

## RANCANG BANGUN SISTEM AGEN CERDAS MONITORING STOK PERUSAHAAN

**Rika Yunitarini**

Jurusan Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo  
Jl. Telang PO. Box 2 Kamal, Bangkalan, Madura, 69162  
E-Mail: ryeca070683@yahoo.com

### *Abstract*

*Stock is one of the most important things in every business and manufacturer's factory. If factories cannot monitor their stocks well, it can disservice the factory by missing out the customers or over stocks. In addition, Stock monitoring is also a trial to monitor stocks in order to keep the stock in safe condition, it means that the stocks can covers the customers needs without lack or excess of stocks. Therefore, if goods is under the determined secure limit then the goods must be reordered. This research show the smart agent system which is used to monitor stock. This system works by monitoring the status of stock, classification, periodic review, Reorder Point System (ROP), exponential finesse prediction, and determination of the safety stock. In this research, the implementation of Smart Agent System can be used to determine the number of goods which is ordered periodically and the goods under the Reorder Point System (ROP) before the period, to determine the exact time in ordering goods to add stock and the number of stock in the secure condition.*

*Key words: Smart agent system, monitoring, stock and Reorder Point System*

### **PENDAHULUAN**

Persediaan diperlukan oleh setiap perusahaan. Dengan adanya persediaan maka perusahaan tidak perlu khawatir lagi dalam menghadapi resiko tidak terpenuhinya permintaan pelanggan pada suatu waktu tertentu. Jika resiko ini terjadi, maka perusahaan tidak akan memperoleh untung yang optimal. Mengingat pentingnya fungsi persediaan, maka diperlukan sistem yang mampu mengawasi dan mengatur jumlah pesanan, waktu pemesanan dan hal-hal lain yang terkait dengan persediaan. Untuk itulah perlu adanya sistem monitoring persediaan barang dalam suatu perusahaan.

Dalam *monitoring* persediaan sendiri terdapat beberapa masalah utama yang harus ditangani seperti jumlah barang yang harus dipesan atau ditambahkan pada persediaan dan waktu yang tepat untuk melakukan penambahan tersebut sehingga untung yang diperoleh dapat optimal.

Selain itu, perlu adanya persediaan pengaman (*safety stock*) yang cukup untuk mengantisipasi fluktuasi permintaan pelanggan. Hal ini penting karena jika perusahaan tidak dapat memenuhi permintaan pelanggan karena kekurangan persediaan, maka perusahaan itu akan mengalami kerugian bahkan kehilangan pelanggan.

Sistem agen cerdas sebenarnya merupakan program yang dapat diberi tugas dan dapat menyelesaikan tugasnya secara mandiri, serta mempunyai inteligensi. Dengan adanya bantuan sistem agen tersebut, maka pekerjaan yang membutuhkan waktu lama seperti menganalisis data dapat diselesaikan dengan baik dan lebih cepat daripada melakukannya secara manual, sehingga beban kerja dapat dikurangi serta efisiensi waktu dan tenaga dapat diperoleh. Oleh karena itu, permasalahan *monitoring* persediaan dapat dibantu dengan membuat suatu software berbasis agen cerdas.

Rumusan masalah yang akan dijadikan penelitian adalah bagaimana membuat sistem

agen cerdas *monitoring* persediaan perusahaan dengan kemampuan agen untuk menentukan jumlah barang yang harus dipesan sehingga tidak terjadi penumpukan atau kekurangan persediaan di gudang, menentukan waktu yang tepat untuk melakukan pemesanan barang untuk menambah persediaan, serta menentukan jumlah persediaan pengaman yang harus ada.

Beberapa batasan yang digunakan dalam penelitian, perusahaan yang dijadikan objek penelitian adalah swalayan yang menjual kebutuhan hidup sehari-hari, diasumsikan hanya terdapat sebuah gudang tempat penyimpanan persediaan, diasumsikan barang yang dibeli dan dijual semuanya bagus (tidak ada retur), diasumsikan setiap barang dibeli minimal satu kali setiap bulannya, diasumsikan perjalanan distribusi barang lancar, serta aplikasi yang dibuat menggunakan *tool visual Basic 6.0*.

Untuk mencapai tujuan penelitian, metodologi yang digunakan meliputi analisis kebutuhan data dan studi literatur, perancangan sistem, serta implementasi dan pengujian. Pada tahap pertama, dilakukan *survey* pada obyek yang dijadikan bahan penelitian, sebagaimana yang disebutkan dalam batasan masalah. Selain itu dilakukan pula studi literatur secara intensif tentang sistem agen cerdas, *monitoring* persediaan, serta ROP. Pada tahap perancangan sistem, hal yang dilakukan pada tahap ini adalah membuat detail perancangan dari hasil analisis, yang meliputi perancangan goal overview diagram, perancangan arsitektur sistem, dan perancangan proses yang terdapat dalam sistem agen cerdas *monitoring* stok perusahaan, dari hasil perancangan yang ada, diimplementasikan ke dalam bahasa pemrograman untuk memperoleh hasil berupa *software* agen cerdas yang kemudian akan diuji cobakan untuk menjaga agar sistem sesuai dengan tujuan penelitian.

## AGEN CERDAS

### Pengertian Agen Cerdas

Agen, yang pada mulanya berasal dari suatu cabang ilmu *artificial intelligence* (AI) yang bernama *distributed artificial intelligence* (DAI), seiring dengan berjalannya waktu memiliki perkembangan dalam hal pengertiannya. Sebagai tambahan, penjelasan

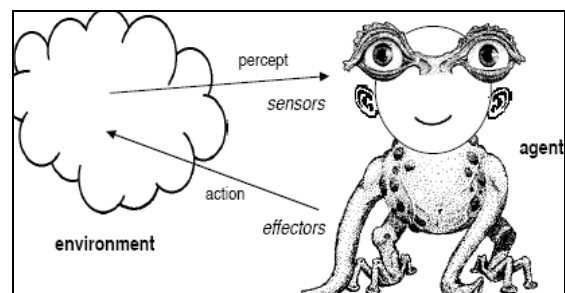
tentang agen juga dapat diperoleh dengan melihat Gambar 1.

Dari Gambar 1 terlihat bahwa agen berada pada lingkungan tertentu dan dapat berinteraksi dengan lingkungannya dengan menggunakan *sensor* dan *effector*. Terdapat proses yang terjadi secara berulang-ulang pada agen yaitu agen menerima masukan (*percept*) dari lingkungan dengan menggunakan *sensor*, lalu agen mencari dan memilih tindakan yang sesuai untuk masukan tadi, kemudian memberikan tindakan (*action*) yang telah dipilih tadi ke lingkungannya dengan menggunakan *effector*. Tindakan yang telah dilakukan oleh agen ke lingkungannya dapat mempengaruhi dan mengubah lingkungannya tersebut.

### Karakteristik Agen Cerdas

Sebuah agen memiliki karakteristik yang menggambarkan kemampuan dari agen itu sendiri. Semakin banyak karakteristik yang dimiliki oleh suatu agen, maka akan semakin cerdas agen tersebut. Ada beberapa karakteristik dari agen yang diberikan [1] yaitu sebagai berikut :

1. *Autonomous*, agen berkemampuan untuk melakukan tugasnya dan mengambil keputusan secara mandiri tanpa adanya intervensi dari luar seperti agen lain, manusia ataupun entitas lain,
2. Reaktif, yaitu kemampuan agen untuk cepat beradaptasi terhadap perubahan informasi yang ada pada lingkungannya.
3. Proaktif, yaitu kemampuan yang berorientasi pada tujuan dengan cara selalu mengambil inisiatif untuk mencapai tujuan.



Gambar 1. Agen dan lingkungannya.

4. Fleksibel, agen harus mempunyai banyak cara dalam mencapai tujuannya.
5. *Robust*. Agen harus dapat kembali ke kondisi semula jika mengalami kegagalan

dalam hal tindakan ataupun dalam menjalankan *plan*.

6. Rasional, yaitu kemampuan untuk bertindak sesuai dengan tugas dan pengetahuannya dengan tidak melakukan hal yang dapat menimbulkan konflik tindakan.
7. Kemampuan berkoordinasi dan berkomunikasi (*Social*). Dalam melakukan tugasnya, agen memiliki kemampuan untuk berkomunikasi dan berkoordinasi baik dengan manusia maupun dengan agen lain.
8. *Situated*. Agen harus berada dan berjalan di lingkungan tertentu.

### Arsitektur Agen

Penelitian ini hanya akan menjelaskan arsitektur agen secara umum. Beberapa arsitektur agen [2] adalah sebagai berikut:

1. *Black Box*. Pada konsep *black box*, agen menerima masukan (*percepts*) dari luar lalu memprosesnya sehingga bisa dihasilkan keluaran (*action*) yang berdasarkan masukan tadi. Konsep *black box* ini dapat digambarkan seperti pada Gambar 2. Tahap selanjutnya dalam *black box* adalah memikirkan tahap-tahapan apa yang terjadi dalam proses tadi. Brenner mengemukakan suatu model untuk proses ini yang berisi tahap-tahapan seperti *interaction*, *information fusion*, *information processing* dan *action*[3].
2. *BDI Agent*. Pada arsitektur ini pengambilan keputusan dilakukan berdasarkan pada tiga hal yakni kepercayaan (*belief*), keinginan (*desire*), dan kehendak (*intention*). Kepercayaan atau *Belief* adalah apa-apa saja yang diketahui dan tidak diketahui oleh agen tentang lingkungannya atau dengan kata lain *belief* merupakan pengetahuan agen atau informasi yang diperoleh agen tentang lingkungannya. Keinginan atau *desire* dapat diartikan sebagai tujuan, tugas yang harus diselesaikan oleh agen atau sesuatu yang ingin dicapai oleh agen. Sedangkan kehendak atau *intention* adalah rencana-rencana yang disusun untuk mencapai keinginan tadi.

### Monitoring Persediaan

*Monitoring* persediaan merupakan upaya pemantauan persediaan untuk menjaga agar

persediaan tersebut selalu dapat mencukupi kebutuhan pelanggan tanpa mengalami kelebihan atau kekurangan. Jika barang berada dibawah batas aman yang telah ditentukan, maka barang harus dipesan kembali.

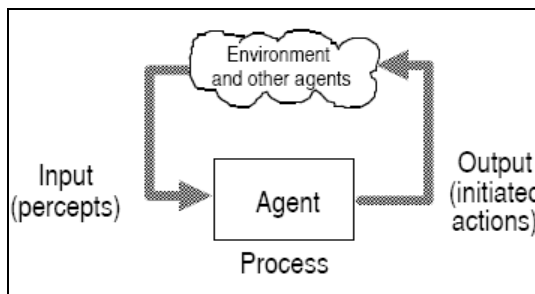
*Monitoring* dapat dilakukan dengan mengatasi masalah-masalah yang biasa terjadi pada persediaan, seperti penentuan jumlah barang yang harus dipesan atau ditambahkan pada persediaan, penentuan waktu untuk melakukan penambahan, serta penentuan jumlah persediaan pengaman yang harus ada.

Dalam *monitoring* persediaan, guna melakukan pengisian kembali terhadap persediaan yang telah berkurang dapat digunakan metode pengisian kembali persediaan yaitu [4] *Reorder Point System*. Dengan menggunakan metode ini, maka pengisian kembali akan dilakukan apabila total persediaan *on-hand* jatuh atau berada pada dibawah titik pemesanan kembali. Titik pengisian kembali adalah suatu tingkat persediaan yang tetap ada dalam *stock* yang jumlahnya sama dengan permintaan selama masa waktu untuk menerima pesanan (disebut dengan *lead time*) [5]. Akan tetapi, karena adanya permintaan yang sifatnya fluktuatif, maka persediaan pengaman perlu ditambahkan ke dalam perhitungan titik pengisian ulang tersebut (Persamaan (1)).

$$ROP = DLT + SS \tag{1}$$

Keterangan:

- *ROP*: titik pemesanan kembali (*reorder point*)
- *DLT*: permintaan selama waktu tunggu
- *SS*: persediaan pengaman (*safety stock*).



Gambar 2. Agen sebagai *Black Box*.

## ANALISA DAN PERANCANGAN SISTEM

### Persediaan Pada Swalayan

Swalayan merupakan salah satu perusahaan yang bertindak sebagai distributor barang-barang kebutuhan hidup sehari-hari kepada konsumen. Sebagai perusahaan dagang, swalayan perlu untuk memperhatikan kepuasan pelanggan, salah satu caranya adalah dengan menyediakan barang yang ingin dicari pelanggan. Hal ini dapat dilakukan apabila barang-barang yang ingin dicari itu terdapat pada persediaan yang dimiliki swalayan. Jika persediaan swalayan itu berkurang atau tidak ada maka permintaan pelanggan tadi tidak akan dapat dipenuhi. Hal ini akan berakibat hilangnya pelanggan swalayan tersebut sehingga secara tak langsung dapat mengurangi laba yang didapat. Oleh karena itulah maka perlu untuk memonitor persediaan tersebut.

Dengan monitoring persediaan, maka akan terdapat mekanisme dimana jika persediaan kurang dari atau sama dengan ROP, maka toko akan melakukan pengisian ulang persediaan dengan memesan barang tersebut kembali. Pengisian ulang tersebut juga dapat dilakukan secara periodik dengan periode waktu yang tetap. Penentuan periode waktu ini dapat dilakukan dengan terlebih dahulu melakukan pengklasifikasian terhadap barang-barang yang ada dalam persediaan.

Untuk keseragaman, maka dalam penelitian ini akan dibuat suatu rancangan khusus untuk proses *monitoring* persediaan pada swalayan dengan keadaan sebagai berikut:

1. Swalayan hanya mempunyai satu gudang dan diasumsikan gudang tersebut cukup untuk menyimpan persediaan yang ada.
2. Pemilihan pemasok berdasarkan waktu terlebih dahulu baru kemudian berdasarkan harga.
3. Sistem pengisian kembali persediaan yang dipakai adalah dengan mengkombinasikan ROP (*reorder point*) system dan *periodic review system*. Persediaan akan diisi setiap periode waktu yang tetap. Periode waktu ini berbeda untuk tiap kelas barang. Jika persediaan suatu barang kurang dari atau sama dengan ROP sebelum periode *review* berakhir, maka agen akan menghitung jumlah pesanan barang tersebut dengan besar pesanan sama dengan sisa hari sebelum periode *review* dikalikan dengan ramalan perhari.

4. Peramalan barang menggunakan model pemulusan eksponensial dengan asumsi data penjualan fluktuatif dan tidak ada kecendrungan. Nilai *alpha* yang digunakan adalah 0,3. Hal ini dikarenakan nilai *alpha* yang sering dipakai adalah antara 0,1 sampai dengan 0,3 [1].
5. Menggunakan metode pengklasifikasian ABC dalam menentukan prioritas tiap-tiap persediaan. Untuk barang kategori A digunakan waktu review 1 bulan, kategori B 2 bulan dan kategori C 3 bulan. Pengklasifikasian dilakukan setiap 6 bulan sekali.
6. Diasumsikan barang yang dijual dan dibeli semuanya bagus (tidak ada retur).

### Spesifikasi Sistem

#### Spesifikasi Tujuan

Tujuan utama dari sistem dapat dicapai dengan menggunakan bantuan beberapa sub tujuan. Selain itu, sub-sub tujuan itu juga dapat diperoleh dengan menjalankan beberapa sub tujuan lagi.

Pada Gambar 3 dapat dilihat bahwa untuk dapat memonitor persediaan, dapat ditempuh dengan cara mendapatkan status untuk tiap-tiap barang. Status ini dapat diperoleh dengan mencatat transaksi masuk dan mencatat transaksi keluar. Selain dengan mendapatkan status barang, *monitoring* dapat dilakukan dengan cara menentukan jumlah barang. Sub tujuan ini dapat ditempuh dengan melakukan beberapa sub tujuan lain seperti menentukan pemasok, mendapatkan prioritas barang, persediaan pengaman, dan meramalkan jumlah permintaan.

#### Spesifikasi Skenario

Skenario ini bertujuan untuk memberikan gambaran tentang bagaimana atau langkah demi langkah jalannya sistem.

Berikut adalah detail dari tiap-tiap skenario:

1. Skenario menentukan Pemasok. Skenario menentukan pemasok melibatkan beberapa tindakan (*action*) yaitu bandingkan harga barang pemasok, bandingkan waktu antar barang, dan menentukan pemasok.
2. Skenario Mencatat Transaksi Masuk. Pada skenario mencatat transaksi masuk

dilibatkan beberapa langkah seperti masukan (*percept*) data barang tiba dan data barang baru serta tindakan (*action*) mencatat barang masuk.

3. Skenario Mencatat Transaksi Keluar. Skenario mencatat transaksi keluar hanya melibatkan dua hal yaitu masukan (*percept*) data barang jual dan tindakan (*action*) mencatat barang keluar.
4. Skenario Mendapatkan Status Barang. Pada skenario mendapatkan status barang melibatkan dua skenario sebelumnya yakni mencatat transaksi masuk dan mencatat transaksi keluar.
5. Skenario Meramalkan Jumlah Permintaan. Skenario meramalkan jumlah permintaan melibatkan beberapa langkah yaitu masukan (*percept*) data penjualan dan tindakan (*action*) melakukan peramalan pemulusan eksponensial.
6. Skenario Mendapatkan Prioritas Barang. Skenario mendapatkan prioritas barang melibatkan beberapa hal yakni masukan (*percept*) data barang dan data penjualan serta tindakan (*action*) melakukan klasifikasi ABC.
7. Skenario Menentukan Jumlah Barang Yang Dipesan. Pada skenario menentukan jumlah barang melibatkan beberapa hal yaitu skenario menentukan pemasok, skenario meramalkan jumlah permintaan, skenario mendapatkan prioritas barang, skenario persediaan pengaman, tindakan (*action*) menentukan jumlah inventori maksimal, dan tindakan menentukan jumlah barang yang akan dipesan.
8. Skenario *Monitoring* Persediaan. Pada skenario *monitoring* persediaan melibatkan beberapa langkah yaitu skenario mendapatkan status barang dan skenario menentukan jumlah barang.

### ***Percept* dan *Action***

Adapun masukan dan tindakan adalah:

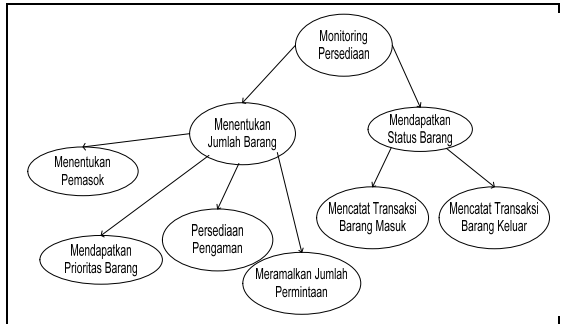
1. *Percept*. *Percept* yang digunakan di dalam sistem antara lain data pemasok baru, data barang jual, data barang beli, dan data barang tiba.
2. *Action*. *Action* yang terlibat adalah bandingkan harga barang pemasok, bandingkan waktu antar barang, tentukan pemasok, menentukan persediaan

pengaman, melakukan klasifikasi ABC, menentukan jumlah barang yang dipesan, melakukan peramalan pemulusan eksponensial, menentukan jumlah inventori maksimal, mencatat barang keluar, dan mencatat barang masuk.

### **Spesifikasi Data**

Adapun data yang terlibat dalam pembuatan sistem ini adalah data barang, data penjualan, data pemasok, data pesanan, data pembelian, dan data temp. Data barang digunakan untuk mencatat informasi barang seperti nama, barcode, jumlah, harga, id pemasok, waktu (periode *review*), dan SS (persediaan pengaman). Data penjualan digunakan untuk mencatat informasi penjualan seperti id penjualan, *barcode* barang, jumlah dan total penjualan serta tanggal transaksi penjualan (hari, bulan, dan tahun). Data pembelian digunakan untuk mencatat informasi pembelian seperti no faktur, *barcode* barang, jumlah, harga, total pembelian, id pemasok, dan status pembelian. Data pemasok digunakan untuk mencatat data pemasok seperti id, nama pemasok, waktu antar, harga, dan *barcode* barang.

Sedangkan data pesanan digunakan untuk mencatat informasi pesanan barang seperti nama barang, jumlah aktual, jumlah pesanan, jumlah persediaan pengaman, nama pemasok, dan status pesanan barang. *Data temp* digunakan untuk mencatat informasi sementara yang berkaitan dengan penghitungan pesanan seperti kapasitas maksimal inventori dan jumlah permintaan berdasarkan ramalan. Data ini berguna untuk menghitung jumlah pesanan barang karena berada dalam level yang tidak aman atau kurang dari sama dengan ROP.



Gambar 3. Goal Overview Diagram.

### Rancangan Arsitektur

Terdapat tiga agen yang digunakan dalam sistem ini yaitu agen perencanaan dan monitoring (*agen\_pm*), agen gudang (*agen\_gudang*), dan agen penjualan (*agen\_jual*).

*Agen\_pm* digunakan untuk melakukan pemantauan terhadap persediaan, menentukan jumlah pesanan, dan menampilkan informasi barang yang telah dihitung jumlah pesannya. *Agen\_gudang* berguna untuk mencatat data pemasok baru, mencatat barang masuk, mencatat data pembelian barang, menampilkan informasi barang, menampilkan informasi pemasok, dan memilih pemasok. Sedangkan *agen\_jual* berhubungan dengan pencatatan barang yang dijual ke pelanggan.

Hubungan yang terjadi antar komponen yang terlibat yaitu :

1. *Agen\_jual* menerima masukan berupa data barang yang dijual, agen ini akan mencatat data tersebut pada *datapenjualan*, *databarang*, dan *datapesanan*. Setelah itu, *agen\_jual* akan mengirimkan pesan ke *agen\_pm* berupa pesan permintaan supaya *agen\_pm* mengecek tanggal atau mengecek barang yang ada.
2. Jika pesan adalah mengecek tanggal, maka setelah menerima pesan tersebut *agen\_pm* akan mengecek tanggal, jika tanggal merupakan merupakan salah satu tanggal dalam periode *review*, maka *agen\_pm* akan berhubungan dengan *agen\_gudang* melalui protokol rencana pembelian dengan mengirimkan pesan permintaan untuk dipikirkan pemasok dari barang yang akan dihitung dengan ketentuan pemasok tersebut merupakan pemasok dengan waktu antar tercepat dan harga termurah.

3. Setelah *agen\_gudang* menerima pesan tersebut, lalu *agen\_gudang* melakukan tindakan-tindakan seperti bandingkan waktu antar pemasok, bandingkan harga pemasok, dan tentukan pemasok berdasarkan informasi yang ada di *datapemasok*. Setelah itu, informasi id pemasok terpilih tadi disimpan di *databarang*.
4. Setelah *agen\_gudang* melakukan tindakan tentukan pemasok, *agen\_gudang* kembali berhubungan dengan *agen\_pm* menggunakan protokol rencana pembelian dengan mengirimkan pesan bahwa memilih pemasok telah selesai dilakukan.
5. Setelah *agen\_pm* menerima pesan tersebut, *agen\_pm* melakukan tindakan melakukan klasifikasi ABC, menentukan persediaan pengaman, melakukan peramalan pemulusan eksponensial, menentukan jumlah inventori maksimal, dan menentukan jumlah barang yang dipesan berdasarkan *datapenjualan*, *databarang*, dan *datapembelian*. Kemudian *agen\_pm* memperbaharui *datapesanan*, *databarang* dan *dataTemp*.
6. Tindakan hitung pesanan karena dibawah ROP baru akan dilakukan *agen\_pm* jika ternyata barang berada dalam kondisi tidak aman atau kurang dari sama dengan ROP.
7. Ketika terdapat masukan berupa *databarang beli* atau data pembelian barang, maka *agen\_gudang* akan melakukan tindakan mencatat pembelian barang pada *datapembelian*.
8. Ketika terdapat masukan berupa *databarang tiba* atau barang yang dipesan telah diterima, maka *agen\_gudang* melakukan tindakan mencatat barang masuk dengan memperbaharui *datapembelian*, *datapesanan* dan *databarang*.
9. Jika terdapat masukan data pemasok baru, maka *agen\_gudang* akan melakukan tindakan mencatat pemasok dengan memperbaharui *datapemasok*.

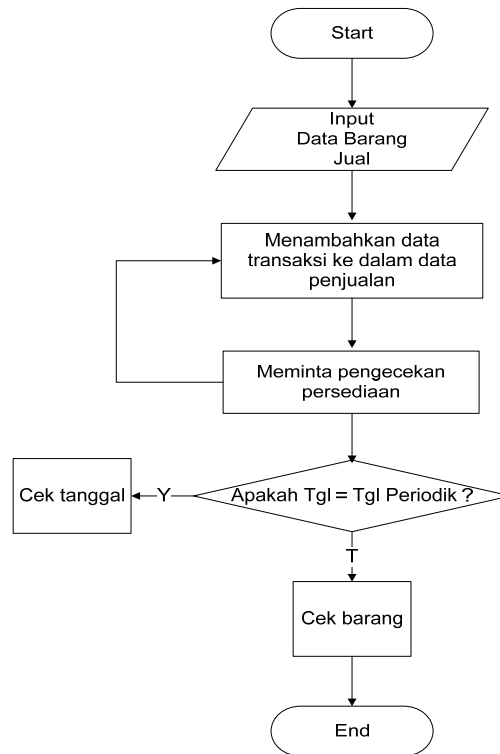
### Rancangan Proses

Rancangan proses digunakan untuk memberikan gambaran yang lebih detail tentang proses-proses yang terjadi di dalam agen.

1. Proses Pencatatan Transaksi Keluar

Proses ini terjadi di dalam *agen\_jual*. Proses yang terjadi dapat digambarkan dengan *flowchart* seperti pada Gambar 4.

Pada Gambar 4 dapat terlihat bahwa proses ini dimulai dengan *agen\_jual* menerima masukan berupa data barang jual. Lalu agen tersebut melakukan dua proses secara paralel yakni menambahkan data transaksi ke dalam data penjualan dan meminta pengecekan persediaan. Di dalam melakukan aktivitas meminta pengecekan persediaan, *agen\_jual* akan mengecek tanggal transaksi, apakah tanggal tersebut merupakan salah satu tanggal pemesanan periodik atau bukan. Jika ternyata tanggal merupakan tanggal pemesanan periodik, maka agen ini akan mengirimkan pesan ke *agen\_pm* berupa pesan cek tanggal, sedangkan jika tidak, maka pesan yang dikirim adalah cek barang. Cek tanggal berisi informasi tanggal, sedangkan cek barang berisi informasi barcode barang dan jumlah barang yang dijual.

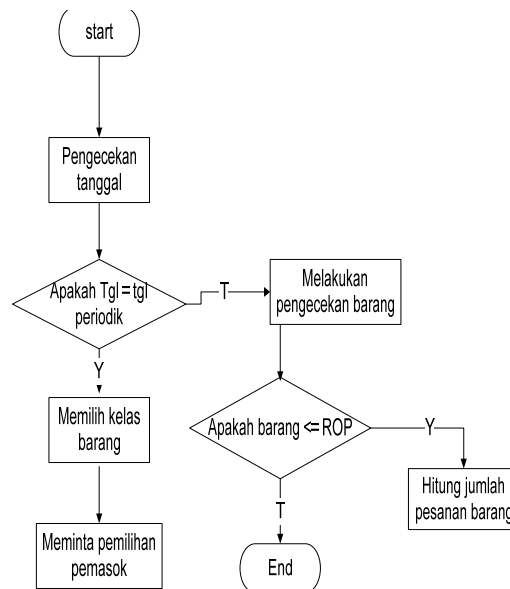


Gambar 4. *Flowchart* Proses Pencatatan Transaksi Keluar dalam *Agen\_jual*.

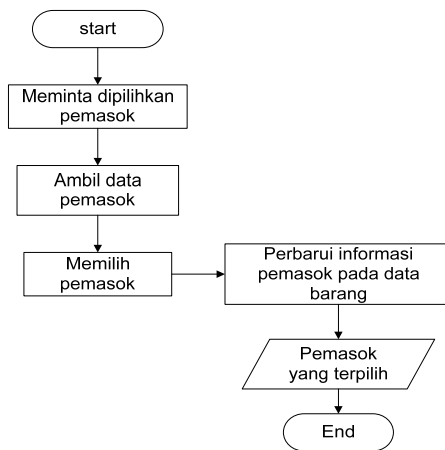
2. Proses Pengecekan Persediaan

Proses ini terjadi di dalam *agen\_pm*. Proses ini diawali dengan *agen\_pm* menerima masukan dari *agen\_jual* berupa pesan cek tanggal atau cek barang. Jika pesan yang diterima adalah cek tanggal, maka *agen\_pm* melakukan aktivitas berupa memilih kelas barang mana yang harus dihitung jumlah pesannya berdasarkan informasi tanggal yang dibawa oleh pesan cek tanggal. Setelah mendapatkan kelas barang yang harus dihitung pesannya, maka agen ini akan mengirimkan pesan ke *agen\_gudang* untuk dipilih pemasok.

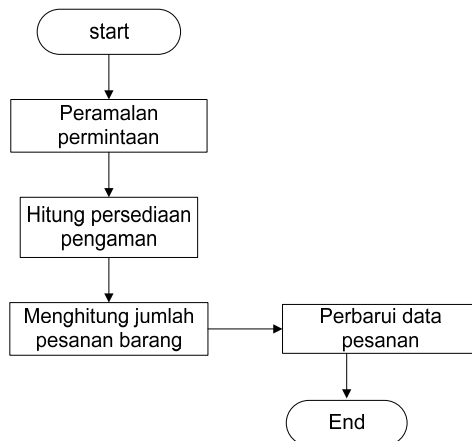
Sedangkan jika pesan adalah cek barang, maka *agen\_pm* akan melakukan aktivitas berupa melakukan pengecekan barang. Apabila barang yang informasinya dikirim lewat pesan cek barang tersebut jumlahnya dalam persediaan kurang dari sama dengan ROP maka agen melakukan aktivitas berupa hitung jumlah pesanan barang. Sedangkan jika tidak, maka proses akan berhenti atau selesai. Untuk lebih jelasnya dapat dilihat pada Gambar 5.



Gambar 5. *Flowchart* Proses Pengecekan Persediaan dalam *Agen\_pm*.



Gambar 6. *Flowchart* Proses Pemilihan Pemasok dalam *Agen\_gudang*.



Gambar 7. *Flowchart* Proses Penghitungan Pesanan dalam *Agen\_pm*.

### 3. Proses Pemilihan Pemasok

Proses ini terjadi di dalam *agen\_gudang*. Proses yang terjadi pada pemilihan pemasok dapat digambarkan dengan *flowchart* seperti pada Gambar 6.

Pada Gambar 6 dapat terlihat bahwa proses ini dimulai dengan *agen\_gudang* menerima masukan berupa pesan meminta dipikirkan pemasok dari *agen\_pm*. Lalu *agen\_gudang* melakukan aktivitas berupa ambil data pemasok. Setelah itu, agen ini akan melakukan aktivitas memilih pemasok. Kemudian, setelah pemasok terpilih, *agen\_gudang* memperbaharui informasi pemasok yang ada pada data barang. Lalu setelah itu agen mengirimkan pesan ke *agen\_pm* untuk memberitahukan

bahwa pemilihan pemasok telah dilakukan.

### 4. Proses Penghitungan Pesanan

Proses ini terjadi di dalam *agen\_pm*. Proses ini diawali dengan *agen\_pm* menerima masukan dari *agen\_gudang* berupa pesan selesai pilih pemasok. Setelah itu, agen melakukan aktivitas berupa ramalkan permintaan, hitung persediaan pengaman, dan Imaks (total persediaan selama periode *review*), menghitung jumlah pesanan barang, dan perbaharui data pesanan. Untuk lebih jelasnya dapat dilihat pada Gambar 7.

## HASIL DAN PEMBAHASAN

Sistem yang dibuat dapat digunakan untuk menyimpan informasi-informasi yang dimasukkan pengguna seperti informasi barang, informasi pemasok, informasi pembelian barang, informasi penjualan, dan informasi penerimaan barang. Terdapat dua buah agen yang bertanggung jawab terhadap hal-hal tersebut yakni *agen\_gudang* dan *agen\_jual*.

### 1. Menyimpan Informasi Barang

Proses ini diawali dengan pengguna memasukkan informasi barang baru pada antarmuka tambah barang. Setelah pengguna menekan tombol tambah, maka *agen\_gudang* akan mengambil informasi tersebut dan menyimpannya ke dalam *file databarang.dat*. Setelah itu, agen ini mengirimkan pesan ke pengguna bahwa tugas yang diberikan telah selesai. Jika terdapat *barcode* barang yang sama dengan yang telah tersimpan, maka data yang lama akan diperbaharui dengan yang baru sedangkan jika *barcode* yang dimasukkan tidak sama maka informasi tersebut disimpan pada *tuple* yang baru pada *file databarang.dat*.

### 2. Menyimpan Informasi Pemasok

Proses menyimpan informasi pemasok diawali dengan pengguna memasukkan informasi pemasok pada antarmuka tambah pemasok. Setelah pengguna menekan tombol tambah, maka *agen\_gudang* akan mengambil informasi tersebut dan menyimpannya ke dalam *file datapemasok.dat*. Setelah itu, agen



melakukan konfirmasi ke pengguna bahwa tugas telah dilaksanakannya.

### 3. Menyimpan Informasi Pembelian

Proses ini dilakukan pada saat pengguna ingin menyimpan informasi pembelian barang dan diawali dengan pengguna memasukkan informasi pembelian barang tersebut pada antarmuka pembelian barang. Setelah pengguna menekan tombol tambah, maka *agen\_gudang* akan mengambil informasi tersebut dan menyimpannya ke dalam *file datapembelian.dat*. Setelah itu, agen ini melakukan konfirmasi ke pengguna bahwa tugas yang diberikan kepadanya telah berhasil.

### 4. Menyimpan Informasi Penerimaan Barang

Proses menyimpan informasi penerimaan barang diawali dengan pengguna memasukkan informasi no faktur pada antarmuka. Setelah pengguna menekan tombol tambah, maka *agen\_gudang* akan mengambil informasi tersebut dan memperbaharui informasi yang ada *file datapembelian.dat*, *databarang.dat*, dan *datapesanan.dat*. Setelah itu, jika faktur pembelian tersebut belum pernah diterima maka agen melakukan konfirmasi ke pengguna bahwa tugas telah dilaksanakannya. Sedangkan jika faktur pembelian tersebut sudah pernah diterima maka agen akan memberikan pesan bahwa data tersebut sudah pernah diproses. Begitu juga jika faktur tersebut tidak terdapat dalam data pembelian, maka agen akan memberitahu pengguna bahwa data tidak ada.

### 5. Menyimpan Informasi Penjualan Barang

Proses ini diawali dengan pengguna memasukkan informasi barang yang dijual ke pelanggan pada antarmuka penjualan barang. Setelah pengguna menekan tombol tambah, maka *agen\_jual* akan mengambil informasi tersebut dan menyimpannya ke dalam *file datapenjualan.dat*. Agen juga akan mengubah informasi jumlah barang yang ada pada *file databarang.dat* dan *datapesanan.dat*. Setelah itu, agen melakukan konfirmasi ke pengguna bahwa tugas telah dilaksanakan.

## Hasil Pengujian Sistem dalam Menginformasikan Data

Sistem dapat menginformasikan beberapa hal seperti informasi data barang, informasi data pemasok, dan informasi data pesanan barang. Agen-agen yang bekerja pada proses ini adalah *agen\_gudang* dan *agen\_pm*.

*Agen\_gudang* bertanggung jawab untuk menampilkan informasi data barang dan informasi data pemasok. Di dalam melakukan proses ini, *agen\_gudang* akan mengambil kedua data tersebut dari *file databarang.dat* dan *datapemasok.dat*. Setelah itu data dikirimkan ke antarmuka untuk kemudian ditampilkan. Lalu agen mengirimkan pesan jika telah sukses mengambil data.

*Agen\_pm* bertanggung jawab untuk menampilkan informasi data pesanan barang. Informasi ini berguna bagi manajer toko untuk melihat hal-hal seperti barang-barang mana saja yang harus segera dipesan, berapa jumlah pesannya, dan ke pemasok mana barang tersebut harus dipesan. Setelah agen sukses mengambil data dan mengirimkannya ke antarmuka, maka agen akan melakukan konfirmasi ke pengguna bahwa telah sukses mengambil data.

## Hasil Pengujian Sistem dalam Monitoring Persediaan

Sistem dapat melakukan *monitoring* persediaan dengan cara mengecek barang yang dijual. Setelah melakukan transaksi barang, *agen\_jual* akan melakukan pengecekan tanggal dan mengirimkan pesan ke *agen\_pm* untuk meminta dilakukan pengecekan tanggal atau pengecekan barang. Pengecekan tanggal dikirimkan jika tanggal pada saat transaksi merupakan salah satu tanggal untuk melakukan pemesanan periodik. Sedangkan pengecekan barang dikirimkan jika tanggal bukan termasuk salah satu tanggal pemesanan periodik. Setelah itu *agen\_jual* akan memberikan informasi tersebut ke pengguna.

Tanggapan yang diberikan oleh *agen\_pm* dan *agen\_gudang* terhadap pesan yang dikirim oleh *agen\_jual* adalah sebagai berikut:

1. Jika tanggal merupakan salah satu tanggal pemesanan periodik.

Setelah menerima pesan dari *agen\_jual*, *agen\_pm* kembali akan mengecek tanggal untuk menentukan kelas barang mana yang harus dihitung jumlah pesannya. Setelah mendapatkan kelas barang mana yang harus

dihitung, maka agen akan memberikan informasi ke pengguna. *Agen\_pm* akan mengirimkan pesan ke *agen\_gudang* untuk dipikirkan pemasok untuk tiap-tiap barang pada kelas terpilih. Setelah menerima pesan tersebut *agen\_gudang* akan melakukan pemilihan pemasok dan selanjutnya *agen\_gudang* akan menginformasikan hal ini ke pengguna dan *agen\_pm* menghitung jumlah barang yang harus dipesan. Perhitungan dilakukan dengan meramalkan permintaan yang akan datang berdasarkan histori penjualan barang. Hasil dari perhitungan tersebut akan dimasukkan ke dalam file *datapesanan.dat*. Lalu setelah hal itu selesai, *agen\_pm* akan menginformasikan hal ini ke pengguna.

2. Jika tanggal bukan merupakan salah satu tanggal pemesanan periodik.

Setelah menerima pesan pengecekan barang dari *agen\_jual*, *agen\_pm* akan mengecek jumlah barang tersebut di dalam persediaan apakah kurang dari sama dengan ROP atau masih berada dalam *level* aman. Jika barang masih berada dalam *level* aman, maka *agen\_pm* akan mengirimkan pesan ke pengguna. Sedangkan jika barang berada pada *level* tidak aman atau kurang dari sama dengan ROP maka agen mengirimkan pesan ke pengguna. Setelah itu agen menghitung jumlah barang yang harus dipesan dan akan kembali mengirimkan pesan.

## SIMPULAN DAN SARAN

Simpulan yang dapat diambil dengan dibangunnya Sistem Agen Cerdas *Monitoring* Persediaan Perusahaan adalah:

1. Sistem agen cerdas *monitoring* persediaan dapat menentukan jumlah barang yang harus dipesan sehingga tidak terjadi penumpukan atau kekurangan persediaan di gudang.
2. Sistem agen cerdas yang dibangun dapat digunakan untuk menentukan waktu yang tepat melakukan pemesanan barang guna menambah persediaan dan sekaligus dapat menentukan jumlah persediaan pengaman yang harus ada.
3. Dengan *monitoring* persediaan, maka terdapat mekanisme dimana jika persediaan kurang dari atau sama dengan ROP, maka toko akan melakukan pengisian ulang persediaan dengan memesan barang tersebut kembali. Pengisian ulang tersebut juga dapat dilakukan secara periodik dengan periode waktu yang tetap. Penentuan periode waktu ini dapat dilakukan dengan terlebih dahulu melakukan pengklasifikasian terhadap barang-barang yang ada dalam persediaan.

Saran yang dapat diajukan untuk pengembangan dan perbaikan Sistem Agen Cerdas *Monitoring* Persediaan Perusahaan adalah:

1. Sistem yang akan datang sebaiknya dibangun agar bisa memesan barang ke pemasok secara otomatis.
2. Sistem sebaiknya dibuat dengan lebih kompleks dan menarik sehingga dapat digunakan pada dunia nyata.
3. Sebaiknya sistem yang akan dibangun selanjutnya dapat menerapkan berbagai macam peramalan seperti peramalan jangka pendek, peramalan jangka menengah, dan peramalan jangka panjang.
4. Sistem agen yang dibangun sebaiknya dapat melakukan pembelajaran.

## DAFTAR PUSTAKA

- [1] Padgham L and Winikoff M. *Developing Intelligent Agent System. A Practical Guide*. United Kingdom: John Wiley & Sons Ltd. 2004.
- [2] Wahono RS. *Pengantar Multi Agent Sistem (MAS)*. 2006. URL: <http://www.ilmukomputer.com/romiagent.pdf>, diakses tanggal 22 Januari 2006.
- [3] Brenner W, Zarnekow R and Wittig H. *Intelligent Software Agents: Foundation and Applications*. New York: Springer-Verlag. 2006.
- [4] Gaspersz V. *Production Planning And Inventory Control*. Jakarta: Gramedia Pustaka Utama. 2006.
- [5] Taylor BW. *Sains Manajemen. Pendekatan Matematika untuk Bisnis*. Jakarta: Selemba Empat. 2004