikasi masalah yang telah didapat dari hasil observasi di CV.Gratia Motorindo maka penjabaran perumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana merancang sistem pengolahan data barang yang sudah terkomputerisasi melalui aplikasi *inventory* berbasis *web* sesuai dengan kebutuhan perusahaan.

2. Bagaimana merancang aplikasi *inventory* berbasis *web* yang dapat melakukan pemesanan barang secara lebih mudah.
3. Bagaimana perancangan sistem informasi *inventory* berbasis *web* pada Gratia Motorindo yang dapat membantu pengolahan data ketika perpindahan barang antar gudang.



- 4.
5. Bagaimana sistem dapat menyajikan stok opname dan laporan yang cepat dan sesuai dengan manajemen perusahaan

1.4. Maksud dan Tujuan

Maksud dari penulisan skripsi ini adalah:

1. Membangun sebuah sistem informasi *inventory* berbasis *web* pada CV.Gratia Motorindo untuk menyelesaikan permasalahan dalam hal pengolahan data, pencatatan, pengajuan, pembelian, transaksi barang masuk dan keluar, dan perpindahan barang antar gudang.
2. Menyajikan stok opname dan laporan secara cepat sesuai kebutuhan perusahaan.
3. Mempermudah karyawan CV.Gratia Motorindo dalam hal kegiatan *inventory* dan mengurangi kesalahan-kesalahan yang biasa dilakukan.
4. Menerapkan dan mengembangkan pengetahuan yang telah didapat baik didalam maupun diluar kampus STMIK Nusa Mandiri Jakarta.

Adapun Tujuan dari penulisan skripsi ini adalah sebagai salah satu syarat kelulusan pada Strata Satu (S1) Program Studi Sistem Informasi Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Nusa Mandiri Jakarta.

1.5. Metode Penelitian

1.5.1. Teknik Pengumpulan Data

A. Observasi

Dalam metode ini penulis melakukan pengumpulan data dan pengamatan secara langsung mengenai sistem *inventory* yang sudah berjalan dan permasalahan dari CV.Gratia motorindo.

B. Wawancara

Dalam hal ini penulis melakukan wawancara untuk melengkapi bahan yang sudah ada selama observasi. Penulis melakukan wawancara kepada bapak Darwin wijaya selaku Direktur Utama Gratio Motorindo yang berkaitan dengan sistem yang diteliti.

C. Studi Pustaka

Metode studi pustaka yang penulis lakukan untuk membantu penulis dalam hal pembuatan skripsi ini adalah dengan beberapa buku jurnal dan literature yang berkaitan dengan materi yang dibuat dalam penyusunan skripsi ini penulis mendapat banyak bahan masukan tentang bagaimana merancang atau mengembangkan suatu sistem informasi menurut para ahlinya. Juga pada metode ini penulis membuka, mengambil, dan mengutip dari beberapa kutipan para ahli yang berdasarkan dari jurnal ilmiah nasional.

1.6. Metode Pengembangan Sistem:

A. Analisa Kebutuhan Sistem

Analisa kebutuhan sistem untuk karyawan yang berhubungan dengan sistem *inventory* pada CV.Gratia motorindo

1. Menyediakan *form login* untuk karyawan sesuai dengan *level* atau otorisasi kewenangan.
2. Menyediakan pengolahan master data yang terdiri dari staff, *customer*, *kategori*, *supplier*, *gudang* & *level* (otorisasi kewenangan)

3. Menyediakan pengolahan transaksi pembelian, yang terdiri dari transaksi pembelian, bukti barang masuk, purchase order dan pembayaran transaksi.
4. Menyediakan pengolahan transaksi penjualan, yang terdiri dari transaksi penjualan, PO, *sales order*, *sales invoice*, dan pembayaran transaksi.
5. Menyediakan menu gudang, untuk mengetahui stok barang..
6. Menyediakan menu pengolahan retur barang, yang terdiri retur pembelian dan retur penjualan.
7. Menyediakan menu pengolahan stok opname
8. Menyediakan menu laporan, yang terdiri dari laporan stok produk, laporan pembelian, laporan penjualan,

B. Desain

Tahap desain sistem adalah tahap dimana penulis melakukan desain sistem ini sesuai dengan kebutuhan karyawan CV.Gratia Motorindo dalam menjalankan sistem *inventory*. Dalam hal ini penulis menggunakan diagram *ERD* dan *LRS* untuk membuat rancangan *database*, *software*, *architechture* menggunakan *UML (Unified Modeling Language)*, dan untuk mendesain user interface menggunakan *CSS (Cascading yle Sheets)*, *Bootstrab*, dan *tamplate aqwatavarius*, sedangkan perangkat lunak yang digunakan untuk mendukung sistem menggunakan *Laravel*, *MySQL PHP 7.1* dengan *web server XAMPP*.

C. Code Generation

Pada tahap ini bahasa pemrograman yg digunakan *PHP* dan *database MySQL* dimana untuk menerapkan aplikasi transaksi sistem informasi *inventory* berbasis

web, teknik pemrograman yang digunakan adalah teknik pemrograman yang berorientasi objek

D. Testing

Pada tahap ini merupakan pengujian terhadap sistem yang telah dibuat, untuk mengetahui bagaimana program yang dibuat, dan mengetahui dimana kekurangan program tersebut, dalam hal ini penulis menggunakan *Black Box Testing*.

E. Support

Pada tahap ini pembuatan sistem adalah menggunakan *hardware* sudah support ke semua program, dan *software* yang digunakan sudah bisa diimplementasikan dalam lingkup yang luas. Dibutuhkan juga dalam pengembangan sistem server local yang terkoneksi dengan internet untuk menghubungkan komputer satu dengan komputer lainnya dan peralatan pendukung lainnya.

1.7. Ruang Lingkup

Dalam penulisan skripsi ini, penulis akan membatasi ruang lingkup penelitian dengan menitikberatkan permasalahan yang akan dibahas, yaitu mengenai sistem informasi *inventory* pada CV.Gratia Motorindo. Adapun proses bisnis yang terkait dengan sistem informasi *inventory* yang akan diteliti meliputi: prosedur, transaksi pembelian, prosedur transaksi penjualan, prosedur retur barang, prosedur stok opname, prosedur laporan, dan prosedur hutang dan piutang dagang. Sistem yang diusulkan oleh penulis yang dijalankan menggunakan jaringan internal atau *intranet* dan tidak membahas keamanan sistem.

BAB II

LANDASAN TEORI

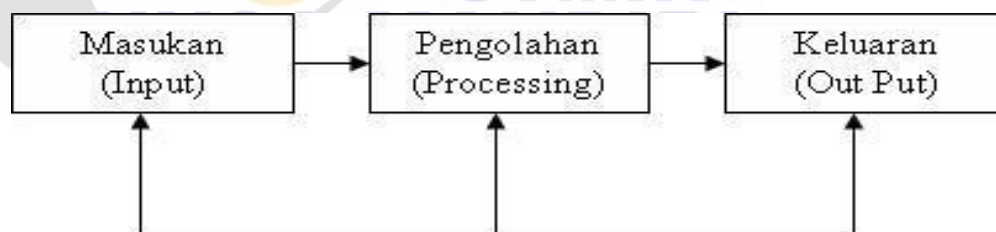
2.1. Tinjauan Pustaka

Teori yang berhubungan dengan skripsi yang penulis dibuat diantaranya :

2.1.1 Konsep Dasar Sistem Informasi

A. Pengertian Sistem

Menurut (Anggraeni, 2017:1) menyatakan bahwa “Sistem adalah kumpulan orang yang saling sistematis dan terstruktur untuk membentuk satu kesatuan yang melaksanakan suatu fungsi untuk mencapai tujuan. Sistem memiliki beberapa karakteristik atau sifat yang terdiri dari komponen sistem, batasan sistem, lingkungan luar sistem, penghubung sistem, masukan sistem, keluaran sistem, pengolahan sistem dan sasaran sistem.



Sumber : Anggraeni (2017:1)

Gambar II.1

Model Sistem

B. Pengertian Informasi

Menurut (Hutahaean, 2014:9) menyatakan bahwa “informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya”.

C. Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem yang didalam terdapat suatu organisasi yaitu yang mempertemukan kebutuhannya pada pengolahan transaksi harian yang mendukung fungsi-fungsi operasi organisasi dan untuk dapat laporan yang diperlukan.(Agusvianto, 2017)

Tujuan dari sistem informasi yaitu untuk menyediakan informasi untuk mengambil suatu keputusan dan menyediakan suatu informasi yang dipergunakan didalam suatu perencanaan, pengendalian, mengevaluasi dan memperbaiki berkelanjutannya.

D. Pengertian Inventory

Inventory adalah kumpulan elemen yang saling berhubungan satu sama lain dengan memberikan masukan (data) untuk mendapatkan suatu keluaran (informasi) dengan fungsi untuk memproses, menyimpan, mendistribusikan informasi serta mendukung pengambilan keputusan. (Rorimpandey, 2015).

Inventory adalah fasilitas yang bersifat tetap, yang dirancang untuk mencapai target tingkat pelayanan dengan total biaya yang paling rendah.inventory dibutuhkan dalam proses kordinasi penyaluran barang, yang muncul sebagai akibat kurang seimbangnya antara proses permintaan dan penawaran mendorong munculnya persediaan, persediaan membutuhkan ruang sebagai tempat penyimpanan sementara yang disebut sebagai gudang. (Riyan Naufal, 2017)

E. Internet

Menurut (Jubilee, 2011:1) mengertikan “Internet merupakan kependekan dari interconnection network adalah seluruh jaringan computer yang saling terhubung menggunakan standar sistem global TCP/IP sebagai protocol pertukaran paket untuk melayani pengguna diseluruh dunia.

F. Web

Menurut (Rerung, 2018:1) kan bahwa “Web adalah jaringan computer yang terdiri dari kumpulan situs internet yang menawarkan teks, grafik, suara dan sumber daya animasi melalui hypertext transfer protocol.

2.1.2 Bahasa Pemrograman

Menurut (Sahyar, 2016:2) menyatakan bahwa “bahasa pemrograman yaitu program yang digunakan untuk menerjemahkan perintah-perintah yang ditulis dalam bahasa pemrograman ke dalam bahasa mesin agar dapat dimengerti oleh komputer”.

Bahasa yang digunakan oleh penulis antara lain :

A. HTML (Hypertext Markup Language)

Menurut (Jubilee, 2011:2) menyatakan bahwa “HTML (*Hypertext Markup Language*) adalah sebuah bahasa yang digunakan untuk membuat halaman web internet, menampilkan berbagai informasi di dalam sebuah penjelajah web internet dan pemformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud terintegrasi.

B. PHP (Personal Home Page)

Menurut (Supono, 2016:3) mengartikan bahwa “PHP adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat mengerti oleh computer yang bersifat *servers-slide* yang dapat ditambahkan ke dalam *HTML*”.

C. CSS (*Cascading Style Sheet*)

Menurut (Prasetio, 2014:252) menyatakan bahwa “CSS (*Cascading Style Sheet*) adalah suatu teknologi yang digunakan untuk memperindah tampilan halaman

website (Situs)”. Singkatnya dengan menggunakan metode CSS ini mudah mengubah secara keseluruhan sekaligus memformat ulang suatu situs.

D. Javascript

Menurut (Edy, 2014:2) menyatakan bahwa “*Javascript* adalah kode pemrograman yang bisa dimasukan atau disisipkan *HTML*”.

E. Adobe Dreamweaver

Menurut (Putratama dan Supono, 2016:4) menyatakan bahwa “Adobe Dreamweaver adalah perangkat lunak yang digunakan untuk membuat atau mengedit halaman web”. Adobe Dreamweaver mendukung beberapa bahasa pemrograman, antara lain : *HTML, PHP, CSS, Javascript, PHP* dan sebagainya.

2.1.3 Basis Data

Menurut(Widodo, 2017:8),”Basis Data adalah model data, kumpulan dari alat-alat konseptual untuk menjelaskan data, hubungan data, semantik data, dan batasan-batasan minimum”.

A. MYSQL (My Structucre Query Language)

Menurut (Anhar, 2010:21) menyatakan bahwa “*MYSQL (My Structucre Query Language)* adalah sebuah perangkat lunak sistem manajemen basis data SQL (Database Management System) atau DBMS dari sistem banyak DBMS”.

Berikut adalah perintah dasar *MYSQL* :

a. Perintah Dasar *MySQL*

1. Create Database, perintah untuk membuat database.
2. User, perintah menggunakan database
3. Create Table, perintah untuk membuat table.
4. Drop Database, perintah untuk menghapus database.

- b. Perintah *MySQL* pada sub *DML*

DML (Database Manipulation Language) adalah paket bahasa yang digunakan untuk melakukan manipulasi terhadap data pada sebuah basis data. Manipulasi data seperti :

- a. Insert, perintah untuk memasukkan data.
- b. Select, perintah untuk menampilkan data.
- c. Update, perintah untuk mengubah atau mengedit data.
- d. Delete, perintah untuk menghapus data.

B. XAMPP

Menurut (Imansyah, 2014:4), menyatakan “*XAMPP* adalah installer yang membundel *apache*, *PHP* dan *MySQL* untuk windows dalam satu paket”. Yang terdapat pada Xampp di antaranya : *apache*, *MySQL*, *PHP*, *FilZilla*, *FTP Server* dan *PHP my Admin*.

2.1.4 Model Pengembangan Perangkat Lunak

A. Pengertian Waterfall

Menurut (Rosa dan Shalahudin, 2015) menyatakan bahwa “*Waterfall* adalah model SDLC air terjun (*waterfall*) sering juga disebut model *sequensial linear* atau alur hidup klasik (*classic life cycle*)”.

Menurut (Shalahuddin dan Sukamdi, 2013:29) menyimpulkan bahwa “

1. Analisa Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.

2. Desain

Desain perangkat lunak adalah multi langkah yang focus pada *desain* pembuatan program perangkat lunak seperti struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.

3. Pembuatan Kode Program

Desain harus ditranlasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program computer sesuai dengan desain yang telah dibuat.

4. Pengujian

Pengujian ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai yang diinginkan.

Model air terjun sangat cocok digunakan bagi kebutuhan pelanggan yang sudah dipahami dan kemungkinan terjadinya perubahan selama pengembangan perangkat lunak kecil.

2.1.5 Teori Pendukung (Supporting Theory)

A. UML (*Unified Modeling Language*)

Menurut (Rosa dan Shalahudin, 2015:133), menyatakan bahwa “UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan requirement, membuat analisa dan desain,serta menggambarkan arsitektur dalam pemrograman objek”. Macam-macam jenis UML sebagai berikut : *Class Diagram* (statis), *Package Diagram* (statis), *Sequence Diagram* (dinamis), *Communication Diagram*, *State Chart Diagram* (dinamis), *Activity Diagram* (dinamis), *Component Diagram* (statis), *Deployment Diagram* (statis).

Namun pada penelitian ini hanya menggunakan 3 diagram, yaitu :

1. Use Case

Menurut Rosa dan Shalahudin (2016:155), “Use Case atau diagram Use Case merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat”.

2. Activity Diagram

Menurut (Rosa dan Shalahudin, 2015:161) menyatakan bahwa “Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

3. Deployment Diagram

Menurut (Rosa dan Shalahudin, 2015:164), menyatakan bahwa “Deployment diagram menunjukan konfigurasi dalam proses eksekusi aplikasi”.

B. ERD (Entity Relationship Diagram)

Menurut (Shalahudin dan Sukamdi, 2013:50), “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram*”. Berikut adalah bentuk contoh hubungan relasi dalam ERD.

1. Binary

Binary (satu relasi menghubungkan dua buah entitas). Tingkat hubungan satu , dinyatakan dengan satu kejadian pada entitas pertama.

2. Ternarys

Ternarys (satu buah relasi menghubungkan banyak entitas). Tingkat hubungan satu ke banyak adalah sama dengan banyak kesatu tergantung dari arah mana berhubungan tersebut dilihat.

3. N-ary

Tingkat hubungan banyak ke banyak terjadi tiap kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya.

C. LRS (Logical Record Structure)

Menurut (Afifah dan Supriyanta 2018) “LRS (*logical Record Structure*) merupakan hasil dari pemodelan ERS (*Entity Relational Ship*) beserta atributnya sehingga bisa terlihat hubungan-hubungan antar entitas”. LRS terdiri dari link-link diantara record. Link ini menunjukkan arah dari satu tipe record lainnya. Banyak link dari LRS yang diberi tanda field-field yang kelihatan pada dua link tipe.

D. Black Box Testing

Menurut (Martudi, 2014) menyatakan bahwa “Black Box Testing bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara pengoperasinya, apakah pemasukan dan keluaran telah berjalan sebagaimana yang diharapkan”.

2.2 Penelitian Terkait

Dalam penyusunan skripsi ini, penulis banyak terinspirasi dan mereferensi dari penelitian-penelitian sebelumnya, penelitian yang berhubungan dengan skripsi ini yaitu perancangan sistem informasi persediaan barang pada PT.Alaisys Sidoarjo merupakan perancangan dari sistem yang sedang berjalan.

Bahwa sebagai perusahaan teknologi distributor, PT.Alaisys selalu melakukan pengawasan dan pencatatan terhadap barang persediaan, pelaporan dari gudang ke kantor pusat dilakukan dengan cara menyalin data kartu ke dalam Ms.Excel. laporan dalam format Ms.Excel tersebut harus dikirim via email atau gadget. Sistem tersebut menjadikan pihak kantor pusat tidak dapat mengetahui data dari gudang dengan efisien. (Agusvianto, 2017:40.)

Inventory merupakan stok bahan baku, pasokan, komponen, barang yang masih dalam pengerjaan, dan barang jadi yang muncul sepanjang proses produksi dan logistic dalam suatu perusahaan. (Chandra & Wiradinata, 2015:125)

“Dengan berbagai kelebihan dan kemudahan yang ditawarkannya tentu akan meningkatkan efisiensi dan efektivitas kerja dari suatu

perusahaan. Salah satu kelebihan dari system komputerisasi ialah menghasilkan informasi secara relevan, tepat waktu dan akurat. Mengelola sebuah persediaan barang dengan teliti dan terjadwal memberikan suatu dampak efektifitas kinerja dari setiap perusahaan dan meningkatkan pelayanan terhadap konsumen agar lebih baik.(Saefudin, 2018).



BAB III

ANALISA SISTEM BERJALAN

3.1 Tinjauan Institusi/Perusahaan

Di tinjau dari suatu perusahaan, maka penulis akan membahas tentang sejarah, struktur organisasi dan fungsi dari masing-masing struktur CV.Gratia Motorindo.

3.1.1 Sejarah Perusahaan

CV.Gratia motorindo adalah salah satu perusahaan di Indonesia yang bergerak dibidang penjualan. Lokasi perusahaan ini di jalan Jelambar selatan ix no.15f Jakarta barat dan didirikan 13 maret 2011. Sasaran utama perusahaan ini di bentuk untuk penjualan ke toko-toko sperpart motor. Perusahaan ini menyediakan barang sperpart motor dan lain-lain.

Di dalam era globalisasi saat ini, Indonesia berada di posisi antara negara satu dengan negara lain, tujuan untuk memperoleh keuntungan dan mendorong perkembangan perindustrian yang terus tumbuh dan menghasilkan keragaman produk dan berkualitas yang mampu bersaing. Dengan pengalaman, pengetahuan, kepercayaan yang diberikan dan dukungan manusia yang professional.

Presentasi keuntungan sekitar 91%-100% dan transaksi pembayaran dengan menggunakan mata uang rupiah.

Setiap perusahaan yang didirikan pasti mempunyai visi dan misi, diantaranya :

1. Visi :

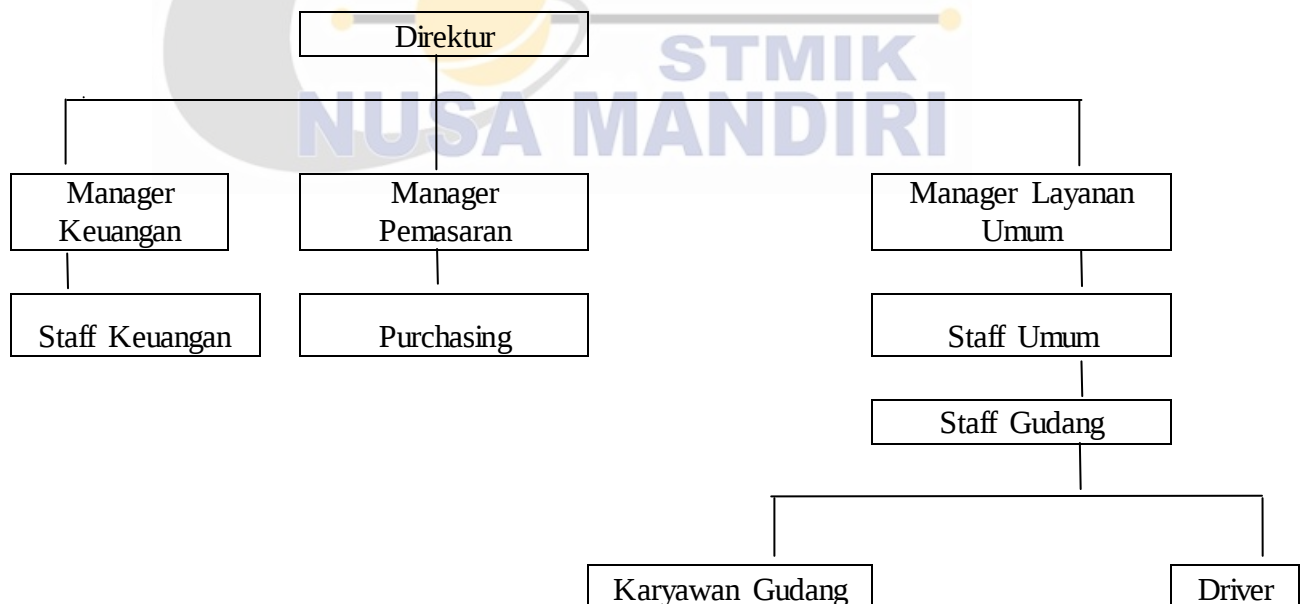
- a. Perusahaan yang memfokuskan diri dalam mengembangkan pasar import dan menjadi mitra dagang bagi perusahaan nasional.
- b. Menciptakan peluang dengan melakukan penyesuaian spesifikasi produk dan harga yang mampi memenuhi permintaan.

2. Misi :

- a. Mengembangkan sumber daya manusia (SDM) yang berkinerja tinggi
- b. Menerapkan teknologi terkini dan aktifitas perusahaan.
- c. (satu relasi menghubungkan dua buah entitas). Tingkat hubungan satu kesatu, dinyatakan dengan satu kejadian pada entitas pertama.

3.1.2 Struktur Organisasi dan Fungsi

Struktur organisasi adalah suatu susunan dan hubungan antar tiap bagian serta posisi pada suatu perusahaan dalam menjalankan suatu kegiatan operasional. Struktur Organisasi menggambarkan dengan jelas pemisahan kegiatan pekerjaan antara yang satu dengan yang lain dan bagaimana hubungan aktifitas dan fungsi dibatasi. Struktur organisasi pada CV.Gratia Motorindo digambar dibawah ini :



Gambar III.1

Struktur Organisasi CV.Gratia Motorindo

Adapun fungsi dari struktur organisasi CV.Gratia Motorindo :

1. Direktur

Direktur adalah suatu jabatan tertinggi dalam perusahaan yang bertindak sebagai pimpinan, sekaligus sebagai pemilik perusahaan. Wewenang direktur adalah sebagai berikut :

- a. Memimpin perusahaan dengan membuat kebijakan-kebijakan di perusahaan.
- b. Memanajemen, memilih, menentukan dan mengawasi pekerjaan yang ada di perusahaan.

2. Staff Keuangan atau *Accounting*

Staff Keuangan atau *Accounting* adalah suatu proses atau mencatat, mengklasifikasi, meringkas, mengolah dan menyajikan data, transaksi serta kejadian yang berhubungan dengan keuangan sehingga dapat digunakan oleh orang dengan tujuan tertentu. Tugas seorang accounting yaitu :

- a. Menyusun dan membuat laporan keuangan, perpajakan perusahaan, anggaran pengeluaran dan pendapatan di suatu perusahaan.
- b. Menyusun dan membuat surat-surat yang berhubungan dengan perbankan dan kemampuan keuangan perusahaan.

3. Manager Keuangan

- a. Membuat laporan keuangan, memimpin, mengontrol, dan membuat keputusan penting mengenai keuangan perusahaan.
- b. Mengkoordinasi kegiatan pencatatan penjualan dan pengeluaran kas perusahaan.

4. Manager Pemasaran

- a. Bertanggung jawab terhadap target penjualan produk dan mengatur jadwal penjualan.
- b. Mencari terobosan perluasan pada market yang mungkin atau potensial untuk di garap.

5. Purchasing

Purchasing atau pembelian adalah salah satu fungsi sangat penting dalam suatu perusahaan yang mendukung kelancaran proses penjualan.

Tugas seorang purchasing yaitu :

- a. Melakukan pembelian barang dan membuat purchase order (PO).
- b. Membantu konsumen, jika terdapat komplain barang

6. Manager Layanan Umum

- a. Mengawasi seluruh staff kantor dan mengontrol disiplin karyawan dan hubungan karyawan dalam perusahaan
- b. Memberikan informasi atau data yang akurat kepada staff dan menetapkan standard dan prosedur operasional

7. Staff Umum

- a. Membantu dan melaksanakan tugas- tugas yang di berikan oleh Koordinator Keuangan dan Umum
- b. Melaksanakan penyelenggaraan administrasi umum

8. Staff Gudang

- a. Bertanggung jawab dan mengatur segala barang yang keluar masuk dari gudang.

- b. Mengecek semua barang dengan jumlah yang ada di laporan persediaan barang dan memastikan bahwa semua barang dalam keadaan baik, tidak rusak atau hilang.

9. Karyawan Gudang

- a. Melakukan pengerjaan barang sesuai permintaan pelanggan
- b. Mempacking barang-barang berdasarkan orderan.

10. *Driver* atau Transportasi

- a. Mengkordinasi barang yang akan dikirim
- b. Mengantar barang yang akan dikirim ke toko-toko atau ekspedisi

3.2 Prosedur sistem berjalan

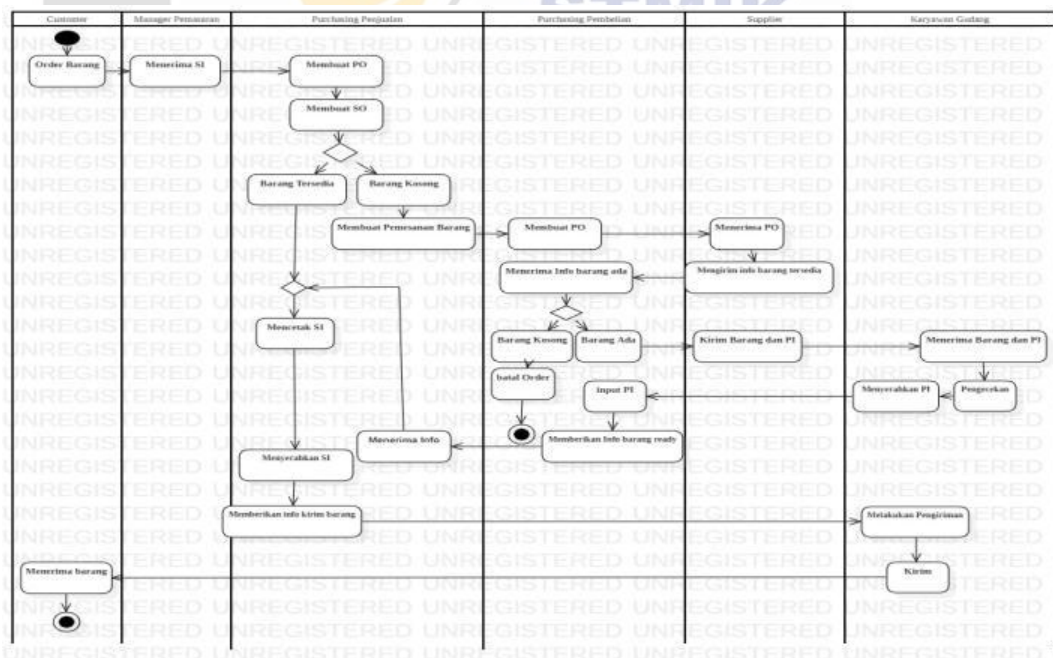
Dari pengamatan yang telah dilakukan proses bisnis persediaan (*inventory*) yang terjadi didalam CV.Gratia Motorindo yang bergerak di bidang perusahaan penjualan sperpart motor sebagai berikut :

Dalam kegiatan persediaan barang, proses dimulai dari sales mendapatkan orderan. Kemudian sales memberikan *orderan* atau *Sales Invoice (SI)* ke bagian *purchasing* penjualan. Setelah *purchasing* penjualan menerima *orderan*, kemudian *purchasing* membuat *Purchasing Order (PO)*. Bagian *purchasing* penjualan langsung membuat *Sales Order (SO)* untuk barang yang tersedia di gudang. *Purchasing* penjualan langsung membuat faktur *Sales Invoice (SI)* sesuai dengan permintaan *customer* dan memberikan faktur *Sales Invoice (SI)* ke karyawan gudang dan memberikan informasi untuk pengiriman. Jika barang sudah siap untuk di kirim maka barang dikirim ke *customer*. Jika barang yang tidak tersedia maka *purchasing* penjualan, memberikan informasi ke bagian *purchasing* pembelian untuk membuat *Purchasing Order (PO)* ke *supplier*. Bagian *purchasing* pembelian

langsung melakukan pembelian barang ke *supplier*. Jika yang dibeli telah tersedia, maka barang segera dikirim ke gudang dan jika barang kosong, maka pihak *supplier* menghubungi pihak *Purchasing* pembelian untuk meminta pembatalan *order*.

Ketika barang datang, dari pihak *supplier* akan memberikan *Purchasing Invoice* (PI) dan karyawan gudang langsung melakukan pengecekan barang. Kemudian *Purchasing Invoice* (PI) diberikan ke bagian *Purchasing* pembelian untuk di input dan memberikan info ke *purchasing* penjualan bahwa barang telah tersedia. Kemudian *purchasing* penjualan membuat *Sales Order* (SO). *purchasing* penjualan langsung membuat faktur *Sales Invoice* (SI) dan memberikan faktur *Sales Invoice* (SI) ke karyawan gudang dan memberikan informasi untuk pengiriman barang. Jika barang sudah siap untuk di kirim maka barang dikirim ke customer.

Adapun *Activity diagram* proses persediaan barang (*inventory*) sistem berjalan yang terjadi pada CV.Gratia Motorindo dapat di lihat sebagai gambar berikut :



Gambar III.2

Activity Diagram Proses Persediaan Barang

3.3 Spesifikasi Dokumen Sistem Berjalan

Spesifikasi dokumen sistem berjalan adalah bentuk dokumen yang berasal dari lingkungan dalam maupun luar organisasi perusahaan, yang mana dokumen itu akan diolah dalam suatu proses di dalam lingkungan gudang agar dapat menghasilkan keluaran yang diinginkan.

Spesifikasi Dokumen Masukan dan Keluaran :

a. Nama Dokumen : Surat jalan

Fungsi : Sebagai bukti penerimaan barang

Sumber : Supplier

Tujuan : Bagian Gudang

Media : Kertas

Jumlah : 1 Rangkap

Frekuensi : Setiap penerimaan barang

Bentuk : Lampiran A-1

b. Nama Dokumen : Purchases Order (PO)

Fungsi : Sebagai bukti permintaan pembelian barang ke supplier

Sumber : Bagian Gudang

Tujuan : Purchasing

Media : Kertas

Jumlah : 1 Rangkap

Frekuensi : Setiap ada permintaan pembelian barang

Bentuk : Lampiran A-3

BAB IV

RANCANGAN SISTEM DAN PROGRAM USULAN

4.1 Analisa Kebutuhan Software

A. Tahapan Analisis

Pada sistem persediaan barang ini ada tiga hak akses yaitu, admin Gudang, *Staff Inventory* dan Pimpinan. Penjelasan sebagai berikut :

- a. Admin adalah bagian yang memiliki hak akses sepenuhnya pada sistem, mulai dari input merek, kategori, *stock* barang, data *supplier*, data *customer*, data pembelian, data penjualan, *return* pembelian, *return* penjualan, *history* pembelian, *history* penjualan, *history return* pembelian, *history return* penjualan, laporan pembelian, laporan penjualan, laporan *return* pembelian, laporan *return* penjualan dan *list user*.
- b. *Staff Inventory* hanya bisa melakukan dari *stock* barang, data *customer*, data *supplier*, data pembelian, data penjualan, *return* pembelian, *return* penjualan, *history* pembelian, *history* penjualan, *history return* pembelian, *history return* penjualan, laporan pembelian, laporan penjualan, laporan *return* pembelian, laporan *return* penjualan.
- c. Pimpinan hanya bisa melakukan pengecekan laporan pembelian, laporan penjualan, laporan *return* pembelian, laporan *return* penjualan.

Berikut ini adalah spesifikasi kebutuhan dari sistem persediaan barang :

Halaman Admin

- A.1 Admin melakukan login
- A.2 Admin melakukan data item
- A.3 Admin mengelola data *customer*

A.4 Admin Mengelola data *supplier*

A.5 Admin mengelola data

A.6 Admin mengelola *return*

A.7 Admin mengelola *history*

A.8 Admin mengelola laporan data

A.9 Admin mengelola laporan return

A.10 Admin mengelola list user

Halaman Staff Inventory

B.1 Staff Inventory melakukan *login*

B.2 Staff Inventory mengelola *item*

B.3 Staff Inventory mengelola data *customer*

B.4 Staff Inventory mengelola data *supplier*

B.5 Staff Inventory mengelola data

B.6 Staff Inventory mengelola data *return*

B.7 Staff Inventory mengelola data *history*

B.8 Staff Inventory mengelola data laporan

B.9 Staff Inventory mengelola laporan *return*

Halaman Login Pimpinan

C.1 Pimpinan melakukan *login*

C.2 Pimpinan melihat laporan pembelian

C.3 Pimpinan melihat laporan penjualan

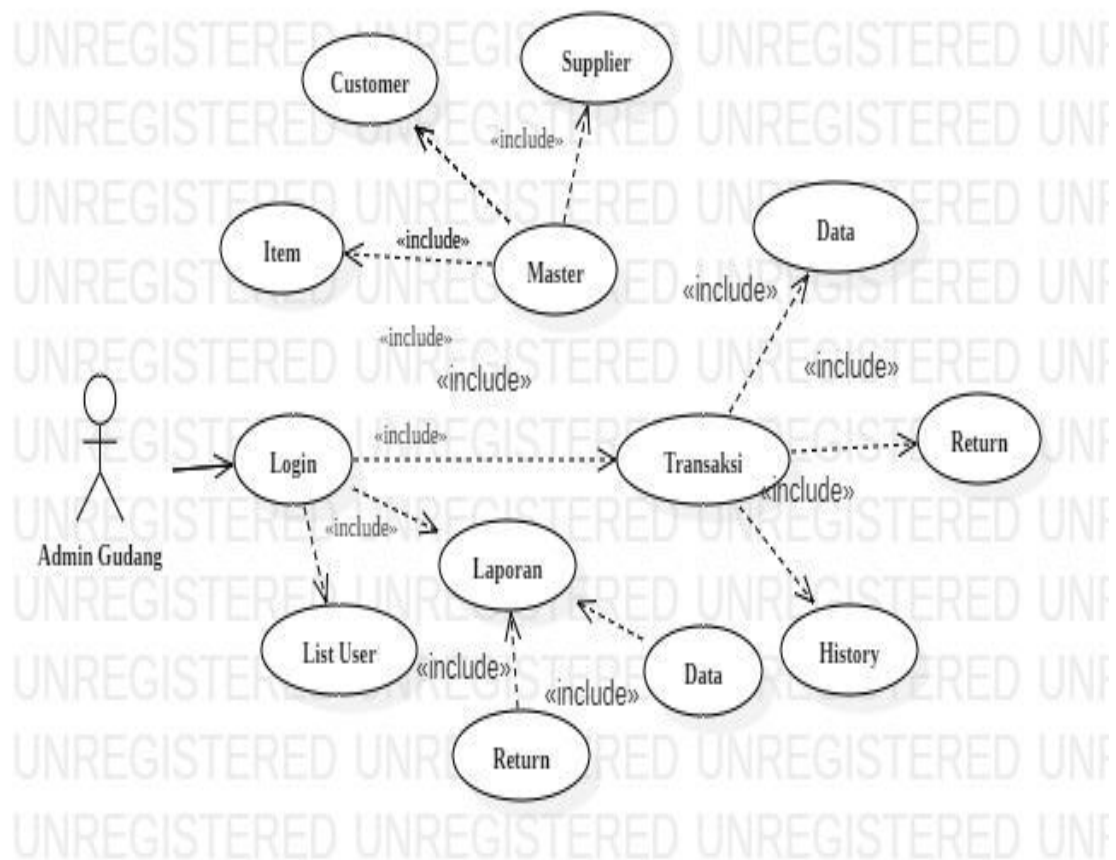
C.4 Pimpinan melihat laporan *return* pembelian

C.5 Pimpinan melihat laporan return pembelian

1. Use Case Diagram

Diagram usecase digunakan untuk mendeskripsikan apa yang seharusnya dilakukan oleh sistem. Diagram use case menyediakan cara mendeskripsikan pandangan *eksternal* terhadap sistem dan interaksi-interaksinya terhadap dunia luar. Dengan cara ini diagram use case menggantikan diagram *konteks* pada pendekatan konvensional.

1) Use Case Diagram Menu Login Admin Gudang



Gambar IV.1

Use Case Diagram Halaman Login Admin Gudang

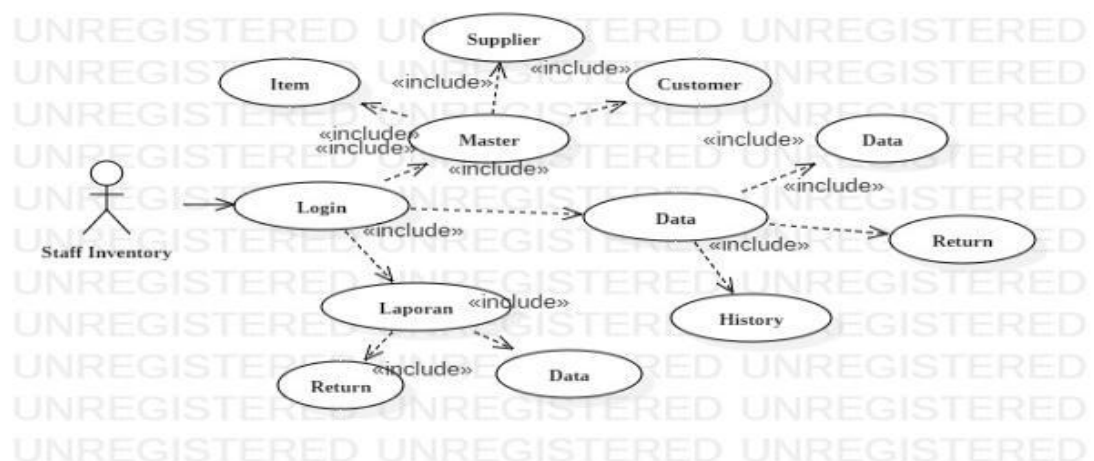
Deskripsi Use Case Diagram Menu login admin gudang

TABLE IV.I

Deskripsi Use Case Diagram Menu login admin gudang

<i>Use Case Name</i>	<i>Diagram User</i>
Requirement	A1-A10
Goal	Admin dapat mengelola data master,data,laporan dan list user
Pre-conditions	Admin telah login
Post-conditions	Admin berhasil mengelola seluruh data master, data, laporan dan list user
Failed and condition	Jika admin salah memasukkan username dan password
Primary Actors	Admin
Main Flow/Basic Path	1. Admin melakukan login. 2. Admin menambah data item. 3. Admin mengelola data customer. 4. Admin mengelola data supplier. 5. Admin mengelola data. 6. Admin mengelola return. 7. Admin mengelola history. 8. Admin mengelola laporan data. 9. Admin mengelola laporan return. 10. Admin mengelola list user.

2) Use Case Diagram Login Staff Inventory



Gambar IV.2

Use Case Diagram Login Staff Inventory

Deskripsi Use Case Diagram Menu Login Staff Inventory

3) Use Case Diagram Login Pimpinan



Use Case Diagram Login Pimpinan

Deskripsi Use Case Diagram login pimpinan

TABLE IV.3

Deskripsi Use Case Diagram Menu Login Pimpinan

Use Case Name	Diagram User
Requirement	C1-C5
Goal	Pimpinan dapat mengelola data laporan yaitu : laporan pembelian, laporan penjualan, laporan return pembelian dan laporan return penjualan.
Pre-conditions	Pimpinan telah <i>login</i>
Post-conditions	Pimpinan berhasil mengelola seluruh data laporan
Failed and condition	Jika Pimpinan salah memasukkan <i>username</i> dan <i>password</i>
Primary Actors	Pimpinan
Main Flow/Basic Path	1. Pimpinan melakukan <i>login</i>. 2. Pimpinan mengelola laporan pembelian. 3. Pimpinan mengelola laporan penjualan. 4. Pimpinan mengelola laporan return pembelian. 5. Pimpinan mengelola laporan return penjualan.

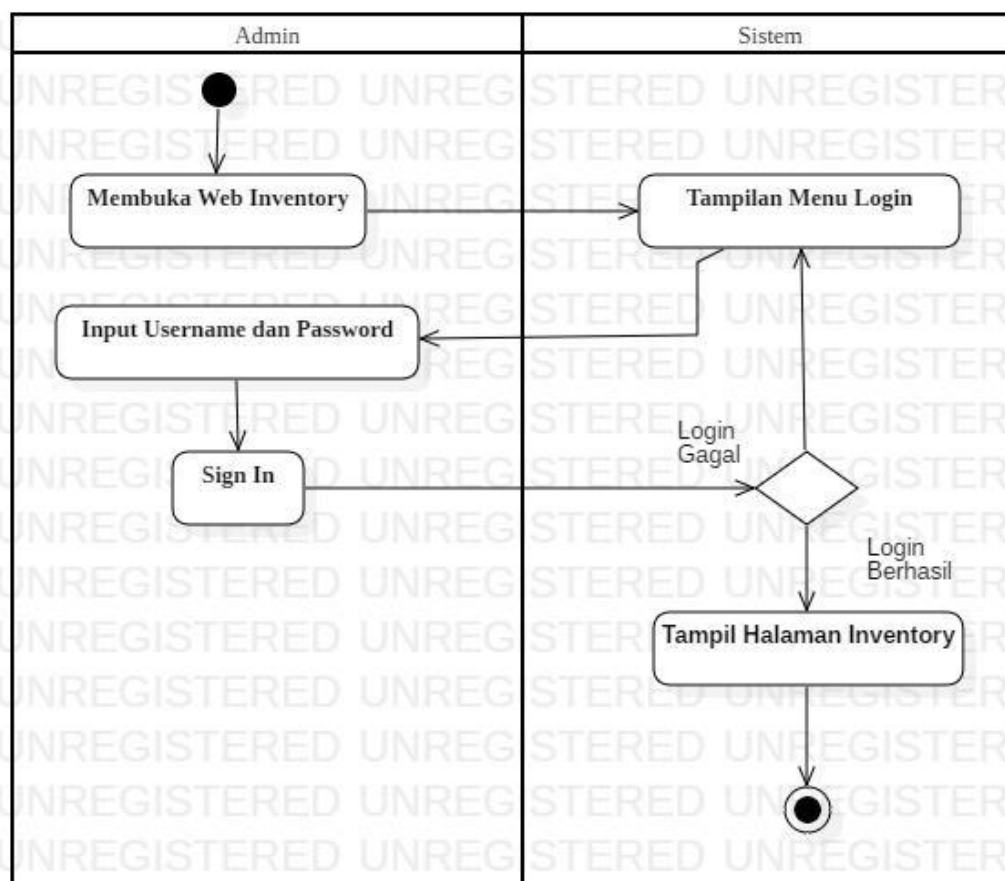
1. Activity Diagram

Activity Diagram merupakan diagram flowchart yang diperluas yang menunjukkan aliran kendali satu aktifitas ke aktifitas lain. Kita dapat menggunakan diagram ini untuk memodelkan aspek dinamis sistem. Berikut Activity Diagram sistem penjualan berbasis web yang diusulkan.

A. Admin Gudang

1. Activity Diagram Halaman Login Admin Gudang

Pada halaman ini admin akan melakukan login terlebih dahulu. Admin membuka *web inventory*, maka akan muncul tampilan halaman *login*. Input *username* dan *password*, pilih *sign in*. jika gagal tampilan akan kembali ke menu login dan jika berhasil, akan tampil tampilan *inventory* dan selesai.



Gambar IV.4
Activity Diagram Halaman Login Admin

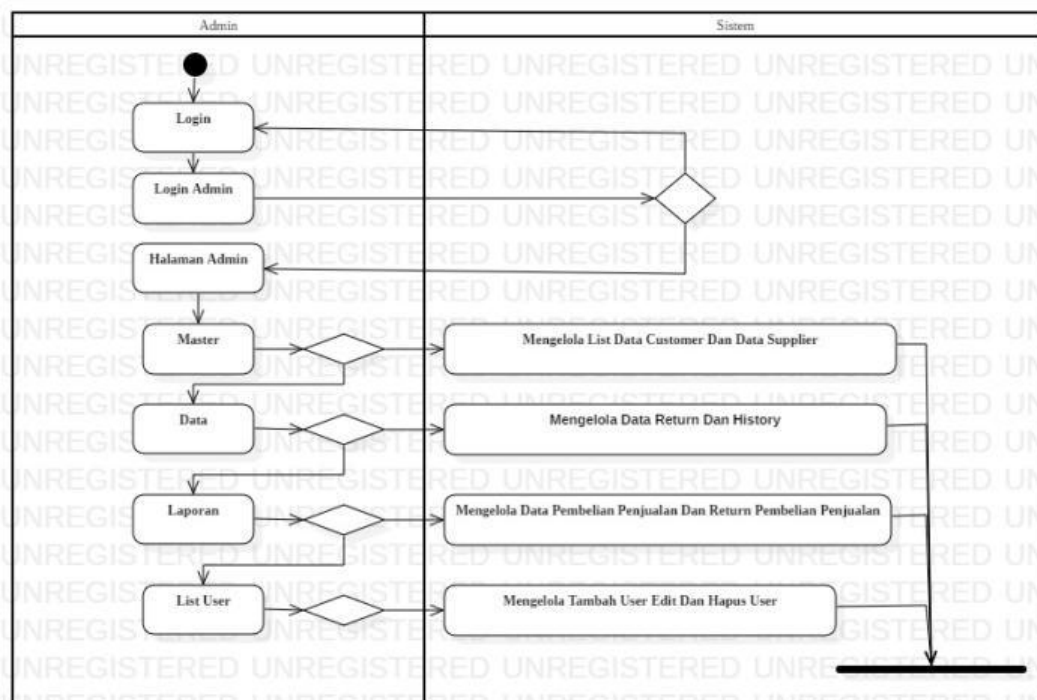
2. Activity Diagram Setelah Login Admin

Pada halaman ini admin dapat melakukan login terlebih dahulu dan masuk ke halaman admin. Kemudian admin dapat memilih menu *master*, data, laporan dan *list user*. Penjelasannya sebagai berikut :

- a. Menu Master terdiri dari halaman list merk, list katagori, *stock* barang, data *customer* dan data *supplier*. Setiap halaman memiliki tambah data, *edit* dan hapus.
- b. Menu Data terdiri dari halaman data penjualan, data pembelian, data *return* barang, data *return* penjualan, data *history* pembelian, data *history* penjualan, data *history* return pembelian dan data *history* return

penjualan. setiap masing-masing halaman bisa memiliki data tambah, *edit* dan hapus.

- c. Menu Lampiran terdiri dari halaman laporan pembelian, penjualan, *return* pembelian dan *return* penjualan. Setiap menu bisa *print* laporan.
- d. Menu *List User* digunakan untuk menambah hak akses *login*.



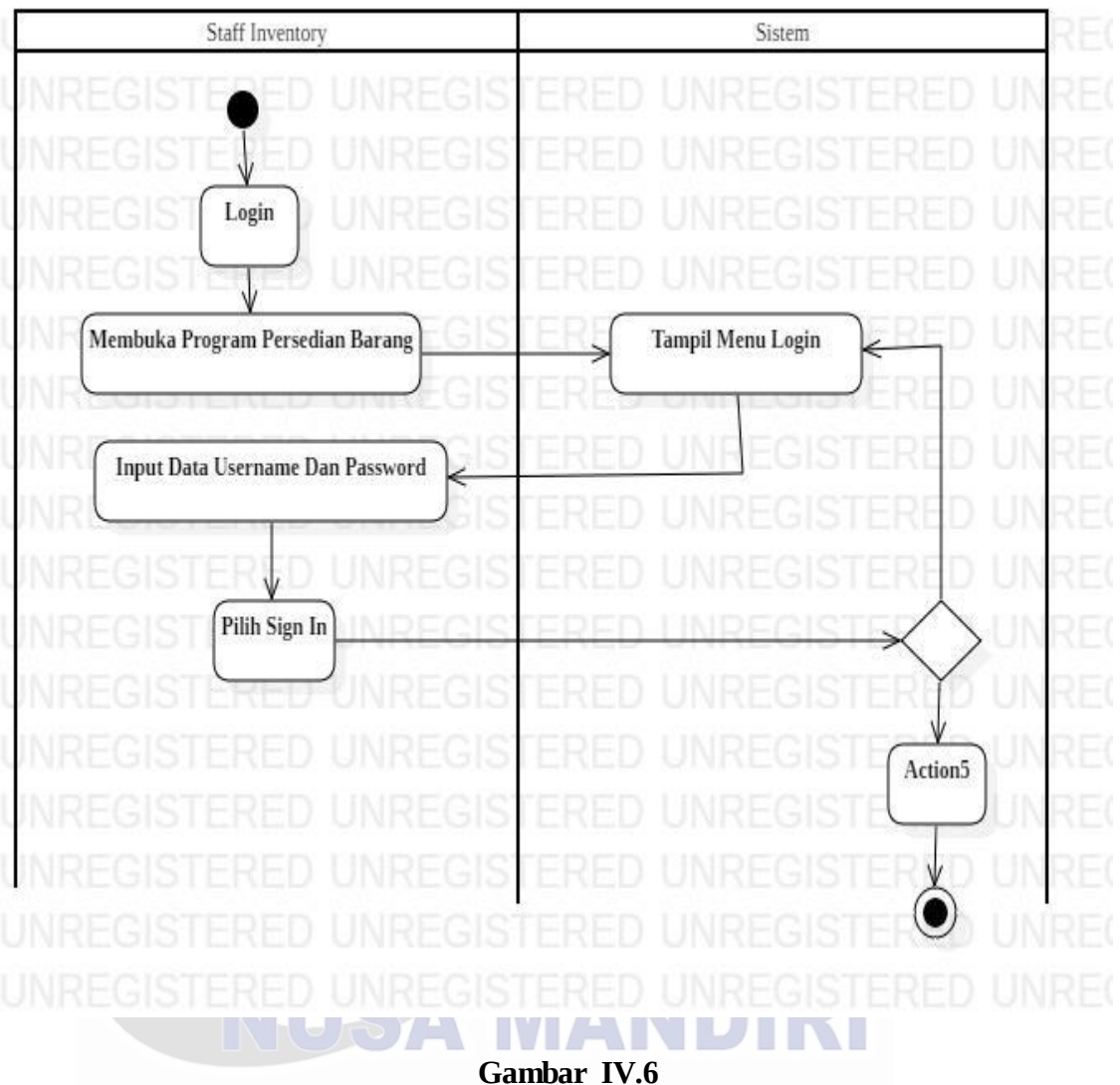
Gambar IV.5

Activity Diagram Halaman Login Menu Master

B. Staff Inventory

1. Activity Diagram login Staff Inventory

Pada halaman ini *staff inventory* akan melakukan login terlebih dahulu. *Staff inventory* membuka *web inventory*, maka akan muncul tampilan halaman *login*. Input *username* dan *password*, pilih sign in, jika gagal tampilan akan kembali ke menu *login* dan jika berhasil, akan tampil halaman *inventory* dan selesai.



Gambar IV.6

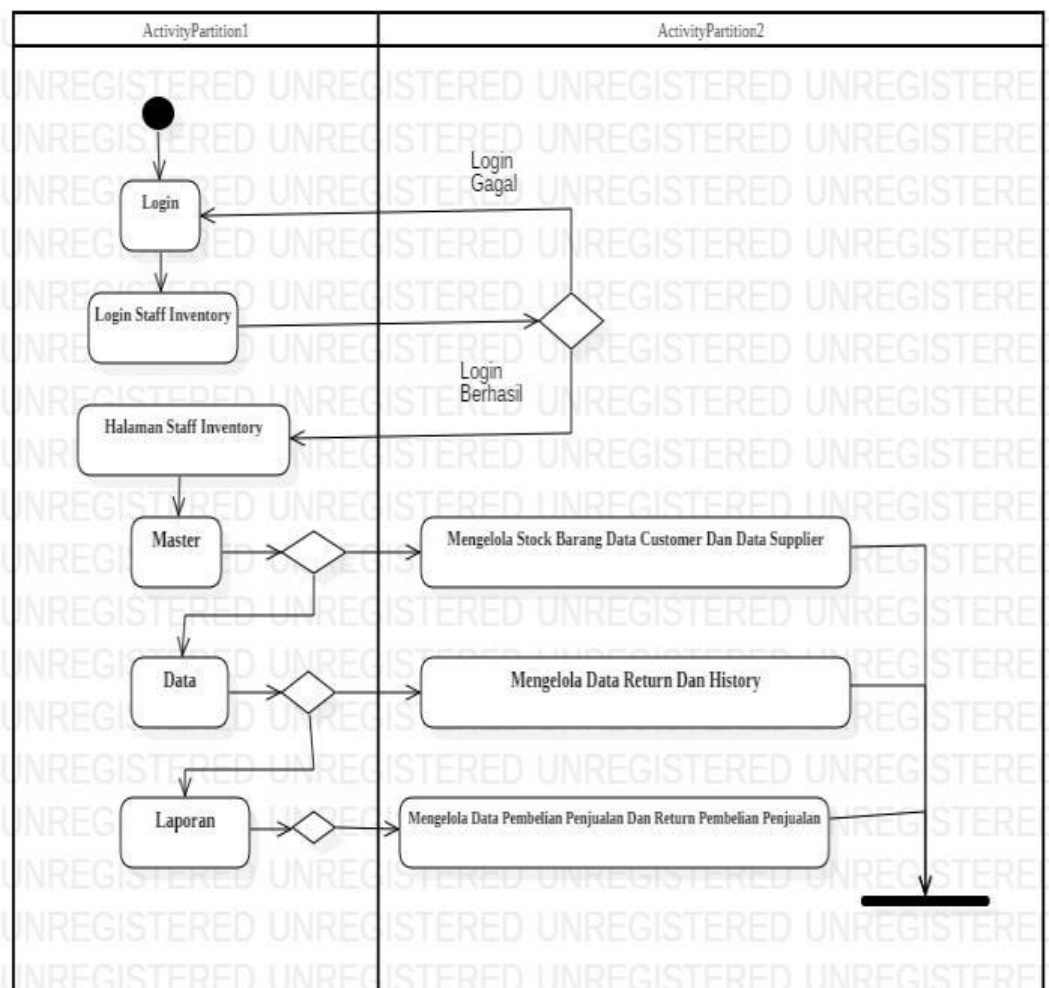
Activity Diagram Halaman Login Staff Inventory

2. Activity Diagram Setelah Login Staff Inventory

Pada halaman ini *staff inventory* dapat melakukan login terlebih dahulu dan masuk ke halaman *staff inventory*. Kemudian staff dapat memilih menu *master*, data laporan. Penjelasan nya sebagai berikut :

- a. Menu Master terdiri dari halaman *stock* barang, data *customer* dan data *supplier*. Setiap menu memiliki tambah data, *edit* dan hapus.

- b. Menu Data terdiri dari halaman data penjualan, data pembelian, data *return* barang, data *return* penjualan, data *history* pembelian, data *history* penjualan, data *history* penjualan, data *history* return pembelian dan data *history* return penjualan. Setiap masing-masing halaman bisa memiliki data tambah, *edit* dan hapus.
- c. Menu Laporan terdiri dari halaman laporan pembelian, penjualan, *return* pembelian dan *return* penjualan. Setiap menu bisa *print* laporan.



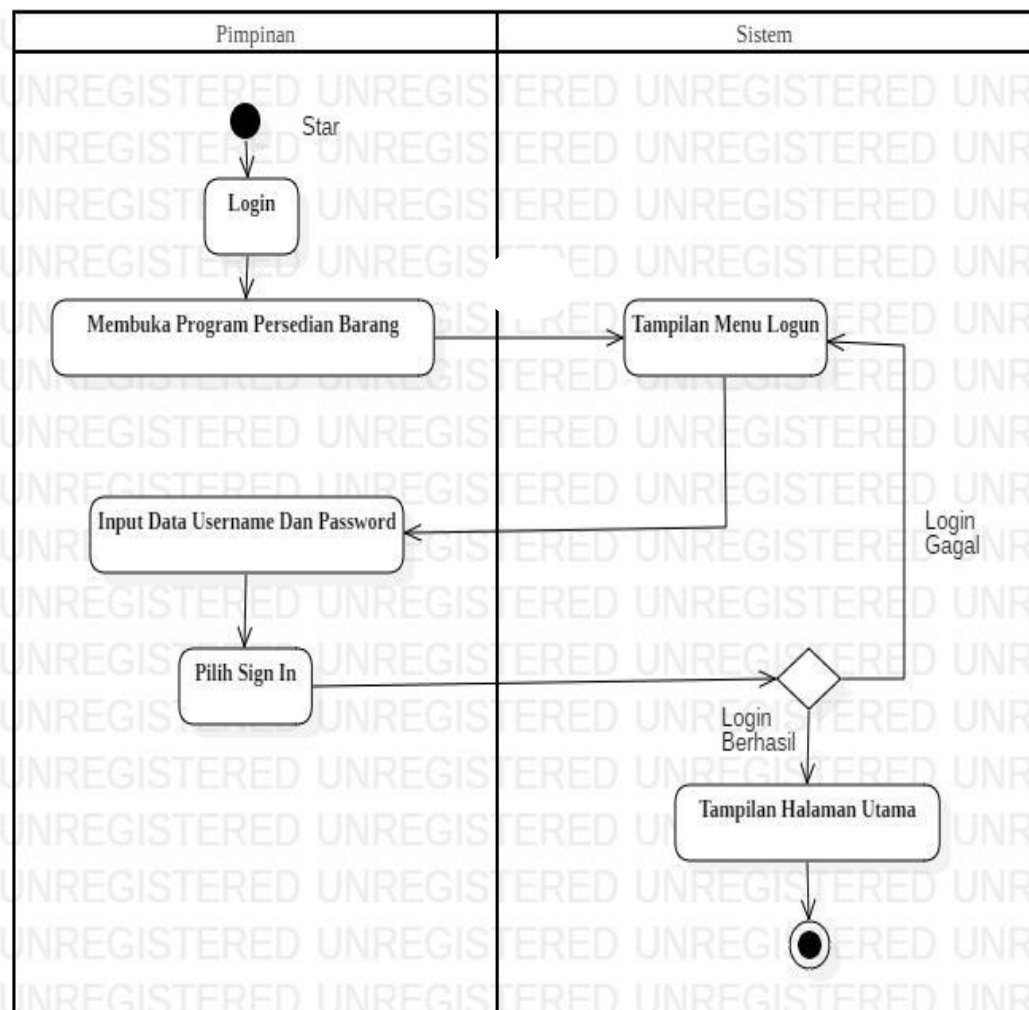
Gambar IV.7

Activity Diagram Halaman Login Staff Inventory

C. Pimpinan

1. Activity Diagram Halaman Login Pimpinan

Pada halaman ini pimpinan akan melakukan login terlebih dahulu. Pimpinan membuka web inventory, maka akan muncul tampilan halaman login. Input username dan password, pilih sign in, jika gagal tampilan akan kembali ke menu login dan jika berhasil, akan tampil halaman inventory dan selesai.

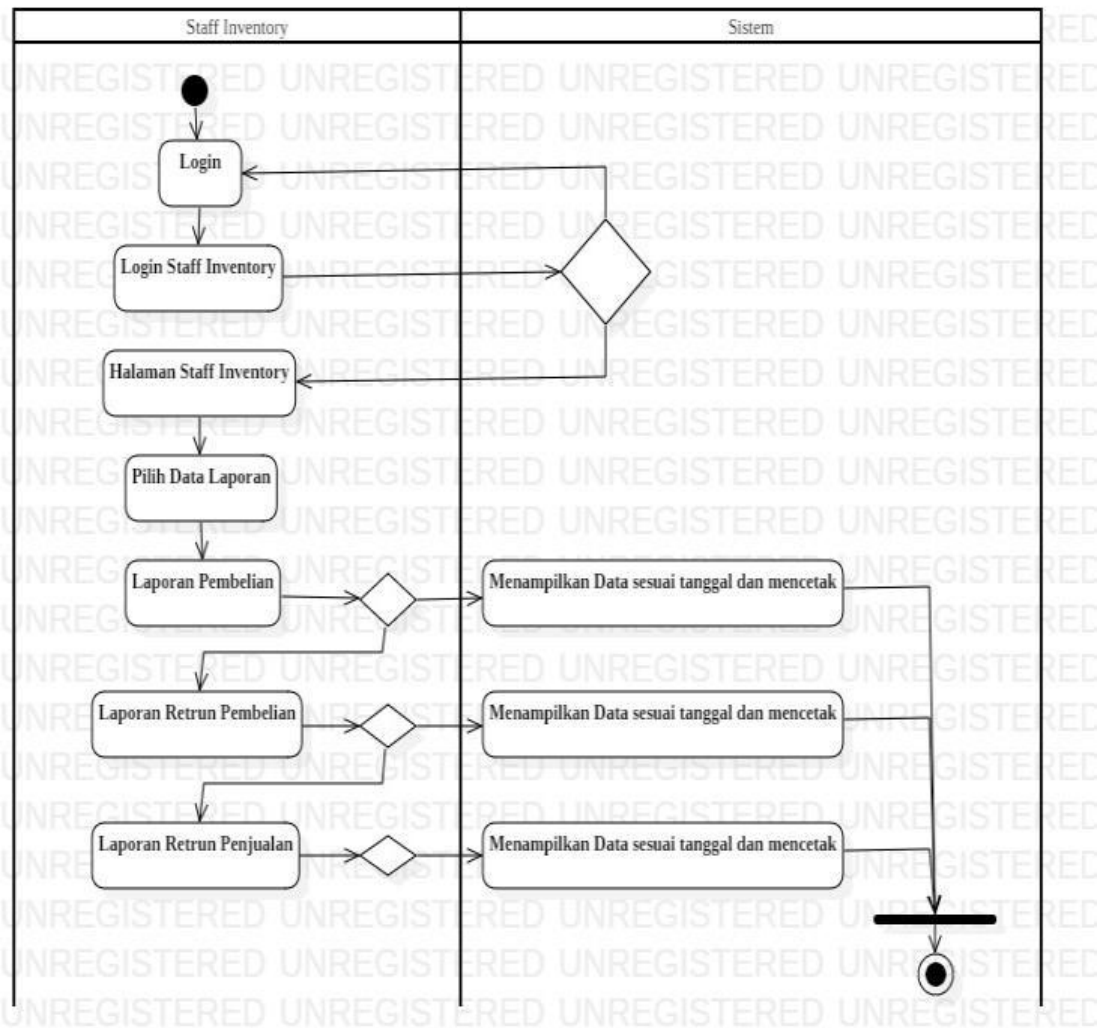


Gambar IV.8

Activity Diagram Halaman Login pimpinan

2. Activity Diagram Halaman Login Pimpinan

Pada halaman ini pimpinan dapat melakukan *login* terlebih dahulu dan masuk ke halaman *login* pimpinan kemudian pimpinan dapat memilih menu laporan. Menu Laporan terdiri dari halaman laporan pembelian, penjualan, *return* pembelian dan *return* penjualan. Setiap menu bisa *print* laporan.



Gambar IV.9

Activity Diagram Setelah Login pimpinan

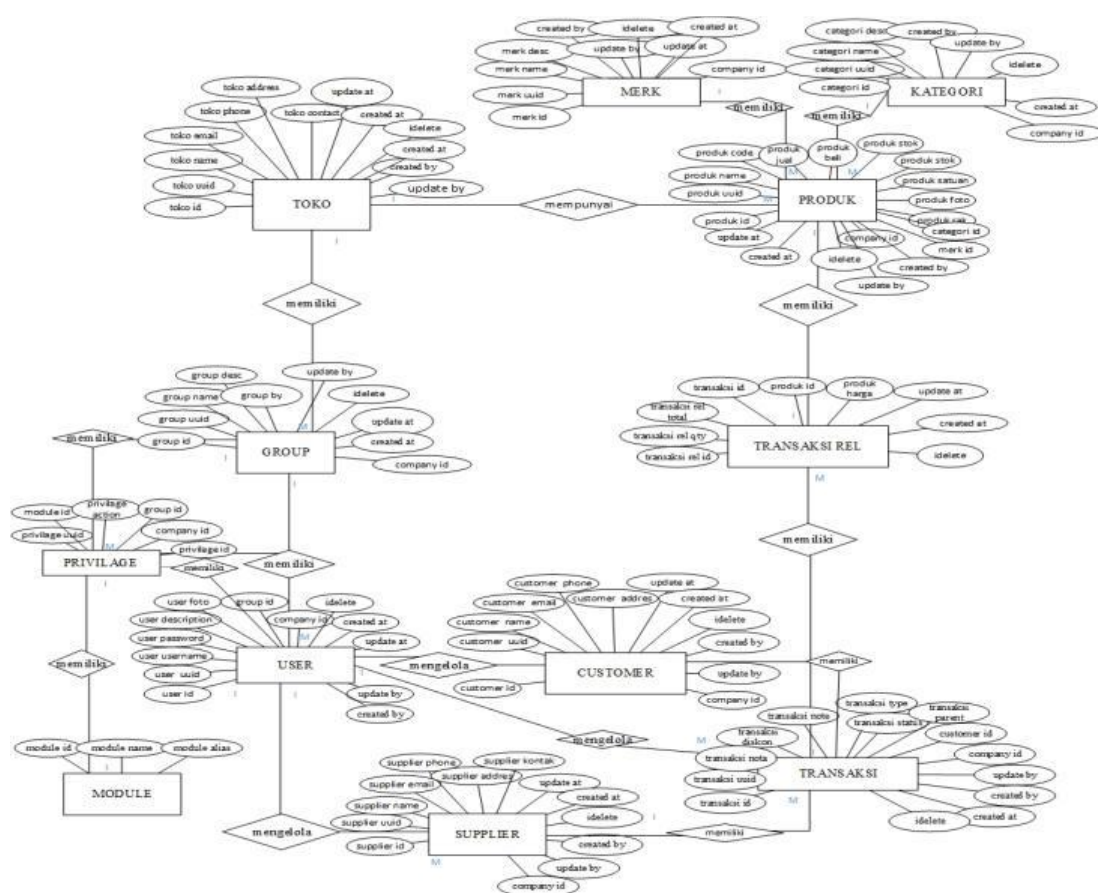
4.2 Desain

Pada tahap ini penulis akan menjelaskan tentang desain databases, desain software architecture dan desain interface dari sistem usulan yang sedang dibuat.

4.2.1 Database

Mengambarkan hubungan antar table yang dibuat beserta relasi antar table. Dalam penggambarannya penulis menggambarkan Entity Relationship Diagram (ERD). Karena dengan menggambarkan diagram ERD maka penulis juga membuat spesifikasi filenya sebagai berikut :

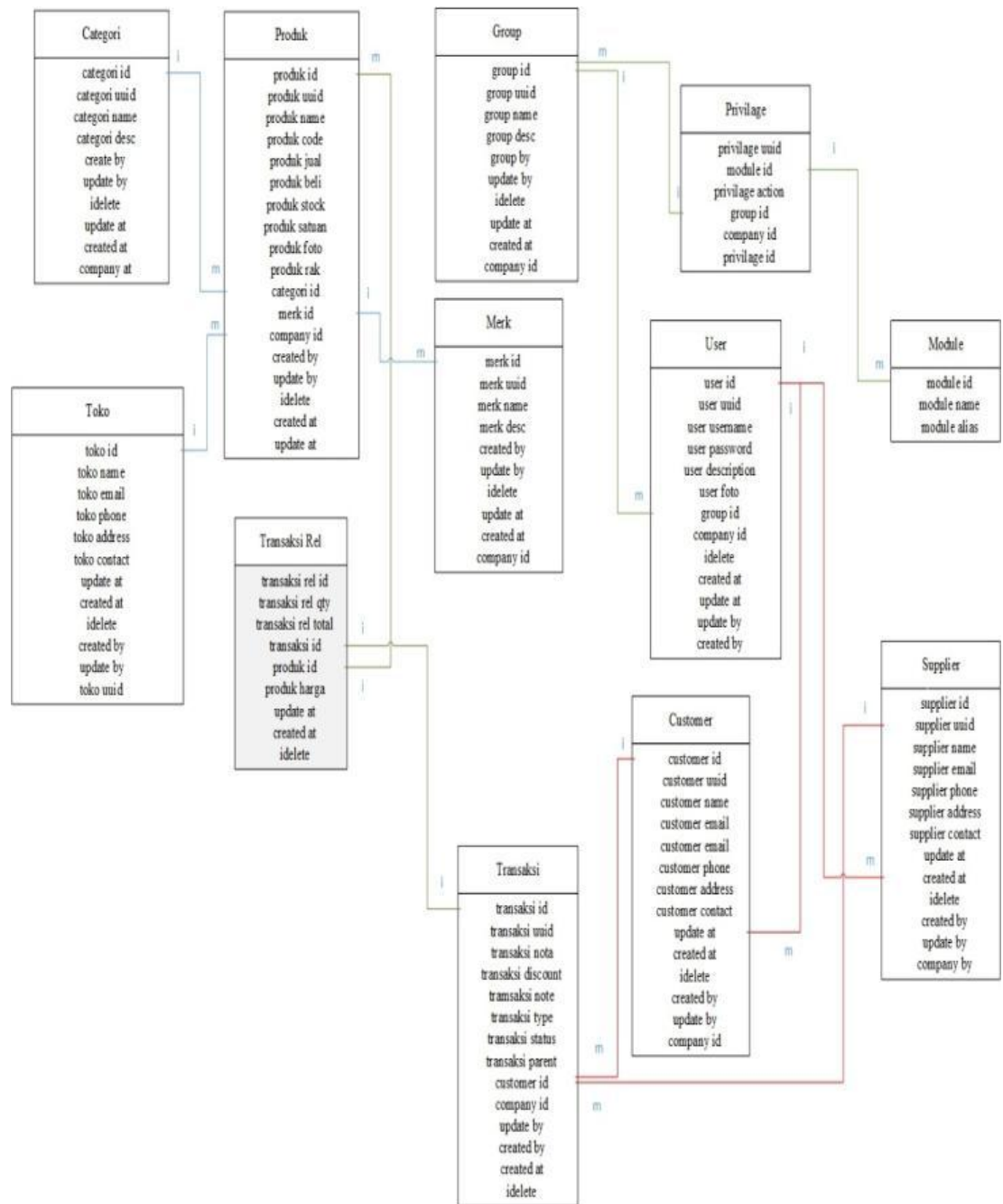
1. Entity Relationship Diagram



Gambar IV.10

Entity Relationship Diagram

2. Logical record structure



Gambar IV.11

Logical record structure

3. Spesifikasi File

a. Spesifikasi file Category

Nama Databases : Gratia

Nama File : Category

Akronim : Category.myd

Tipe File : Master

Akses file : Random

Panjang Record : 155

Kunci Field : primary key

TABEL IV.4

Spesifikasi File Tabel Category

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id kategori	kategori id	Int	11	Primary key
2	id unik kategori	kategori uuid	Varchar	50	unik key
3	nama kategori	kategori name	Varchar	50	
4	deskripsi kategori	kategori desc	Text		
5	di buat oleh	created by	int	11	
6	perbarui oleh	updated by	Int	11	
7	Hapus	Idelete	Int	11	
8	di perbarui	updated at	datetime		
9	di buat	created at	datetime		
10	id perusahaan	company id	Int	11	

b. Spesifikasi file Customer

Nama Databases : Gratia

Nama File : customer

Akronim : customer.myd

Tipe File : Master

Akses file : Random

Panjang Record : 275

Kunci Field : Primary key

TABEL IV.5

Spesifikasi File Tabel Customer

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id pelanggan	customer id	Int	11	<i>primary key</i>
2	id unik pelanggan	customer uuid	varchar	50	
3	nama pelanggan	customer name	varchar	50	
4	email pelanggan	customer email	varchar	50	
5	tlp pelanggan	customer phone	varchar	20	
6	alamat pelanggan	customer address	text		
7	kontak pelanggan	customer contact	varchar	50	
8	di perbarui	updated at	datetime		
9	di tulis	created at	datetime		
10	Hapus	Idelete	Int	11	
11	di buat oleh	created by	Int	11	
12	di perbarui oleh	updated by	Int	11	
13	id perusahaan	company id	Int	11	

c. Spesifikasi file Group

Nama Databases : Gratia

Nama File : Group

Akronim : Group.myd

Tipe File : Master

Akses file : Random

Panjang Record : 155

Kunci Field : Primary key

TABEL IV.6

Spesifikasi File Tabel Group

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id grup	group id	Int	11	<i>primary key</i>
2	id unik grup	group uuid	Varchar	50	<i>unik key</i>
3	nama grup	group name	Varchar	50	
4	deskripsi grup	group desc	Text		
5	di buat oleh	created by	int	11	
6	perbarui oleh	updated by	Int	11	
7	Hapus	Idelete	Int	11	
8	di perbarui	updated at	datetime		
9	di buat	created at	datetime		
10	id perusahaan	company id	Int	11	

d. Spesifikasi *file Merk*

Nama Databases : Gratia

Nama *File* : *Merk*

Akronim : *Merk.myd*

Tipe *File* : Master

Akses *file* : *Random*

Panjang *Record* : 155

Kunci *Field* : Primary key

TABEL IV.7

Spesifikasi File Tabel *Merk*

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id merk	merk id	Int	11	<i>primary key</i>
2	id unik merk	merk uuid	Varchar	50	<i>unik key</i>
3	nama merk	merk name	Varchar	50	
4	deskripsi merk	merk desc	Text		
5	di buat oleh	created by	int	11	
6	perbarui oleh	updated by	Int	11	
7	Hapus	Idelete	Int	11	
8	di perbarui	updated at	Datetime		
9	di buat	created at	Datetime		
10	id perusahaan	company id	Int	11	

e. Spesifikasi *file Produk*

Nama Databases : Gratia

Nama File : *Produk*

Akronim : *Produk.myd*

Tipe File : Master

Akses file : *Random*

Panjang Record : 430

Kunci Field : Primary key

TABEL IV.8

Spesifikasi File Tabel Produk

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id produk	produk id	int	11	<i>Primary key</i>
2	id unik produk	produk uuid	varchar	50	
3	nama produk	produk name	varchar	50	
4	kode produk	produk code	int	11	
5	jual produk	produk jual	int	11	
6	beli produk	produk beli	int	11	
7	stok produk	produk stock	int	11	
8	satuan produk	produk satuan	varchar	20	
9	foto produk	produk foto	varchar	100	
10	rak produk	produk rak	varchar	50	
11	id katagori	category id	int	11	
12	id merk	merk id	int	11	
13	id perusahaan	company id	int	50	
14	di buat oleh	created by	int	11	
15	perbarui oleh	updated by	int	11	
16	Hapus	Idelete	int	11	
17	di perbarui	updated at	datetime		
18	di buat	created at	datetime		

f. Spesifikasi file Supplier

Nama Databases : Gratia

Nama File : Supplier

Akronim : Supplier.myd

Tipe File : Master

Akses file : Random

Panjang Record : 275

Kunci Field : Primary key

TABEL IV.9

Spesifikasi File Tabel *Supplier*

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id supplier	supplier id	Int	11	<i>primary key</i>
2	id unik supplier	supplier uuid	varchar	50	
3	nama supplier	supplier name	varchar	50	
4	email supplier	supplier email	varchar	50	
5	tlp supplier	supplier phone	varchar	20	
6	alamat supplier	supplier address	Text		
7	kontak supplier	supplier contact	varchar	50	
8	di perbarui	updated at	datetime		
9	di buat	created at	datetime		
10	Hapus	Idelete	Int	11	
11	di buat oleh	created by	int	11	
12	perbarui oleh	updated by	Int	11	
13	id perusahaan	company id	Int	11	

g. Spesifikasi *file* Toko

Nama Databases : Gratia

Nama *File* : Toko

Akronim : Toko.myd

Tipe *File* : MasterAkses *file* : *Random*Panjang *Record* : 264Kunci *Field* : *Primary key*

TABEL IV.10

Spesifikasi File Tabel Toko

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id toko	toko id	Int	11	<i>primary key</i>
2	id unik toko	toko uuid	varchar	50	
3	nama toko	toko name	varchar	50	
4	email toko	toko email	varchar	50	
5	tlp toko	toko phone	varchar	20	
6	alamat toko	toko address	Text		
7	kontak toko	toko contact	varchar	50	
8	di perbarui	updated at	datetime		
9	di buat	created at	datetime		
10	Hapus	Idelete	Int	11	
11	di buat oleh	created by	int	11	
12	perbarui oleh	updated by	Int	11	

h. Spesifikasi file Transaksi

Nama Databases : Gratia

Nama File : Transaksi

Akronim : Transaksi.myd

Tipe File : Master

Akses file : Random

Panjang Record : 312

Kunci Field : Primary key

TABEL IV.11

Spesifikasi File Tabel

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id transaksi	transaksi id	Int	11	<i>primary key</i>
2	id unik transaksi	transaksi uuid	varchar	100	
3	nota transaksi	transaksi nota	varchar	100	
4	total transaksi	transaksi total	Int	11	
5	diskon transaksi	transaksi discount	Int	11	
6	catatan transaksi	transaksi note	Text		
7	tipe transaksi	transaksi type	varchar	3	
8	status transaksi	transaksi status	varchar	10	
9	induk transaki	transaksi parent	Int	11	
10	id pelanggan	customer id	Int	11	
11	id perusahaan	company id	Int	11	
12	di buat oleh	created by	int	11	
13	perbarui oleh	updated by	Int	11	
14	di perbarui	updated at	datetime		
15	di buat	created at	datetime		
16	Hapus	Idelete	Int	11	

i. Spesifikasi *file Transaksi rel*

Nama Databases : Gratia

Nama File : *Transaksi Rel*Akronim : *Transaksi Rel.myd*

Tipe File : Master

Akses file : *Random*

Panjang Record : 88

Kunci Field : *Primary key*

TABEL IV.12

Spesifikasi File Tabel Transaksi Rel

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id detail transaksi	transaksi rel id	int	11	<i>primary key</i>
2	kuantitas detail transaksi	transaksi rel qty	int	11	
3	total detail transaksi	transaksi rel total	int	11	
4	sisanya detail transaksi	transaksi rel remaining	int	11	
5	id transaksi	transaksi id	int	11	
6	id produk	produk id	int	11	
7	harga produk	produk harga	int	11	
8	di perbarui	updated at	datetime		
9	di buat	created at	datetime		
10	Hapus	Idelete	int	11	

j. Spesifikasi file User

Nama Databases : Gratia

Nama File : User

Akronim : User.myd

Tipe File : Master

Akses file : Random

Panjang Record : 371

Kunci Field : Primary key

TABEL IV.13

Spesifikasi File Tabel User

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	id user	user id	Int	11	<i>primary key</i>
2	id unik user	user uuid	Varchar	75	
3	nama pengguna user	user username	Varchar	30	
4	kata sandi user	user password	Varchar	100	
5	deskripsi user	user description	Text		
6	foto user	user foto	Varchar	100	
7	id grup	group id	Int	11	
8	id perusahaan	company id	Int	11	
9	Hapus	Idelete	Int	11	
10	di buat	created at	Datetime		
11	di perbarui	updated at	Datetime		
12	perbarui oleh	updated by	Int	11	
13	di buat oleh	created by	int	11	

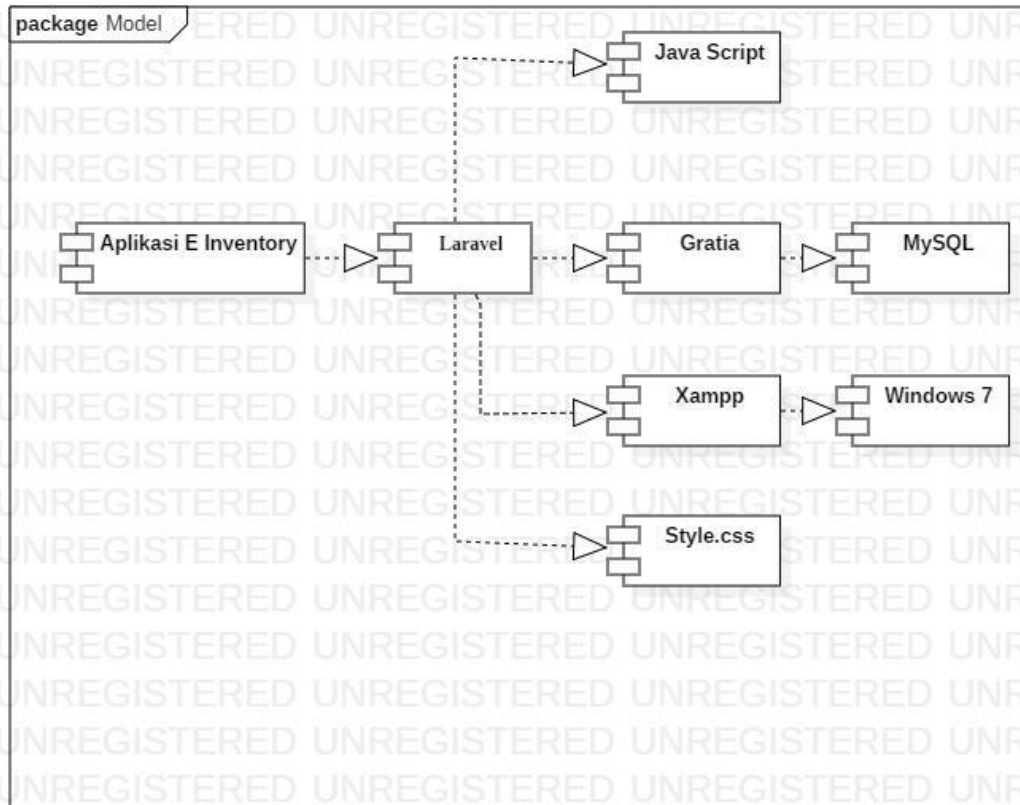
4.2.2 Software Architecture

A. Component Diagram

Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya, komponen piranti lunak adalah model berisi *code*, baik berisi *source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time*, *link time*, maupun *run time*.

Komponen dapat juga berupa interface, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

Component diagram berikut gambar diagramnya.



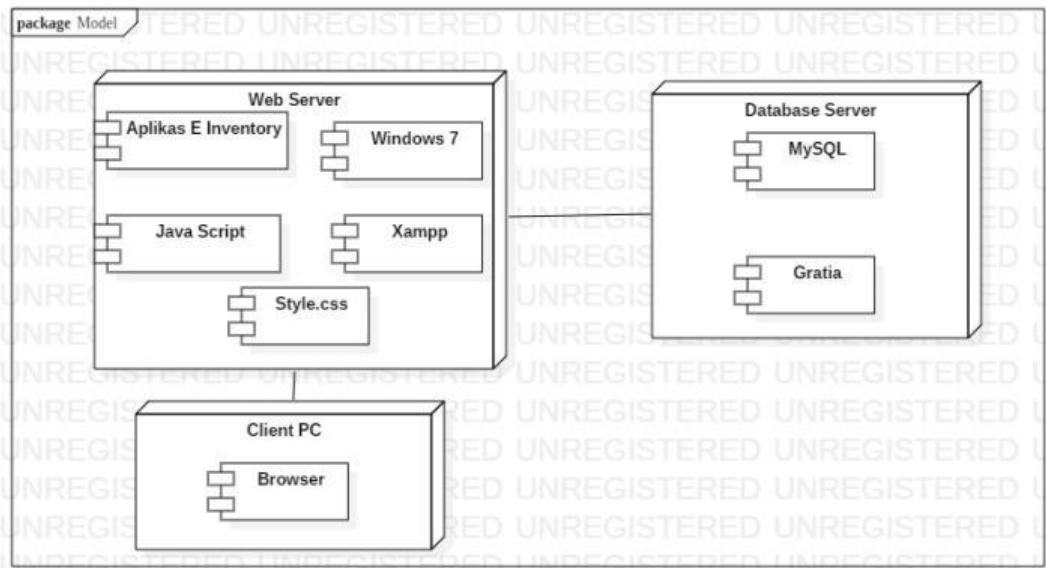
Gambar IV.12
Component diagram

B. Deployment Diagram

Diagram deployment mewakili pandangan pengembangan sistem sehingga hanya akan ada satu *diagram deployment* untuk satu sistem. *Diagram deployment* terdiri dari node dan node merupakan perangkat keras fisik yang digunakan untuk menyebarkan aplikasi.

Tiap node pada diagram *deployment* mewakili satu unit komputasi sistem yang dalam banyak hal merupakan bagian dari perangkat keras.

Deployment Diagram berikut gambar diagramnya.



Gambar IV.13

Deployment diagram

4.2.3 User Interface

Berikut adalah tampilan E-Inventory Barang Pada CV.Gratia Motorindo dari program yang diusulkan :

1. Admin Login

CV. Gratia Motorindo

Sign In To Admin

admin|

.....

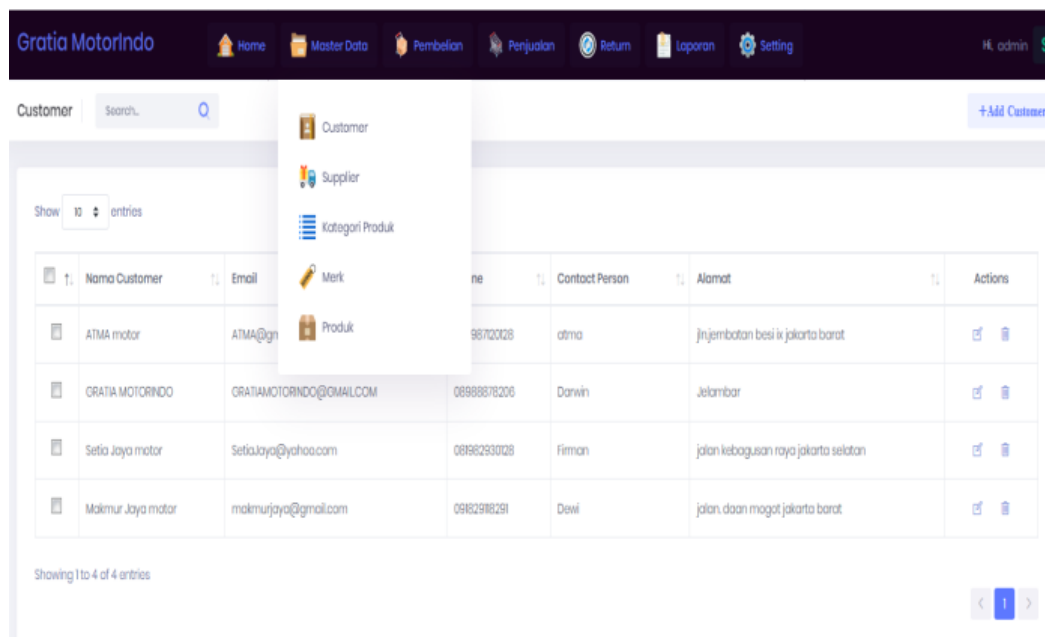
☐ Remember me [Forget Password ?](#)

Sign In

Gambar IV.14

Tampilan Halaman Login

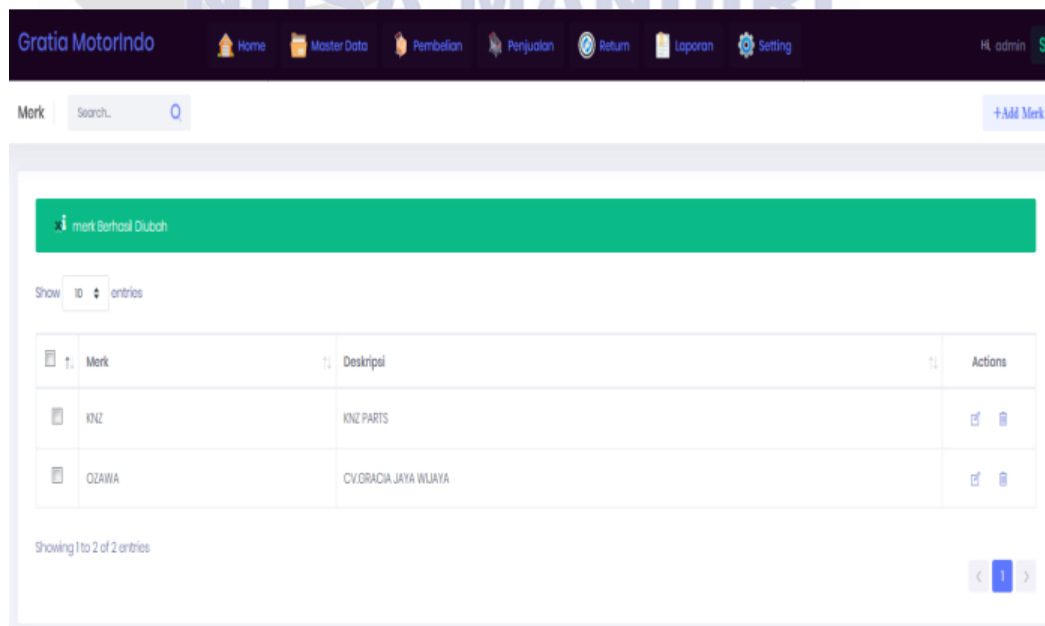
2. Tampilan Menu Data Master



Gambar IV.15

Tampilan Menu Data Master

3. Tampilan List Merk



Gambar IV.16

Tampilan Halaman List Merk

4. Tampilan List Kategori

Category Search... [+Add Category](#)

Show 10 entries

Kategori	Deskripsi	Actions
AKI	ACCU	Edit Delete
BLOCK SEHER ASSY	BLOCK SEHER	Edit Delete
BOHLAM DEPAN	BOHLAM DEPAN	Edit Delete
BOHLAM STOP	BOHLAM STOP	Edit Delete

Showing 1 to 4 of 4 entries

Gambar IV.17

Tampilan Halaman List Kategori

5. Tampilan Halaman Stock Barang

Produk Search... [+Add Produk](#)

Produk Berhasil Dubah

Show 10 entries

Kode	Nama Produk	Merk	Kategori	Harga Beli	Harga Jual	Stock	No Rak	Actions
X01	AKI semi kering GTZ-55 (tharisma) KNZ	QDAWA	AKI	Rp. 82,900	Rp. 172,500	102 box	01	Edit Delete
P02	Bahlam Depan Grand 12V 35/35W KNZ	KNZ	BOHLAM DEPAN	Rp. 1,500	Rp. 2,600	500 PCS	02	Edit Delete
P01	Block Seher Assy Mio KNZ	KNZ	BLOCK SEHER ASSY	Rp. 200,000	Rp. 345,000	22 BOX	03	Edit Delete
p03	Bahlam Stop GOLD SERIES KNZ	KNZ	BOHLAM STOP	Rp. 1,500	Rp. 2,700	0 pcs	04	Edit Delete

Showing 1 to 4 of 4 entries

Gambar IV.18

Tampilan Halaman Stock Ba

Gratia MotorIndo Home Master Data Pembelian Penjualan Return Laporan Setting HI, admin

Tambah Produk [Save Produk](#)

Kode: Silahkan Input Kode Produk

Nama Produk: Silahkan Input Nama Produk

Mark: Silahkan Pilih Mark Produk

Kategori: Silahkan Pilih Kategori Produk

Harga Beli: Silahkan Input Harga Beli Produk

Harga Jual: Silahkan Input Harga Jual Produk

Stock: Silahkan Input Stock Produk

Satuan: Silahkan Input Satuan Produk

No Rak: Silahkan Input No Rak Produk

Foto: [Browse](#) Silahkan Input Foto Produk

Gambar IV.19

Tampilan Tambah Produk

6. Tampilan Halaman Data Customer

Gratia MotorIndo Home Master Data Pembelian Penjualan Return Laporan Setting HI, admin

Customer [+Add Customer](#)

Show 10 entries

	Nama Customer	Email	Phone	Contact Person	Alamat	Actions
	ATMA motor	ATMA@gmail.com	08998720028	atma	Jln jembatan besi ix jakarta barat	Edit Delete
	GRATIA MOTORINDO	GRATIAMOTORINDO@GMAIL.COM	08988878206	Darwin	Jelambar	Edit Delete
	Setia Jaya motor	SetiaJaya@yahoo.com	08182930028	Firman	Jalan kebogusan raya jakarta selatan	Edit Delete
	Makmur Jaya motor	makmurjaya@gmail.com	08182918291	Dewi	Jalan daan mogot jakarta barat	Edit Delete

Showing 1 to 4 of 4 entries

Gambar IV.20

Tampilan Halaman Data Customer

Gratia MotorIndo

Home Master Data Pembelian Penjualan Return Laporan Setting Hi, admin

Tambah Customer Save Customer

Nama Customer:

Silahkan Input Nama Customer

Email:

Silahkan Input Email Customer

Phone:

Silahkan Input No Telephone Customer

Contact Person:

Silahkan Input Contact Person Customer

Alamat:

Silahkan Input Customer Alamat

Gambar IV.21

Tampilan Tambah Customer

7. Tampilan Halaman Data Supplier

Gratia MotorIndo

Home Master Data Pembelian Penjualan Return Laporan Setting Hi, admin

Supplier +Add Supplier

Show 10 entries

	Nama Supplier	Email	Phone	Contact Person	Alamat	Actions
	KNZ	knzpart@gmail.com	08912828301	Lia	Jakarta	
	cv.gracia jaya wijaya	gracia@gmail.com	0828902922	Firman	jakarta	
	SG AUTOPARTA	SG12@gmail.com	0881293882093	Dewi	jakarta	
	DKI	dkl82@gmail.com	081272830729	Deni	jakarta	

Showing 1 to 4 of 4 entries

< 1 >

Gambar IV.22

Tampilan Halaman Data Supplier

Gratia Motorindo Home Master Data Pembelian Penjualan Return Laporan Setting HL admin S

Tambah Supplier Save Supplier

Nama Supplier:
Silahkan Input Nama Supplier

Email:
Silahkan Input Email Supplier

Phone:
Silahkan Input No Telephone Supplier

Contact Person:
Silahkan Input Contact Person Supplier

Alamat:
Silahkan Input Alamat & Alamat

Gambar IV.23

Tampilan Tambah Supplier

8. Tampilan Menu Data Pembelian

Gratia Motorindo Home Master Data Pembelian Penjualan Return Laporan Setting HL admin S

Pembelian Search... + Add Pembelian

Purchase Order
Purchase Invoice

Show 10 entries

	Tanggal	No Nota	Supplier	Status	Total	Actions
	2019-08-15 11:59:48	POB-1508160004	KIZ	finish	Rp. 10.975.000	
	2019-08-15 11:55:51	POB-1508160003	KIZ	finish	Rp. 2.000.000	
	2019-08-15 11:05:43	POB-1508160002	cv gratia jaya wijaya	finish	Rp. 1.500.000	
	2019-08-15 10:58:10	POB-1508160001	KIZ	finish	Rp. 880.000	

Showing 1 to 4 of 4 entries

Gambar IV.24

Tampilan Menu Data Pembelian

Gratia MotorIndo Home Master Data Pembelian Penjualan Return Laporan Setting Hi, admin

Tambah Pembelian

Informasi Nota

No. Nota

POB-180890001

Informasi Supplier

Supplier Tambah Baru ?

-- Umum --

Telp / HP
Tidak ada

Alamat
Tidak ada

09:18:07 PM Refresh Halaman

#	Kode Barang	Nama Barang	Harga	Qty	Sub Total	
1	Ketik Kode / Nama Barang					X

+ Baris Baru (F7) **Total : Rp. 0**

Catatan Transaksi (Jika Ada)

Simpan (F10)

Shortcut Keyboard :
F7 = Tambah baris baru
F10 = Simpan Transaksi

Gambar IV.25

Tampilan Data Pembelian

9. Tampilan Menu Data Penjualan

Gratia MotorIndo Home Master Data Pembelian **Penjualan** Return Laporan Setting Hi, admin

Penjualan Search...

+ Add Penjualan

Show 10 entries

	Tanggal	No Nota	Customer	Status	Total	Actions
☐	2019-08-15 12:01:57	POJ-150890005	ATMA motor	finish	Rp. 1,725,000	
☐	2019-08-15 11:17:32	POJ-150890004	Setia Jaya motor	pending	Rp. 3,450,000	Print Edit Delete
☐	2019-08-15 11:12:37	POJ-150890003	Makmur Jaya motor	finish	Rp. 1,867,500	
☐	2019-08-15 11:06:26	POJ-150890002	Umum	finish	Rp. 3,000,000	
☐	2019-08-15 10:58:53	POJ-150890001	ATMA motor	finish	Rp. 1,335,000	
☐	2019-08-13 14:39:01	POJ-130890001	Umum	pending	Rp. 100,000	Print Edit Delete
☐	2019-08-05 15:15:20	POJ-050890001	ATMA motor	finish	Rp. 650,000	

Showing 1 to 7 of 7 entries

Gambar IV.26

Tampilan Menu Data Penjualan

Gratia MotorIndo

Home Master Data Pembelian Penjualan Return Laporan Setting HI admin

Tambah Penjualan

09:20:59 PM Refresh Halaman

#	Kode Barang	Nama Barang	Harga	Qty	Sub Total	
1	Ketik Kode / Nama Barang					X

+ Baris Baru (F7) Total : Rp. 0

Catatan Transaksi (Jika Ada) Simpan (F10)

Shortcut Keyboard :
 F7 = Tambah baris baru
 F10 = Simpan Transaksi

Informasi Nota

No. Nota
 POJ-1808180001

Informasi Customer

Customer Tambah Baru ?
 -- Umum --

Telp / HP
 Tidak ada

Alamat
 Tidak ada

Gambar IV.27

Tampilan Data Penjualan

10. Tampilan Halaman Return

Gratia MotorIndo

Home Master Data Pembelian Loading... Return Laporan Setting HI admin

Tambah Sales Return

09:23:33 PM Refresh Halaman

#	Kode Barang	Nama Barang	Harga	Qty	Sub Total	
1	Ketik Kode / Nama Barang					X

+ Baris Baru (F7) Total : Rp. 0

Catatan Transaksi (Jika Ada) Simpan (F10)

Shortcut Keyboard :
 F7 = Tambah baris baru
 F10 = Simpan Transaksi

Informasi Nota

No. Nota
 SR-1808180001

Informasi Customer

Customer Tambah Baru ?
 -- Umum --

Telp / HP
 Tidak ada

Alamat
 Tidak ada

Gambar IV.28

Tampilan Halaman Return

11. Tampilan History Penjualan

Gratia MotorIndo

Home Master Data Pembelian Penjualan Return Laporan Setting

Hi, admin

Sales Invoice

Search...

+Add Sales Invoice

Show 10 entries

#	Tanggal	No Nota	No Ref SO	Customer	Status	Total	Actions
1	2019-08-15 12:02:16	SI-1508190004	SO-1508190004	ATMA motor	finish	Rp.1725,000	 
2	2019-08-15 11:25:2	SI-1508190003	SO-1508190003	Makmur Jaya motor	finish	Rp.1887500	 
3	2019-08-15 10:06:41	SI-1508190002	SO-1508190002	Umum	finish	Rp.3,000,000	 
4	2019-08-15 10:58:09	SI-1508190001	SO-1508190001	ATMA motor	finish	Rp.1335,000	 
5	2019-08-05 15:15:37	SI-0508190001	SO-0508190001	ATMA motor	finish	Rp.650,000	 

Showing 1 to 5 of 5 entries

Gambar IV.29

Tampilan History Penjualan

12. Tampilan Menu Laporan

Gratia MotorIndo

Home Master Data Pembelian Penjualan Return Laporan Setting

Hi, admin

Laporan Penjualan

Dari: hh / bb / tttt

Sampai: hh / bb / tttt

Cetak Cancel

- Laporan Users
- Laporan Penjualan
- Laporan Pembelian
- Laporan Sales Return
- Laporan Purchase Return

Gambar IV.30

Tampilan Menu Laporan

13. Tampilan Menu Halaman Laporan Pembelian


CV.Gratia Motorindo

PT. Gratia Motorindo
alamat: Pergudangan Cetrul Kosambi blog H2N-H2M

No	Tanggal	No Nota	No Ref PO	Supplier	Status	Total
1	2019-08-31 08:19:46	PI-3108190001	POB-3108190001	KNZ	finish	Rp. 829,000
2	2019-08-26 11:52:50	PI-2608190001	POB-2608190001	Umum	finish	Rp. 82,900
3	2019-08-15 12:00:16	PI-1508190004	POB-1508190004	KNZ	finish	Rp. 18,975,000
4	2019-08-15 11:15:56	PI-1508190003	POB-1508190003	KNZ	finish	Rp. 2,000,000
5	2019-08-15 11:05:48	PI-1508190002	POB-1508190002	cv.gracia jaya wijaya	finish	Rp. 1,500,000
6	2019-08-15 10:58:16	PI-1508190001	POB-1508190001	KNZ	finish	Rp. 890,000
Total :						Rp. 24,276,900

Gambar IV.31

Tampilan Menu Halaman Laporan Pembelian

14. Tampilan List User

Gratia Motorindo
[Home](#)
[Master Data](#)
[Pembelian](#)
[Penjualan](#)
[Retur](#)
[Laporan](#)
[Setting](#)
Hi, admin S

User [+ Add User](#)

Show 10 entries

	Username	Group	Deskripsi	Actions
	pimpinan	Pimpinan	pimpinan level	Edit Delete
	admin	Administrator	admin	Edit Delete
	knzadmin	Pimpinan	asdasd	Edit Delete

Showing 1 to 3 of 3 entries

<
1
>

Gambar IV.31

Tampilan Menu List User

4.3 Code Generation

Karena program yang dibuat adalah program yang terstruktur, maka pada tahapan ini penulis akan menampilkan listin program pada form yang terkait dengan proses bisnis utamanya. Berikut ini adalah code generation pada E-Inventory barang pada CV.Gratia Motorindo Jakarta Barat.

a) Menu Login

```
<?php
```

```
namespace App\Http\Controllers;
```

```
use Illuminate\Http\Request;
```

```
use App\Users;
```

```
use App\Modules\Karyawan\Models\Karyawan;
```

```
use Illuminate\Support\Facades\Session;
```

```
use Illuminate\Support\ServiceProvider;
```

```
use Illuminate\Support\Facades\Hash;
```

```
class AuthController extends Controller
```

```
{
```

```
    public function index(Request $request){
```

```
        if ($request->session()->has('iid_users')){
```

```
            return redirect('dashboards');
```

```
        }
```

```
        return view('login');
```

```
    }
```

```

public function login(Request $request)
{
    $data = Users::where('user_username', $request->username)->first();

    if (count($data)>0)
    {
        if (Hash::check($request->password , $data->user_password))
        {
            $request->session()->put(['iid_users' => $data->user_id,
                                     'uuid_users' => $data->user_uuid,
                                     'username' => $data->user_username,
                                     'group_id' => $data->group_id,
                                     'company_id' => $data->company_id,
                                     'cfoto_users' => $data->user_foto
            ]);

            return redirect('dashboards');
        }
    }
    else
    {
        $message = 'password salah';
    }
}
else

```

```
{
    $message = 'username salah';
}
```

```
return redirect('/')->with('message', $message)
```

```
->with('id', "alert-danger")
```

```
->with('icon', "fa fa-info")
```

```
->with('class', "alert alert-danger");
```

```
}
```

```
public function logout(Request $request)
```

```
{
```

```
    $request->session()->flush();
```

```
    return redirect('/');
```

```
}
```

```
}
```

b) Menu Master

```
<?php
```

```
namespace App\Modules\Supplier\Controllers;
```

```

use Illuminate\Http\Request;

use App\Http\Controllers\Controller;

use Yajra\DataTables\DataTables;

use Ramsey\Uuid\Uuid;

use Session;

class SupplierController extends Controller
{
    var $module = 'Supplier';

    var $table_module = 'Supplier';

    function __construct(){
        $this->middleware('login');
    }

    public function index()
    {
        $primarykey = 'supplier_id';

        $fieldlist = array('Nama Supplier' => 'supplier_name',
                           'Email' => 'supplier_email',
                           'Phone' => 'supplier_phone',
                           'Contact Person' => 'supplier_contact',
                           'Alamat' => 'supplier_address',
                           'Action ' => 'actions'
                           );
    }

```

```
$view = 'layouts.list-data';
```

```
$action = url('supplier/create');
```

```
$view_data = [
```

```
    'key' => $primarykey,
```

```
    'fieldlist' => $fieldlist,
```

```
    'apilink'  => 'supplier/api',
```

```
    'module'   => ucfirst($this->table_module),
```

```
    'action'   => $action,
```

```
];
```

```
if (view()->exists($view)) {
```

```
    return view($view, $view_data)->render();
```

```
}
```

```
}
```

```
function apiSupplier()
```

```
{
```

```
    $sys_api = sys_api($this->module);
```

```
    $get_supplier = $sys_api->get_all_supplier();
```

```
    $var = Datatables::of($get_supplier)
```

```
        ->addColumn('actions', function($get_supplier){
```

```
            return '<a href="'.URL('supplier/edit/'.$get_supplier['supplier_uuid']).'"
```

```
class="btn btn-sm btn-clean btn-icon btn-icon-md" title="Edit">
```

```

        <i class="la la-edit"></i>

        </a>&nbsp;

        <a href="'.URL('supplier/delete/'.$get_supplier['supplier_uuid']).'"

class="btn btn-sm btn-clean btn-icon btn-icon-md" title="View">

        <i class="la la-trash"></i>

        </a>;

    })

    ->rawColumns(['action'])

    ->escapeColumns(['action'])

    ->make(true);

return $var;
}

/**
 * Show the form for creating a new resource.
 *
 * @return \Illuminate\Http\Response
 */

public function create()
{
    $action    = array('link' => url('supplier/store'),
        'name' => 'Tambah');

    $data = array('action'    => $action,

```

```

'module'    => $this->module,

'data'      => array('supplier_id' => "));

```

```

$view = 'Supplier::form';

```

```

if (view()->exists($view)) {

    return view($view, $data)->render();

}

}

```

```

/**

```

```

 * Store a newly created resource in storage.

```

```

 *

```

```

 * @param \Illuminate\Http\Request $request

```

```

 * @return \Illuminate\Http\Response

```

```

 */

```

```

public function store(Request $request)

```

```

{

```

```

    $input = $request->all();

```

```

    $company_id = Session::get('company_id');

```

```

    $data = [

```

```

        'supplier_uuid'      => Uuid::uuid4(),

```

```

        'supplier_name'      => $input['nama'],

```

```

        'supplier_email'     => $input['email'],

```

```

        'supplier_phone'     => $input['phone'],

```

```

'supplier_contact'      => $input['contact'],
'supplier_address'      => $input['address'],
'company_id'           => $company_id
];

$sys_api = sys_api($this->module);
$result = $sys_api->save_supplier($data);

if ($result == true)
{
    $message = "Supplier Berhasil Ditambahkan";
    return redirect('/supplier')->with('message', $message)
        ->with('id', "alert-success")
        ->with('icon', "fa fa-info")
        ->with('class', "alert alert-success");
}
else
{
    $message = "Supplier gagal ditambahkan";
    return redirect('/supplier/create')->with('message', $message)
        ->with('id', "alert-danger")
        ->with('icon', "fa fa-info")
        ->with('class', "alert alert-danger");
}

```

```

/**
 * Display the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function show($id)
{

}

/**
 * Show the form for editing the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function edit($uuid)
{
    $sys_api = sys_api($this->module);
    $company_id = Session::get('company_id');
    $find      = $sys_api->get_one_supplier($uuid, $company_id);

    $action    = array('link' => url('/supplier/update/'.$uuid),
                       'name' => "Edit");

```

```

$data      = array('action'    => $action,
                   'module'    => $this->module,
                   'data'      => $find,
                   'method'    => 'PATCH',
                   'lastview'  => isset($request['lastview'])? $request['lastview']:''
                );

```

```

$view = 'Supplier::form';

```

```

if (view()->exists($view)) {
    return view($view, $data)->render();
}
}

/**
 * Update the specified resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function update(Request $request, $uuid)
{
    $input = $request->all();

    $data = [

```

```

'supplier_name'      => $input['nama'],
'supplier_email'     => $input['email'],
'supplier_phone'     => $input['phone'],
'supplier_contact'   => $input['contact'],
'supplier_address'   => $input['address'],
];

```

```
$sys_api = sys_api($this->module);
```

```
$result = $sys_api->update_supplier($data, $uuid);
```

```

if ($result == true)
{
    if ($request['lastview'] != "")
    {
        $message = "Supplier Berhasil Diubah";

        return redirect('/supplier/detail/'.$id)->with('message', $message)

            ->with('id', "alert-success")

            ->with('icon', "fa fa-info")

            ->with('class', "alert alert-success");
    }
    else
    {
        $message = "Supplier Berhasil Diubah";

        return redirect('/supplier')->with('message', $message)

            ->with('id', "alert-success")

```

```

        ->with('icon','fa fa-info")

        ->with('class',"alert alert-success");

    }

}

else

{

    if ($request['lastview'] != "")

    {

        $message = "Supplier Gagal Diubah";

        return redirect('/supplier/detail/'.$id)->with('message', $message)

            ->with('id', "alert-success")

            ->with('icon',"fa fa-info")

            ->with('class',"alert alert-success");

    }

    else

    {

        $message = "Supplier gagal diubah";

        return redirect('/supplier/edit/'.$id)->with('message', $message)

            ->with('id', "alert-danger")

            ->with('icon',"fa fa-info")

            ->with('class',"alert alert-danger");

    }

}

}

}

```

```

/**
 * Remove the specified resource from storage.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

public function destroy($uuid)
{
    $data = [
        'delete' => 1,
    ];

    $sys_api = sys_api($this->module);
    $result = $sys_api->update_supplier($data, $uuid);

    if ($result == true)
    {
        $message = "Supplier Berhasil Dihapus";

        return redirect('/supplier')->with('message', $message)
            ->with('id', "alert-success")
            ->with('icon', "fa fa-info")
            ->with('class', "alert alert-success");
    }

    else
    {

```

```

$message = "Supplier gagal dihapus";

return redirect('/supplier')->with('message', $message)

    ->with('id', "alert-danger")

    ->with('icon', "fa fa-info")

    ->with('class', "alert alert-danger");

}

}

public function ceksupplier(Request $request)
{
    $input = $request->all();
    $id = $input['id_pelanggan'];
    $company_id = Session::get('company_id');

    $sys_api = sys_api($this->module);

    $find      = $sys_api->get_one_supplier_by_id($id, $company_id);

    if (count($find) > 0)
    {
        $find = $find->toArray();

        $data['supplier'] = ['data' => $find,

            'status' => 1];
    }

    else

    {

```

```

        $data['supplier'] = ['pesan' => 'data Customer Salah !',
            'status' => 0];
    }

    echo json_encode($data);
}
}

```

c) Menu Transaksi

```
<?php
```

```
namespace App\Modules\Pembelian\Controllers;
```

```
use Illuminate\Http\Request;
```

```
use App\Http\Controllers\Controller;
```

```
use Yajra\DataTables\DataTables;
```

```
use Ramsey\Uuid\Uuid;
```

```
use Session;
```

```
use App\Modules\Pembelian\Models\Pembelian;
```

```
class PembelianController extends Controller
```

```
{
```

```
    var $module = 'Pembelian';
```

```
    var $table_module = 'transaksi';
```

```
function __construct(){
    $this->middleware('login');
}
```

```
public function index()
{
    $primarykey = 'transaksi_id';

    $fieldlist = array('Tanggal' => 'created_at',
        'No Nota' => 'transaksi_nota',
        'Supplier' => 'supplier_name',
        'Status' => 'transaksi_status',
        'Total' => 'transaksi_total',
        'Action ' => 'actions'
    );
```

```
$view = 'layouts.list-data';
```

```
$action = url('pembelian/create');
```

```
$view_data = [
    'key' => $primarykey,
    'fieldlist' => $fieldlist,
    'apilink' => 'pembelian/api',
    'module' => ucfirst($this->module),
    'action' => $action,
];
```

```

if (view()->exists($view)) {

    return view($view, $view_data)->render();

}

}

function apiPembelian()

{

    $sys_api = sys_api($this->module);

    $company_id = Session::get('company_id');

    $get_pembelian = $sys_api->get_all_pembelian($company_id);

    $var = Datatables::of($get_pembelian)

        ->addColumn('actions', function($get_pembelian)
        {
            if ($get_pembelian['transaksi_status'] != 'finish')
            {

                return '<a

href="'.URL('pi/store/'.$get_pembelian['transaksi_uuid']).'" class="btn btn-sm

btn-clean btn-icon btn-icon-md" title="Terima">

                <i class="la la-send"></i>

                </a>&nbsp;'.

                '<a

href="'.URL('pembelian/edit/'.$get_pembelian['transaksi_uuid']).'" class="btn

btn-sm btn-clean btn-icon btn-icon-md" title="Edit">

                <i class="la la-edit"></i>

```

```
</a>&nbsp;
```

```
<a
```

```
href="'.URL('pembelian/delete/'.$get_pembelian['transaksi_uuid']).'" class="btn
```

```
btn-sm btn-clean btn-icon btn-icon-md" title="View">
```

```
<i class="la la-trash"></i>
```

```
</a>';
```

```
}
```

```
}}
```

```
->editColumn('supplier_name', function($get_pembelian)
```

```
{
```

```
if (isset($get_pembelian['supplier_name']))
```

```
{
```

```
$customer = $get_pembelian['supplier_name'];
```

```
}
```

```
else
```

```
{
```

```
$customer = 'Umum';
```

```
}
```

```
return $customer;
```

```
}}
```

```
->editColumn('transaksi_total', function($get_pembelian){
```

```
return 'Rp. '.number_format($get_pembelian['transaksi_total']);
```

```
}}
```

```
->rawColumns(['action'])
```

```
->escapeColumns(['action'])
```

```

        ->make(true);

return $var;
}

/**
 * Show the form for creating a new resource.
 *
 * @return \Illuminate\Http\Response
 */
public function create()
{
    $action = array('link' => url('pembelian/store'),
                    'name' => "Tambah");
    $company_id = Session::get('company_id');
    $auto_code = notanumber('POB');

    $sys_supplier = sys_api('Supplier');
    $supplier = $sys_supplier->get_all_supplier($company_id);

    $data = array('action' => $action,
                  'module' => $this->module,
                  'supplier' => $supplier,
                  'auto_code' => $auto_code,
                  'uuid_transaksi' => "",

```

```
'data' => array('customer_id' => "));
```

```
$view = 'Pembelian::form';
```

```
if (view()->exists($view)) {
    return view($view, $data)->render();
}
}
```

```
/**
```

```
 * Store a newly created resource in storage.
```

```
 *
```

```
 * @param \Illuminate\Http\Request $request
```

```
 * @return \Illuminate\Http\Response
```

```
 */
```

```
public function store(Request $request)
```

```
{
```

```
    $input = $request->all();
```

```
    $users_id = Session::get('iid_users');
```

```
    $company_id = Session::get('company_id');
```

```
    $sys_api = sys_api($this->module);
```

```
    $result = $sys_api->save_pembelian($input, $company_id, $users_id);
```

```
    echo json_encode($result);
```

```
}
```

```

/**
 * Display the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

public function show($uuid)
{
    $sys_api = sys_api($this->module);
    $company_id = Session::get('company_id');
    $find = $sys_api->get_one_pembelian($uuid, $company_id);
    $detail = $sys_api->get_pembelian_rel($find['transaksi_id']);

    $data = ['transaksi' => $find,
            'details' => $detail];

    return view('Pembelian::detail',$data);
}

/**
 * Show the form for editing the specified resource.
 *
 * @param int $id

```

```

* @return \Illuminate\Http\Response

*/

public function edit($uuid)
{
    $sys_api = sys_api($this->module);

    $company_id = Session::get('company_id');

    $find = $sys_api->get_one_pembelian($uuid, $company_id);

    $detail = $sys_api->get_pembelian_rel($find['transaksi_id']);

    $sys_supplier = sys_api('Supplier');

    $supplier = $sys_supplier->get_all_supplier($company_id);

    $action = array('link' => url('/pembelian/update/'.$uuid),
                    'name' => "Edit");

    $data = array('auto_code' => $find['transaksi_not'],

                  'action' => $action,

                  'module' => $this->module,

                  'data' => $find,

                  'detail' => $detail,

                  'supplier' => $supplier,

                  'from_create' => "",

                  'uuid_transaksi'=> $uuid,

                  'method' => 'PATCH',

```

```
'lastview'    => isset($request['lastview'])?$request['lastview']:''
);
```

```
$view = 'Pembelian::form';
```

```
if (view()->exists($view)) {
    return view($view, $data)->render();
}
}
```

```
/**
 * Update the specified resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @param int $id
 *
 * @return \Illuminate\Http\Response
 */
```

```
public function update(Request $request, $uuid)
{
    $input = $request->all();

    $users_id = Session::get('iid_users');

    $company_id = Session::get('company_id');

    $sys_api = sys_api($this->module);

    $result = $sys_api->update_pembelian($input, $company_id, $users_id);
```

```

echo json_encode($result);
}

/**
 * Remove the specified resource from storage.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function destroy($uuid)
{
    $data = [
        'delete' => 1,
    ];

    $sys_api = sys_api($this->module);

    $company_id = Session::get('company_id');

    $result = Pembelian::where('transaksi_uuid','=',$uuid)->update($data);

    if ($result == true)
    {
        $message = "Pembelian Berhasil Dihapus";

        return redirect('/pembelian')->with('message', $message)
            ->with('id', "alert-success")
            ->with('icon',"fa fa-info")
    }
}

```

```

        ->with('class','alert alert-success');
    }
else
{
    $message = "Pembelian gagal dihapus";
    return redirect('/pembelian')->with('message', $message)
        ->with('id', "alert-danger")
        ->with('icon',"fa fa-info")
        ->with('class','alert alert-danger');
}
}

public function cekcode(Request $request){
    $input = $request->all();
    $keyword = $input['keyword'];
    $register = $input['registered'];

    $cekcode = cekkode($keyword, 'POB');

    if ($cekcode != "")
    {
        echo json_encode([ 'status' => 1, 'datanya' => $cekcode ]);
    }
else
{

```

```

        echo json_encode([ 'status' => 0, 'datanya' => array() ]);
    }

}

public function cekstock(Request $request)
{
    $input = $request->all();

    $kode_barang = $input['kode_barang'];
    $jumlah_beli = $input['stok'];

    $product = Product::select(DB::raw('ckode_products, istock_products'))
        ->Where('ckode_products','=',$kode_barang)
        ->first();

    if (count($product) > 0)
    {
        $product = $product->toArray();

        if ($product['istock_products'] < $jumlah_beli )
        {
            $data = ['pesan' => '<h4 align="center">Stock Tidak Cukup <br> Sisa Stock
            '.$product['istock_products'].'</h4>'];

            'status' => 0];
        }
    }
    else

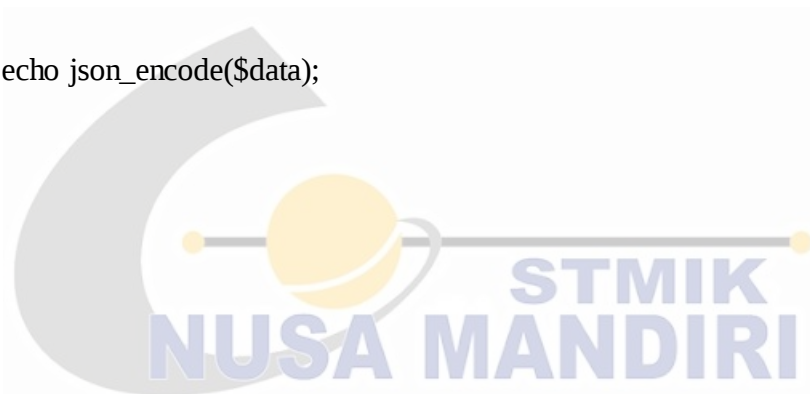
```

```

{
    $data = ['status' => 1];
}
}
else
{
    $data = ['pesan' => 'No Part Salah !',
            'status' => 0];
}

echo json_encode($data);
}
}

```



d) Menu Laporan

```
<?php
```

```
namespace App\Modules\Laporan\Controllers;
```

```
use Illuminate\Http\Request;
```

```
use App\Http\Controllers\Controller;
```

```
use App\Modules\Karyawan\Models\Karyawan;
```

```
class LaporanController extends Controller
```

```

{

    var $module = 'Laporan';

    var $table_module = 'laporan';


    function __construct(){

        $this->middleware('login');

    }


    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {

        // $data = Karyawan::where('karyawan_delete','=','0')->get();

        $action    = array('link' => url('si/cetak'),

            'name' => "Penjualan");

        $data = array('action'    => $action,

            'module'    => $this->module,

            // 'data'    => $data

        );


        $view = $this->module.'::laporan';

```

```

if (view()->exists($view)) {

    return view($view, $data)->render();

}

}

```

```

public function pembelian()

{

    $action    = array('link' => url('pi/cetak'),

                        'name' => "Pembelian");

    $data = array('action'  => $action,

                  'module'  => $this->module,

                  // 'data'  => $data

    );

```

```

$view = $this->module.'::laporan';

```

```

if (view()->exists($view)) {

    return view($view, $data)->render();

}

}

```

```

public function sr()

{

    $action    = array('link' => url('sr/cetak'),

```

```

        'name' => "Sales Return");

$data = array('action'  => $action,

        'module'  => $this->module,

        // 'data'  => $data

    );

$view = $this->module.'::laporan';

if (view()->exists($view)) {

    return view($view, $data)->render();

}

}

public function pr()
{
    $action  = array('link' => url('pr/cetak'),

        'name' => "Purchase Return");

    $data = array('action'  => $action,

        'module'  => $this->module,

        // 'data'  => $data

    );

    $view = $this->module.'::laporan';

    if (view()->exists($view)) {

```

```

        return view($view, $data)->render();
    }
}

/**
 * Show the form for creating a new resource.
 *
 * @return \Illuminate\Http\Response
 */
public function create()
{
    //
}

/**
 * Store a newly created resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @return \Illuminate\Http\Response
 */
public function store(Request $req
{
    //
}

```

```

/**
 * Display the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

```

```

public function show($id)
{
    //
}

```

```

/**
 * Show the form for editing the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

```

```

public function edit($id)
{
    //
}

```

```

/**
 * Update the specified resource in storage.
 *

```

```

* @param \Illuminate\Http\Request $request
* @param int $id
* @return \Illuminate\Http\Response
*/

public function update(Request $request, $id)
{
    //
}

/**
 * Remove the specified resource from storage.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function destroy($id)
{
    //
}
}

```

e) Code List User

```
<?php
```

```
namespace App\Modules\User\Controllers;
```

```

use Illuminate\Http\Request;

use App\Http\Controllers\Controller;

use Yajra\DataTables\DataTables;

use Ramsey\Uuid\Uuid;

use Hash;

use Session;

use PDF;

```

```

class UserController extends Controller

```

```

{
    var $module = 'User';
    var $table_module = 'User';

    function __construct(){
        $this->middleware('login');
    }

```

```

    public function index()

```

```

    {
        $primarykey = 'user_id';

        $fieldlist = array('Username'      => 'user_username',
                           'Group'        => 'group_name',
                           'Deskripsi'    => 'user_description',
                           'Action '      => 'actions'
        );
    }

```

```
$view = 'layouts.list-data';
```

```
$action = url('user/create');
```

```
$view_data = [
```

```
    'key' => $primarykey,
```

```
    'field list' => $field list,
```

```
    'apilink'  => 'user/api',
```

```
    'module'   => ucfirst($this->table_module),
```

```
    'action'   => $action,
```

```
];
```

```
if (view()->exists($view)) {
```

```
    return view($view, $view_data)->render();
```

```
}
```

```
}
```

```
function apiUser()
```

```
{
```

```
    $sys_api = sys_api($this->module);
```

```
    $company_id = Session::get('company_id');
```

```
    $get_user = $sys_api->get_all_user($company_id);
```

```
$var = Datatables::of($get_user)
```

```
    ->editColumn('user_username', function($get_users)
```

```
    {
```

```

        if (isset($get_users['user_foto']))
        {
            $user_foto = '';
        }
        else
        {
            $user_foto = '<div class="kt-badge kt-badge--xl kt-badge--
primary">'.ucfirst(substr($get_users['user_username'], 0,1)).'</div>';
        }
        return '<span style="width: 200px;">
            <div class="kt-user-card-v2">
                <div class="kt-user-card-v2__pic">
                    '.$user_foto.'
                </div>
                <div class="kt-user-card-v2__details">
                    <a class="kt-user-card-v2__name"
href="'.URL('user/detail/'.$get_users['users_uuid']).'">'.$get_users['user_usernam
e'].'</a>
                </div>
            </div>
        </span>';
    })
    ->addColumn('actions', function($get_user){

```

```

        return '<a href="'.URL('user/edit/'.$get_user['user_uuid']).'"
class="btn btn-sm btn-clean btn-icon btn-icon-md" title="Edit">

        <i class="la la-edit"></i>

        </a>&nbsp;

        <a href="'.URL('user/delete/'.$get_user['user_uuid']).'" class="btn
btn-sm btn-clean btn-icon btn-icon-md" title="View">

        <i class="la la-trash"></i>

        </a>;

    })

    ->rawColumns(['action'])

    ->escapeColumns(['action'])

    ->make(true);

return $var;
}

```

```
/**
```

```
 * Show the form for creating a new resource.
```

```
 *
```

```
 * @return \Illuminate\Http\Response
```

```
 */
```

```
public function create()
```

```
{
```

```
    $action    = array('link' => url('user/store'),
```

```
        'name' => 'Tambah');
```

```

$sys_group = sys_api('Group');

$company_id = Session::get('company_id');

$group = $sys_group->get_all_group($company_id);

```

```

$data = array('action' => $action,
              'module' => $this->module,
              'group' => $group,
              'data' => array('group_id' => ",));

```

```

$view = 'User::form';

```

```

if (view()->exists($view)) {
    return view($view, $data)->render();
}
}

```

```

/**

```

```

 * Store a newly created resource in storage.

```

```

 *

```

```

 * @param \Illuminate\Http\Request $request

```

```

 * @return \Illuminate\Http\Response

```

```

 */

```

```

public function store(Request $request)

```

```

{

```

```

    $input = $request->all();

```

```

$company_id = Session::get('company_id');

$data = [

    'user_uuid'      => Uuid::uuid4(),

    'user_username'   => $input['username'],

    'user_password'   => Hash::make($input['password']),

    'user_description' => $input['deskripsi'],

    'group_id'        => $input['group'],

    'company_id'      => $company_id

];

```

```

if($request->hasFile('foto_user')){

    $data['user_foto'] = upload('','user',$request['foto_user']);

}

$sys_api = sys_api($this->module);

$result = $sys_api->save_user($data);

```

```

if ($result == true)

{

    $message = "User Berhasil Ditambahkan";

    return redirect('/user')->with('message', $message)

        ->with('id', "alert-success")

        ->with('icon', "fa fa-info")

        ->with('class', "alert alert-success");

}

```

```

else

{

$message = "User gagal ditambahkan";

return redirect('/user/create')->with('message', $message)

->with('id', "alert-danger")

->with('icon', "fa fa-info")

->with('class', "alert alert-danger");

}

}

```

```

/**
 * Display the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

```

```

public function show($id)

{


}

```

```

/**
 * Show the form for editing the specified resource.
 *
 * @param int $id

```

```

* @return \Illuminate\Http\Response

*/

public function edit($uuid)
{
    $sys_api = sys_api($this->module);

    $company_id = Session::get('company_id');

    $find      = $sys_api->get_one_user($uuid, $company_id);

    $sys_group = sys_api('Group');

    $group = $sys_group->get_all_group($company_id);

    $action    = array('link' => url('/user/update/'.$uuid),
                        'name' => "Edit");

    $data      = array('action' => $action,
                        'module'  => $this->module,
                        'group'   => $group,
                        'data'    => $find,
                        'method'  => 'PATCH',
                        'lastview' => isset($request['lastview'])?$request['lastview']:''
                    );

    $view = 'User::form';

    if (view()->exists($view)) {

```

```

        return view($view, $data)->render();
    }
}

/**
 * Update the specified resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function update(Request $request, $uuid)
{
    $input = $request->all();
    $company_id = Session::get('company_id');
    $data = [
        'user_description' => $input['deskripsi'],
        'group_id'         => $input['group'],
    ];

    if($request->hasFile('foto_user'))
    {
        $sys_api = sys_api($this->module);
        $find    = $sys_api->get_one_user($uuid, $company_id);
        $path = public_path('img/'.$this->module);
    }
}

```

```

if(isset($find['user_foto']))
{
    unlink($path.'/'.$find['user_foto']);
    unlink($path.'/500/'.$find['user_foto']);
}

$data['user_foto'] = upload('','user',$request['foto_user']);
}

$sys_api = sys_api($this->module);
$result = $sys_api->update_user($data, $uid);

if ($result == true)
{
    if ($request['lastview'] != "")
    {
        $message = "User Berhasil Diubah";

        return redirect('/user/detail/'.$id)->with('message', $message)
            ->with('id', "alert-success")
            ->with('icon',"fa fa-info")
            ->with('class',"alert alert-success");
    }
    else
    {
        $message = "User Berhasil Diubah";
    }
}

```

```

return redirect('/user')->with('message', $message)

->with('id', "alert-success")

->with('icon',"fa fa-info")

->with('class',"alert alert-success");

}

}

else

{

if ($request['lastview'] != "")

{

$message = "User Gagal Diubah";

return redirect('/user/detail/'.$id)->with('message', $message)

->with('id', "alert-success")

->with('icon',"fa fa-info")

->with('class',"alert alert-success");

}

else

{

$message = "User gagal diubah";

return redirect('/user/edit/'.$id)->with('message', $message)

->with('id', "alert-danger")

->with('icon',"fa fa-info")

->with('class',"alert alert-danger");

}

}

```

```

    }
}

/**
 * Remove the specified resource from storage.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function destroy($uuid)
{
    $data = [
        'delete' => 1,
    ];

    $sys_api = sys_api($this->module)

    $result = $sys_api->update_user($data, $uuid);

    if ($result == true)
    {
        $message = "User Berhasil Dihapus";

        return redirect('/user')->with('message', $message)

            ->with('id', "alert-success")

            ->with('icon',"fa fa-info")

            ->with('class',"alert alert-success");
    }
}

```

```

    }

    else

    {

        $message = "User gagal dihapus";

        return redirect('/user')->with('message', $message)

            ->with('id', "alert-danger")

            ->with('icon',"fa fa-info")

            ->with('class',"alert alert-danger");

    }

}

```

```

public function cetak(Request $request)
{
    $sys_api = sys_api('User');
    $company_id = Session::get('company_id');
    $data = $sys_api->get_all_user($company_id);

    $data = array('data' => $data,

        );

    // Send data to the view using loadView function of PDF facade

    PDF::setOptions(['dpi' => 150, 'defaultFont' => 'sans-serif']);

    $pdf = PDF::loadView('User::cetak_user', $data);

    $pdf->setPaper('a4', 'landscape');

```

```

// If you want to store the generated pdf to the server then you can use the store
function

// $pdf->save(storage_path().'_filename.pdf');

// Finally, you can download the file using download function

return $pdf->stream('users.pdf');

}

}

```

4.4 Code Generation

Tabel IV.14

Hasil Pengujian Black Box Testing From Login Admin

No	Skenarion Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengosongkan semua isian lalu menekan tombol "Login"	Username : (kosong) Password : (kosong)	Sistem akan menolak akses login dan menampilkan pesan "Login Gagal, Cek Kombinasi Username & Password"	Sesuai Harapan	Valid
2	Hanya mengisi salah satu kolom isian lalu menekan tombol "Login"	Username : (admin) Password : (kosong)	Sistem akan menolak akses login dan menampilkan pesan "Login Gagal, Cek Kombinasi Username & Password"	Sesuai Harapan	Valid

3	Mengisi password yang salah lalu menekan tombol "Login"	Username : (admin) Password : (admin2)	Sistem akan menolak akses login dan menampilkan pesan "Login salah, silahkan coba lagi"	Sesuai Harapan	Valid
4	Menekan semua data dengan benar lalu menekan tombol "Login"	Username : (admin) Password : (admin)	sistem menerima dan menampilkan menu halaman User	Sesuai Harapan	Valid

Tabel IV.15

Hasil Pengujian Black Box Testing From Tambah Barang

No	Skenarion Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Tidak menginput kode barang,nama barang,kategori,merk, stock, harga jual, harga beli, keterangan "Simpan"	kode barang : (kosong) nama barang : (kosong) kategori : (kosong) merk : (kosong) stock : (kosong) harga jual : (kosong) harga beli : (kosong) keterangan (kosong)	sistem akan menolak permohonan dan menampilkan pesan "Kode barang #1 harus di isi ! Nama barang #1 harus di isi ! Kategori #1 harus di isi ! Sock #1 harus di isi ! Harga #1 harus di isi ! Harga #1 harus di isi !	Sesuai Harapan	Valid

2	Menginput kode barang, nama barang, kategori, stock, harga jual, harga beli, keterangan "Simpan"	kode barang : (x01) nama barang : (Aki) kategori : (Aki) merk : (KNZ) stock : (10) harga jual : (172.500) harga beli : (82.900) keterangan (baik)	Sistem menerima dan menampilkan data staff baru pada menu data staff	Sesuai Harapan	Valid
---	--	---	--	----------------	-------

Tabel IV.16

Hasil Pengujian Black Box Testing From Tambah Pelanggan

No	Skenarion Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Tidak menginput nama, alamat, nomor telepon dan info tambahan lainnya "simpan data"	Nama : (kosong) Alamat : (kosong) Nomor telepon : (kosong) Info Pelanggan lainnya : (kosong)	Sistem akan menolak menampilkan pesan Nama harus di isi ! Alamat harus di isi ! Telepon/Han dphone harus di isi	Sesuai Harapan	Valid
2	Menginput nama, alamat, dan info tambahan lainnya "simpan"	Nama : (Atma) Alamat : (jakarta) Nomor telepon : (kosong) Info Pelanggan lainnya : (kosong)	sistem akan menolak menampilkan pesan Telepon/Han dphone harus di isi !	Sesuai Harapan	Valid

3	Menginput tambah pelanggan yang benar dan mengisi semua	Nama : (Atma) Alamat : (jakarta) Nomor telepon : (08998712 0128) Info Pelanggan lainnya : (kosong)	Sistem menerima dan menampilkan faktur transaksi pembelian	Sesuai Harapan	Valid
---	---	--	--	----------------	-------

Tabel IV.17

Hasil Pengujian Black Box Testing From Tambah Supplier

No	Skenarion Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Tidak menginput nama, alamat, nomor telepon dan info tambahan lainnya "simpan data"	Nama : (kosong) Alamat : (kosong) Nomor telepon : (kosong) Info Pelanggan lainnya : (kosong)	Sistem akan menolak menampilkan pesan Nama harus di isi ! Alamat harus di isi ! Telepon/Hand phone harus di isi	Sesuai Harapan	Valid
2	Menginput nama, alamat, dan info tambahan lainnya "simpan"	Nama : (CV.Gratia) Alamat : (jelambar) Nomor telepon : (kosong) Info Pelanggan lainnya : (kosong)	sistem akan menolak menampilkan pesan Telepon/Hand phone harus di isi !	Sesuai Harapan	Valid

3	Menginput tambah pelanggan yang benar dan mengisi semua isian lalu menekan tombol "simpan"	Nama : (KNZ) Alamat : (jakarta) Nomor telepon : (0891282830 1) Info Pelanggan lainnya : (kosong)	Sistem menerima dan menampilkan faktur transaksi pembelian	Sesuai Harapan	Valid
---	--	---	--	----------------	-------

Tabel IV.18

Hasil Pengujian Black Box Testing From Login Staff Inventory

No	Skenarion Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengosongkan semua isian lalu menekan tombol "Login"	Username : (kosong) password : (kosong)	Sistem akan menolak akses login dan menampilkan pesan "Login Gagal, cek kombinasi Username & Password"	Sesuai Harapan	Valid
2	Hanya mengisi salah satu kolom isian lalu menekan tombol "Login"	Username : (kosong) password : (kosong)	Sistem akan menolak akses login dan menampilkan pesan "Login Gagal, cek kombinasi Username & Password"	Sesuai Harapan	Valid
3	Mengisi password yang salah lalu menekan tombol "Login"	Username : (admin) password : (admin2)	Sistem akan menolak akses login dan menampilkan pesan "Login salah, silahkan coba lagi"	Sesuai Harapan	Valid

4	Memasukan semua data dengan benar lalu menekan tombol "Login"	<i>Username :</i> (admin2) <i>password :</i> (admin2)	Sistem menerima dan menampilkan menu halaman <i>User</i>	Sesuai Harapan	Valid
---	---	--	--	----------------	-------

Tabel IV.19

Hasil Pengujian *Black Box Testing* From Login Pimpinan

No	Skenarion Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengosongkan semua isian lalu menekan tombol "Login"	<i>Username :</i> (kosong) <i>password :</i> (kosong)	Sistem akan menolak akses login dan menampilkan pesan "Login Gagal, Cek Kombinasi Username & Password"	Sesuai Harapan	Valid
2	Hanya mengisi salah satu kolom isian lalu menekan tombol "Login"	<i>Username :</i> (kosong) <i>password :</i> (kosong)	Sistem akan menolak akses login dan menampilkan pesan "Login Gagal, cek kombinasi Username & Password"	Sesuai Harapan	Valid
3	Mengisi password yang salah lalu menekan tombol "Login"	<i>Username :</i> (admin) <i>password :</i> (admin2)	Sistem akan menolak akses login dan menampilkan pesan "Login salah, silahkan coba lagi"	Sesuai Harapan	Valid

4	Memasukan semua data dengan benar lalu menekan tombol "Login"	<i>Username :</i> (admin3) <i>password :</i> (admin3)	Sistem menerima dan menampilkan menu halaman <i>User</i>	Sesuai Harapan	Valid
---	---	--	--	----------------	-------

5.5 Support

5.5.1 Publikasi Web

Mengakses website ini bisa dengan menggunakan *browser* yang tersedia seperti *Mozilla firefox, opera, google chroom* atau *internet explorer* dengan cara mengetik alamat website yang dituju kedalam *address* yang terdapat pada *browser* yang digunakan. Publikasi situs web aplikasi sistem informasi *inventory* berbasis web pada CV. Gratia Motorindo dengan alamat *URL localhost//gratia*

4.5.2 Spesifikasi *Hardware* Dan *Software*

Di dalam mengimplementasikan perancangan dan pembuatan situs web yang penulis jadikan sebagai penulisan skripsi tentunya membutuhkan sarana-sarana pendukung atau tools yaitu yang terdiri dari perangkat keras dan perangkat lunak. Berikut ini gambaran umum struktur hubungan antara komponen dari sistem keseluruhan secara fisik.

Tabel IV.20*Spesifikasi Hardware Dan Software*

Kebutuhan	Keterangan
Sistem Operasi	Windows 7
Processor	Intel Inside
RAM	2 GB
Hardisk	320 GB
Monitor	14"
Keyboard	108 Key Standar
Mouse	Google Chrome, Mozilla Firefox
Web Editor	PHP
Bahasa Pemrograman	Sumlime text, Note
Server Module	PHP Myadmin
Web Server	Xampp
Database	Mysql
UML Designer	Start UML dan Microsoft Visio

5.6 Spesifikasi Dokumen Sistem Susulan

Spesifikasi Dokumen Sistem usulan merupakan pembahasan mengenai bentuk dari dokumen-dokumen sistem usulan yang diusulkan dengan tujuan untuk meningkatkan dalam proses transaksi inventory di CV. Gratia Motorindo.

1. Nama Dokumen : Laporan Pembelian

Fungsi : Sistem

Tujuan : Pembelian (*Purchasing*)

Media : From

Jumlah : Rangkap Satu

Frekuensi : Setiap ingin menampilkan laporan pembelian

Bentuk : Lampiran B-1

2. Nama dokumen : Laporan penjualan (*Purchasing Invoice*)

Fungsi : Sebagai Laporan Penjualan

Sumber : Sistem

Tujuan : Pembelian (*Purchasing*)

Media : Form

Jumlah : Rangkap Satu

Frekuensi : Setiap ingin menampilkan Laporan Penjualan

Bentuk : Lampiran B-2

3. Nama dokumen : Laporan Return Pembelian

Fungsi : Sebagai Laporan Return Pembelian

Sumber : Sistem

Tujuan : Pembelian (*Purchasing*)

Media : Form

Jumlah : Rangkap Satu

Frekuensi : Setiap ingin menampilkan Laporan Return Pembelian

Bentuk : Lampiran B-2

4. Nama dokumen : Laporan Return Penjualan

Fungsi : Sebagai Laporan Return Penjualan

Sumber : Sistem

Tujuan : Pembelian (*Purchasing*)

Media : Form

Jumlah : Rangkap Satu

Frekuensi : Setiap ingin menampilkan Laporan Return Penjualan

Bentuk : Lampiran B-4

BAB V

PENUTUPAN

5.1 Kesimpulan

Untuk mengatasi masalah yang ada pada CV.Gratia Motorindo, maka di perlukan suatu bentuk aplikasi pemrograman penanganan baru yang menggunakan computer sebagai salah satu alat hasil kemajuan teknologi saat ini yang nantinya diharapkan agar dapat memberikan informasi yang efektif dan efisien. Dengan aplikasi ini diharapkan banyak membantu para pekerja yang terlibat langsung dalam proses pengolahan data barang, data supplier, data customer, data pembelian, data penjualan dan data laporan return.

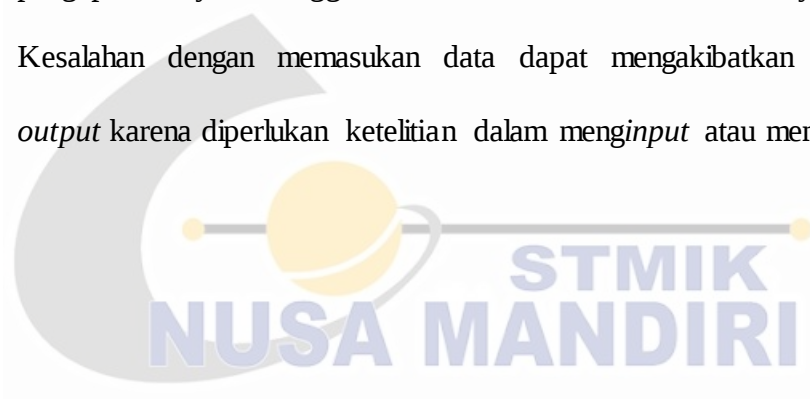
Kelebihan yang dapat diberikan dalam pnggunaan aplikasi pemrograman aplikasi ini sebagai berikut :

- a. Aplikasi program sebagai alat alternative pemecahan masalah dalam membantu mengatasi permasalahan yang dihadapi pada sistem lama.
- b. Sistem manual setiap kegiatannya banyak memerlukan pencatatan pada setiap lembar kertas, Sehingga kegiatan tersebut mempunyai resiko yang cukup tinggi baik dalam penyimpanan data atau pembuatan laporan.
- c. Kesalahan yang terjadi dalam pembuatan laporan-laporan dapat diperkecil dibandingkan dengan cara manual.

5.2 Saran-Saran

Dengan adanya masalah-masalah diatas, maka penulis dapat memberikan saran sebagai berikut :

- a. Melatih tenaga administrasi agar dapat menguasai program aplikasi inventory.
- b. Agar program ini berjalan dengan baik, sebaiknya menggunakan perangkat keras yang mendukung baik sarana dan prasarana.
- c. Di perlukan pemeliharaan secara rutin didalam pelaksanaan dan pengoprasiannya, sehingga akan mudah diketahui kesalahannya.
- d. Kesalahan dengan memasukan data dapat mengakibatkan kesalahan pada *output* karena diperlukan ketelitian dalam *menginput* atau memasukan data.



DAFTAR PUSTAKA

- Affah,Supriyanta. 2018. Analisa Dan Perancangan Sistem Informasi Untuk Keunggulan Bersaing Perusahaan Dan Organisasi Modern. Yogyakarta. CV,Andi Offset.
- Agusvianto, Hendra. 2017. Sistem informasi Inventory Gudang Untuk Mengontrol Persediaan Barang Pada Gudang Studi Kasus. PT.Alaisys Sidoarjo.
- Anggraeni,Elisabet Yunaeti .Irvani Rita. 2017. Pengantara Sistem Informasi. Yogyakarta. CV.Andi Offset.
- Anhar, 2010 Panduan menguasai PHP Dan MySQL. Secara Otodidak. Jagakarsa. Mediakit.
- Chandra & Wiradinata, 2015 Rancang Bangun Sistem Informasi Inventori ,Surabaya.
- Edy Winarto, 2014in1. Javascript, jQuery, dan jQuery Mobile. Jakarta. PT.Alex Media Komputindi.
- Hutahaeen, 2014 Konsep Sistem Informasi. Yogyakarta. CV.Budi Utama.
- Imansyah, 2014 Membangun Toko Online Dengan Wordpress. Jakarta. PT.Alex Media Komputindo.
- Jubilee, 2011 Step by Step HTML. Jakarta. PT.Elex Media Komputindo.
- Martudi, 2014 Membangun Toko Online Dengan Wordpress. Jakarta. CV.Budi Utama.
- Prasetio, 2014 Buku Sakti Webmaster. Jagakarsa. Mediakita.
- Rerung, Rintho Rante. 2018. Pemrograman Web Dasar. . Yogyakarta. CV.Budi Utama.
- Rorimpandey, Gladys C. 2015. Analisa Perancangan Sistem Inventory Produk

Non-BBM. PT.Pertamina (Perseto). Manado. Universitas Manado.

Rosa dan Shalahudin, 2015 Rekayasa perangkat lunak (berstruktur dan berorientasi Objek Bandung . informatika.

Saefudin, 2018 Rekayasa perangkat lunak (berstruktur dan berorientasi Objek). Bandung . informatika.

Sahyar, 2016 Algoritma Dan Pemrograman Menggunakan Matlab (Matrix Laboratory). Jakarta. Kencana.

Soepomo, 2014Perancangan Sistem Informasi Inventory Sparepart Elektornik Berbasis WEB PHP.Yogyakarta CV. Human Global Service.

Sukamdi, 2013. Rekayasa Perangkat Lunak. Bandung. Informatika.

Supono, 2016Pemrograman Web Dengan Menggunakan PHP Dan Framework Codeigniter Surabaya. Universitas Negeri Surabaya.

Widodo, Agus Wahyu. Kurnianing Tyas. 2017. Sistem Basis Data. Malang. UB Press Enterprise.

STMIK
NUSA MANDIRI



CV. Gratia Motorindo

Jalan. Jelambar Selatan IX No.15f Grogol petamburan,
Jakarta Barat, 11460 Telp.021-5679106/08988878206

Email : Gratiamotorindo@gmail.com

SURAT KETERANGAN RISET

Yang bertanda tangan dibawah ini :

Nama : Darwin Wijaya

Jabatan : Direktur Utama

Dengan ini menyatakan bahwa :

Nama : Siti wiyarti

Nim : 11145368

Alamat : Jl.jelambar timur rt.011/09 kelurahan jelambar baru
Kecamatan grogol petamburan

Dari Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri Jurusan Sistem Informasi benar telah melakukan kegiatan riset tentang sistem informasi Inventory Barang pada CV.Gratia Motorindo pada tanggal 01 April 2019 sampai 06 Mey 2019 dan yang bersangkutan telah melakukan tugas dengan baik dan tanggung jawab.

Demikian surat keterangan riset ini dibuat dengan sebenar-benarnya dan untuk dapat dipergunakan sebagaimana mestinya.

Jakarta, 06 Mey 2019

Direktur utama

Darwin Wijaya



PPT. Mitra Anugerah
Jln.RE. Martadinata 10 Block B/AE Jakarta Utara
Telp. 0857 11799 583

Surat Jalan

Kepada Yth.
CV.Gratia Motorindo
Jln.jelambar Selatan IX No.15F
Jakarta Barat

Telah dikirim dari : PT. Mitra Anugerah

No	Surat Jalan	Deskripsi Barang	Qty		Total
			Pcs	Ctn	
1	MA 197355	Mitran Pack- OPP Tape 48mm (Bening)		144	144
Total				144	

Barang-barang tersebut telah diterima dengan baik sesuai dengan keadaan sebenarnya.

Jakarta, 26 Juli 2019

Dikirim Oleh,


(IPAN)

Diterima Oleh


CV.Gratia Motorindo
(Ferman)



CV. Gratia Motorindo

Jalan. Jelambar Selatan IX No.15F
 Jakarta Barat, 11460 Telp.021-5679106/08988878206
 Email : Gratiamotorindo@gmail.com

PURCHASE ORDER

20190727/XI/PO-ER

TO : Kurnia Jaya
 Attn : Bpk. Kevin

Jakarta, 27 Juli 2019

No	Item Description	Packing	Qty	Harga/Carton	Total Rupiah
1	Block Seher + Ring Vixion KNZ	8	8	364,000	2,912,000.00
2	Block Seher + Ring Mio M3 KNZ	10	10	392,000	3,920,000.00
3	Block Seher + Ring RX King KNZ	8	8	720,000	5,760,000.00
Total Faktur					12,592,000.00

Note : Claim/Pengaduan hanya dilayani paling lambat 7 hari dari tanggal faktur

Jakarta, 27 Juli 2019

Dikirim Oleh,

Diterima Oleh



CV. Gratia Motorindo
 Jl. Jelambar selatan IX No. 15F
 Jakarta Barat, 11460 Telp. 021-5679106/0898878206
 Email : Gratiamotorindo@gmail.com

No	Tanggal	No Nota	No Ref PO	Supplier	Status	Total
1	2019-08-31 08:19:46	PI-3108190001	POB-3108190001	KNZ	finish	Rp. 829,000
2	2019-08-26 11:52:50	PI-2608190001	POB-2608190001	Umum	finish	Rp. 82,900
3	2019-08-15 12:00:16	PI-1508190004	POB-1508190004	KNZ	finish	Rp. 18,975,000
4	2019-08-15 11:15:56	PI-1508190003	POB-1508190003	KNZ	finish	Rp. 2,000,000
5	2019-08-15 11:05:48	PI-1508190002	POB-1508190002	cv.gracia jaya wijaya	finish	Rp. 1,500,000
6	2019-08-15 10:58:16	PI-1508190001	POB-1508190001	KNZ	finish	Rp. 890,000
Total :						Rp. 24,276,900

Lampiran B,2

Dokumen Sistem Usulan Laporan Penjualan



CV. Gratia Motorindo
 Alamat: Jl. Jelambar selatan IX No. 15F
 Jakarta Barat, 11460 Telp. 021-5679106/0898878206
 Email : Gratiamotorindo@gmail.com

No	Tanggal	No Nota	No Ref SO	Customer	Status	Total
1	2019-08-15 12:02:16	SI-1508190004	SO-1508190004	ATMA motor	finish	Rp. 1,725,000
2	2019-08-15 11:12:52	SI-1508190003	SO-1508190003	Makmur Jaya motor	finish	Rp. 1,867,500
3	2019-08-15 11:06:41	SI-1508190002	SO-1508190002	Umum	finish	Rp. 3,000,000
4	2019-08-15 10:59:09	SI-1508190001	SO-1508190001	ATMA motor	finish	Rp. 1,335,000
5	2019-08-05 15:15:37	SI-0508190001	SO-0508190001	ATMA motor	finish	Rp. 650,000
Total :						Rp. 8,577,500