

# RANCANG BANGUN APLIKASI *INVENTORY* DENGAN METODE *EOQ* DAN *ROP* BERBASIS WEB

Pramudita Cahya Firdaus<sup>1</sup>, Rustiyana<sup>2</sup>, Eky M Mufthi<sup>3</sup>  
<sup>1,2,3</sup> Fakultas Teknologi Informasi Universitas Bale Bandung

<sup>1</sup>pramudita132@gmail.com

<sup>2</sup>Rustiyana@unibba.ac.id

<sup>3</sup>ekymmufthi@unibba.ac.id

Jln. R.A.A Wiranatakusumah Baleendah Bandung

**Abstract**— In doing this final project, the composer aims to design and build an inventory application for the Asep Groceries Groceries. Where this study aims to design and build an application through which the wholesale owner can monitor and control the inventory (inventory) that is at the intended wholesaler. Inventory is materials or goods (organizational resources) that are stored, which will be used to meet certain objectives, for example: for the production or assembly process, for parts of equipment, or for sale, a company does not can be called a company if it does not have inventory (inventory). *EOQ* is the most economical volume or amount of purchases to be carried out at each time of purchase. So by applying the *EOQ* model in purchasing, ordering costs and storage costs can be reduced, after the number of customers with the most economical costs is determined, then re-order is done. Reorder Points explain when to re-stock, so that the receipt of ordered materials is just the right amount and also on time. Based on the statement above, the composer aims to make an application that can streamline the wholesale owner in managing inventory so that it is expected that through the application that will be made, it can prevent a run out of goods (inventory) or excess goods when making a re-ordering process. In doing this research, the composer will use *MDD* (Model Drive Development) as the basis of the research methodology, *EOQ* (Economic Oerder Quantity) method and *ROP* (Re-order Point) to control inventory, *UML Diagram* (Unived Modeling Language) to make system design. For the programming language used is *Hypertext Preprocessor* (PHP) with *MySQL* as its database.

**Keyword**— Grocery Wholesale, inventory, *MDD*, *EOQ* & *ROP*, *UML diagram*, *PHP*, *MySQL*

**Abstrak**— Dalam mengerjakan tugas akhir ini penyusun bertujuan untuk merancang dan membangun aplikasi *inventory* (persediaan barang) pada grosir sembako pak asef. Dimana penelitian ini bertujuan untuk merancang dan membangun suatu aplikasi yang mana melalui aplikasi tersebut pemilik grosir dapat memantau dan mengendalikan *inventory* (persediaan barang) yang ada pada grosir yang di maksud. Persediaan (*Inventory*) adalah bahan-bahan atau barang (sumberdaya-sumberdaya organisasi) yang di simpan, yang akan di pergunakan untuk

memenuhi tujuan tertentu, misalnya: untuk proses produksi atau perakitan, untuk suku cadang dari peralatan, maupun untuk dijual, suatu perusahaan tidak dapat di sebut perusahaan apabila tidak memiliki persediaan (*inventory*). *EOQ* merupakan volume atau jumlah pembelian yang paling ekonomis untuk di laksanakan pada setiap kali pembelian. Sehingga dengan menerapkan model *EOQ* dalam pembelian, biaya pemesanan dan biaya penyimpanan dapat ditekan, setelah jumlah pemesan dengan biaya yang paling ekonomis di tentukan maka di lakukan lah re-order. Reorder Point menjelaskan kapan saat seharusnya diadakan pemesanan lagi (*re-stock*), sehingga penerimaan bahan yang dipesan tepat jumlahnya dan juga tepat pada waktunya. Berdsasarkan pernyataan di atas penyusun bertujuan untuk membuat suatu apliikasi yang dapat mengefisiensikan pemilik grosir dalam megelola *inventory* sehingga di harapkan melalui aplikasi yang akan di buat dapat mencegah terjadinya kehabisan barang (persediaan) atau kelebihan barang saat melakukan proses pemesanan kembali. Dalam mengerjakan penelitian ini penyusun akan menggunakan *MDD* (Model Drive Developement) sebagai dasar metodologi penelitian, metode *EOQ* (Economic Oerder Quantity) dan *ROP* (Re-order Point) untuk mengendalikan persediaan, *UML Diagram* (Unived Modeling language) untuk membuat perancangan sistem. Untuk bahasa pemograman yang di gunakan adalah *Hypertext Preprocessor* (PHP) dengan *MySQL* sebagai *database*-nya.

**Kata Kunci**— Grosir Sembako, *inventory*, *MDD*, *EOQ* & *ROP*, *UML Diagram*, *PHP*, *MySQL*.

## I. PENDAHULUAN

Salah satu jenis usaha wiraswasta yang bergerak dalam bidang dagang (*retailer*), harus memiliki *inventory* (persediaan barang). Tujuan utama dari penyimpanan persediaan adalah untuk memenuhi kebutuhan konsumen, di sisi lain pemilik usaha di tuntut untuk mampu mengendalikan *inventory* (persediaan barang) untuk menghindari terjadinya *over stock*, sehingga pemilik usaha mendapatkan keuntungan yang optimal.

Grosir Sembako adalah salah satu jenis usaha wiraswasta yang bergerak dalam bidang dagang (*retailer*) yang menjual kebutuhan pokok hidup seperti penjualan mie instan, kopi, beras, alat tulis dan kebutuhan hidup lainnya. Pemilik Grosir bertugas mengelola segala aktivitas yang menyangkut pada produksi produk. saat ini Grosir Sembako menyediakan lebih dari 100 jenis produk dengan rata-rata produksi per minggu mencapai 100 hingga 200 pcs untuk sebagian jenis produk.

Dalam operasional pengendalian *inventory* (persediaan barang), untuk menentukan jumlah produk yang akan disediakan ulang, Pemilik Grosir melihat dari penjualan perhari, sedangkan untuk menentukan kapan dilakukan pemesanan (*restock*), Pemilik Grosir melihat rata-rata penjualan produk dari hari sebelumnya, jika jumlah produk mencapai jumlah rata-rata penjualan dan produk barang yang di sediakan hampir habis terjual maka akan dilakukan penyediaan kembali. Dengan proses pengendalian persediaan yang dilakukan seperti saat ini, Pemilik Grosir Sembako mengalami berbagaimacam permasalahan. Seperti lupa mencatat barang yang harus di sediakan kembali sehingga dapat menimbulkan kosongnya salah satu produk yang seharusnya di sediakan dan tidak mampu memenuhi kebutuhan konsumen.

Penelitian sebelumnya telah di lakukan oleh (Tomi lukmana, Diana Trivena Y, 2015) dengan judul “Penerapan Metode EOQ dan ROP (Studi Kasus: PD. BARU)”. Penelitian ini memiliki kelemahan yakni aplikasi ini hanya bisa di akses oleh satu *user* yaitu oleh bagian *inventory*, sehingga tidak dapat di akses oleh pemilik perusahaan dari luar kator.

Berdasarkan masalah yang ada di atas Grosir Sembako membutuhkan sebuah aplikasi *inventory* (persediaan barang) yang mampu menentukan berapa banyaknya jumlah item produk yang akan dipesan pada suatu waktu dan kapan dilakukan pemesanan ulang terhadap item produk tersebut.

## II. LANDASAN TEORI

### A. *Inventory (Persediaan Barang)*

Persediaan (*Infentory*) adalah bahan-bahan atau barang (sumberdaya-sumberdaya organisasi) yang di simpan, yang akan di pergunakan untuk memenuhi tujuan tertentu, misalnya: untuk proses produksi atau perakitan, untuk suku cadang dari peralatan, maupun untuk dijual. Walaupun persediaan hanya merupakan suatu sumber dan yang meganggur, akan tetapi dapat di katakan tidak ada perusahaan yang beroperasi tanpa persediaan. (Hengki Fitrayco dan Habibi, 2009).

### B. *Pengendalian Persediaan*

Pengendalian persediaan adalah suatu usaha yang di lakukan suatu perusahaan untuk menjaga stabilitas produksi, sehinga dapat memenuhi kebutuhan kosumen. Dengan dilakukannya pengendalian persediaan jumlah

produk yang di sediakan akan stabil sehingga dapat meghemat biaya produksi dan memaksimalkan keuntungan perusahaan.

Tujuan utama dari pengendalian persediaan adalah agar perusahaan selalu mempunyai persediaan dalam jumlah yang tepat, pada waktu yang tepat dan dalam spesifikasi atau mutu yang telah di tentukan, sehinga kontinuitas usaha dapat terjamin (tidak terganggu) (Hengki Fitrayco dan Habibi, 2009).

### C. *Economic Order Quantity*

*Economic Order Quantity* (EOQ) atau *Economic Lot Size* (ELS) merupakan suatu metode manajemen persediaan paling terkenal dan paling tua. Diperkenalkan oleh FW. Harris sejak tahun 1914. Model ini dapat dipergunakan baik untuk persediaan yang dibeli maupun yang dibuat sendiri, dan banyak digunakan sampai saat ini karena penggunaannya relatif mudah. Model ini mampu untuk menjawab pertanyaan tentang kapan pemesanan/pembelian harus dilakukan dan berapa banyak jumlah yang harus dipesan agar biaya total (penjumlahan antara biaya pemesanan dengan biaya penyimpanan) menjadi minimum.

Pengertian EOQ sebenarnya merupakan volume atau jumlah pembelian yang paling ekonomis untuk di laksanakan pada setiap kali pembelian. Sehingga dengan menerapkan model EOQ dalam pembelian, biaya pemesanan dan biaya penyimpanan dapat ditekan (Tri Muhamad Hani, 2011).

EOQ dapat di tentukan dengan rumus sebagai berikut :

$$EOQ = \sqrt{\{(2D \times S) / H\}}$$

Keterangan :

D : Kebutuhan barang per periode

Adalah jumlah barang yang di butuhkan dalam satu periode.

S : Biaya order pesan

Adalah biaya tranfortasi, telepon dan lain-lain yang di kieluarkan ketika hendak melakukan peyediaan kembali (*Re-Stock*).

H : Biaya penyimpanan

Adalah biaya yang di butuhkan untuk penyimpanan barang atau persediaan yang di butuhkan, Contoh : untuk menyimpan *Ice Cream* di butuhkan *Freezer* dalam kasus ini biaya untuk membayar listrik termasuk kedalam biaya penyimpanan

### D. *Re-order Point*

Setelah jumlah bahan yang di beli dengan biaya yang minimal di tentukan, masalah selanjutnya adalah kapan perusahaan harus memesan kembali agar perusahaan tidak kehabisan bahan atau persediaan. Titik dimana perusahaan harus memesan kembali agar kedatangan bahan baku yang di pesan tepat pada saat persediaan bahan diatas *safety stock* sama dengan nol disebut *Reorder Point*.

*Reorder Point* menjelaskan kapan saat seharusnya diadakan pemesanan lagi, sehingga penerimaan bahan yang dipesan tepat pada waktunya. Ada dua faktor yang menentukan *Reorder Point*, yaitu :

1. Safety Stock adalah persediaan minimal yang ada dalam perusahaan.
2. Kebutuhan barang per periode.

Untuk menghitung *Reorder Point*, dirumuskan sebagai berikut :

$$\text{Rumus ROP} = \text{Safety stock} + (D/365)$$

Jika permintaan atas produk tidak diketahui dengan pasti, maka terdapat kemungkinan terjadinya kehabisan persediaan. Untuk mengatasi hal tersebut pihak gudang sering sekali memilih untuk menyimpan persediaan pengaman (*Safety Stock*), (Haposan Naingolan, 2010).

#### D. Rancang

Perancangan atau rancang merupakan serangkaian prosedur untuk menterjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen *system* di implementasikan (R. S. Presman, 2010).

#### F. Bangun

Pengertian Pembangunan atau bangun sistem adalah kegiatan menciptakan *sistem* baru maupun mengganti atau memperbaiki *system* yang telah ada secara keseluruhan (R. S. Presman, 2010).

#### G. Aplikasi

Aplikasi adalah program siap pakai yang dapat di gunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan pembuatan aplikasi tersebut, aplikasi adalah program komputer yang di tulis dalam suatu bahasa pemrograman dan di penggunaan untuk menyelesaikan masalah tertentu (Robi Muhamad, 2009).

#### H. Website

Secara *basic*, *website* di gunakan untuk publikasi informasi. Adapun informasi yang akan di sediakan adalah beraneka ragam dari *profil* pribadi hingga *company profil*. *Website* sering juga di sebut *web*, dapat diartikan suatu kumpulan halaman-halaman yang menampilkan berbagai macam informasi teks, data gambar diam ataupun bergerak, data dinamis, suara maupun gabungan dari semuanya, baik itu yang bersifat statis maupun dinamis yang dimana membentuk suatu rangkaian bangunan yang saling berkaitan dimana masing-masing dihubungkan dengan jaringan halaman atau *hyperlink* (Nurhadi, 2017).

Pada dasarnya *website* dibagi menjadi dua bagian, yaitu :

##### a. Website Statis

Merupakan *web* yang halamannya tidak berubah, biasanya untuk melakukan perubahan di lakukan secara manual dengan mengubah kode. *Website* statis informasinya merupakan informasi satu arah, yakni hanya berasal dari pemilik *software*nya saja, hanya bisa di *update* oleh pemiliknya saja, contoh dari pengertian *website* statis ini adalah *profil* perusahaan.

##### b. Website Dinamis

Merupakan *web* yang halamannya selalu *update*, biasanya terdapat halman *backend* (halaman administrator) yang di gunakan untuk menambah atau merubah konten. *Web* dinamis membutuhkan *database* untuk menyimpan data. *Web* dinamis mempunyai arus informasi dua arah, yakni berasal dari pengguna dan pemilik, sehingga peng-*update*-an dapat di lakukan oleh pengguna dan pemilik *website*. Contoh dari pengertian *website* dinamis ini yaitu *friendster*, *Multiply*, *Facebook* dan lain-lain.

#### I. My Structured Query Language (MySQL)

MySQL adalah Sebuah program *database server* yang mampu menerima dan mengirimkan datanya sangat cepat, *multi user* serta menggunakan perintah dasar SQL (*Structured Query Language*). MySQL merupakan dua bentuk lisensi, yaitu *FreeSoftware* dan *Shareware*. MySQL yang biasa kita gunakan adalah MySQL *FreeSoftware* yang berada dibawah Lisensi GNU/GPL (*General Public License*), (Haris Spuro, 2012).

MySQL Merupakan sebuah *database server* yang *free*, artinya kita bebas menggunakan database ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya. MySQL pertama kali dirintis oleh seorang *programmer database* bernama Michael Widenius, Selain *database server*, MySQL juga merupakan program yang dapat mengakses suatu *database* MySQL yang berposisi sebagai *Server*, yang berarti program kita berposisi sebagai *Client*. Jadi MySQL adalah sebuah *database* yang dapat digunakan sebagai *Client* maupun *server*.

#### J. Hypertext Preprocessor (PHP)

Bahasa pemrograman PHP merupakan bahasa pemrograman yang cukup populer dan banyak di gunakan oleh para *programmer* di dunia. PHP atau yang memiliki kepanjangan *Hypertext Preprocessor*, merupakan suatu bahasa pemrograman yang di fungsikan untuk membangun suatu *website* dinamis, PHP menyatu dengan kode HTML. HTML digunakan sebagai pembangun atau pondasi dari kerangka *layout web*, sedangkan PHP difungsikan sebagai prosesnya, sehingga dengan adanya PHP tersebut sebuah web akan mudah di *maintenance*, (Agus Saputra, 2013).

PHP berjalan pada sisi *server* sehingga PHP disebut juga sebagai bahasa *Sever Side Scripting*, artinya bahwa dalam setiap/unuk menjalankan PHP, akan mebutuhkan *web server* untuk menjalankannya. PHP ini bersifat *open source* sehingga dapat di pakai secara gratis, dan mampu lintas *platform*, yaitu dapat berjalan pada sistem operasi *widow* maupun *linux*.

#### K. Unified Modeling Language (UML)

Menurut (Ivar Jacobson, 2010). *Unified Modeling Language* (UML) adalah bahasa standar untuk menulis cetak biru perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, membuat, dan mendokumentasikan artefak dari sistem

intensif perangkat lunak. UML hanya bahasa dan hanya satu bagian dari metode pengembangan perangkat lunak.

#### L. Model Driven Development

Teknik pengembangan berbasis model (MDD) menekankan gambar model untuk membantu memvisualisasikan dan menganalisis masalah, mendefinisikan kebutuhan bisnis, dan merancang sistem informasi. Analisis dan desain sistem terstruktur - berpusat pada proses Teknik informasi (IE) - berpusat pada data Analisis dan desain berorientasi obyek (OOAD) - terpusat pada objek (integrasi data dan masalah proses) Rute model driven development.

Tahapan yang dipakai pada penelitian adalah sebagai berikut:

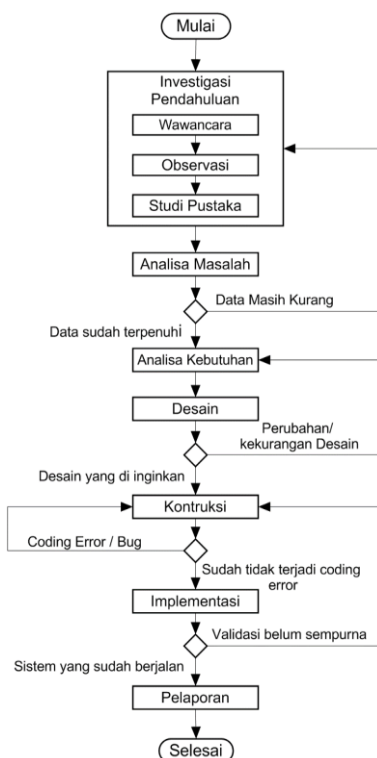
- 1) *Preliminary investigation* (investigasi awal)
- 2) *Problem analysis* (Analisis masalah)
- 3) *Requirements analysis* (Analisis Kebutuhan)
- 4) *Design* (Desain)
- 5) *Construction* (Kontruksi)
- 6) *Implementation* (implementasi)

### III. PEKERJAAN DAN DISKUSI HASIL

#### A. Proses Pekerjaan

##### 1. Kerangka Berfikir

Berikut ini adalah langkah-langkah yang di lakukan untuk mencapai tujuan dari penelitian, ditunjukkan dengan gambar dibawah ini:

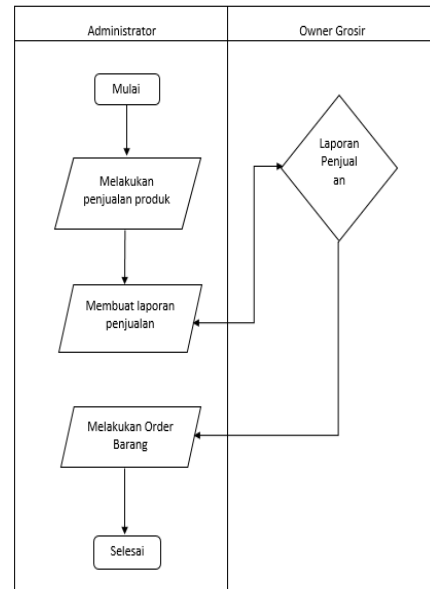


Gambar 1. Gambaran Umum Penelitian

#### 2. Analisis Sistem

Analisis system ini bertujuan untuk membandingkan sistem yang sedang berjalan saat ini dan sistem usulan yang akan dilakukan dalam penelitian ini.

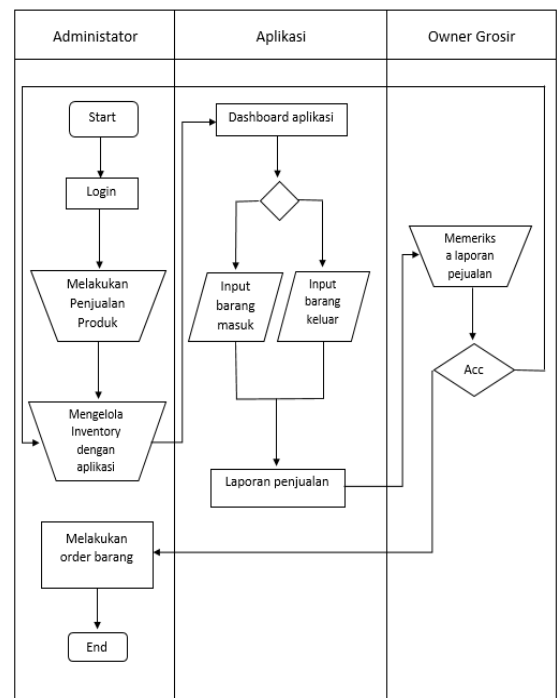
##### a. Analisis Sisem yang Sedang Berjalan



Gambar 2. Analisis Sistem Yang Sedang Berjalan

Pada gambar di atas dapat di simpulkan bahwa segala aktifitas di lakukan secara manual, tidak menggunakan sistem komputasi komputer.

##### b. Anlisis sistem yang Diusulkan



Gambar 3. Analisis Sistem Yang diusulkan

Pada gambar di atas dapat di simpulkan bahwa Admin Tidak perlu lagi membuat laporan penjualan secara

manual karena, laporan penjualan telah di buat oleh aplikasi yang di usulkan.

### c. Analisis Kebutuhan

Analisis kebutuhan mencakup pekerjaan-pekerjaan penentuan kebutuhan atau kondisi yang harus di penuhi dalam suatu produk baru atau perubahan produk yang mempertimbangkan berbagai kebutuhan, dari hasil analisis ini harus dapat di laksanakan, diukur, diuji, terkait dengan kebutuhan bisnis yang teridentifikasi, serta di definisikan sampai tingkat detail yang memadai untuk detail sistem. Berikut adalah beberapa software dan hardware yang di butuhkan :

#### 1. Kebutuhan Sistem

Tabel 1. Kebutuhan Sistem

No	Perangkat Keras	Perangkat Lunak
1	Seperangkat PC / Laptop	Windows 7 32 bit
2	-	Xampp
3	-	Sublime text
4	-	Mozilla Fire fox
5	-	Microsoft word 2013
6	-	Star UML
7	-	Balsamiq mockups 3 V.3.5.17
8	-	Framework CSS

#### 2. Kebutuhan Hardware

Tabel 2. Kebutuhan Hardware

No	Spesifikasi	Komputer
1	Processor	Intel Celeron 1.80 GHz (2CPUs)
2	Ram (Random Acces Memory)	2 GB
3	Hard Disk	320 GB

#### 3. Kebutuhan Software

Tabel 3. Kebutuhan Software

No	Spesifikasi	Komputer
1	Sistem Operasi	Windows 7 32 bit
2	Bahasa Pemograman	Bahasa Pemograman PHP

### 3. Fitur-Fitur

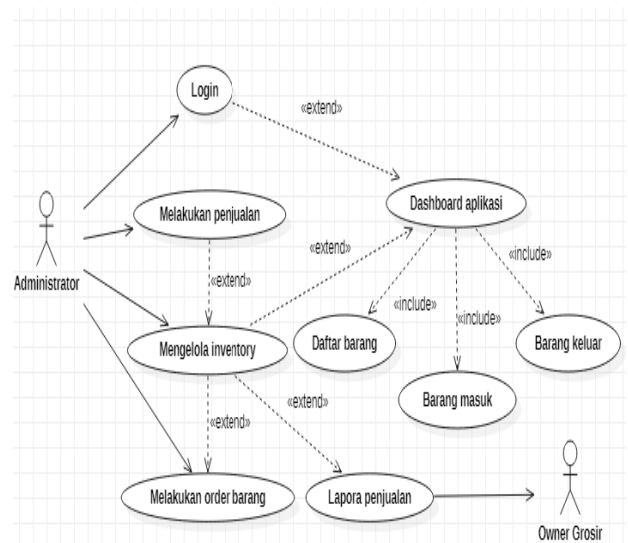
Fitur-fitur yang di gunakan dalam aplikasi ini di maksudkan agar *user* dapat dengan mudah mengoperasikan-nya, di tunjang dengan metode EOQ dan ROP untuk memberikan biaya paling ekonomis dalam pengelolaan *inventory*, bagian terpenting dari fitur-fitur yang ada adalah sebagai berikut :

- Login user
- Stok alert
- Penghitungan EOQ otomatis
- Penghitungan ROP otomatis
- Biaya-biaya dan *leadtime* bisa di *custom*
- Dashboard dan grafik sederhana
- Calculate stock (bukan *trigger*)

### 4. Perancangan Sistem

Pada tahap ini perangkat lunak dideskripsikan dengan model analisis penggunaan diagram *Use Case*. Analisis digunakan untuk pemetaan awal mengenai perilaku yang diisyaratkan sistem aplikasi kedalam elemen-elemen pemodelan. Untuk membantu perancangan dan melengkapi dokumentasi perancangan.

#### a. Use Case Diagram



Gambar 4. Usecase Diagram

Diagram Use Case di atas menunjukkan Actor yang terlibat di dalam Aplikasi pengelolaan Inventory ada 2 Actor, yang masing-masing mempunyai hak sebagai berikut :

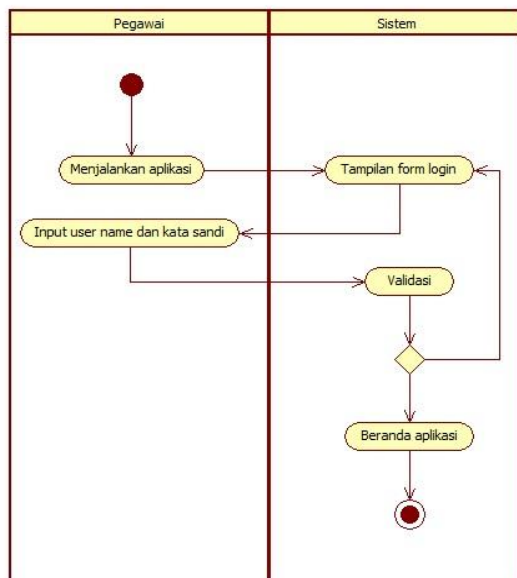
#### 1. Definisi Aktor

- Pegawai  
Pegawai adalah staf Grosir Sembako yang diberikan kewenangan untuk melakukan pengelolaan (*maintenance*) aplikasi. Dalam sistem yang akan di kembangkan peran administrator di pegang oleh pegawai. Pegawai dapat melihat dan mengubah semua data sistem, dan dapat membenahi kesalahan yang terjadi dalam sistem.

- *Owner grosir*  
Owner Grosir dapat melakukan berbagai hal yang di lakukan pegawai, akan tetapi segala hal yang bersangkutan dengan pengelolaan sistem di serahkan kepada pegawai, sehingga Owner Grosir dapat menerima berbagai macam laporan dari pegawai, dan apabila Owner Grosir menyetujui laporan order barang yang telah di serahkan pegawai, pegawai dapat melakukan order barang.

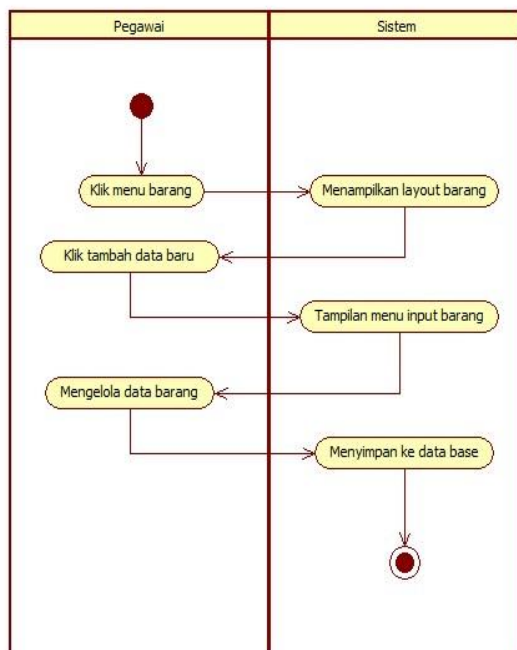
## b. Actifity Diagram

### 1) Actifity Diagram Login



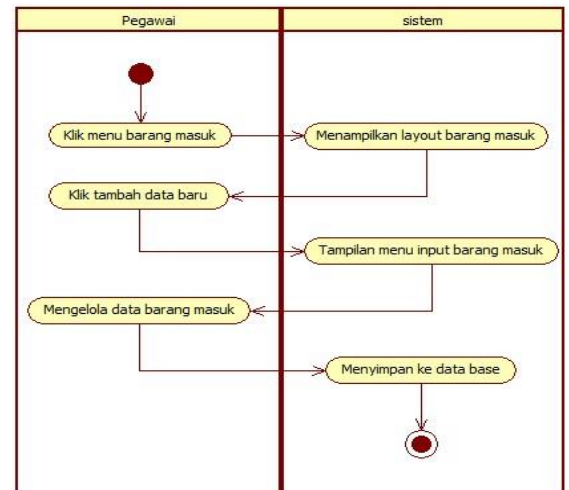
Gambar 5. Actifity diagram login

### 2) Actifity diagram olah data barang



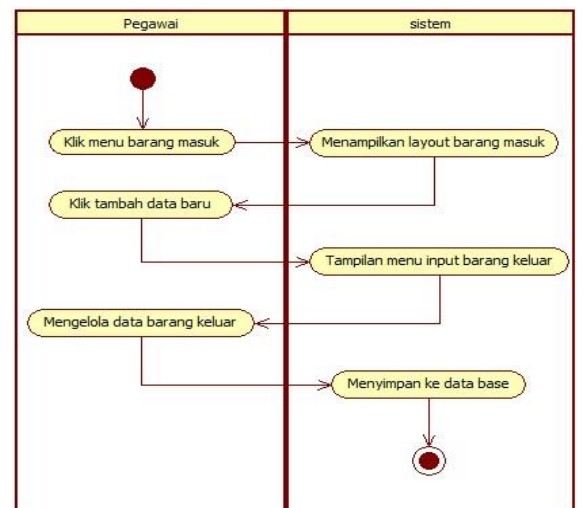
Gabar 6. Actifity diagram olah data barang

### 3) Actifity diagram olah data barang masuk



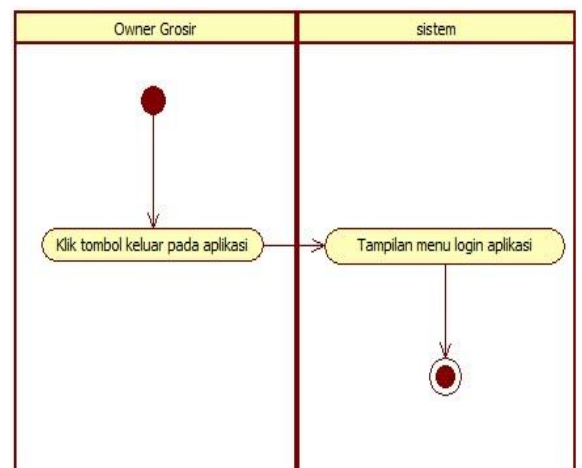
Gambar 7. Actifity diagram olah data barang masuk

### 4) Actifity diagram olah data barang keluar



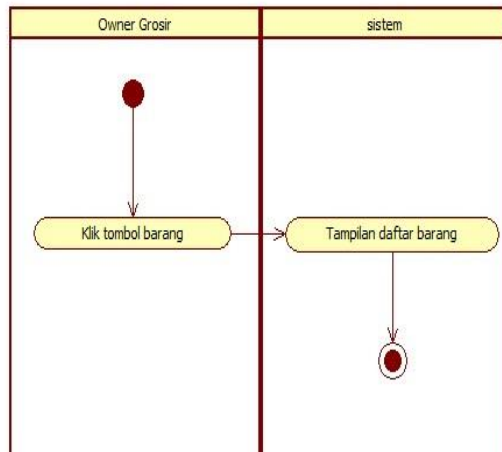
Gambar 8. Actifity diagram olah data barang keluar

### 5) Actifity diaram Llogout



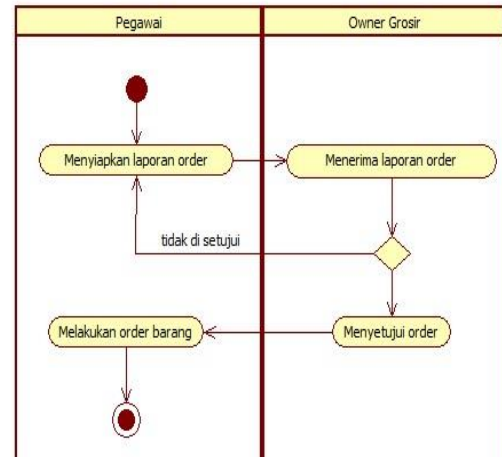
Gambar 9. Actifity diagram logout

6) *Activity diagram* laporan data barang



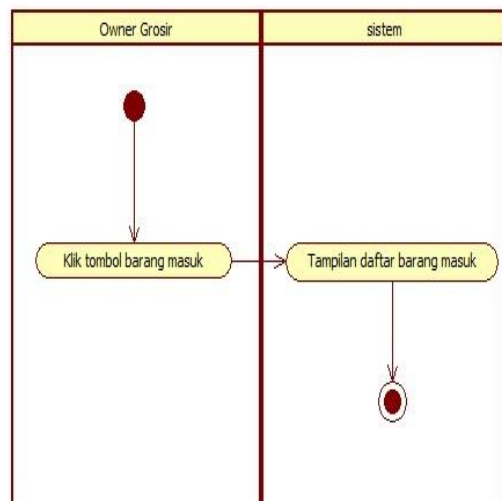
Gambar 9. Activity diagram laporan data barang

9) *Activity diagram* order barang



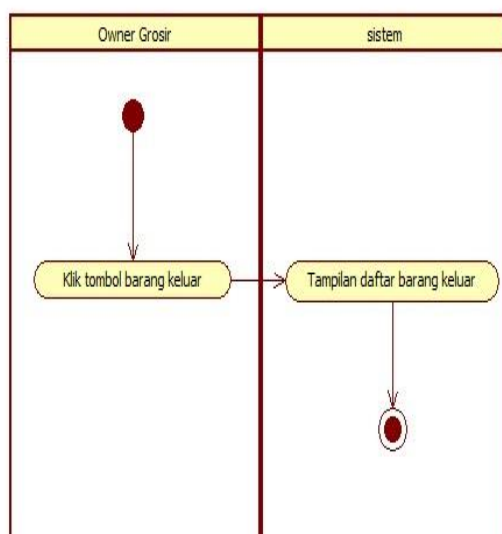
Gambar 12. Activity diagram order barang

7) *Activity diagram* data barang masuk



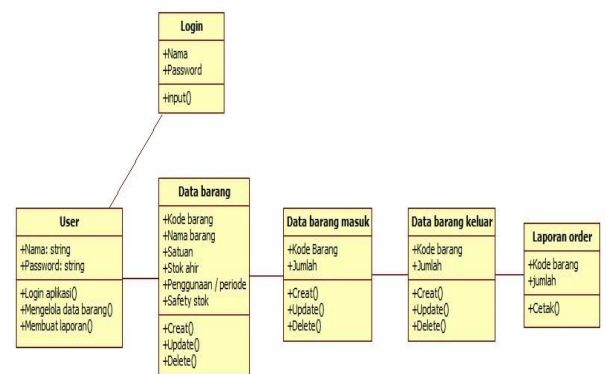
Gambar 10. Activity diagram data barang masuk

8) *Activity diagram* data barang keluar



Gambar 11. Activity diagram data barang keluar

c. *Class Diagram*



Gambar 13. Class diagram aplikasi pengendalian inventory

Pada gambar 4.11 *Class diagram* (diagram kelas) ini menggambarkan struktur sistem aplikasi pengendalian *inventory* yang diusulkan dari segi pendefinisian kelas-kelas yang akan dibuat untuk membuat aplikasi pengendalian *inventory* pada grosir sembako.

d. Implementasi EOQ dan ROP

Sebagaimana yang tertera pada halaman judul tugas akhir ini, penyusun menggunakan metode *Economic order Quantity* (EOQ) dan *Re-order Point* (ROP) untuk mengelola *inventory* pada grosir sembako. Maka dari itu penyusun menyisipkan rumus EOQ dan ROP pada aplikasi yang di buat.

```

115 {
116     $model->attributes-$POST['Barang'];
117     $model->attributes-$POST['Barang'];
118     $rop = $model->safety_stok + ($model->penggunaan_tahun/365);
119     $model->rop = round($rop, 0, PHP_ROUND_HALF_UP);
120     $r_eoq = ( 2 * $model->penggunaan_tahun * (($op/100) * $model->harga)) / (($bpe/100) * $
    model->harga);
121     $model->eoq = round(sqrt($r_eoq), 0, PHP_ROUND_HALF_UP);
122     if($model->save())
123         $this->redirect(array('manage'));
124 }
125
  
```

Gambar 14. Implementasi EOQ dan ROP



Pada gambar di atas dapat di lihat penyusun menggunakan rumus EOQ dan ROP, pada baris 118 terdapat rumus ROP :  $SS + (D/365)$ , dan pada baris 120 terdapat rumus EOQ :  $\sqrt{\{(2D \times S) / H\}}$ . setelah memastikan rumus pada aplikasi selanjutnya adalah pembuktian penghitungan EOQ dan ROP pada aplikasi.

Kode Barang	Nama Barang	Stok Akhir	Harga	Penggunaan/Tahun	Safety Stok	Rop	Eoq
001	Djarum Coklat	20	12200	18000	20	69	120

Gambar 15. Hasil penghitungan EOQ dan ROP

Pada gambar di atas dapat di lihat aplikasi menampilkan ROP : 69 dan EOQ : 120 untuk produk Djarum Coklat. Berikut adalah penghitungan secara manual :

- a. Menghitung EOQ

$$\text{Rumus : } \sqrt{\{(2D \times S) / H\}}$$

**D** = Permintaan *product* / Tahun, **S** = biaya pemesanan, **H** = biaya penyimpanan per periode.  
D = 18.000, S = 10%, H = 25%

$$\text{EOQ} = \sqrt{(2) (18.000) (10\%) / (25\%) = \sqrt{14400} = 120$$

- b. Menghitung ROP

$$\text{Rumus : } SS + (D/365)$$

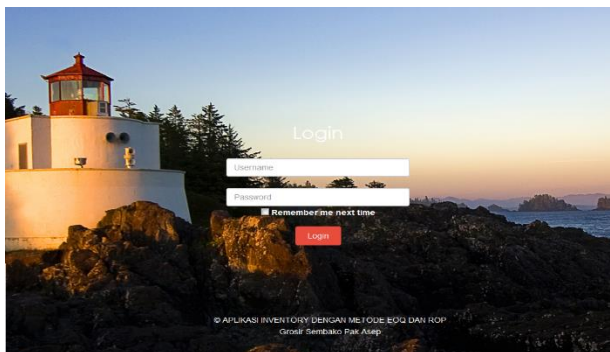
**SS** = *Safety Stock*, **D** = Penggunaan / Tahun

$$SS = 20, PT = 18000 \text{ pcs}$$

$$20 + (18000 / 365) = 69$$

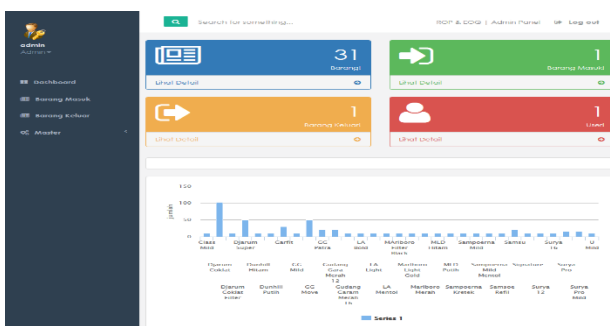
- d. Implementasi antar muka

- 1) Form *Login* aplikasi



Gambar 16. Form login aplikasi

- 2) *Dashboard* Aplikasi



Gambar 17. Dashboard aplikasi

- 3) Halaman Data barang

Gabar 18. Halama data barang

- 4) Halaman form input data barang

Gambar 19. Halamn *Form input* data barang

- 5) Halaman barang masuk

Gambar 20. Halaman barang masuk

- 6) *Form* isi data barang masuk



No	Tujuan	Input	Output diharapkan	Output sistem
1	Lihat detail barang	Klik lihat detail pada menu barang	Menampilkan detail barang yang ada	*Sukses *Detail barang tampil sesuai dengan <i>output</i>
2	Lihat detail barang masuk	Klik lihat detail pada menu barang masuk	Menampilkan detail barang masuk	*Sukses *Detail barang masuk tampil sesuai dengan <i>output</i>
3	Lihat detail barang keluar	Klik lihat detail pada menu barang keluar	Menampilkan detail barang keluar	*Sukses *Detail barang keluar tampil sesuai dengan <i>output</i>
4	Lihat detail <i>user</i>	Klik lihat detail pada meu user	Menampilkan detail data user	*Sukses *Detail data user tampil sesuai dengan <i>output</i>
5	Menuju menu master	Klik menu master	Menampilkan menu master	*Sukses *Detail menu master tampil sesuai dengan <i>output</i>
No	Tujuan	Input	Output diharapkan	Output sistem
1	Lihat detail barang	Klik lihat detail pada menu barang	Menampilkan detail barang yang ada	*Sukses *Detail barang tampil sesuai dengan

Gambar 21. Form isi data barang masuk

## 7) Halaman barang keluar

ID	Kode Barang	Jumlah	Tanggal
4	0001	5	2019-08-07
5	0005	3	2019-08-15
6	0023	2	2019-08-15
7	0019	10	2019-08-15

Gambar 22. Halaman barang keluar

## 8) Halaman isi data barang keluar

Gambar 23. Halaman isi dat barang keluar

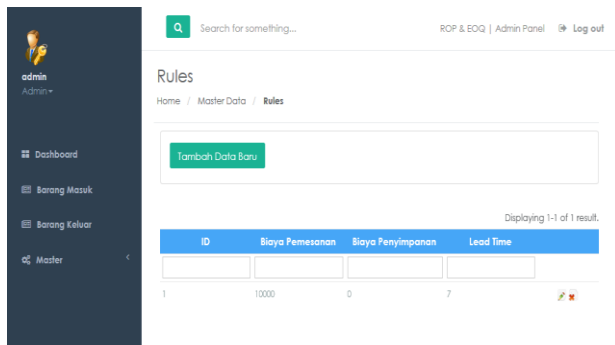
## 9) Halaman *form user*

GROUP_ID	User Name
Admin	admin

Gambar 24. Halaman *form user*

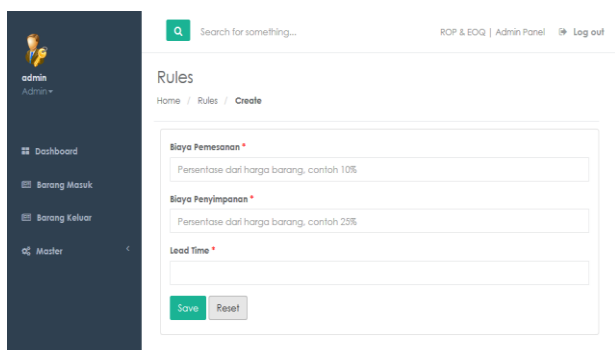
## 10) Halaman rules

3	Lihat detail barang keluar	Klik lihat detail pada menu barang keluar	Menampilkan detail barang keluar	*Sukses *Detail barang keluar tampil sesuai dengan <i>output</i>
4	Lihat detail <i>user</i>	Klik lihat detail pada meu user	Menampilkan detail data user	*Sukses *Detail data user tampil sesuai dengan <i>output</i>
5	Menuju menu master	Klik menu master	Menampilkan menu master	*Sukses *Detail menu master tampil sesuai dengan <i>output</i>
No	Tujuan	<i>Input</i>	<i>Output</i> diharapkan	<i>Output</i> sistem
1	Lihat detail barang	Klik lihat detail pada menu barang	Menampilkan detail barang yang ada	*Sukses *Detail barang tampil sesuai dengan <i>output</i>
2	Lihat detail barang masuk	Klik lihat detail pada menu barang masuk	Menampilkan detail barang masuk	*Sukses *Detail barang masuk tampil sesuai dengan <i>output</i>
3	Lihat detail barang keluar	Klik lihat detail pada menu barang keluar	Menampilkan detail barang keluar	*Sukses *Detail barang keluar tampil sesuai dengan <i>output</i>



Gambar 25. Halaman *rules*

## 11) Halaman isi data rules



Gambar 26. Halaman isi data *rules*

## IV KESIMPULAN

Berdasarkan penelitian yang di lakukan penulis melalui beberapa tahapan yang di lakukan pada bab-bab sebelumnya maka penulis dapat menyimpulkan :

1. Dengan terselesaikannya penelitian ini penulis dapat membuat suatu aplikasi yang mana tujuan dari aplikasi tersebut adalah memudahkan pemilik grosir dalam mengelola *inventory*
2. Dari uji coba yang di lakukan pada Grosir Sembako, penulis dapat menyimpulkan bahwa aplikasi ini dapat mengoptimalkan pemilik grosir dalam mengelola *inventory*.
3. Dengan adanya aplikasi ini penulis dapat memahami bagaimana cara mengelola *inventory* menggunakan metode EOQ dan ROP pada Grosir Sembako, yang mana melalui metode EOQ (*Economic Order Quantity*) pemilik grosir dapat menentukan kuantitas pemesanan barang paling ekonomis, di lengkapi dengan metode ROP (*Re-order Point*), metode ini menentukan kapan barang harus di sediakan kembali sehingga mencegah terjadinya ketidak tersediaan produk

## REFERENSI

- [1]. Agus Saputra. (2013). *Smarty PHP OOP Engine for PHP Template*. Bandung: PT.Elex Media Computindo.
- [2]. Hapusan Naingolan. (2010). Perhitungan metode Economic Order Quantity (EOQ), Reorder Point (ROP) dan Safety Stock (SS)

dalam mengendalikan persediaan bahan baku pada PT Nutune Batam. Bandung.

- [3]. Haris Spuro. (2012). Modul Pembelajaran Praktek Basis Data MySQL.
- [4]. Hengki Fitrayco dan Habibi. (2009). Makalah Pengendalian Persediaan (Infentory Control). 3.
- [5]. Hengki Fitrayco dan Habibi. (2009). Makalah Pengendalian Persediaan (Infentory Control). 2.
- [6]. Ivar Jacobson. (2010). *The Univied Modelig Languge Reverence Manual*.
- [7]. Nurhadi. (2017). *Pondasi Dasar Pemograman Website*.
- [8]. R. S. Presman. (2010). *Software Engineering A Practioner's approach 7th ed*. MC Grow Hill.
- [9]. Robi Muhamad. (2009). Makalah Aplikasi Komputer. .
- [10]. Taufik Iqbal, D. A. (2016). Aplikasi mamagement Persediaan Barang Berbasis Economic Order Quantity.
- [11]. Tomi Lukmana, Diana Trivena. (2015). Penerapan Metode EOQ dan ROP (Studi Kasus: PD. BARU). *Jurnal Teknik Informatika dan Sistem Informasi Volume 1 Nomor 3*.
- [12]. Tri Muhamad Hani. (2011). Metode EOQ (Econonomic Order Quantity).
- [13]. Turmudi, M. (2017). Aplikasi Optimasi dan Evisiensi Persediaan Menggunakan Metode EOQ, Savety Stock dan ROP di Toko Batik Nusantara.
- [14]. Wijaya. (2009). *Model Driven Developement* .