



**RANCANG BANGUN APLIKASI PENGENDALIAN PERSEDIAAN  
MENGUNAKAN METODE ROP DAN EOQ PADA PT PRANA  
ARGENTUM CORPORATION**

**TUGAS AKHIR**



**Program Studi  
S1 SISTEM INFORMASI**

**UNIVERSITAS  
Dinamika**

**Oleh:**

**FAREL FALERI**

**19410100005**

---

**FAKULTAS TEKNOLOGI DAN INFORMATIKA**

**UNIVERSITAS DINAMIKA**

**2023**

RANCANG BANGUN APLIKASI PENGENDALIAN PERSEDIAAN  
MENGUNAKAN METODE ROP DAN EOQ PADA PT PRANA  
ARGENTUM CORPORATION

TUGAS AKHIR

Diajukan sebagai salah syarat untuk menyelesaikan  
Program Sarjana



UNIVERSITAS  
**Dinamika**

Oleh:

Nama : Farel Faleri  
NIM : 19410100005  
Program Studi : S1 Sistem Informasi

FAKULTAS TEKNOLOGI DAN INFORMATIKA  
UNIVERSITAS DINAMIKA

2023

Tugas Akhir

RANCANG BANGUN APLIKASI PENGENDALIAN PERSEDIAAN  
MENGUNAKAN METODE ROP DAN EOQ PADA PT PRANA  
ARGENTUM CORPORATION

Dipersiapkan dan disusun oleh

Farel Faleri

NIM: 19410100005

Telah diperiksa, dibahas dan disetujui oleh Dewan Pembahas

Pada: 12 Juli 2023

Susunan Dewan Pembahas

Pembimbing

I. Pantjawati Sudarmaningtyas, S.Kom., M.Eng.

NIDN. 0712066801

II. Dr. Eng. Valentinus Roby Hananto, S.Kom., M.Sc.

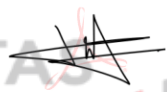
NIDN. 0715028903

Pembahas

I. Ir. Henry Bambang Setyawan, M.M.

NIDK. 8973650022

  
Digitally signed by  
Universitas  
Dinamika  
Date: 2023.07.31  
20:31:53 +07'00'



  
Digitally signed  
by Henry  
Bambang S

Tugas Akhir ini telah diterima sebagai salah satu persyaratan

untuk memperoleh gelar Sarjana:



Digitally signed by  
Universitas Dinamika  
Date: 2023.08.02  
10:55:28 +07'00'

Tri Sagirani, S.Kom., M.MT.

NIDN. 0731017601

Dekan Fakultas Teknologi dan Informatika  
UNIVERSITAS DINAMIKA

*Never can we suspend the time  
Having to leave the tracks behind  
There is a longer way ahead*

*After all*

- Farel Faleri



UNIVERSITAS  
**Dinamika**



*Kupersembahkan kepada*  
*Keluarga,*  
*Bapak Ibu Dosen,*  
*Civitas Universitas Dinamika*  
*Teman, sahabat dan almamater*  
*Universitas Dinamika yang kubanggakan.*

**SURAT PERNYATAAN**  
**PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH**

Sebagai mahasiswa Universitas Dinamika, Saya :

Nama : Farel Faleri  
NIM : 19410100005  
Program Studi : S1 Sistem Informasi  
Fakultas : Fakultas Teknologi dan Informatika  
Jenis Karya : Tugas Akhir  
Judul Karya : **RANCANG BANGUN APLIKASI PENGENDALIAN PERSEDIAAN MENGGUNAKAN METODE ROP DAN EOQ PADA PT PRANA ARGENTUM CORPORATION**

Menyatakan dengan sesungguhnya bahwa :

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, Saya menyetujui memberikan kepada Universitas Dinamika Hak Bebas Royalti *Non-Eksklusif (Non-Exclusive Royalty Free Right)* atas seluruh isi/sebagian karya ilmiah Saya tersebut diatas untuk disimpan, dialihmediakan, dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama Saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut diatas adalah hasil karya asli Saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya, atau pendapat orang lain yang ada dalam karya ilmiah ini semata-mata hanya sebagai rujukan yang dicantumkan dalam Daftar Pustaka Saya.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiasi pada karya ilmiah ini, maka Saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada Saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 20 Juni 2023



**Farel Faleri**  
NIM : 19410100005

## ABSTRAK

PT Prana Argentum Corporation merupakan perusahaan yang berfokus pada jasa *pest control*, salah satu kegiatan dari *pest control* yaitu fumigasi. Fumigasi merupakan teknik pengendalian hama dengan memasukan *gas methyl bromide* (CH<sub>3</sub>Br) kedalam ruang kedap udara dengan tingkat konsentrasi yang cukup dan waktu yang sudah ditentukan. Untuk melakukan aktivitas fumigasi diperlukan beberapa peralatan yaitu *gas methyl bromide* (CH<sub>3</sub>Br), gas oksigen, *cartridge gas mask*, plastik fumigasi, *sand snake*, dan kertas sertifikat fumigasi. Namun berdasarkan hasil wawancara kepada pihak perusahaan, seringkali terjadi *out of stock* pada persediaan tertentu. Dalam satu bulan terdapat 95 *container* dari total 625 *container* atau 14% jumlah *container* mengalami penundaan aktivitas fumigasi diakibatkan kekurangan persediaan untuk melakukan aktivitas fumigasi. Penelitian ini bertujuan untuk mengatasi permasalahan *out of stock* dengan menggunakan metode *reorder point* (ROP) dan *economic order quantity* (EOQ). *Reorder point* digunakan untuk mengetahui kapan waktu yang tepat untuk memesan persediaan. Lalu *economic order quantity* untuk mengetahui berapa jumlah persediaan yang dipesan untuk menghindari *overstock* pada persediaan. Untuk pengujian pada penelitian ini, menggunakan *whitebox testing* dan membandingkan hasil perhitungan *manual* dengan hasil perhitungan aplikasi. Dari hasil pengujian *whitebox testing* pada aplikasi terbukti dengan menggunakan rumus *cyclomatic complexity*, perhitungan pada aplikasi menghasilkan 2 alur path. Lalu 2 alur path melalui proses *test case* dan menyimpulkan bahwa 2 alur path sudah berjalan dengan baik pada aplikasi. Hasil pengujian perhitungan metode *reorder point* dan *economic order quantity* menunjukkan konsistensi antara perhitungan yang dilakukan menggunakan aplikasi dengan perhitungan *manual*. Hal ini membuktikan bahwa aplikasi untuk metode *reorder point* dan *economic order quantity* berjalan dengan benar dan sesuai dengan harapan peneliti.

Kata Kunci : *Reorder Point*, *Economic Order Quantity*, *Out Of Stock*, Fumigasi.



## KATA PENGANTAR

Segala syukur dan puji hanya bagi Tuhan Yesus Kristus, oleh karena anugerah-Nya yang melimpah, kemurahan dan kasih setia yang besar akhirnya penulis dapat menyelesaikan penulisan tugas akhir dengan judul “Rancang Bangun Aplikasi Pengendalian Persediaan Menggunakan Metode ROP dan EOQ pada PT Prana Argentum Corporation”. Laporan Tugas Akhir disusun sebagai salah satu syarat menyelesaikan program strata satu di Universitas Dinamika.

Penyelesaian Tugas Akhir ini tidak terlepas dari kontribusi berbagai pihak yang telah memberikan berbagai masukan, nasihat, saran, kritik, dan dukungan baik secara moral maupun materiil kepada penulis. Oleh karena itu, dengan tulus penulis menyampaikan rasa terima kasih kepada:

1. Ibu, Bapak dan keluarga tercinta keluarga yang selalu mendoakan, memberi dukungan, dan memberikan semangat pada setiap langkah yang dilalui dan aktivitas penulis.
2. Bapak Dr. Anjik Sukmaaji, S.Kom., M.Eng. selaku Ketua Program Studi S1 Sistem Informasi Universitas Dinamika yang telah memberikan izin kepada penulis untuk melaksanakan Tugas Akhir.
3. Ibu Pantjawati Sudarmaningtyas, S.Kom., M.Eng, selaku Dosen S1 Sistem Informasi sekaligus dosen pembimbing pertama dalam kegiatan Tugas Akhir selalu memberikan dedikasi waktu dan pemikirannya untuk membimbing, mendukung, dan memberikan arahan kepada penulis dalam seluruh proses penyelesaian Tugas Akhir.
4. Bapak Dr. Eng. Valentinus Roby Hananto, S.Kom., M.Sc.. selaku Dosen S1 Sistem Informasi sekaligus dosen pembimbing kedua dalam kegiatan Tugas Akhir selalu menunjukkan kesabaran dan ketekunan dalam memberikan arahan kepada penulis.
5. Bapak Ir. Henry Bambang Setyawan, M.M . selaku Dosen SI sekaligus dosen penguji yang telah memberikan saran serta masukan dalam penyelesaian Tugas Akhir ini.



6. *Owner* PT Prana Argentum Corporation sekaligus salah satu penyemangat penulis untuk menyelesaikan tugas akhir ini, terima kasih Bapak Doddy Prahastho.
7. Teman-teman tercinta yang memberikan bantuan dan dukungan dalam menyelesaikan Tugas Akhir ini.
8. Pihak-pihak lain yang tidak dapat disebutkan satu-persatu yang telah memberikan bantuan dan dukungan kepada penulis.

Semoga Tuhan YME senantiasa melimpahkan rahmat dan hidayah-Nya. Pada akhirnya, penulis berharap skripsi ini dapat memberikan manfaat, baik bagi penulis secara khusus maupun bagi semua yang membutuhkannya secara umum.

Surabaya, 12 Juli 2023

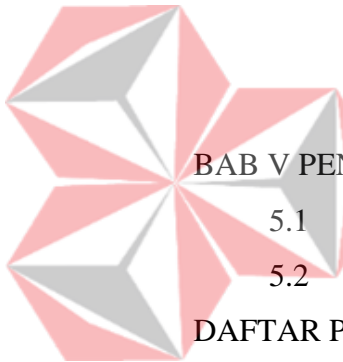


UNIVERSITAS  
**Dinamika** Penulis

## DAFTAR ISI

	Halaman
ABSTRAK .....	vii
KATA PENGANTAR .....	viii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR .....	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	5
1.5 Manfaat.....	5
BAB II LANDASAN TEORI .....	6
2.1 Penelitian Terdahulu .....	6
2.2 Pengendalian Persediaan.....	7
2.3 <i>System Development Life Cycle (SDLC)</i> .....	8
2.4 <i>Reorder Point (ROP)</i> .....	9
2.5 <i>Economic Order Quantity (EOQ)</i> .....	9
2.6 <i>Lead Time</i> .....	10
2.7 <i>Safety Stock</i> .....	11
2.8 <i>Website</i> .....	11
2.9 <i>Object Oriented Analysis And Design (OOAD)</i> .....	11
2.10 <i>White Box Testing</i> .....	11
BAB III METODOLOGI PENELITIAN.....	14
3.1 <i>Planning</i> .....	14
3.1.1 Observasi.....	14
3.1.2 Wawancara.....	15
3.1.3 Studi Literatur .....	16
3.1.4 Analisis Kebutuhan Sistem .....	16
3.1.5 Diagram <i>Input, Proses dan Output (IPO)</i> .....	21

3.2	<i>Design</i> .....	25
3.2.1	<i>Use Case Diagram</i> .....	25
3.2.2	<i>Activity Diagram</i> .....	26
3.2.3	<i>Sequance Diagram</i> .....	28
3.2.4	<i>Class Diargam</i> .....	29
3.2.5	<i>Desain Antarmuka</i> .....	30
3.3	<i>Coding</i> .....	30
3.4	<i>Testing</i> .....	31
3.5	<i>Release</i> .....	31
BAB IV HASIL DAN PEMBAHASAN .....		32
4.1	<i>Coding</i> .....	32
4.1.1	Implementasi Aplikasi .....	32
4.2	<i>Testing</i> .....	34
4.2.1	<i>White Box</i> .....	34
4.2.2	Implementasi Perhitungan ROP.....	39
4.2.3	Implementasi Perhitungan EOQ .....	40
BAB V PENUTUP.....		42
5.1	Kesimpulan.....	42
5.2	Saran.....	42
DAFTAR PUSTAKA .....		43
LAMPIRAN.....		45



UNIVERSITAS  
Dinamika

## DAFTAR TABEL

	Halaman
Tabel 1.1. <i>Bill Of Material</i> .....	1
Tabel 1.2. Permasalahan. ....	2
Tabel 2.1. Penelitian Terdahulu .....	6
Tabel 2.2. Notasi <i>Flow Graph</i> .....	12
Tabel 3.1. Identifikasi Masalah.....	18
Tabel 3.2. Peran dan Tanggung jawab pengguna. ....	19
Tabel 3.3. Identifikasi Kebutuhan Fungsional. ....	20
Tabel 3.4. Identifikasi Non-Fungsional. ....	20
Tabel 3.5. Identifikasi Perangkat. ....	21
Tabel 3.6. Perhitungan EOQ. ....	27
Tabel 3.7. Perhitungan ROP. ....	28
Tabel 4.1. <i>Graph Matrix</i> ROP .....	37
Tabel 4.2. <i>Graph Matrix</i> EOQ .....	37
Tabel 4.3. Alur <i>path</i> ROP .....	38
Tabel 4.4. Alur <i>path</i> EOQ.....	38
Tabel 4.5. Hasil <i>Test Case</i> ROP.....	38
Tabel 4.6. Hasil <i>Test Case</i> EOQ .....	38
Tabel 4.7. Rekap Data Permintaan Persediaan Dalam Periode Maret-April-Mei	39
Tabel 4.8. Perhitung Manual ROP Pada Periode Maret-April-Mei.....	39
Tabel 4.9. Perhitungan Manual EOQ Dalam Periode Maret-April-Mei.....	40

## DAFTAR GAMBAR

	Halaman
Gambar 3.1. Metodologi Penelitian. ....	14
Gambar 3.2. Diagram BPMN.....	15
Gambar 3.3. Diagram BPMN Menggunakan Aplikasi .....	17
Gambar 3.4. Diagram IPO . ....	21
Gambar 3.5. Diagram IPO. ....	22
Gambar 3.6. Diagram IPO. ....	23
Gambar 3.7. Use Case Diagram. ....	26
Gambar 3.8. <i>Activity Diagram</i> Perhitungan EOQ.....	27
Gambar 3.9. <i>Activity Diagram</i> Perhitungan ROP. ....	28
Gambar 3.10. <i>Sequance Diagram</i> Perhitungan EOQ.....	29
Gambar 3.11. <i>Sequance Diagram</i> Perhitungan ROP. ....	29
Gambar 3.12. <i>Class Diagram</i> .....	30
Gambar 4.1 Implementasi Perhitungan ROP.....	32
Gambar 4.2 Implementasi Perhitungan EOQ. ....	33
Gambar 4.3. Implementasi <i>Dashboard</i> .....	33
Gambar 4.4. <i>Source Code</i> ROP .....	34
Gambar 4.5. <i>Source Code</i> EOQ .....	35
Gambar 4.6. <i>Flow Graph</i> Perhitungan ROP .....	35
Gambar 4.7. <i>Flow Graph</i> Perhitungan EOQ.....	36
Gambar 4.8. Hasil Perhitungan Aplikasi ROP.....	40
Gambar 4.9. Hasil Perhitungan Aplikasi EOQ .....	41

## DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Hasil Wawancara .....	45
Lampiran 2. Activity Diagram .....	46
Lampiran 3. Sequence Diagram .....	66
Lampiran 4. Hasil <i>Desain</i> Antarmuka .....	79
Lampiran 5. Implementasi Aplikasi .....	89
Lampiran 6. Data Penggunaan Dalam 1 Periode .....	96
Lampiran 7. Hasil Cek Plagiasi .....	99
Lampiran 8. Biodata Penulis .....	101



UNIVERSITAS  
**Dinamika**

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Pada revolusi industri 4.0 perkembangan teknologi begitu cepat sehingga segala kegiatan yang dikerjakan oleh manusia memerlukan bantuan dari sebuah teknologi terutama dari teknologi informasi. PT Prana Argentum Corporation merupakan perusahaan yang melakukan *pest control* dengan menyediakan layanan jasa fumigasi, analisis, *survey*, dan *packing* yang berlokasi di tanjung perak pada jalan Ikan Mungsing V. Fumigasi itu sendiri dapat diartikan sebagai teknik pengendalian hama dengan memasukan gas *Methyl Bromide* (CH<sub>3</sub>Br) ke dalam ruangan kedap udara dengan tingkat konsentrasi yang cukup dan waktu yang sudah ditentukan. Fumigasi untuk 1 *container* membutuhkan 3-4 kg gas *Methyl Bromide* (CH<sub>3</sub>Br). PT Prana Argentum Corporation memberikan layanan jasa fumigasi untuk 2 jenis ukuran *container* yaitu 20ft dan 40ft. Dalam memberikan layanan tersebut PT Prana Argentum Corporation memiliki standar kompetensi dan standar operasi yang selalu di audit setiap tahunnya oleh badan karantina pertanian/tumbuhan Indonesia (PT Prana Argentum Corporation, 2016). Tabel 1.1 berisi rincian bahan atau alat yang digunakan untuk melakukan proses fumigasi.

Tabel 1.1. *Bill Of Material*.

No	Jenis Container	Nama Persediaan	Penggunaan
1	Container 20ft	<i>Gas Methyl Bromide</i> (CH <sub>3</sub> Br)	3,5 KG
		Gas Oksigen	2 Tabung
		<i>Cartridge Gas Mask</i>	2 <i>Cartridge</i>
		Plastik Fumigasi	1 Plastik Fumigasi
		<i>Sand Snake</i>	34 <i>Sandsnake</i>
		Kertas Sertifikat Fumigasi	9 Lembar
2	Container 40ft	<i>Gas Methyl Bromide</i> (CH <sub>3</sub> Br)	5,5 KG
		Gas Oksigen	2 Tabung
		<i>Cartridge Gas Mask</i>	2 <i>Cartridge</i>
		Plastik Fumigasi	1 Plastik Fumigasi
		<i>Sand Snake</i>	44 <i>Sandsnake</i>
		Kertas Sertifikat Fumigasi	9 Lembar

PT Prana Argentum Corporation memberikan layanan fumigasi secara B2B (*business to business*), proses bisnis diawali adanya pesanan fumigasi dari perusahaan pemilik *container* ekspor maupun impor, selanjutnya dilakukan proses fumigasi terhadap *container* milik pemesan. Setelah melalui tahap fumigasi



selanjutnya adalah tahap penyiapan dokumen dan pengeluarnya sertifikat sebagai bukti *container* sudah siap dikirim dan diserahkan ke *shipper* untuk dikirim ke tempat tujuan.

Permasalahan yang terjadi pada saat ini adalah perusahaan sering kali mengalami kendala keterlambatan dalam melakukan fumigasi dikarenakan terjadinya permasalahan pada kegiatan pengadaan barang ataupun bahan yang digunakan saat melakukan fumigasi. Barang yang sering kali kekurangan adalah *gas Methyl Bromide* (CH<sub>3</sub>Br), gas oksigen, *cartridge gas mask*, plastik fumigasi, *sand snake* dan kertas sertifikat. Untuk detail dari barang yang sering kehabisan dapat dilihat pada Tabel 1.2.

Tabel 1.2. Permasalahan.

Nama Barang	Penyebab	Akibat	Penggunaan Per Bulan	Kekurangan Per Bulan
<i>Gas Methyl Bromide</i> (CH <sub>3</sub> Br).	Dikarenakan penggunaan <i>gas Methyl Bromide</i> (CH <sub>3</sub> Br) pada setiap melakukan proses fumigasi tidak tentu tergantung dari kondisi suhu saat melakukan fumigasi. Ketika suhu lebih rendah dari yang seharusnya maka gas yang digunakan akan semakin banyak, sehingga terjadi penggunaan gas yang berlebih.	Menyebabkan penggunaan <i>gas Methyl Bromide</i> (CH <sub>3</sub> Br) pada setiap <i>container</i> tidak pasti dan dapat menggunakan lebih dari ekspektasi awal, dari melebihi ekspektasi awal dapat terjadinya <i>out of stock</i> , dan menunda proses fumigasi.	2500 KG.	80 KG.
Gas Oksigen.	Kurangnya monitoring persediaan di depo, sehingga tidak mengetahui berapa banyak sisa persediaan yang dimiliki.	Terjadi <i>out of stock</i> , dan tertundanya proses fumigasi jika melakukan proses fumigasi tanpa gas oksigen dapat melanggar <i>safety measure</i> dari pemerintah.	60 tabung.	9 tabung.
<i>Cartridge Gas Mask</i> .	Kurangnya monitoring persediaan di depo, sehingga tidak mengetahui berapa banyak	Terjadi <i>out of stock</i> dan tertundanya proses fumigasi jika melakukan proses fumigasi tanpa <i>Cartridge Gas Mask</i> dapat	32 <i>cartridge</i> .	3 <i>cartridge</i> .

Nama Barang	Penyebab	Akibat	Penggunaan Per Bulan	Kekurangan Per Bulan
	sisa persediaan yang dimiliki.	melanggar <i>safety measure</i> dari pemerintah.		
Plastik Fumigasi.	Kurangnya monitoring persediaan di depo, sehingga tidak mengetahui berapa banyak sisa persediaan yang dimiliki.	Terjadi <i>out of stock</i> dan dapat menunda proses fumigasi jika melakukan proses fumigasi tanpa plastik fumigasi dapat membahayakan pekerja saat menjalankan proses fumigasi dan dapat melanggar <i>safety measure</i> dari pemerintah.	84 plastik fumigasi.	11 plastik fumigasi.
<i>Sand Snake.</i>	Kurangnya monitoring persediaan di depo, sehingga tidak mengetahui berapa banyak sisa persediaan yang dimiliki.	Terjadi <i>out of stock</i> dan dapat menunda proses fumigasi jika fumigasi tetapi dijalankan dapat membahayakan pekerja sekitar saat menjalankan proses fumigasi dan dapat melanggar <i>safety measure</i> dari pemerintah.	680 <i>sandsnake.</i>	63 <i>sandsnake.</i>
Kertas Sertifikat Fumigasi.	Kurangnya monitoring persediaan, sehingga tidak mengetahui berapa banyak sisa persediaan yang dimiliki.	Terjadi <i>out of stock</i> dan menghambat proses dokumentasi untuk pesanan fumigasi.	27 kertas sertifikat fumigasi.	3 kertas sertifikat fumigasi.

Dapat diperkirakan dalam 1 bulan terdapat 95 *container* dari total 625 *container* atau 15% total *container* yang mengalami penundaan proses fumigasi diakibatkan kekurangan persediaan untuk melakukan proses fumigasi.

Apabila proses fumigasi ditunda akan menimbulkan masalah bagi perusahaan berupa denda dari depo dan *shipper*. Hal ini akan menyebabkan pengeluaran tak terduga pada perusahaan hingga 10% dari pendapatan perbulannya. Jika aktivitas fumigasi tetap dijalankan tanpa menggunakan *safety measure*, perusahaan terkena sanksi dari pemerintah karena membahayakan fumigator dan pekerja yang berkerja di sekitar area fumigasi yang mengakibatkan perusahaan tidak dapat melakukan aktivitas fumigasi atau tidak dapat beroperasi hingga jangka

waktu yang ditentukan oleh pemerintah. Hal ini dapat menyebabkan kerugian pada perusahaan dari tidak adanya pemasukan, pembatalan kegiatan fumigasi serta pertanggungjawaban terhadap karyawan.

Metode *Economic Order Quantity* (EOQ) digunakan untuk mengelola *stock* atau persediaan dengan tujuan untuk menentukan jumlah barang yang optimal untuk meminimalisir biaya persediaan. Penerapan metode EOQ dapat mengurangi risiko terjadinya kekurangan *stock* atau kelebihan *stock* yang dapat mengganggu proses operasional perusahaan. Dengan adanya metode EOQ yang efektif, perusahaan dapat menghemat biaya persediaan yang dikeluarkan dan menjaga efisiensi dalam manajemen persediaan. (Ismunandar, Hendriadi, & Garno, 2019).

*Reorder Point* (ROP) merupakan titik kritis di mana perlu dilakukan pemesanan persediaan guna menghindari *out of stock*. ROP dihitung berdasarkan tingkat penggunaan persediaan dan waktu pemesanan, dengan tambahan *safety stock*. Tujuan perhitungan ROP adalah untuk memastikan ketersediaan persediaan yang cukup sebelum terjadi kehabisan *stock* (Rafliana & Suteja, 2018).

Solusi dari permasalahan yang dialami oleh PT Prana Argentum Corporation adalah dengan merancang dan membangun sebuah aplikasi yang berfungsi untuk mengatasi masalah *out of stock* dan *overstock* persediaan menggunakan metode ROP yang berfungsi untuk mengetahui kapan pemesanan persediaan, dan metode EOQ yang berfungsi untuk mengetahui berapa banyak persediaan yang harus dibeli untuk meminimalisir biaya persediaan.

## 1.2 Rumusan Masalah

Berdasarkan uraian permasalahan yang ada pada latar belakang, maka rumusan masalahnya yaitu, bagaimana merancang dan membangun aplikasi pengendalian persediaan dengan Metode EOQ dan ROP pada PT Prana Argentum Corporation.

## 1.3 Batasan Masalah

Berdasarkan uraian diatas, maka aplikasi ini hanya mengakomodir sistem pengendalian persediaan pada PT Prana Argentum Corporation.

1. Pada aplikasi hanya digunakan untuk persediaan gas *methyl bromide* (CH<sub>3</sub>Br), gas oksigen, *cartridge gas mask*, plastik fumigasi, *sand snake*, dan kertas sertifikat fumigasi.
2. Aplikasi hanya berfokus pada kapan membeli persediaannya dan berapa banyak persediaan yang dibeli.
3. Aplikasi hanya memiliki 3 jenis user yaitu, fumigator, manager, dan staff administrasi.
4. Aplikasi tidak dapat membedakan jenis ukuran *container* pada pesanan *customer*.

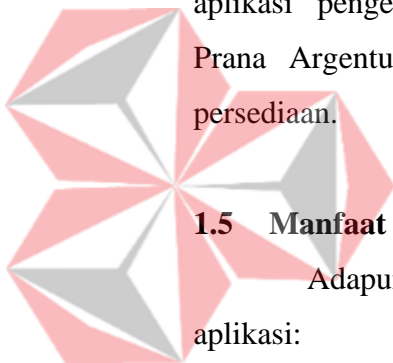
#### 1.4 Tujuan

Berdasarkan uraian dari latar belakang dan rumusan masalah, maka dapat disesuaikan bahwa tujuan dari penelitian ini mampu menghasilkan rancang bangun aplikasi pengelolaan persediaan menggunakan metode EOQ dan ROP pada PT Prana Argentum untuk membantu pihak perusahaan dalam mengendalikan persediaan.

#### 1.5 Manfaat

Adapun manfaat dari penelitian yang dilaksanakan, berikut manfaat dari aplikasi:

1. Membantu perusahaan dalam pemesanan persediaan pada kapan melakukan pemesanan persediaan dengan metode ROP.
2. Membantu perusahaan dalam pemesanan persediaan pada berapa banyak persediaan yang perlu dipesan kepada *supplier* dengan metode EOQ.
3. Mempermudah perusahaan saat pengecekan persediaan dikarenakan semua sudah tercatat pada komputer.



## BAB II

### LANDASAN TEORI

Dalam pelaksanaan penelitian ini, digunakan alur ilmiah sebagai landasan teori dan referensi dalam implementasi tahapan *extreme programming* untuk mengembangkan aplikasi. Oleh karena itu, penelitian ini didasarkan pada pemahaman teori yang luas. Beberapa konsep teori yang menjadi dasar dalam penelitian ini antara lain pengendalian persediaan, *system development life cycle* (SDLC), *economic order quantity* (EOQ), *reorder point* (ROP), *lead time*, *safety stock*, *website*, *object oriented analysis dan design* (OOAD), serta *whitebox testing*. Pemahaman teori ini menjadi landasan penting dalam proses penelitian ini.

#### 2.1 Penelitian Terdahulu

Dalam penelitian ini, akan digunakan beberapa penelitian terdahulu sebagai acuan untuk membandingkan perbedaan dengan penelitian yang dilakukan. Tabel 2.1 berisi ringkasan penelitian-penelitian terdahulu yang relevan.

Tabel 2.1. Penelitian Terdahulu

No	Judul Penelitian	Nama Peneliti	Tahun	Hasil Penelitian	Perbedaan Penelitian Sekarang
1	Rancang Bangun Sistem Pengelolaan Stok Berbasis Website Dengan Metode EOQ pada Toko Murah Mojokerto.	Aldo Julianto.	2022	Menghasilkan aplikasi pengelolaan <i>stock</i> pada Toko Murah Mojokerto yang memiliki sistem transaksi yang bisa mencatat semua transaksi pada Toko Murah, dan terdapat mengetahui kapan pengisian persediaan berikutnya serta banyak persediaan yang perlu di beli.	Perbedaan penelitian ini dan penelitian sebelumnya yaitu, pada penelitian sebelumnya tidak menggunakan metode <i>safety stock</i> dan memiliki rumus ROP yang berbeda, aplikasi hanya memiliki 2 <i>rule(user)</i> , lalu pada penelitian sebelumnya menggunakan <i>blackbox</i>

No	Judul Penelitian	Nama Peneliti	Tahun	Hasil Penelitian	Perbedaan Penelitian Sekarang
2	Rancang Bangun Sistem Persediaan Gudang pada PT Abadi Jaya Manunggal Menggunakan Metode EOQ ( <i>Economy Order Quantity</i> ).	Wildana Lathif Mahmudi, Dedy Kurniadi, ST., M.Kom, Ir. Agus Adhi Nugroho, M.T., IPM.	2019	Menghasilkan sebuah Aplikasi persediaan Gudang pada PT abadi jaya manunggal yang dapat mengetahui jumlah pengisian persediaan yang efektif untuk setiap pembelian persediaan serta fitur untuk rekap <i>stock</i> perbulannya.	<i>testing</i> sebagai tahap <i>testing</i> . Perbedaan penelitian ini dan penelitian sebelumnya yaitu, pada penelitian sebelumnya hanya berfokus pada metode EOQ, lalu pada penelitian sebelumnya menggunakan <i>blackbox testing</i> sebagai tahap <i>testing</i> .
3	Implementasi Metode EOQ untuk Persediaan Organik Pada Koperasi Unit Desa (KUD) Menranti.	Adeliana Putri, Fauriatun Helmiyah, Sri Rezeki Maulina Azmi.	2022	Menghasilkan sebuah aplikasi yang dapat meningkatkan kinerja karyawan dan membantu karyawan gudang dalam meningkatkan kebutuhan koperasi unit dan dapat mengontrol jumlah persediaan yang ada serta dapat menghasilkan laporan dengan cepat dan mudah.	Perbedaan penelitian ini dan penelitian sebelumnya yaitu, pada penelitian sebelumnya hanya berfokus pada metode EOQ, memiliki lama periode yaitu 12 bulan.

Dari beberapa penelitian terdahulu diatas, memiliki beberapa perbedaan dengan penelitian ini yaitu, data yang akan digunakan untuk melakukan perhitungan ROP, EOQ, dan *Safety Stock* diambil dari jangka waktu 3 bulan terakhir permintaan persediaan, lalu ROP memiliki perbedaan rumus dari 3 penelitian diatas, dan yang terakhir berfokus pada 6 alat yang akan digunakan untuk melakukan proses fumigasi.

## 2.2 Pengendalian Persediaan

Persediaan memiliki peran yang krusial dalam operasional perusahaan. Secara umum, persediaan dapat didefinisikan sebagai *stock* bahan baku yang digunakan untuk memenuhi permintaan *customer*. Dengan demikian, dapat

disimpulkan bahwa persediaan merupakan komponen yang penting dalam menjalankan proses produksi dan dapat memberikan kepuasan yang lebih tinggi terhadap permintaan *customer* (Dewi, Herawati, & Wahyuni, 2019). Persediaan yang digunakan dalam penelitian ini, yaitu gas *methyl bromide* (CH<sub>3</sub>Br), gas oksigen, *cartridge gas mask*, kertas sertifikat fumigasi, plastik fumigasi, dan *sand snake*.

### 2.3 System Develoment Life Cycle (SDLC)

*System Development Life Cycle* (SDLC) merupakan suatu kerangka kerja yang terdiri dari tahapan analisis kebutuhan perangkat lunak, desain sistem, pembangunan sistem, pengujian sistem, dan implementasi. Pengembangan SDLC membutuhkan sebuah sistem yang memiliki kemampuan untuk membuat dan menyimpan struktur basis data. (Setiany, et al., 2021). Model yang sering digunakan oleh *developer* untuk membuat sebuah sistem biasanya menggunakan model *extreme programming*.

*Extreme Programming* (XP) adalah suatu model pengembangan aplikasi yang menyederhanakan beberapa tahapan dalam pengembangan sistem, dengan tujuan untuk meningkatkan efisiensi, adaptabilitas, dan fleksibilitas. Model ini memiliki beberapa nilai dasar yang menjadi landasan, antara lain *communication*, *courage*, *simplicity*, *feedback*, dan *quality work* (Budiarti & Risyanto, 2020).

#### A. Planning

Tahap awal dimulai dengan melakukan analisis kebutuhan yang diperlukan untuk merencanakan desain aplikasi melalui melakukan wawancara langsung dengan pemilik PT Prana Argentum, dan mengidentifikasi permasalahan yang muncul dalam sistem saat sistem sedang beroperasi.

#### B. Design

Pada tahap ini, dimulai dengan melakukan perancangan sistem menggunakan *software design*, *system design*, dan *prototyping*. Hasil dari tahap ini adalah sebuah *blueprint* yang akan digunakan pada tahap selanjutnya.



### C. Coding

Setelah mendapatkan *blueprint* dari tahap sebelumnya, tahap berikutnya adalah memulai pembangunan sistem/aplikasi berbasis website sesuai dengan *blueprint* yang telah disusun.

### D. Testing

Pada tahap ini, dilakukan pengujian terhadap sistem yang telah dibangun pada tahap sebelumnya untuk memastikan bahwa sistem berjalan sesuai dengan perencanaan. Jika hasil pengujian tidak memenuhi harapan, maka dilakukan evaluasi kembali pada tahap perencanaan.

### E. Release

Apabila tahap pengujian berjalan sesuai dengan perencanaan, maka aplikasi dapat di-*hosting* dan siap untuk digunakan.

## 2.4 Reorder Point (ROP)

*Reorder Point* (ROP) adalah titik di mana perlu dilakukan pemesanan persediaan agar persediaan yang telah dipesan tiba tepat waktu dan persediaan *safety stock* mencapai nol. Pemesanan persediaan ini dilakukan secara berulang. (Djalamang, Qosim, & Hasan, 2021).

$$ROP = (LT \times DS) + SS \quad (1)$$

Keterangan:

ROP = *Reorder Point* (Kg)

LT = *Lead Time* (hari)

DS = rata rata *Demand* per satuan waktu (Kg)

SS = *Safety Stock* (Kg)

## 2.5 Economic Order Quantity (EOQ)

*Economic Order Quantity* (EOQ) merupakan jumlah optimal dari persediaan yang dibeli pada setiap periode dengan tujuan mencapai efisiensi yang maksimal. Metode EOQ bertujuan untuk meminimalkan persediaan, mengurangi biaya, dan meningkatkan kualitas. Penerapan metode EOQ dalam perencanaan persediaan perusahaan dapat mengurangi risiko kehabisan *stock* yang dapat

mengganggu aktivitas perusahaan, serta menghemat biaya persediaan. Dengan menerapkan rancangan metode EOQ, perusahaan dapat mengoptimalkan persediaan, mengurangi biaya penyimpanan, memaksimalkan penggunaan ruang gudang, dan menghindari masalah yang dapat timbul akibat penumpukan persediaan yang berlebihan. Analisis EOQ merupakan alat yang mudah digunakan dan praktis dalam pengelolaan persediaan. (Ismunandar, Hendriadi, & Garno, 2019).

Rumus perhitungan EOQ yang akan digunakan pada aplikasi pengelolaan persediaan pada PT Prana Argentum yaitu sebagai berikut:

$$EOQ = \sqrt{\frac{2DS}{H}} \quad (2)$$

Keterangan:

EOQ = Jumlah pemesanan dengan biaya inventori minimal (Kg).

S = Biaya pesan untuk per pesanan (Rp).

D = Jumlah permintaan per periode (Kg).

H = Biaya penyimpanan per unit per periode (Rp).

Untuk mengestimasi siklus permintaan persediaan per periode, kita menggunakan jumlah permintaan persediaan dalam periode tersebut. Periode yang dipilih adalah 3 bulan terakhir, dengan tujuan untuk memperoleh data yang optimal. Pemilihan periode 3 bulan ini didasarkan pada pertimbangan untuk mengoptimalkan permintaan data yang akurat. Fluktuasi pesanan dari bulan ke bulan dapat berbeda-beda, dengan adanya periode 3 bulan, kita dapat mengatasi fluktuasi tersebut dan memperoleh gambaran yang lebih stabil. Berikut ini adalah rumus yang digunakan untuk menghitung permintaan dan biaya pemesanan yang akan digunakan dalam perhitungan EOQ.

## 2.6 Lead Time

*Lead Time* adalah interval waktu antara pemesanan persediaan hingga penerimaan persediaan yang telah dipesan. Ketika proses pengisian persediaan memiliki *lead time* yang lebih lama, dapat menyebabkan pemborosan dalam perusahaan karena meningkatnya waktu pemrosesan yang tidak efisien (Nurwulan, Taghsya, Astuti, Fitri, & Nisa, 2021).

## 2.7 Safety Stock

*Safety Stock* adalah persediaan tambahan yang digunakan untuk menjaga potensi kekurangan persediaan. Keberadaan *safety stock* membantu mengurangi risiko kerugian yang timbul akibat kehabisan persediaan. Dengan adanya *safety stock*, perusahaan dapat mengurangi biaya pemisahan yang timbul ketika persediaan mencapai tingkat yang kritis. (Chusminah, Haryati, & Nelfiant, 2019).

$$SS = (\text{Permintaan tertinggi} - \text{Permintaan Rata - Rata}) \times LT \quad (3)$$

Keterangan:

SS = *Safety Stock* (Kg).

LT = *Lead Time* (hari).

## 2.8 Website

*Website* adalah sebuah platform digital yang berfungsi sebagai wadah untuk menyajikan informasi yang telah diolah oleh individu atau kelompok. Umumnya, website *dihosting* di server web dan diakses melalui internet. (Suhendar & Rosmalina, 2020).

## 2.9 Object Oriented Analysis And Design (OOAD)

*Object Oriented Analysis And Design* (OOAD) adalah sebuah metode yang bertujuan untuk melakukan analisis dan perancangan sistem dengan menggunakan pendekatan berorientasi objek. Pada tahap analisis, identitas objek digunakan untuk membedakan objek tersebut dari objek lain, sedangkan perilaku objek dijelaskan melalui kejadian yang terjadi. Saat melakukan perancangan, identitas objek memungkinkan objek untuk saling berinteraksi dan berbagi operasi. Selain itu, objek juga dapat mempengaruhi objek lainnya dalam sistem (Purwaningtias, 2018).

## 2.10 White Box Testing

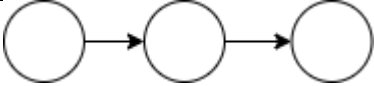
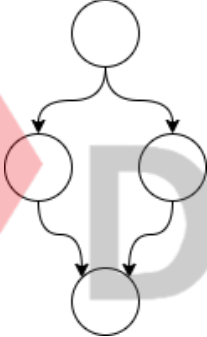
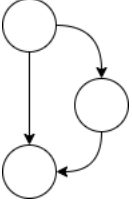
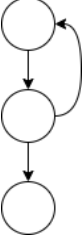
*White box*, juga dikenal sebagai *clear box* atau *glass box*, adalah metode pengujian perangkat lunak di mana pengujian dilakukan dengan pemahaman terperinci tentang struktur internal, desain, dan implementasi dari sistem yang sedang diuji. Dalam konteks pengujian perangkat lunak, *white box testing* sering

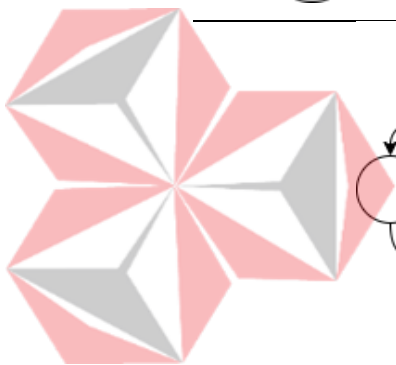
digunakan sebagai kebalikan dari *black box testing*. Dalam *white box testing*, penguji memiliki akses penuh ke kode sumber, struktur data, aliran eksekusi, dan komponen internal lainnya dari sistem yang sedang diuji (Sie1, Musdar, & Bahri, 2022).

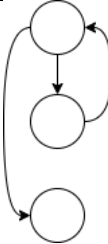
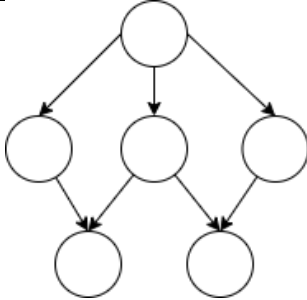
### 1. Flow Graph

*Flow graph* merupakan representasi visual dari aliran kontrol dalam suatu program. *Flow graph* menggambarkan bagaimana program berpindah dari satu pernyataan ke pernyataan lainnya melalui cabang keputusan *sequence*, *if-else*, *if-then*, *while*, *untils*, dan *case*. Pada Tabel 2.2 akan ditunjukkan jenis-jenis notasi yang digunakan dalam *flow graph* untuk menggambarkan alur program.

Tabel 2.2. Notasi *Flow Graph*

Notasi	Tipe
	<i>Sequence</i>
	<i>If-Else</i>
	<i>If-Then</i>
	<i>While</i>



Notasi	Tipe
	<i>Untils</i>
	<i>Case</i>

## 2. Cyclomatic Complexity

*Cyclomatic complexity* merupakan hasil perhitungan dari *flow graph* dengan menghitung jumlah putusan keputusan independen dalam suatu program. Putusan keputusan independen ini mencakup cabang keputusan *sequence*, *if-else*, *if-then*, *while*, *untils*, dan *case*.

$$\text{Cyclomatic Complexity } V(G) = E - N + 2 \quad (4)$$

Keterangan:

E = Jumlah *edge* (panah) dalam *flow graph*

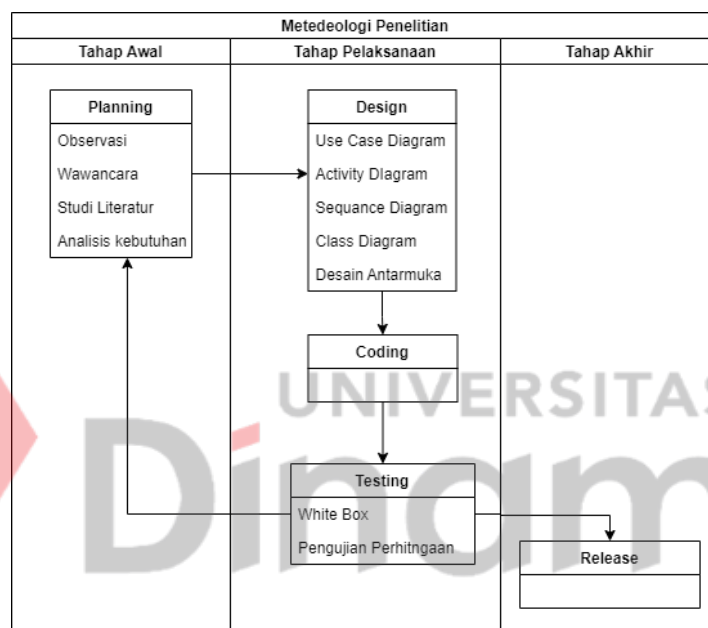
N = Jumlah *node* (titik) pada *flow graph*

## 3. Graph Matrix

*Matrix graph*, yang juga dikenal sebagai *matrix adjacency*, digunakan untuk menggambarkan hubungan antara simpul dan sisi dalam *flow graph*. Matriks ini mengandung entri-entri yang menunjukkan keberadaan *edge* yang menghubungkan dua *node* dalam *flow graph*.

## BAB III METODOLOGI PENELITIAN

Penelitian ini akan direalisasikan melalui tiga tahap utama, yaitu tahap awal, tahap pelaksanaan, dan tahap akhir. Tahap awal melibatkan *planning* yang matang, kemudian dilanjutkan dengan tahap pelaksanaan yang mencakup desain, *coding*, dan *testing*. Setelah aplikasi melalui proses *testing*, aplikasi akan siap untuk *release*. Rincian mengenai tahapan penelitian ini dapat dilihat pada Gambar 3.1.



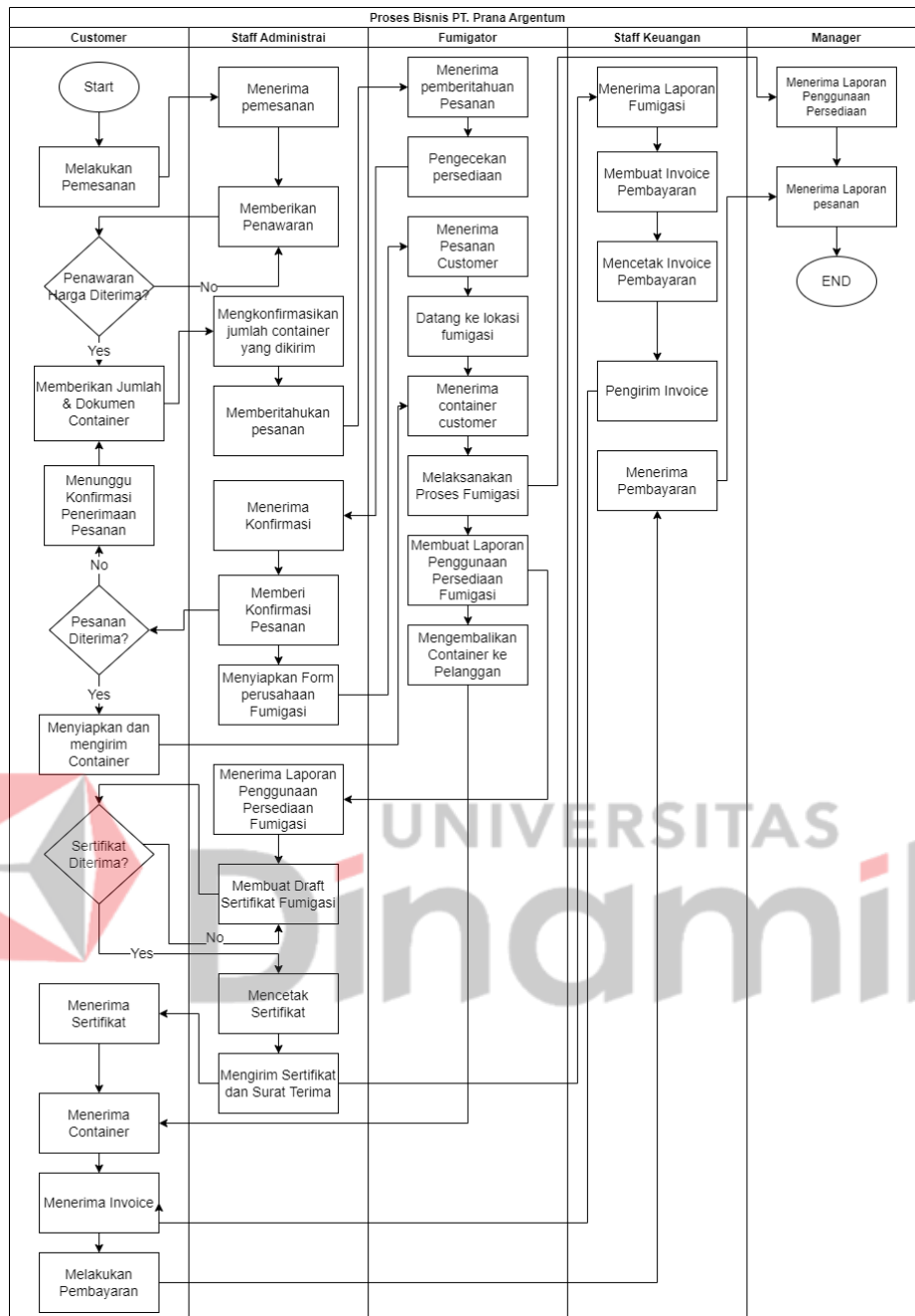
Gambar 3.1. Metodologi Penelitian.

### 3.1 *Planning*

#### 3.1.1 **Observasi**

Pada tahap observasi, peneliti mengunjungi lokasi perusahaan dengan tujuan memperoleh izin untuk melakukan penelitian. Peneliti secara teliti mengamati proses masuk dan keluarnya persediaan gas yang digunakan selama fumigasi di PT Prana Argentum Corporation. Observasi ini bertujuan untuk mendapatkan pemahaman yang mendalam mengenai persyaratan dan data yang diperlukan dalam pengelolaan persediaan PT Prana Argentum Corporation.

Untuk hasil observasi yang telah dilakukan, peneliti memahami alur proses bisnis saat ini, lalu proses bisnis tersebut dibuat dalam bentuk diagram yang akan digunakan untuk mencari alternatif solusi, dapat dilihat pada gambar 3.2.



Gambar 3.2. Diagram BPMN.

### 3.1.2 Wawancara

Pada tahap wawancara, peneliti menyusun serangkaian pertanyaan yang dirancang untuk mendukung proses penelitian. Pertanyaan tersebut menggali informasi terkait proses masuk dan keluar persediaan yang digunakan di perusahaan. Wawancara dilakukan dengan pihak perusahaan, dan hasil dari wawancara tersebut terlampir dalam Lampiran 1.



### 3.1.3 Studi Literatur

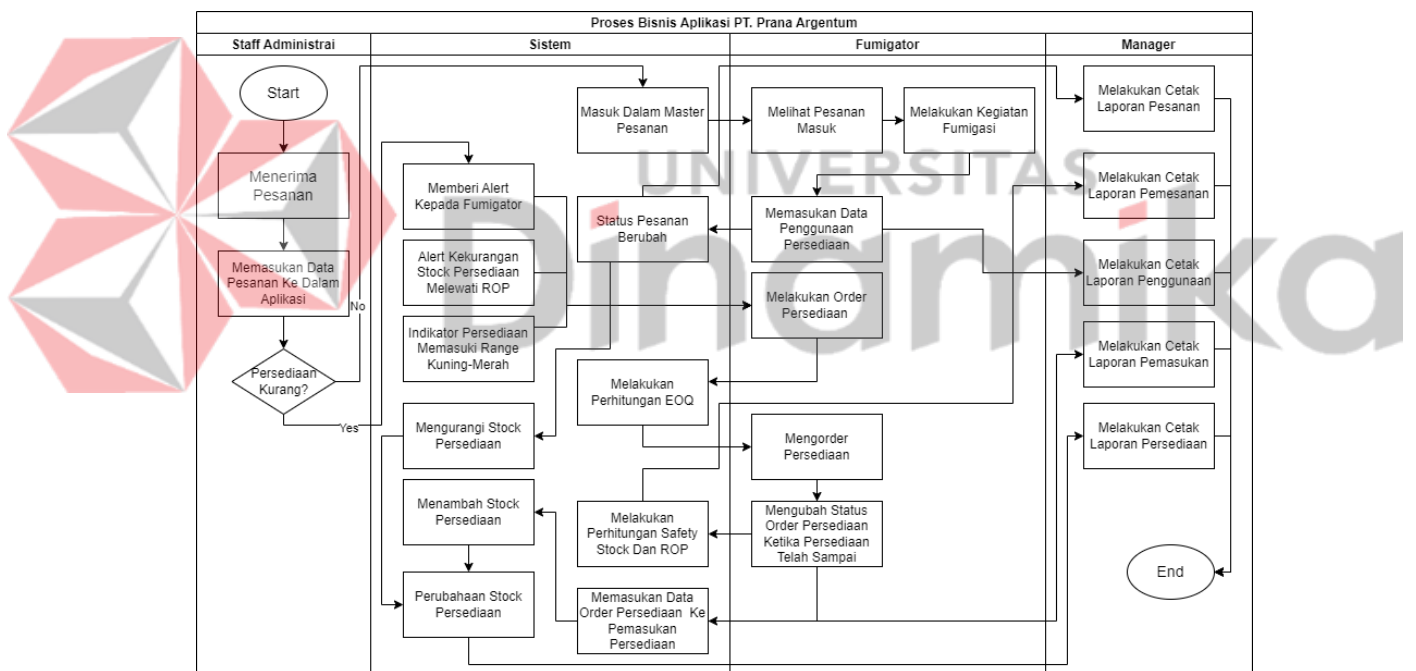
Studi literatur dilakukan dalam penelitian ini untuk mengidentifikasi permasalahan yang dihadapi oleh PT Prana Argentum Corporation. Tujuan dari studi literatur adalah untuk memperoleh pemahaman yang mendalam tentang masalah tersebut, serta menggali pengetahuan, metode, dan solusi yang relevan. Melalui studi literatur, diharapkan dapat menemukan berbagai teori yang terkait dengan permasalahan yang ada, yang nantinya dapat digunakan sebagai referensi dalam memberikan solusi. Hasil dari studi literatur ini terdokumentasikan dalam Bab 2, sebagai sumber informasi yang telah dipelajari.

### 3.1.4 Analisis Kebutuhan Sistem

#### A. Analisis Proses Bisnis

Penelitian ini dimulai dengan mengidentifikasi permasalahan yang ada. dari hasil wawancara dengan pihak perusahaan, sehingga menghasilkan identifikasi masalah. identifikasi kebutuhan fungsional dan identifikasi kebutuhan non-fungsional, untuk proses bisnis dimulai *customer* melakukan pesanan kepada staff administrasi, lalu dilanjutkan staff administrasi memberi penawaran harga dari pesanan kepada *customer*, jika *customer* menolak penawaran harga dari staff administrasi, maka staff administrasi akan memberi tawaran harga yang lain kepada *customer*, jika *customer* menerima penawaran harga dari staff administrasi, *customer* akan memberi detail dari pesanan seperti jumlah *container* dan dokumen yang berhubungan dengan *container*, setelah staff administrasi mengetahui *detail* pesanan *customer*, staff administrasi memberitahukan *detail* pesanan kepada fumigator untuk pengecekan persediaan, fumigator akan memberitahu kepada staff administrasi apakah pesanan dapat diterima atau tertundah, setelah staff administrasi mendapatkan informasi dari fumigator, staff administrasi akan memberi konfirmasi kepada *customer* mengenai status pesanan, lalu staff akan menyiapkan *form* perusahaan fumigasi, setelah *customer* mendapat status pesanan, *customer* bisa mengirimkan *container* sesuai *detail* pesanan ke depo dan diterima oleh fumigator, setelah fumigator menerima *container* milik *customer*, langsung melakukan proses fumigasi, setelah proses fumigasi selesai fumigator akan mengirim *container* ke tempat *shipper*, lalu akan membuat laporan mengenai

permintaan persediaan, setelah selesai pembuatan laporan tersebut akan dikirim ke manager dan staff administrasi, setelah staff administrasi menerima laporan penggunaan, maka dilanjutkan pembuatan sertifikat fumigasi untuk setiap *container*, setelah selesai pembuatan sertifikat fumigasi akan dikirim ke *customer* untuk melakukan validasi sertifikat, jika *customer* tidak memersetujui sertifikatnya, maka staff administrasi melakukan revisi pada sertifikat, jika *customer* memersetujui sertifikat, maka staff administrasi akan mencetak sertifikat tersebut dan mengirim kepada *customer* dan staff keuangan, setelah staff keuangan menerima sertifikat fumigasi, akan membuat *invoice* pembayaran yang akan dikirim kepada *customer* untuk melakukan proses pembayaran, dan setelah *customer* membayar *invoice* tersebut, maka akan masuk kedalam laporan pesanan dan diterima oleh manager. Untuk diagram BPMN yang menggunakan aplikasi dapat dilihat gambar 3.3.



Gambar 3.3. Diagram BPMN Menggunakan Aplikasi

## B. Identifikasi Permasalahan

Pembuatan aplikasi pengendalian persediaan menggunakan metode ROP dan EOQ pada PT Prana Argentum Corporation dimulai dengan melakukan identifikasi masalah yang bertujuan untuk menyimpulkan masalah yang dihadapi dan solusi yang akan diberikan. Untuk mencapai tujuan ini, tahap-tahap yang harus dilalui meliputi melakukan wawancara dengan pihak terkait. Melalui wawancara

tersebut, didapatkan pemahaman tentang permasalahan yang dihadapi oleh perusahaan, sehingga memungkinkan untuk menemukan alternatif solusi yang dapat mengatasi masalah yang ada dalam perusahaan.

Tabel 3.1. Identifikasi Masalah.

<b>Identifikasi Masalah</b>	
<b>Masalah Yang Ditemukan</b>	<b>Dampak</b>
15% pesanan <i>customer</i> tertunda yang disebabkan dari kurangnya persediaan untuk melaksanakan proses fumigasi dikarenakan kurangnya monitoring persediaan yang diperlukan untuk melaksanakan proses fumigasi.	Terjadinya <i>out of stock</i> pada persediaan yang dibutuhkan untuk melaksanakan proses fumigasi dan banyaknya penundaan pesanan proses fumigasi, lalu jika melakukan proses fumigasi tanpa mematuhi <i>safety measure</i> dapat melanggar <i>safety measure</i> dari pemerintah
<b>Solusi</b>	
Membuat aplikasi pengendalian persediaan menggunakan metode ROP dan EOQ yang bertujuan untuk memberitahukan pemesanan persediaan yang tepat untuk menghindari <i>out of stock</i> dan jumlah persediaan yang dibeli.	

### C. Identifikasi Pengguna

Pada fase ini, peneliti mengidentifikasi pengguna dan kebutuhan data berdasarkan observasi, wawancara, dan hasil spesifik masalah untuk menentukan pengguna dan data pada sistem, untuk pengguna aplikasi, terdapat fumigator, staff administrasi, dan Manager.

### D. Identifikasi Kebutuhan Fungsional

Setelah tahap sebelumnya, maka selanjutnya adanya identifikasi kebutuhan fungsional, fungsi sistem informasi yang akan dibuat sebagai berikut:

1. Fungsi Pengelolaan Data
  - A. Fungsi Mengelola Data Penggunaan Persediaan.
  - B. Fungsi Mengelola Data Pesanan.
  - C. Fungsi Mengelola Data Pemesanan Persediaan
2. Fungsi Pengelolaan Informasi
  - A. Informasi Rekap Persediaan.
  - B. Informasi Rekap Pemasukan Persediaan.
  - C. Informasi Rekap Penggunaan Persediaan.
  - D. Informasi Rekap Pemesanan Persediaan
  - E. Informasi Rekap Pesanan.
  - F. Informasi Reorder Point.
  - G. Informasi Economic Order Quantity.

### E. Identifikasi kebutuhan Data

Setelah melakukan proses identifikasi masalah dan pengguna, langkah selanjutnya adalah identifikasi kebutuhan data. Untuk aplikasi ini, terdapat beberapa jenis data yang diperlukan, antara lain data persediaan yang terdiri dari beberapa jenis alat seperti *Gas Methyl Bromide (CH<sub>3</sub>Br)*, Gas Oksigen, *Cartridge Gas Mask*, Plastik Fumigasi, *Sand Snake*, dan Kertas Sertifikat Fumigasi. Selain itu, data-data lain yang diperlukan mencakup data permintaan, data pemasukan, data pesanan, dan data yang diperlukan untuk menjalankan metode ROP dan EOQ. Siklus pengumpulan data dilakukan dalam periode 1 periode = 3 bulan.

### F. Analisis Kebutuhan Pengguna

Berdasarkan hasil wawancara dan identifikasi pengguna, dapat menghasilkan kebutuhan pengguna yang akan dianalisis agar dapat mengetahui kebutuhan dari setiap pengguna yang ada, berikut merupakan peran dan tanggung jawab pengguna yang dapat di lihat pada tabel 3.2.

Tabel 3.2. Peran dan Tanggung jawab pengguna.

Pengguna	Peran	Kebutuhan Data	Kebutuhan Informasi	Kebutuhan Dokumen
Fumigator.	Mengelola pemesanan dan penggunaan persediaan	<ul style="list-style-type: none"> <li>•Data Persediaan.</li> <li>•Data pemasok.</li> <li>•Data permintaan persediaan.</li> <li>•Data Pemesanan Persediaan.</li> <li>•Data pesanan.</li> </ul>	<ul style="list-style-type: none"> <li>•Titik ROP untuk setiap periode.</li> <li>•Jumlah EOQ untuk setiap periode.</li> <li>•Pesanan yang diterima .</li> <li>•Pesanan yang tertundah.</li> <li>•Detail Pesanan.</li> </ul>	
Staff Administrasi.	Mengelola Pesanan yang masuk	<ul style="list-style-type: none"> <li>•Data pesanan yang diterima.</li> <li>•Data pesanan yang tertundah.</li> <li>•Data detail pesanan.</li> </ul>	<ul style="list-style-type: none"> <li>•Jumlah pesanan yang diterima</li> <li>•Jumlah pesanan yang tertundah</li> <li>•Konfirmasi pesanan selesai</li> </ul>	
Manager.	Melihat Laporan .	<ul style="list-style-type: none"> <li>•Data Persediaan.</li> <li>•Data Pemasukan Persediaan.</li> <li>•Data permintaan persediaan.</li> <li>•Data pesanan.</li> </ul>	<ul style="list-style-type: none"> <li>•Rekap Data Persediaan.</li> <li>•Rekap Data Pemasukan Persediaan.</li> <li>•Rekap Data permintaan Persediaan.</li> <li>•Rekap Data Pesanan.</li> </ul>	<ul style="list-style-type: none"> <li>•Laporan Persediaan.</li> <li>•Laporan Penggunaan Persediaan.</li> <li>•Laporan Pemasukan Persediaan.</li> <li>•Laporan Pesanan.</li> <li>•Laporan Pemesanan Persediaan</li> </ul>

### G. Analisis Kebutuhan Fungsional

Pada fase identifikasi kebutuhan fungsional, dilakukan pengumpulan rangkaian kebutuhan yang diperlukan untuk memenuhi solusi yang telah diidentifikasi pada tahap identifikasi masalah. Tabel 3.3 merupakan tabel identifikasi yang memuat daftar kegiatan yang akan dilakukan oleh pengguna.

Tabel 3.3. Identifikasi Kebutuhan Fungsional.

No	Pengguna	Kebutuhan Fungsional
1.	Fumigator	<ul style="list-style-type: none"> <li>• Mengecek Data Persediaan.</li> <li>• Mengelola Data Penggunaan Persediaan.</li> <li>• Mengelola Data Pemesanan Persediaan</li> <li>• Mengecek Data Pemasukan Persediaan.</li> <li>• Mengecek Data Pesanan.</li> <li>• Perhitungan EOQ</li> <li>• Perhitungan ROP</li> <li>• Mendapatkan Biaya Pemesanan</li> <li>• Perhitungan Safety Stock</li> <li>• Mendapatkan Lead time</li> <li>• Mendapatkan Biaya Penyimpanan</li> <li>• Perhitungan Rata Rata Permintaan</li> <li>• Perhitungan Jumlah Permintaan Per Periode</li> <li>• Mendapatkan Permintaan Max</li> <li>• Perhitungan Rata Rata Permintaan Per Satuan Waktu</li> </ul>
2	Staff Administrasi	<ul style="list-style-type: none"> <li>• Melihat Dashboard</li> <li>• Mengelola Data Pesanan.</li> <li>• Melihat Dashboard</li> <li>• Mencetak Laporan Pesanan.</li> <li>• Mencetak Laporan Persediaan.</li> <li>• Mencetak Laporan Penggunaan Persediaan.</li> </ul>
3	Manager	<ul style="list-style-type: none"> <li>• Mencetak Laporan Pemasukan Persediaan.</li> <li>• Mencetak Laporan pemesanan Persediaan</li> <li>• Melihat Dashboard</li> </ul>

### H. Analisis Kebutuhan Non Fungsional

Pada fase identifikasi kebutuhan non-fungsional, terdapat sejumlah kebutuhan yang tidak terkait langsung dengan fungsionalitas aplikasi, namun memiliki peran penting dalam memastikan performa dan keamanan aplikasi. Hal ini bertujuan untuk menciptakan pengalaman pengguna yang nyaman saat menggunakan aplikasi tersebut. Tabel 3.4 memuat daftar identifikasi kebutuhan non-fungsional yang telah diidentifikasi.

Tabel 3.4. Identifikasi Non-Fungsional.

No	Non-Fungsional	Kebutuhan
1.	<i>Performance.</i>	<ul style="list-style-type: none"> <li>• Setiap proses data tidak boleh lebih dari 3 detik.</li> <li>• Sistem harus <i>real time.</i></li> </ul>
2.	<i>Security.</i>	<ul style="list-style-type: none"> <li>• Akses sistem diperlukan login terlebih dahulu.</li> </ul>

### I. Analisis Kebutuhan *Hardware, Software, dan Jaringan*

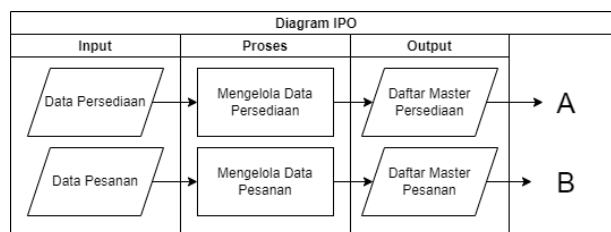
Perangkat lunak yang dibutuhkan untuk menjalankan aplikasi pengendalian persediaan menggunakan metode ROP dan EOQ pada PT Prana Argentum Corporation, yaitu:

Tabel 3.5. Identifikasi Perangkat.

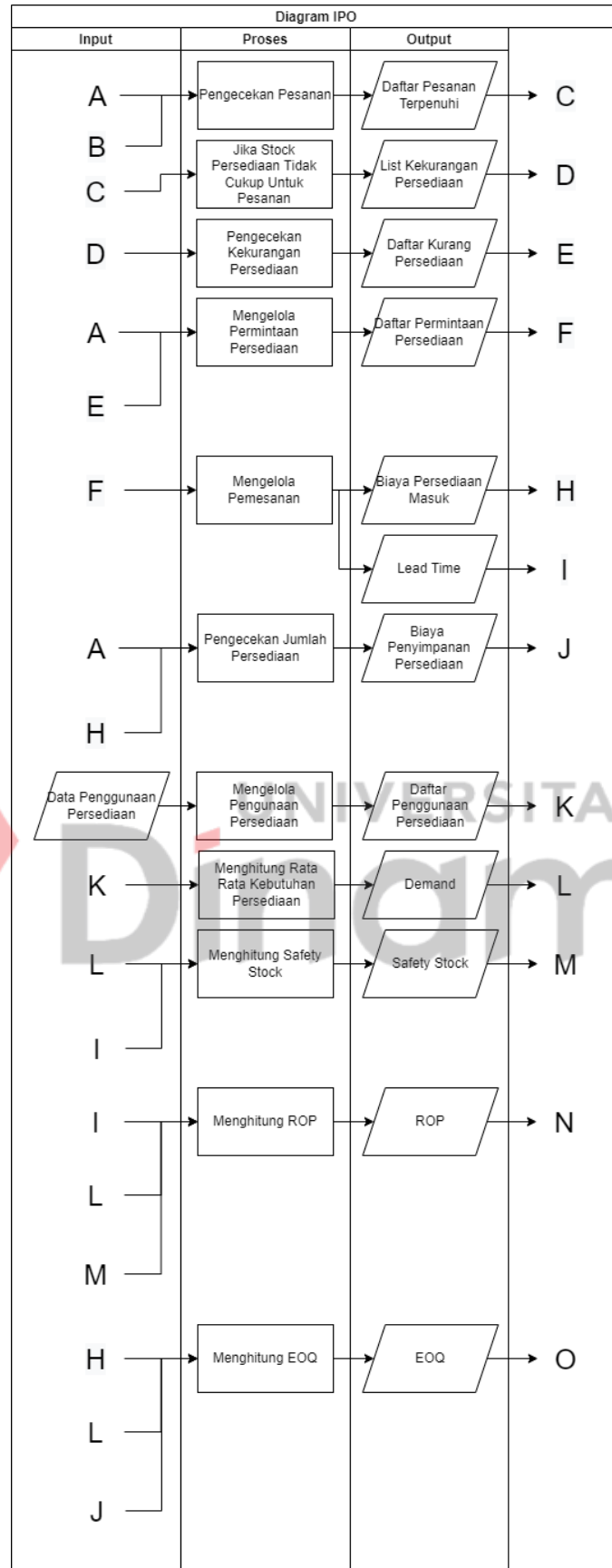
Jenis Kebutuhan	Keterangan
Kebutuhan Perangkat Lunak.	<p>Kebutuhan optimal untuk perangkat lunak yang dibutuhkan server antara lain:</p> <ul style="list-style-type: none"> <li>• <i>Domain Hosting.</i></li> <li>• <i>Visual Studio Code.</i></li> <li>• <i>Google Chrome.</i></li> <li>• <i>XAMPP.</i></li> </ul>
Kebutuhan Perangkat Keras.	<p>Kebutuhan optimal perangkat keras yang dibutuhkan oleh server antara lain:</p> <ul style="list-style-type: none"> <li>• RAM minimal 4 GB.</li> <li>• <i>Harddisk</i> minimal 256 GB.</li> <li>• <i>Processor</i> minimal core i3.</li> <li>• Jaringan Internet.</li> </ul>
Kebutuhan Jaringan.	<p>Kebutuhan jaringan yang dibutuhkan dapat menggunakan LAN(<i>Local Area Network</i>).</p>

### 3.1.5 Diagram *Input, Proses dan Output (IPO)*

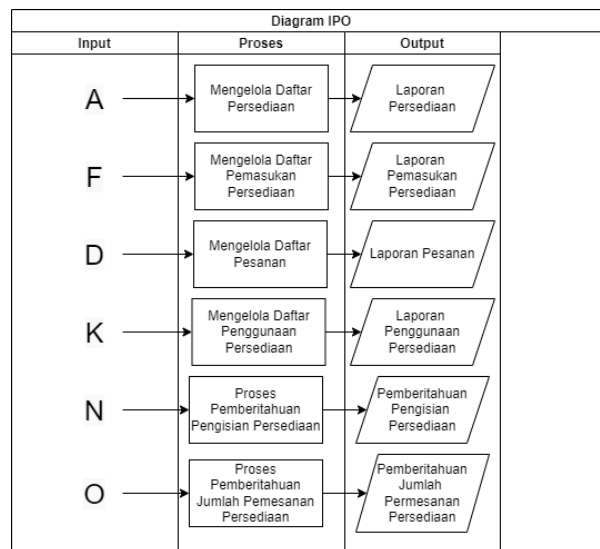
Pada diagram ini, berisikan awal *input* alur data dari *proses* yang akan dijalankan lalu menghasilkan *output* yang ditampilkan ke *admin*, berikut merupakan IPO untuk penelitian.



Gambar 3.4. Diagram IPO Tahap Awal.



Gambar 3.5. Diagram IPO Tahap Pelaksanaan.



Gambar 3.6. Diagram IPO Tahap Akhir.

### Input

- Data Persediaan = Data Persediaan berisikan nama persediaan dan jumlah persediaan.
- Data Pesanan = Data pesanan berisikan nama perusahaan yang memesan, jumlah *container* yang akan melalui fumigator, tanggal diterimanya pesanan, tanggal deadline pesanan, dan status pesanan.
- Data Penggunaan Persediaan = Data penggunaan persediaan berisikan salah satu persediaan yang ada pada data persediaan, jumlah penggunaan persediaan tersebut, dan tanggal penggunaan persediaan tersebut.

### Proses

- Mengelola Data Persediaan = proses ini mengelola data persediaan yang dimana untuk mengetahui informasi jumlah persediaan.
- Mengelola Data Pesanan = proses ini mengelola data pesanan yang dimana untuk mengetahui informasi jumlah persediaan.
- Pengecekan Pesanan = proses ini untuk mengecek data pesanan sudah tepat.
- Pengecekan Kekurangan Persediaan = proses ini mengecek pada persediaan apakah persediaan memiliki kekurangan *stock* atau tidak.
- Mengelola Permintaan Persediaan = proses ini mengelola data permintaan persediaan yang akan dipesan kepada *supplier*.
- Mengelola Persediaan Masuk = proses ini mengelola data persediaan yang telah sampai di depo dari pemesanan *supplier*.
- Pengecekan Jumlah Persediaan = proses ini mengecek jumlah persediaan setelah persediaan masuk di depo.
- Mengelola Data Penggunaan Persediaan = proses ini mengelola data penggunaan persediaan yang digunakan saat fumigasi.
- Menghitung Rata Rata Kebutuhan Persediaan = proses ini menghitung rata-rata kebutuhan dalam 1 periode.
- Menghitung *Safety Stock* = proses ini menghitung jumlah cadangan persediaan.
- Menghitung ROP = proses ini menghitung penentuan kapan persediaan harus dipesan.



- l. Menghitung EOQ = proses ini menghitung penentuan berapa banyak persediaan yang perlu dipesan.
- m. Mengelola Data Pesanan = proses ini adalah mengelola laporan pesanan apa saja yang masuk dalam 1 periode.
- n. Mengelola Daftar Persediaan = proses ini adalah mengelola laporan perubahan *stock* persediaan dalam 1 periode.
- o. Mengelola Daftar Pemasukan Persediaan = proses ini adalah mengelola laporan pemasukan persediaan yang berisikan persediaan apa saja yang dipesan dan masuk kepada *stock* persediaan dalam 1 periode.
- p. Mengelola Daftar Penggunaan Persediaan = proses ini adalah laporan penggunaan persediaan yang berisikan persediaan apa saja yang telah digunakan saat fumigasi sehingga mengurangi *stock* persediaan dalam 1 periode.
- q. Proses Pemberitahuan Pengisian Persediaan = proses ini memberitahukan kepada fumigator kapan melakukan pengisian persediaan.
- r. Proses Pemberitahuan Jumlah Pemesanan Persediaan = proses ini memberitahukan kepada fumigator berapa banyak persediaan yang perlu dipesan.

### Output

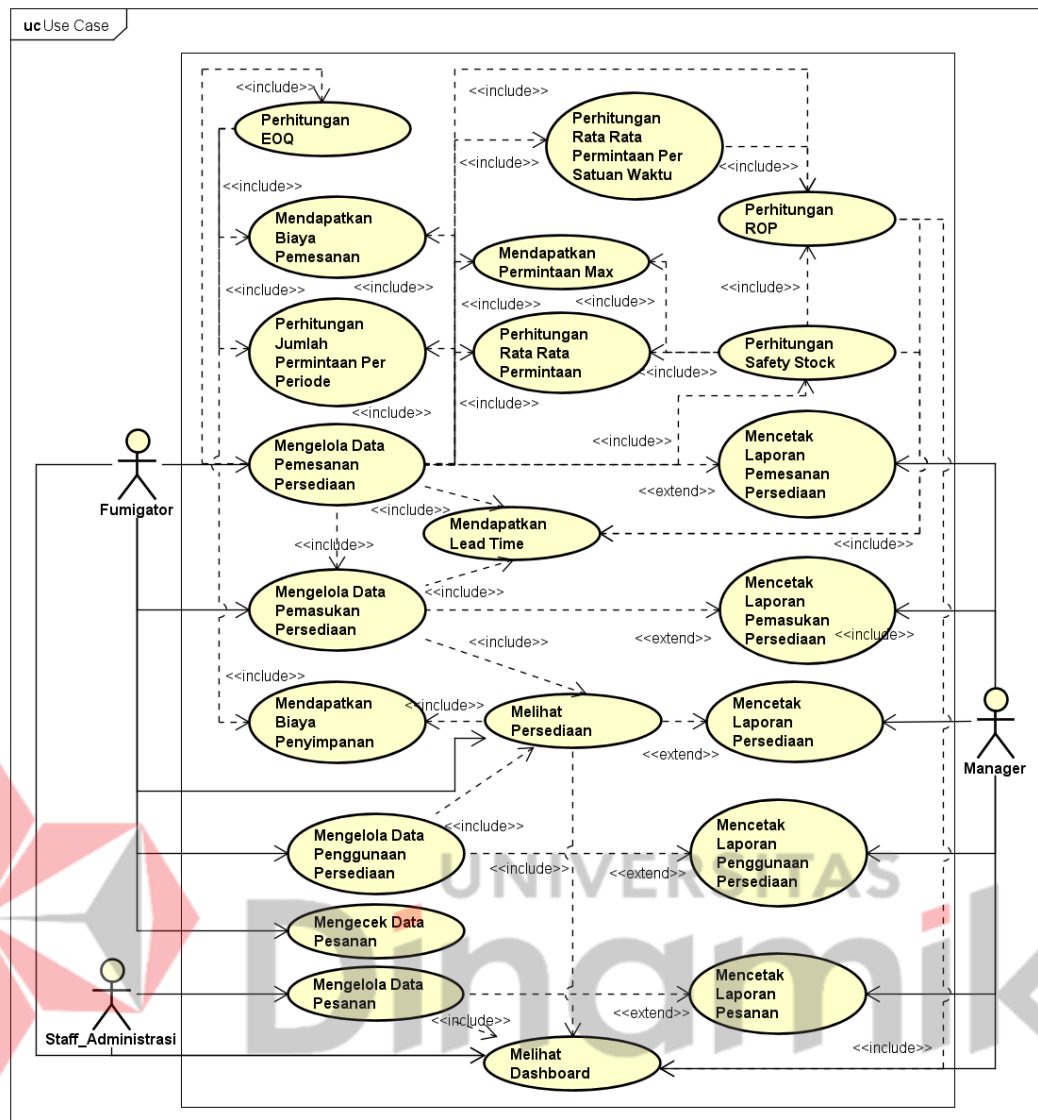
- a. Daftar Master Persediaan = Output ini luaran dari proses data persediaan. Output ini berfungsi sebagai detail informasi persediaan yang masih tersedia.
- b. Daftar Master Pesanan = Output ini luaran dari proses data pesanan. Output ini berfungsi sebagai detail informasi pesanan.
- c. Daftar Pesanan Terpenuhi = Output ini luaran dari proses pengecekan pesanan, Output ini berfungsi untuk mengetahui pesanan yang telah terpenuhi.
- d. Daftar Pesanan Tertunda = Output ini luaran dari proses pengecekan pesanan, Output ini berfungsi untuk mengetahui pesanan yang tertunda.
- e. Daftar Kurang Persediaan = Output ini luaran dari proses pengecekan kekurangan persediaan, output ini berfungsi sebagai detail informasi terkait dengan jumlah kekurangan pada setiap persediaan.
- f. Daftar Pemasukan Persediaan = Output ini luaran dari proses mengelola permintaan persediaan, output ini berfungsi sebagai detail informasi terkait pemasukan persediaan.
- g. Biaya Persediaan Masuk = Output ini luaran dari proses mengelola persediaan masuk, output ini berfungsi sebagai detail informasi pengeluaran yang digunakan untuk memesan persediaan
- h. Lead Time = Output ini luaran dari proses mengelola persediaan masuk, output ini berfungsi sebagai lama pengiriman persediaan dari awal memesan sampai persediaan sampai ke depo.
- i. Biaya Penyimpanan Persediaan = Output ini luaran dari proses pengecekan jumlah persediaan, output ini berfungsi sebagai detail informasi mengenai biaya yang dikeluarkan untuk penyimpanan persediaan.
- j. Daftar Penggunaan Persediaan = Output ini luaran dari proses pengecekan jumlah persediaan, output ini berfungsi sebagai detail informasi mengenai persediaan yang digunakan saat menjalankan fumigasi.
- k. Demand = Output ini luaran dari proses menghitung rata – rata kebutuhan persediaan, output ini berfungsi sebagai salah satu variable yang digunakan untuk menghitung *safety stock*, ROP, dan EOQ.

- l. *Safety Stock* = *Output* ini luaran dari proses menghitung *safety stock*, *output* ini berfungsi sebagai salah satu variable yang digunakan untuk menghitung ROP.
- m. ROP = *Output* ini luaran dari proses menghitung ROP, *output* ini berfungsi untuk memberi peringatan ketika kapan harus mengisi persediaan yang kurang.
- n. EOQ = *Output* ini luaran dari proses menghitung EOQ, *output* ini berfungsi untuk memberi jumlah persediaan yang perlu dipesan untuk menghindari *overstock*.
- o. Laporan Persediaan = *Output* ini luaran dari proses mengelola daftar persediaan, *output* ini berfungsi sebagai laporan yang menunjukkan perubahan *stock* persediaan dalam 1 periode.
- p. Laporan Pemasukan Persediaan = *Output* ini luaran dari proses mengelola daftar pemasukan persediaan, *output* ini berfungsi sebagai laporan persediaan apa saja yang masuk pada 1 periode.
- q. Laporan Pesanan = *Output* ini luaran dari proses mengelola daftar pesanan, *output* ini berfungsi sebagai laporan yang menunjukkan pesanan apa saja yang masuk pada 1 periode.
- r. Laporan Penggunaan Persediaan = *Output* ini luaran dari proses mengelola daftar penggunaan persediaan, *output* ini berfungsi sebagai laporan persediaan apa saja yang digunakan saat fumigasi pada 1 periode.
- s. Pemberitahuan pengisian Persediaan = *Output* ini luaran dari proses pemberitahuan pengisian persediaan, *output* ini berfungsi untuk memberitahu kepada fumigator untuk kapan untuk mengisi persediaan yang kurang.
- t. Pemberitahuan Jumlah Pemesanan Persediaan = *Output* ini luaran dari proses pemberitahuan jumlah pemesanan persediaan, *output* ini berfungsi untuk memberitahu kepada fumigator banyak jumlah persediaan yang perlu dipesan untuk menghindari *overstock* persediaan.

## 3.2 Design

### 3.2.1 Use Case Diagram

Pada pembuatan *use case diagram* menggunakan dari hasil identifikasi kebutuhan sistem yang dihasilkan di tahap sebelumnya, diagram ini berisikan interaksi apa saja yang bisa dilakukan oleh admin pada sistem. Pada *use case diagram* ini terdapat 3 aktor, yang diperankan oleh fumigator, staff administrasi dan manager, aktivitas yang dimiliki oleh fumigator yaitu, mengecek data persediaan, mengelola data penggunaan persediaan, mengecek data pesanan, mengecek data pemasukan persediaan, dan mengelola pemesanan persediaan. Lalu aktivitas yang dimiliki oleh staff administrasi yaitu, mengelola data pesanan, dan yang terakhir manager yang memiliki kegiatan untuk mencetak laporan persediaan, laporan pemasukan persediaan, laporan penggunaan persediaan, laporan pemesanan persediaan, dan laporan pesanan, *use case diagram* dapat dilihat pada gambar 3.7.



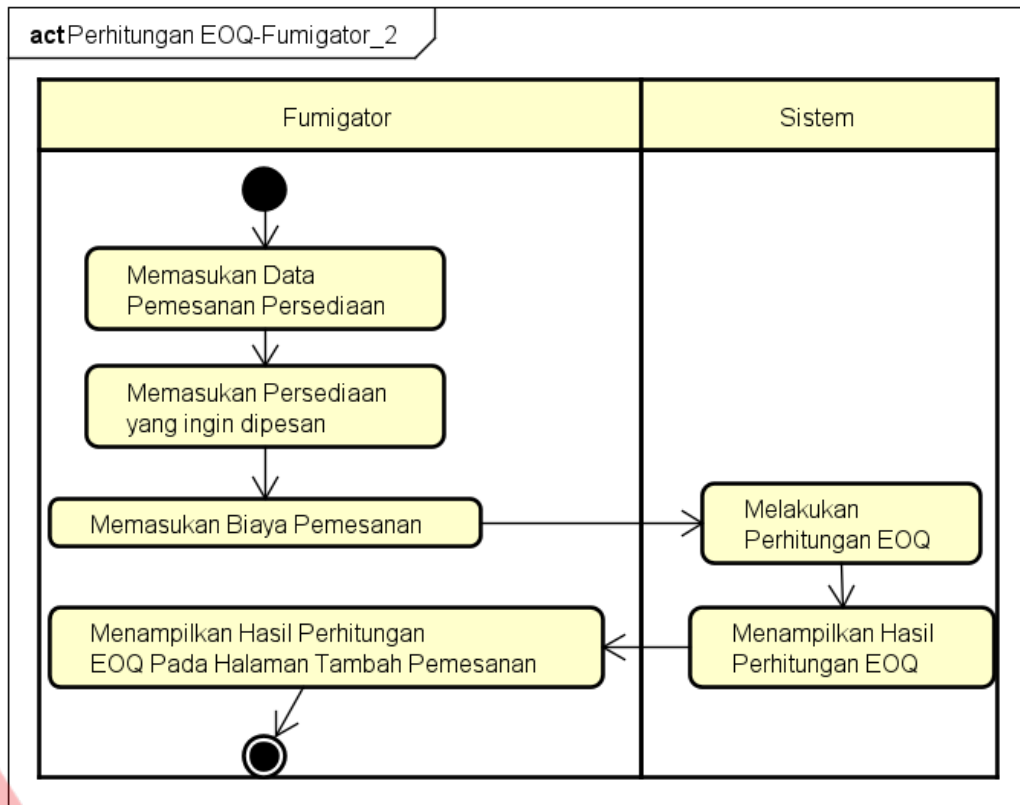
Gambar 3.7. Use Case Diagram.

### 3.2.2 Activity Diagram

Pada pembuatan *activity diagram* berisikan interaksi dari admin ke sistem dari *use case diagram*, di *activity diagram* memperlihatkan alur jalan sistem dari admin saat berinteraksi dengan sistem.

#### A. Perhitungan EOQ

Pada *activity diagram* Perhitungan EOQ dilakukan oleh fumigator, dalam aktivitas ini fumigator dapat melihat hasil perhitungan EOQ, yang dapat dilihat pada gambar 3.8.



Gambar 3.8. Activity Diagram Perhitungan EOQ.

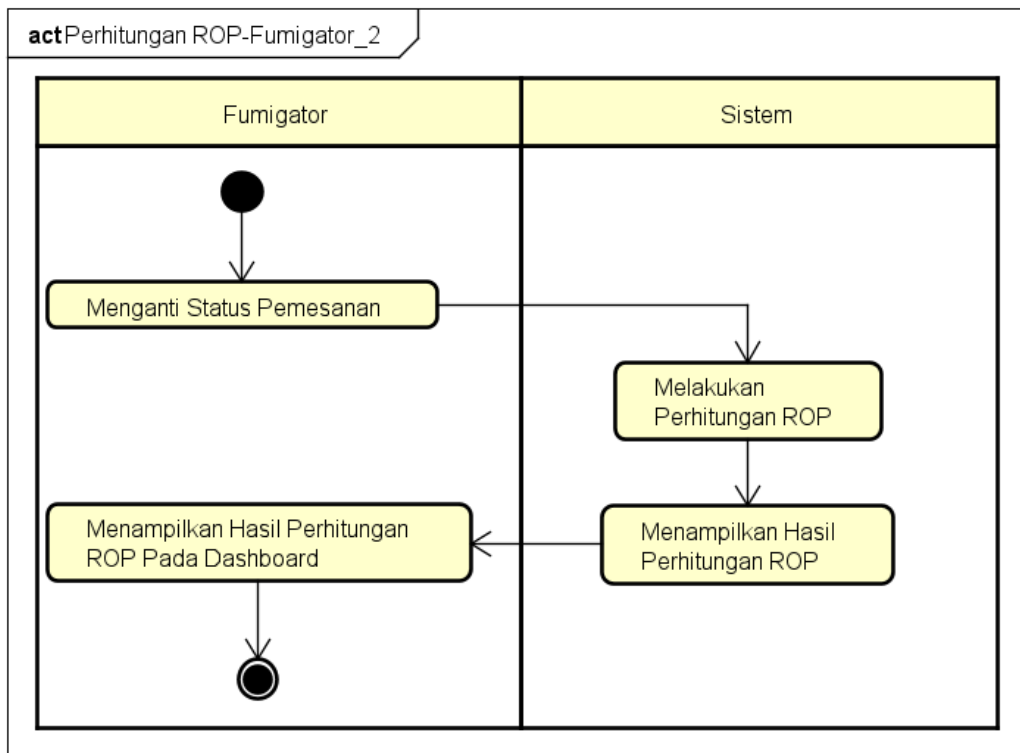
Pada gambar 3.8 menjelaskan proses perhitungan EOQ, fumigator dapat Perhitungan EOQ, yang dapat dilihat pada tabel 3.6.

Tabel 3.6. Perhitungan EOQ.

Nama Use Case	Perhitungan EOQ	
Deskripsi Singkat	Fumigator Mendapatkan Hasil Perhitungan EOQ.	
Alur Normal	Aktor	Sistem
	Menghitung EOQ	
	Ketika aktor menambah data pada pemesanan persediaan, maka perhitungan EOQ akan dimulai, untuk hasil perhitungan EOQ itu sendiri dapat dilihat pada halaman tambah pemesanan.	Ketika sistem menerima data pemesanan persediaan baru, akan mulai proses perhitungan EOQ, setelah selesainya proses perhitungan EOQ, akan ditunjukkan kepada aktor di halaman tambah pemesanan.
Kondisi Awal	Fumigator menambah data pemesanan persediaan.	
Kondisi Akhir	Fumigator dapat melihat hasil perhitungan EOQ.	

### B. Perhitungan ROP

Pada activity diagram Perhitngan ROP dilakukan oleh fumigator, dalam aktivitas ini fumigator dapat melihat hasil perhitungan ROP, yang dapat dilihat pada gambar 3.9.



Gambar 3.9. Activity Diagram Perhitungan ROP.

Pada gambar 3.9 menjelaskan proses perhitungan ROP, fumigator dapat Perhitungan ROP, yang dapat dilihat pada tabel 3.7.

Tabel 3.7. Perhitungan ROP.

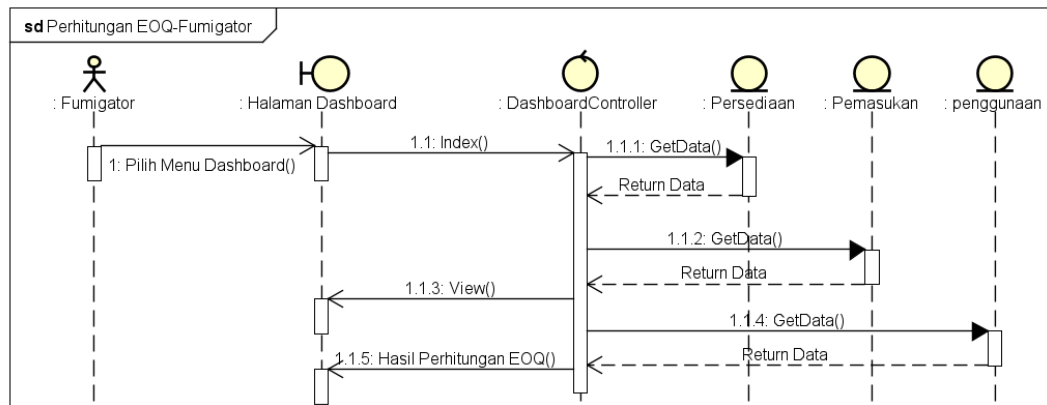
Nama Use Case	Perhitungan ROP	
Deskripsi Singkat	Fumigator Mendapatkan Hasil Perhitungan ROP.	
Alur Normal	Aktor	Sistem
	Menghitung ROP	
	Ketika aktor mengganti status pemesanan, maka perhitungan ROP akan dimulai, untuk hasil perhitungan ROP itu sendiri dapat dilihat pada halaman dashboard dan halaman persediaan.	Ketika sistem menerima data pemasukan persediaan baru, akan mulai proses perhitungan ROP, setelah selesainya proses perhitungan ROP, akan ditunjukan kepada aktor di halaman dashboard dan halaman persediaan.
Kondisi Awal	Fumigator mengganti status pemesanan persediaan.	
Kondisi Akhir	Fumigator dapat melihat hasil perhitungan ROP.	

### 3.2.3 Sequence Diagram

Pada pembuatan *sequence diagram* berisikan interaksi dari *use case diagram*, di *sequence diagram* memperlihatkan alur jalan interaksi objek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan terturun.

### A. Perhitungan EOQ

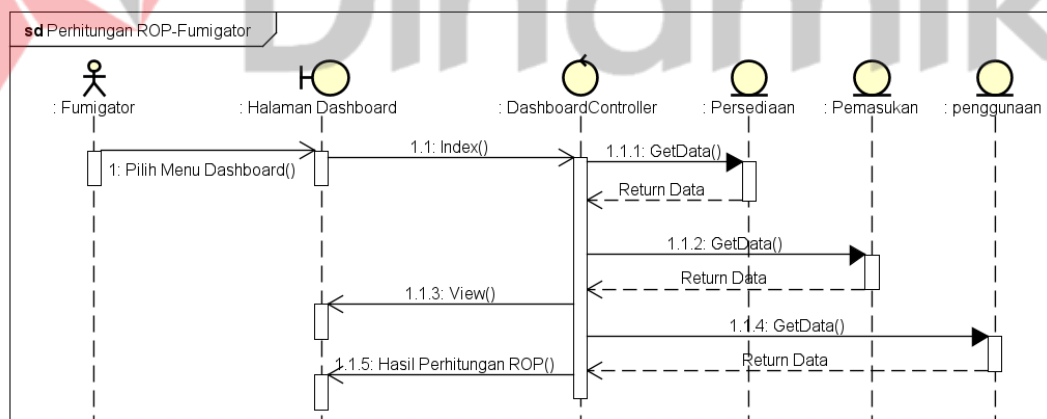
*Sequence diagram* perhitungan EOQ menjelaskan detail dari proses pada *activity diagram* perhitungan EOQ. detail alur yang ada pada proses perhitungan EOQ digambarkan dalam gambar 3.10.



Gambar 3.10. *Sequence Diagram* Perhitungan EOQ.

### B. Perhitungan ROP

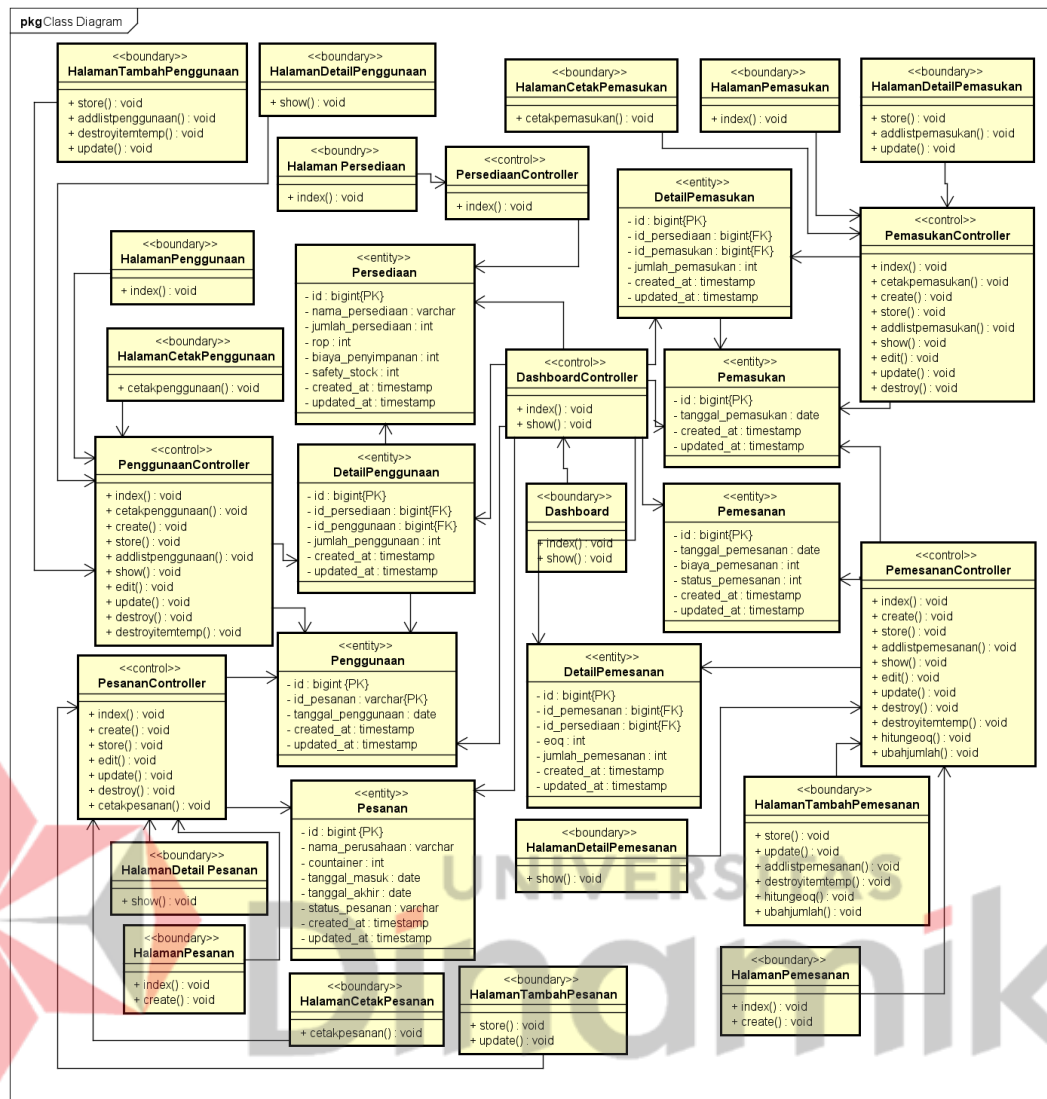
*Sequence diagram* perhitungan ROP menjelaskan detail dari proses pada *activity diagram* perhitungan ROP. detail alur yang ada pada proses perhitungan ROP digambarkan dalam gambar 3.11.



Gambar 3.11. *Sequence Diagram* Perhitungan ROP.

### 3.2.4 Class Diagram

Pada pembuatan *class diagram* bertujuan untuk membuat visualisasi struktur kelas dari sebuah sistem yang dapat memperlihatkan semua kebutuhan yang diperlukan sistem, dapat dilihat pada gambar 3.12.



Gambar 3.12. Class Diagram.

### 3.2.5 Desain Antarmuka

Setelah menyelesaikan proses perancangan keseluruhan sistem, tahap selanjutnya adalah pembuatan *desain* antarmuka. *Desain* antarmuka ini bertujuan untuk memberikan panduan saat melakukan *coding*, yang mana panduan ini akan digunakan sebagai acuan dalam merancang aplikasi pada tahap berikutnya. Hasil dari desain antarmuka dapat ditemukan pada lampiran 4.

### 3.3 Coding

Pada tahap ini, pengembangan sistem akan menggunakan *framework* Laravel dan MySQL sebagai basis data untuk sistem. Selama proses



pengembangan, rancangan sistem dan desain antarmuka akan digunakan sebagai acuan utama dalam mengimplementasikan fungsionalitas dan tampilan aplikasi.

### 3.4 *Testing*

Pada tahap ini, dilakukan proses pengujian aplikasi yang telah dikembangkan pada tahap sebelumnya. Pengujian dilakukan dengan menggunakan metode *whitebox testing* dengan tujuan untuk memverifikasi apakah aplikasi berjalan sesuai dengan rancangan sistem yang telah dibuat pada tahap *desain*..

### 3.5 *Release*

Pada tahap terakhir, dilakukan proses *hosting* website dengan tujuan agar aplikasi dapat segera digunakan di PT Prana Argumen. Tahap ini bertujuan untuk memfasilitasi penggunaan aplikasi oleh staff PT Prana Argumen dan membantu pengendalian persediaan perusahaan.



UNIVERSITAS  
**Dinamika**



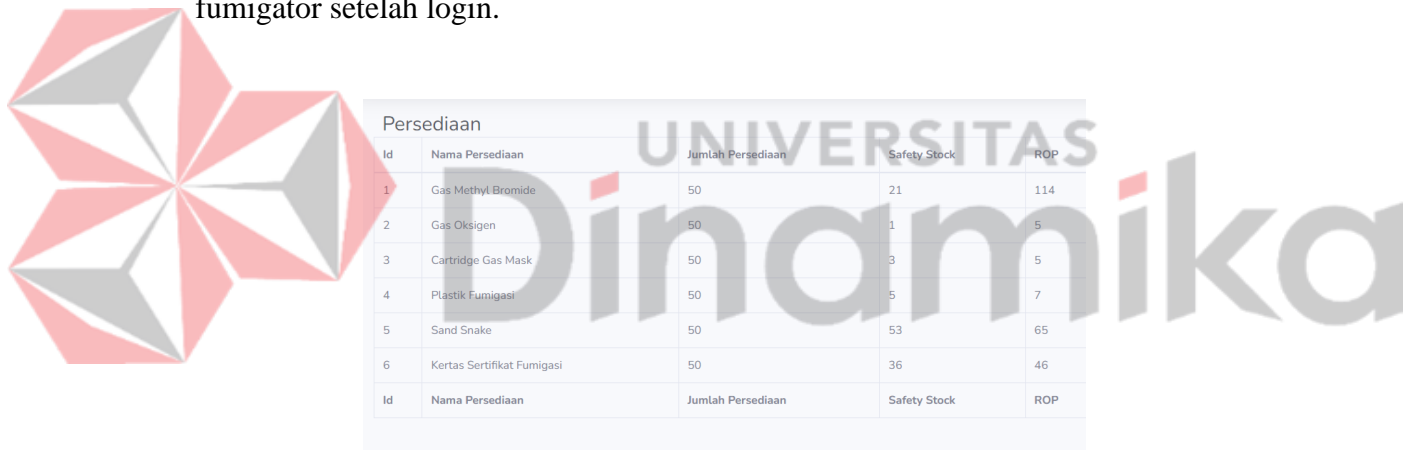
## BAB IV HASIL DAN PEMBAHASAN

### 4.1 Coding

#### 4.1.1 Implementasi Aplikasi

##### A. Perhitungan ROP

Pada hasil evaluasi perhitungan ROP menunjukkan hasil perhitungan Berada pada Halaman persediaan, perhitungan ROP dihitung Ketika status pemesanan pada halaman pemesanan dari “Dari Pengiriman” menjadi “Persediaan Telah Sampai”, Ketika status pemesanan “Persediaan Telah Sampai” sistem akan mulai menghitung ROP menggunakan data penggunaan bulan sebelumnya, Pada gambar 4.1 adalah halaman melihat perhitungan ROP yang dapat diakses oleh fumigator setelah login.



Persediaan				
Id	Nama Persediaan	Jumlah Persediaan	Safety Stock	ROP
1	Gas Methyl Bromide	50	21	114
2	Gas Oksigen	50	1	5
3	Cartridge Gas Mask	50	3	5
4	Plastik Fumigasi	50	5	7
5	Sand Snake	50	53	65
6	Kertas Sertifikat Fumigasi	50	36	46
Id	Nama Persediaan	Jumlah Persediaan	Safety Stock	ROP

Gambar 4.1 Implementasi Perhitungan ROP.

##### B. Perhitungan EOQ

Pada hasil evaluasi perhitungan EOQ menunjukkan hasil perhitungan EOQ Ketika user melakukan pemesanan persediaan, jumlah pemesanan sama dengan EOQ dikarena fungsi utama EOQ untuk menghitung berapa banyak persediaan yang perlu dibeli, untuk tombol hapus digunakan untuk menghapus persediaan yang ingin dibeli, dan untuk tombol edit jumlah digunakan Ketika persediaan yang ingin dipesan oleh user, tidak sama dengan hasil perhitungan EOQ Pada gambar 4.2 adalah halaman perhitungan EOQ dapat diakses oleh fumigator setelah login.

List Penggunaan Persediaan Simpan

Show  entries Search:

No	Nama Persediaan	EOQ	Jumlah Pemesanan	Aksi
1	Gas Methyl Bromide	528	528	<span style="color: red;">Hapus</span> <span style="color: teal;">Edit Jumlah</span>
2	Gas Oksigen	112	112	<span style="color: red;">Hapus</span> <span style="color: teal;">Edit Jumlah</span>
3	Cartridge Gas Mask	82	82	<span style="color: red;">Hapus</span> <span style="color: teal;">Edit Jumlah</span>
4	Plastik Fumigasi	93	93	<span style="color: red;">Hapus</span> <span style="color: teal;">Edit Jumlah</span>
5	Sand Snake	202	202	<span style="color: red;">Hapus</span> <span style="color: teal;">Edit Jumlah</span>
6	Kertas Sertifikat Fumigasi	187	187	<span style="color: red;">Hapus</span> <span style="color: teal;">Edit Jumlah</span>

Gambar 4.2 Implementasi Perhitungan EOQ.

### C. Dashboard

Pada halaman *Dashboard* aplikasi ini, akan ditampilkan beberapa informasi penting, seperti jumlah pesanan yang masuk, jumlah pesanan yang sedang dalam proses pengerjaan, jumlah *container* yang telah selesai melalui proses fumigasi, dan jumlah *container* yang saat ini sedang dalam proses fumigasi. Selain itu, terdapat juga tabel indikator persediaan untuk setiap jenis persediaan, yang ditandai dengan tiga warna berbeda. Warna hijau akan muncul jika persediaan melebihi ROP dan *safety stock*. Warna kuning akan muncul jika persediaan kurang dari ROP namun masih melebihi *safety stock*, dan warna merah akan muncul jika persediaan kurang dari ROP dan *safety stock*. Indikator warna kuning dan merah dapat di-klik untuk membuka halaman tambah pemesanan. Untuk melihat tampilan dashboard secara detail, dapat dilihat pada gambar 4.3.

Dashboard

JUMLAH PESANAN MASUK	JUMLAH PESANAN DIALANKAN	JUMLAH CONTAINER SELESAI	JUMLAH CONTAINER DIALANKAN
32	0	387	0

Id	Nama Persediaan	Stock Persediaan	ROP	Safety Stock	Indikator
1	Gas Methyl Bromide	500	87	27	<span style="color: green;">Aman</span>
2	Gas Oksigen	50	6	3	<span style="color: green;">Aman</span>
3	Cartridge Gas Mask	50	9	9	<span style="color: green;">Aman</span>
4	Plastik Fumigasi	80	30	27	<span style="color: green;">Aman</span>
5	Sand Snake	110	117	105	<span style="color: orange;">Segera Memesan</span>
6	Kertas Sertifikat Fumigasi	60	66	60	<span style="color: red;">Lakukan Pemesanan</span>
Id	Nama Persediaan	Stock Persediaan	ROP	Safety Stock	Indikator

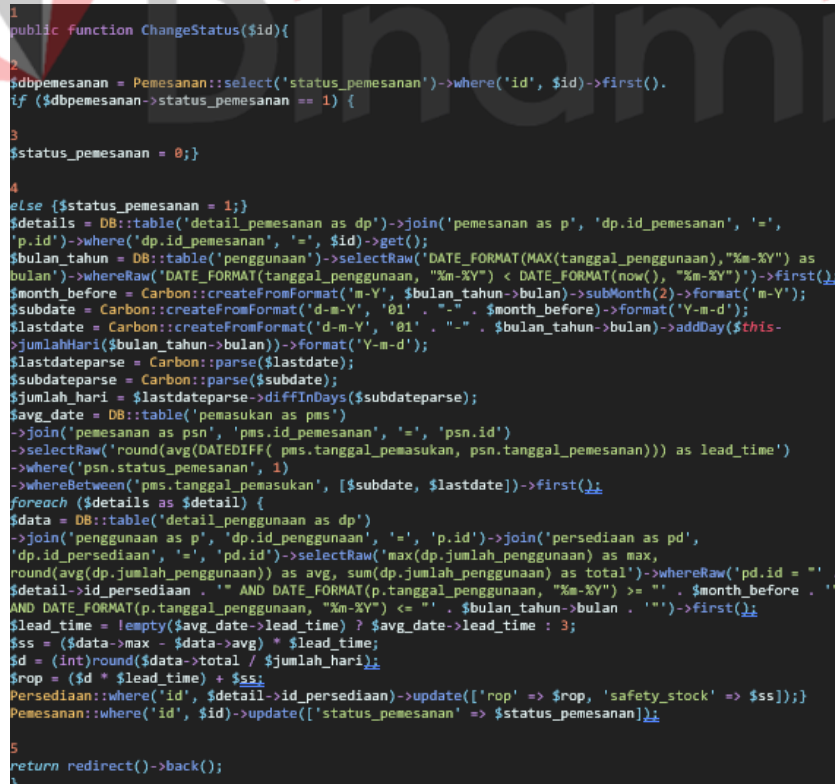
Gambar 4.3. Implementasi *Dashboard*

## 4.2 Testing.

### 4.2.1 White Box

Untuk melakukan pengujian aplikasi, akan dilakukan pengujian menggunakan metode *whitebox testing* yang melibatkan pemeriksaan terperinci terhadap logika dan struktur internal aplikasi. Pengujian aplikasi ini difokuskan pada pengujian metode perhitungan ROP dan EOQ. Tujuan dari pengujian ini adalah memastikan bahwa metode yang diimplementasikan dalam sistem berfungsi dengan benar dan menghasilkan perhitungan yang akurat. Melalui metode *whitebox testing*, penguji dapat memeriksa setiap langkah yang dilakukan dalam proses perhitungan dalam sistem. Hal ini membantu dalam mengidentifikasi potensi kesalahan dalam implementasi metode aplikasi serta memastikan bahwa sistem menghasilkan output yang sesuai dengan harapan.

Sebelum memasuki langkah perhitungan *Cyclomatic Complexity*, Langkah pertama yaitu mengubah kode menjadi *script code* yang digunakan saat aplikasi menjalankan metode ROP dan EOQ. Setelah tahap sebelumnya dijalankan, kode tersebut diubah dalam bentuk gambar *basis path*. Untuk *source code* metode perhitungan ROP dan EOQ dapat dilihat pada gambar 4.4 dan gambar 4.5.



```

1 public function ChangeStatus($id){
2
3 $dbpemesanan = Pemesanan::select('status_pemesanan')->where('id', $id)->first().
4 if ($dbpemesanan->status_pemesanan == 1) {
5
6 $status_pemesanan = 0;}
7
8 else {$status_pemesanan = 1;}
9 $details = DB::table('detail_pemesanan as dp')->join('pemesanan as p', 'dp.id_pemesanan', '=',
10 'p.id')->where('dp.id_pemesanan', '=', $id)->get();
11 $bulan_tahun = DB::table('penggunaan')->selectRaw('DATE_FORMAT(MAX(tanggal_penggunaan), "%m-%Y") as
12 bulan')->whereRaw('DATE_FORMAT(tanggal_penggunaan, "%m-%Y") < DATE_FORMAT(now(), "%m-%Y")')->first();
13 $month_before = Carbon::createFromFormat('m-Y', $bulan_tahun->bulan)->subMonth(2)->format('m-Y');
14 $subdate = Carbon::createFromFormat('d-m-Y', '01' . "-" . $month_before)->format('Y-m-d');
15 $lastdate = Carbon::createFromFormat('d-m-Y', '01' . "-" . $bulan_tahun->bulan)->addDay($this-
16 >jumlahHari($bulan_tahun->bulan))->format('Y-m-d');
17 $lastdateparse = Carbon::parse($lastdate);
18 $subdateparse = Carbon::parse($subdate);
19 $jumlah_hari = $lastdateparse->diffInDays($subdateparse);
20 $avg_date = DB::table('pemasukan as pms')
21 ->join('pemesanan as psn', 'pms.id_pemesanan', '=', 'psn.id')
22 ->selectRaw('round(avg(DATEDIFF( pms.tanggal_pemasukan, psn.tanggal_pemesanan))) as lead_time')
23 ->where('psn.status_pemesanan', 1)
24 ->whereBetween('pms.tanggal_pemasukan', [$subdate, $lastdate])->first();
25 foreach ($details as $detail) {
26 $data = DB::table('detail_penggunaan as dp')
27 ->join('penggunaan as p', 'dp.id_penggunaan', '=', 'p.id')->join('persediaan as pd',
28 'dp.id_persediaan', '=', 'pd.id')->selectRaw('max(dp.jumlah_penggunaan) as max,
29 round(avg(dp.jumlah_penggunaan) as avg, sum(dp.jumlah_penggunaan) as total')->whereRaw('pd.id = ' .
30 $detail->id_persediaan . ' ' AND DATE_FORMAT(p.tanggal_penggunaan, "%m-%Y") >= ' . $month_before . ' '
31 AND DATE_FORMAT(p.tanggal_penggunaan, "%m-%Y") <= ' . $bulan_tahun->bulan . ' ')->first();
32 $lead_time = !empty($avg_date->lead_time) ? $avg_date->lead_time : 3;
33 $ss = ($data->max - $data->avg) * $lead_time;
34 $d = (int)round($data->total / $jumlah_hari);
35 $rop = ($d * $lead_time) + $ss;
36 Persediaan::where('id', $detail->id_persediaan)->update(['rop' => $rop, 'safety_stock' => $ss]);
37 Pemesanan::where('id', $id)->update(['status_pemesanan' => $status_pemesanan]);
38
39 return redirect()->back();
40 }

```

Gambar 4.4. Source Code ROP

```

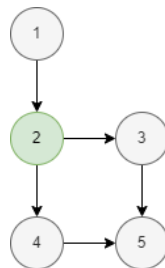
1 public function hitungEOQ(Request $request)
2 {
3
4 $request->validate([
5     'biaya_pemesanan' => 'required',
6 ]);
7
8 $bulan_tahun = DB::table('penggunaan')
9     ->selectRaw('DATE_FORMAT(MAX(tanggal_penggunaan),'m-%Y') as bulan')
10    ->whereRaw('DATE_FORMAT(tanggal_penggunaan, "%m-%Y") < DATE_FORMAT(now(), "%m-%Y")')
11    ->first();
12 $temporary_pemesanan = session("temporary_pemesanan");
13 $s = $request->biaya_pemesanan;
14 $no = 1;
15 $month_before = Carbon::createFromFormat('m-Y', $bulan_tahun->bulan)->subMonth(2)->format('m-Y');
16 foreach ($temporary_pemesanan as $temp) {
17     $data = DB::table('detail_penggunaan as dp')
18     ->join('penggunaan as p', 'dp.id_penggunaan', '=', 'p.id')
19     ->join('persediaan as ps', 'dp.id_persediaan', '=', 'ps.id')
20     ->selectRaw('max(dp.jumlah_penggunaan) as max, round(avg(dp.jumlah_penggunaan)) as avg,
21     sum(dp.jumlah_penggunaan) as total')
22     ->whereRaw('ps.id = "' . $temp['id'] . '" AND DATE_FORMAT(p.tanggal_penggunaan, "%m-%Y") >= "' .
23     $month_before . '" AND DATE_FORMAT(p.tanggal_penggunaan, "%m-%Y") <= "' . $bulan_tahun->bulan . '"')
24     ->first();
25
26     $persediaan = Persediaan::findorfail($temp['id']);
27     $temporary_pemesanan[$temp['id']]['eoq'] = $data->total > 0 ? round(sqrt((2 * $data->total * $request-
28     >biaya_pemesanan) / $persediaan->biaya_penjimpanan)) : 0;
29     $temporary_pemesanan[$temp['id']]['jumlah_pemesanan'] = $data->total > 0 ? round(sqrt((2 * $data-
30     >total * $request->biaya_pemesanan) / $persediaan->biaya_penjimpanan)) : 0;
31     $temporary_pemesanan[$temp['id']]['biaya_pemesanan'] = $request->biaya_pemesanan;
32     session(["temporary_pemesanan" => $temporary_pemesanan]);
33 }
34 $temporary_pemesanan = session("temporary_pemesanan");
35
36 return redirect()->route('tambah.pemesanan');

```

Gambar 4.5. Source Code EOQ

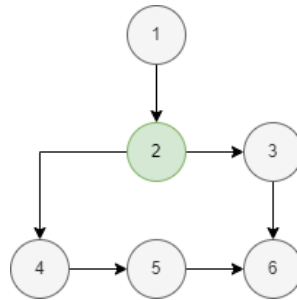
### 1. Flow Graph

Setelah *Source code* perhitungan ROP dan EOQ selesai, lanjut dalam pembuatan *flow graph* perhitungan ROP dan EOQ, hasil *flow graph* perhitungan ROP dan EOQ dapat dilihat pada gambar 4.6 dan gambar 4.7.



Gambar 4.6. Flow Graph Perhitungan ROP

Pada *flow graph* perhitungan ROP memiliki 5 *node*, pada *node* 1 berisikan tempat awalnya *fuction* untuk menghitung ROP, lalu *note* 2 untuk mengecek status pemesanan pada database pemesanan, *node* 3 untuk memeriksa jika status pemesanan 0 maka akan langsung ke *node* 5, *node* 4 memeriksa jika status pemesanan 1 maka akan melakukan perhitungan ROP, dan yang terakhir *node* 5 untuk kembali ke halaman sebelumnya.



Gambar 4.7. *Flow Graph* Perhitungan EOQ

Pada *flow graph* perhitungan EOQ memiliki 6 *node*, pada *node* 1 berisikan tempat awalnya *fuction* untuk menghitung EOQ, lalu *node* 2 untuk mengecek apakah user sudah memasukan biaya pemesanan, *node* 3 untuk memeriksa jika user tidak memasukan biaya pemasukan langsung ke *node* 6, *node* 4 memeriksa jika user memasukan biaya pemasukan, melakukan *query* untuk perhitungan EOQ, *node* 5 untuk melakukan perhitungan EOQ, dan yang terakhir *node* 6 untuk kembali ke halaman tambah pemesanan.

## 2. *Cyclomatic Complexity*

Setelah pembuatan *flow graph*, dilanjutkan melakukan perhitungan *cyclomatic complexity* berdasarkan *flow graph* yang dibuat pada tahap sebelumnya, berikut perhitungan yang diperoleh.

### *Cyclomatic Complexity Reorder Point*

Diketahui:

$E = 5$  *edge*.

$N = 5$  *nodes*.

$P = 1$  *predicate*.

Perhitungan *Reorder Point* dengan Rumus *edge* dan *nodes*.

$$V(G) = 5 - 5 + 2$$

$$V(G) = 0 + 2$$

$$V(G) = 2$$

Perhitungan *Reorder Point* dengan Rumus *Predicate*.

$$V(G) = 1 + 1$$

$$V(G) = 1$$

### *Cyclomatic Complexity Economic Order Quantity*

Diketahui:

$E = 6$  edge.

$N = 6$  nodes.

$P = 1$  predicate.

Perhitungan *Economic Order Quantity* dengan Rumus *edge* dan *nodes*.

$$V(G) = 6 - 6 + 2$$

$$V(G) = 0 + 2$$

$$V(G) = 2$$

Perhitungan *Economic Order Quantity* dengan Rumus *Predicate*.

$$V(G) = 1 + 1$$

$$V(G) = 2$$

Setelah mendapatkan hasil perhitungan *cyclomatic complexity*, diketahui bahwa ditemukan 2 *region*, setiap *region* memiliki *path* masing masing berasal dari *flow graph*. Pada tabel 4.3 dan tabel 4.4 adalah alur *path* dihasilkan dari setiap *region*. *graph matrix* bertujuan untuk memvisualisasikan jalur *independent* dapat dilihat pada tabel 4.1 dan tabel 4.2.

Tabel 4.1. *Graph Matrix* ROP

	1	2	3	4	5	
1		1				$1 - 1 = 0$
2			1	1		$2 - 1 = 1$
3					1	$1 - 1 = 0$
4					1	$1 - 1 = 0$
5						
						$1 + 1 = 2$

Tabel 4.2. *Graph Matrix* EOQ

	1	2	3	4	5	6	
1		1					$1 - 1 = 0$
2			1	1			$2 - 1 = 1$
3						1	$1 - 1 = 0$
4					1		$1 - 1 = 0$
5						1	$1 - 1 = 0$
6							
							$1 + 1 = 2$

Tabel 4.3. Alur *path* ROP

<i>Path</i>	<i>Alur Basic Path</i>
<i>Path</i> ke-1	1,2,3,5
<i>Path</i> ke-2	1,2,4,5

Tabel 4.4. Alur *path* EOQ

<i>Path</i>	<i>Alur Basic Path</i>
<i>Path</i> ke-1	1,2,3,6
<i>Path</i> ke-2	1,2,4,5,6

Pada tahap berikut merupakan *test case* dengan menguji 2 jalur dari proses perhitungan untuk menyesuaikan hasil *output* pada setiap jalur sesuai dengan apa yang diharapkan. Pada tabel 4.5 dan tabel 4.6 menunjukkan hasil dari pengujian pada 2 jalur dalam proses perhitungan ROP dan EOQ.

Tabel 4.5. Hasil *Test Case* ROP

<b>Path</b>	<b>Skenario Uji</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Uji</b>	<b>Hasil Alur</b>
1.	Ketika Status Pemesanan "Dalam Perjalanan"	Tidak Melakukan Perhitungan	Berhasil	<input checked="" type="checkbox"/> Terlewati <input type="checkbox"/> Tidak Terlewati
2.	Ketika Status Pemesanan Berubah Menjadi "Persediaan Telah Sampai"	Mendapatkan Hasil Perhitungan ROP	Berhasil	<input checked="" type="checkbox"/> Terlewati <input type="checkbox"/> Tidak Terlewati

Tabel 4.6. Hasil *Test Case* EOQ

<b>Path</b>	<b>Skenario Uji</b>	<b>Hasil Yang Diharapkan</b>	<b>Hasil Uji</b>	<b>Hasil Alur</b>
1.	Tidak Melakukan Pemesanan	Tidak Melakukan Perhitungan EOQ	Berhasil.	<input checked="" type="checkbox"/> Terlewati <input type="checkbox"/> Tidak Terlewati
2.	Memasukan Biaya Pemesanan	Mendapatkan Hasil Perhitungan EOQ	Berhasil.	<input checked="" type="checkbox"/> Terlewati <input type="checkbox"/> Tidak Terlewati

Berdasarkan hasil *test case* dari tabel 4.5 dan tabel 4.6, maka dapat disimpulkan bahwa struktur internal dan logika program berjalan dengan baik, serta hasil dari perhitungan *cyclomatic complexity* dan *grahp matrix* memiliki hasil yang

sama, sehingga dengan hasil *whitebox testing* yang baik, aplikasi layak digunakan dalam perusahaan.

#### 4.2.2 Implementasi Perhitungan ROP

Pada tahap ini akan melakukan proses implementasi perhitungan ROP, pada tabel 4.7 berisikan hasil rekap data permintaan persediaan dalam 1 periode pada bulan maret-april-mei yang akan digunakan untuk mengimplementasi perhitungan ROP.

Tabel 4.7. Rekap Data Permintaan Persediaan Dalam Periode Maret-April-Mei

Nama Persediaan	Permintaan Max	Permintaan Rata Rata(K)	Lead Time(LT)	Total Permintaan(D)
<i>Gas Methyl Bromide</i>	96	88,75	3	2840
Gas Oksigen	4	3,90625	3	125
<i>Cartridge Gas Mask</i>	2	2	3	64
Plastik Fumigasi	10	8,285714286	3	58
<i>Sand Snake</i>	120	102,5	3	410
Kertas Sertifikat Fumigasi	70	58,33333333	3	350

Setelah data telah direkap dapat mulai perhitungan dari menghitung *demand* pada 1 periode dan menghitung *safety stock*. Setelah variable lengkap akan memulai perhitungan ROP, rumus-rumus yang dihitung pada tabel 4.8 berada dalam bab 2.

Tabel 4.8. Perhitung *Manual* ROP Pada Periode Maret-April-Mei

Nama Persediaan	<i>Safety Stock</i> (SS)	ROP
<i>Gas Methyl Bromide</i>	21,75	114,3586957
Gas Oksigen	0,28125	4,357336957
<i>Cartridge Gas Mask</i>	0	2,1
Plastik Fumigasi	5,142857143	7,034161491
<i>Sand Snake</i>	52,5	65,86956522
Kertas Sertifikat Fumigasi	35	46,41304348

Berdasarkan hasil perhitungan ROP pada tabel 4.8, untuk melakukan pengujian perhitungan antara perhitungan aplikasi dan perhitungan *manual*, pada gambar 4.8 menampilkan hasil perhitungan ROP pada aplikasi yang memiliki data yang sama seperti perhitungan *manual*, lalu dapat membandingkan perhitungan sehingga hasil perhitungan aplikasi dan *manual* sama, hal ini dilakukan karna perlunya verifikasi implementasi hasil perhitungan sesuai dengan harapan.



Nama Persediaan	Jumlah Persediaan	Safety Stock	ROP
Gas Methyl Bromide	50	21	114
Gas Oksigen	50	1	5
Cartridge Gas Mask	50	3	5
Plastik Fumigasi	50	5	7
Sand Snake	50	53	65
Kertas Sertifikat Fumigasi	50	36	46

Gambar 4.8. Hasil Perhitungan Aplikasi ROP

Berdasarkan hasil perhitungan *manual* ROP pada tabel 4.8 dan Hasil perhitungan aplikasi ROP pada gambar 4.8 memiliki hasil perhitungan ROP yang sama sehingga membuktikan bahwa hasil perhitungan aplikasi ROP sudah berjalan dengan benar sesuai harapan.

#### 4.2.3 Implementasi Perhitungan EOQ

Pada tahap ini akan melakukan proses implementasi perhitungan EOQ, pada tabel 4.7 berisikan hasil rekap data permintaan persediaan dalam 1 periode yang akan digunakan untuk mengimplementasi perhitungan EOQ. Setelah data telah direkap dapat mulai perhitungan dari memperoleh biaya pemesanan dan memperoleh biaya penyimpanan. Setelah variable lengkap akan memulai perhitungan EOQ, rumus-rumus yang dihitung pada tabel 4.9 berada dalam bab 2.

Tabel 4.9. Perhitungan *Manual* EOQ Dalam Periode Maret-April-Mei

Nama Persediaan	Biaya Pemesanan(S)	Biaya Penyimpanan(H)	EOQ
Gas Methyl Bromide	100000	2000	532,9165038
Gas Oksigen	100000	2000	111,8033989
Cartridge Gas Mask	100000	2000	80
Plastik Fumigasi	100000	2000	76,15773106
Sand Snake	100000	2000	202,4845673
Kertas Sertifikat Fumigasi	100000	2000	187,0828693

Berdasarkan hasil perhitungan EOQ pada tabel 4.9, untuk melakukan pengujian perhitungan antara perhitungan aplikasi dan perhitungan *manual*, pada gambar 4.9 menampilkan hasil perhitungan EOQ pada aplikasi yang memiliki data

yang sama seperti perhitungan *manual*, lalu dapat membandingkan perhitungan sehingga hasil perhitungan aplikasi dan *manual* sama, hal ini dilakukan karna perlunya verifikasi implementasi hasil perhitungan sesuai dengan harapan.

No	Nama Persediaan	EOQ	Jumlah Pemesanan
1	Gas Methyl Bromide	528	528
2	Gas Oksigen	112	112
3	Cartridge Gas Mask	82	82
4	Plastik Fumigasi	93	93
5	Sand Snake	202	202
6	Kertas Sertifikat Fumigasi	187	187

Gambar 4.9. Hasil Perhitungan Aplikasi EOQ

Berdasarkan hasil perhitungan *manual* EOQ pada tabel 4.9 dan Hasil perhitungan aplikasi EOQ pada gambar 4.9 memiliki hasil perhitungan EOQ yang sama sehingga membuktikan bahwa hasil perhitungan aplikasi EOQ sudah berjalan dengan benar sesuai harapan.

## BAB V PENUTUP

### 5.1 Kesimpulan

Berdasarkan hasil penelitian, disimpulkan dari implementasi metode ROP dan EOQ dalam aplikasi pengendalian persediaan pada PT Prana Argentum berhasil dilaksanakan, dan dari hasil pengujian aplikasi dapat disimpulkan bahwa:

1. Setelah aplikasi melalui *whitebox testing* dapat disimpulkan bahwa alur perhitungan ROP dan EOQ dalam aplikasi sudah berjalan baik, benar, dan jalan seperti harapan peneliti.
2. Dari hasil pengujian perhitungan ROP dan EOQ bahwa perhitungan aplikasi memiliki hasil yang sama seperti hasil perhitungan *manual*, yang membuktikan bahwa perhitungan aplikasi sudah berjalan benar dan sesuai yang diharapkan oleh peneliti.

### 5.2 Saran

Dari hasil penelitian ini, Adapun beberapa masukan saran yang dapat mengembangkan system aplikasi ke depannya, yaitu:

1. Adanya *refresh* otomatis pada aplikasi untuk memberi informasi kepada *user* secara *real time*.
2. Memiliki *UI/UX* yang lebih baik, dengan tujuan membuat *user* aplikasi lebih nyaman saat menggunakan aplikasi tersebut.
3. Memiliki fitur untuk membedakan 2 jenis ukuran *container* yang akan dilakukan proses fumigator.

## DAFTAR PUSTAKA

- Budiarti, Y., & Risyanto. (2020). Implementasi Metode Extreme Programming Untuk Merancang Sistem Informasi Pendaftaran Siswa Baru Berbasis Web Pada Smk Multimedia Mandiri Jakarta. *Jurnal Ilmiah Fakultas Sains dan Teknologi*, 3-4.
- Chusminah, Haryati, A., & Nelfiant, F. (2019). Efektifitas Pengelolaan Persediaan Barang Dengan Sistem Safety Stock Pada Pt X Di Jakarta. *Jurnal Economic Resource*, 2.
- Dewi, P. C., Herawati, T. N., & Wahyuni, A. M. (2019). Analisis Pengendalian Persediaan Dengan Metode (Eoq) Economic Order Quantity Guna Optimalisasi Persediaan Bahan Baku Pengemas Air Mineral. *Jurnal Akuntansi Profesi*, 55.
- Djalambang, P. J., Qosim, N., & Hasan. (2021). Analisis Persediaan Beras Pada Toko Bali Yasa. *Jurnal Ekonomi Trend*, 40.
- Henim, S. A., & Sari, R. K. (2020). Evaluasi User Experience Sistem Informasi Akademik Mahasiswa Pada Perguruan Tinggi Menggunakan User Experience Questionnaire. *Jurnal Polietnik Caltex Riau*, 71 - 72.
- Ismunandar, R., Hendriadi, A. A., & Garno. (2019). Analisis Pengendalian Persediaan Dengan Metode (Eoq) Economic Order Quantity Guna Optimalisasi Persediaan Bahan Baku Pengemas Air Mineral. *Jurnal Akuntansi Profesi*, 56.
- Nurwulan, R. N., Taghsya, A. A., Astuti, D. E., Fitri, A. R., & Nisa, K. R. (2021). Pengurangan Lead Time dengan Lean Manufacturing: Kajian Literatur. *Journal of Industrial and Manufacture Engineering*, 34.
- PT Prana Argentum Corporation. (2016, agustus 13). *Company Profile PT Prana Argentum Corporation*. Retrieved from PT Prana Argentum Corporation: <http://fumigasi.id>
- Purwaningtias, F. (2018). E-Commerce Penjualan Berbasis Metode Ooad. *Jurnal Cendikia*, 2.
- Rafliana, T., & Suteja, R. B. (2018). Penerapan Metode EOQ dan ROP untuk Pengembangan Sistem Informasi Inventory Bengkel MJM berbasis Web. *Jurnal Teknik Informatika dan Sistem Informasi*, 346.
- Setiany, P. A., Noviyanto, D., Irfansyahfalah, M., Aisah, S., Aries, Saifudin, & Kusyadi, I. (2021). Penggunaan Metode System Development Life Cycle (SDLC) dalam Analisis dan Perancangan Sistem Informasi Penerimaan Kas Sekolah. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 180.

Suhendar, M., & Rosmalina. (2020). Pembuatan Profil Desa Gunungleutik Berbasis Web. *Jurnal Sistem Informasi*, 3.



UNIVERSITAS  
**Dinamika**