



**UBAYA**  
UNIVERSITAS SURABAYA



# **PROSIDING SNASTIA 2008**

## **KEBANGKITAN TEKNOLOGI INFORMASI DAN KOMUNIKASI NASIONAL**



Surabaya, 31 Mei 2008

**ITFEST**

8.	Program Aplikasi Penunjang Pengambilan Keputusan Pemilihan Tanaman Obat dengan Metode Electre <i>Susana Limanto, Ellysa Tjandra</i>	196
9.	Pembuatan Program Aplikasi untuk Menjalankan Program Komputer Melalui Suara <i>Budi Hartanto, Melissa Angga, Andre Sutanto</i>	203
10.	Perancangan Aplikasi <i>Two-Phase Translation Method</i> pada <i>Query-Based Transitive Translation</i> <i>Adi Heru Utomo, Joko Lianto Buliali, Fuad Cholisi</i>	209
11.	Perencanaan Pengembangan Aplikasi Penggalian <i>Top-K Frequent Closed Constrained Gradient Itemsets</i> pada Basis Data Retail <i>Dhiani Tresna Absari, Arif Djunaidy</i>	220
12.	Sistem Simulasi Berbasis SIG <i>Daniel Hary Prasetyo</i>	230
13.	Auto Matching Antar Dokumen Dengan Metode Cosine Measure <i>Wiwin Sulisty, Riyanarto Sarno</i>	237
14.	Pembuatan Perangkat Lunak Editor Jadwal dan Penelusuran Proyek dengan Metode Earned Value Analysis <i>Ellysa Tjandra, Susana Limanto</i>	248
15.	Tools Simulasi Antrian Pada Supermarket <i>Liliana, Benny Santoso, Imelda Yapitro</i>	257
16.	Virtual Class Berbasis Intelligent Tutoring System <i>Bernard Renaldy Suteja, Sri Hartati</i>	267
IV.	NETWORK AND MOBILE COMPUTING	
1.	Perancangan Antena Mikrostrip Dengan Impedance Matching Untuk Handphone 3G <i>Joko Nugroho, F.X. Hendra Pra</i>	278
2.	Prediksi Jangkauan Jaringan Nirkabel HF Untuk Sistem Peringatan Dini Bencana di Indonesia <i>Wismanu Susetyo, Gamantyo Hendratoro, Achmad Affan</i>	285
3.	The Interaction Forms in Distance Education: "The Synchronous, Asynchronous, and Hybrid Systems" <i>Nani Sri Handayan</i>	293
4.	Implementasi Enkripsi Metode MD5 Pada Simulasi Token dan Server E-Banking <i>Yohanes Priyo Atmojo, Muhammad Rusli, Komang Oka Saputra</i>	301
5.	Key Issues in Combating Cyber Crime <i>Leo Willyanto Santoso</i>	307

# TOOLS SIMULASI ANTRIAN PADA SUPERMARKET

**Liliana, Benny Santoso, Imelda Yapitro**

*Teknik Informatika, Teknik, Universitas Surabaya / Raya Kalirungkut Surabaya*

*Email: [liliana@if.ubaya.ac.id](mailto:liliana@if.ubaya.ac.id)*

*Email: [us6163@fox.ubaya.ac.id](mailto:us6163@fox.ubaya.ac.id)*

*Email: [ed\\_linq@yahoo.com](mailto:ed_linq@yahoo.com)*

## ABSTRAK

*The number of sales in supermarket is difficult to be predicted, therefore the number of cashier have to be provided by the supermarket is complicated to be made. The reason is that there are lots of factors should be entered into consideration.*

*Simulation as a tool for creating the model to describe the system's behavior could be used to approach this problem. This paper explains the steps to create the tools for simulating the queuing process in supermarket's cashiers.*

**Kata kunci:** *simulation, supermarket simulation, supermarket*

### 1. Pendahuluan

Penjualan di supermarket cenderung sulit untuk diprediksi karena dipengaruhi oleh beberapa faktor. Antara lain pola kedatangan pelanggan selalu bervariasi. Ketika pembelian meningkat maka sering terjadi antrian yang panjang. Untuk menangani hal tersebut supermarket mencoba memecahkan masalah dengan menambah jumlah kasir pada semua waktu. Faktor lainnya terdapat pada bagian inventori dimana sering terjadi kesalahan-kesalahan dalam memprediksi jumlah barang yang masuk dan keluar, dan kesalahan ini dapat mengakibatkan dua hal yakni pertama stok digudang berlebih sehingga akan mendatangkan kerugian karena adanya biaya penyimpanan digudang dan hal kedua yakni stok yang ada kurang dari keinginan pelanggan untuk membeli sehingga terjadi kehilangan opportunity. Untuk bisa memprediksi dengan baik harus memperhatikan kebutuhan pelanggan karena kebutuhan pelanggan berbeda-beda mengenai barang yang satu dengan yang lain. Oleh karena itu setiap barang harus diperkirakan polanya.

Meskipun perkiraan bisa dilakukan namun kadangkala terjadi perubahan yang tidak terduga sebelumnya. Perubahan tersebut dipengaruhi oleh dua hal yakni sistemnya berubah dan adanya pola yang bisa berubah dari waktu ke waktu. Karena terjadi dua hal tersebut maka diperlukan sistem yang dapat mengatasi jika terjadi perubahan tidak akan mensimulasikan dari awal.

Oleh karena itu, dibutuhkan suatu aplikasi yang dapat membantu memperkirakan jumlah kasir yang harus disediakan ketika pelanggan datang dengan pola distribusi tertentu atau pada event tertentu atau memperkirakan jumlah stok yang harus disediakan pada waktu tertentu, untuk menghindari *lost sales*.

### 2. Tinjauan Pustaka

Simulasi merupakan suatu teknik permodelan yang menggambarkan hubungan sebab akibat suatu sistem untuk menghasilkan perilaku sistem yang hampir sama dengan perilaku sistem yang sebenarnya. Simulasi dapat digunakan untuk menghasilkan suatu catatan historis yang aktual dan kesimpulan statistik dari semua aktivitas yang terjadi selama periode waktu didesain.

Simulasi digunakan untuk menganalisa suatu sistem secara numerik. Hal ini dilakukan apabila sistem yang ada terlalu kompleks untuk dievaluasi secara analitik. Jika hubungan yang ada dalam suatu sistem sederhana, maka sistem tersebut dapat digambarkan dengan menggunakan suatu metode matematika untuk menghasilkan informasi yang nyata. Cara ini

termasuk cara penyelesaian secara analitik. Tetapi pada kenyataannya, hampir semua sistem yang ada terlalu rumit untuk dievaluasi secara analitik, sehingga sistem harus dipelajari dengan bantuan simulasi.

Simulasi merupakan alat analisa yang kuat yang digunakan untuk membantu para ahli dalam membuat keputusan yang baik, dan tepat waktu pada desain dan operasi sistem. Ada tiga hal dalam mengklasifikasikan model-model simulasi [2], yaitu:

a. Model simulasi statik dan dinamik

Model simulasi statik menggambarkan keadaan suatu sistem pada suatu waktu tertentu, contohnya model simulasi Monte Carlo. Sedangkan model simulasi dinamik menggambarkan keadaan suatu sistem sesuai dengan perubahan yang terjadi sepanjang waktu, contohnya sistem pembawa barang pada suatu pabrik.

b. Model simulasi deterministik dan stokastik

Model simulasi deterministik merupakan suatu model simulasi yang tidak memiliki komponen yang bersifat probabilistik. Pada model ini, nilai input untuk suatu perhitungan hanya satu (tertentu), dan output ditentukan pada waktu seluruh input sudah ditentukan. Tetapi pada kenyataannya, kebanyakan sistem yang ada memiliki beberapa komponen input yang random, sehingga digunakan model simulasi stokastik, contohnya sistem antrian dan inventori. Model simulasi ini menghasilkan output yang random dan output ini dianggap sebagai suatu perhitungan karakteristik model yang benar.

c. Model simulasi kontinu dan diskrit

Model simulasi diskrit menggambarkan perubahan variabel state yang tiba-tiba pada periode waktu yang acak. Sedangkan model simulasi kontinu menggambarkan perubahan variabel state yang konstan pada periode waktu yang tetap. Keputusan untuk menggunakan model simulasi kontinu dan diskrit untuk suatu sistem tertentu tergantung pada objek yang akan dipelajari. Sebagai contoh suatu model arus lalu lintas pada suatu jalan raya bisa merupakan model diskrit jika karakteristik dan perpindahan tiap mobil dianggap penting. Tetapi jika mobil-mobil yang ada dianggap sebagai suatu kumpulan maka model ini merupakan model simulasi yang kontinu.

Inventori dapat didefinisikan sebagai sejumlah barang yang disimpan perusahaan untuk kemudian digunakan. Istilah ini juga dapat didefinisikan sebagai segala sesuatu atau sumber organisasi yang disimpan dalam antisipasinya terhadap pemenuhan permintaan konsumen [1]. Hal ini meliputi persediaan bahan mentah, barang dalam proses, barang jadi atau produk akhir, bahan pelengkap dan komponen yang menjadi bagian keluaran produk perusahaan. Pengertian inventori tidak terbatas pada jenis itu saja tetapi juga dapat berupa uang, bangunan pabrik, peralatan, dan tenaga kerja yang digunakan untuk memenuhi permintaan akan produk dan jasa.

Sedangkan sistem inventori merupakan suatu sistem dinamis yang mengatur keluar masuknya barang dari suatu tempat penyimpanan. Contohnya gudang penyimpanan yang dimiliki toko komputer. Bentuk kebijakan sistem inventori adalah  $(s,S)$ .  $S$  merupakan jumlah maksimum produk yang ditampung oleh gudang penyimpanan. Sedangkan  $s$  merupakan jumlah produk minimum yang dijadikan signal untuk melakukan order ke supplier. Jika suatu saat jumlah produk kurang dari  $s$  maka harus melakukan order. Hal-hal yang mempengaruhi inventori antara lain [2] :

- Menentukan event

- Ada 4 jenis event yang dibutuhkan dalam menjalankan simulasi ini :
  - Datangnya order
  - Permintaan dari konsumen

- Selesaiya simulasi setelah  $n$  bulan
- Evaluasi inventori (pemeriksaan jumlah produk di gudang) secara periodik
- Penentuan variabel state
  - Inventori level. Ada dua jenis inventori level yakni jumlah produk yang ada digudang dan jumlah produk yang seharusnya ada (karena permintaan konsumen) tetapi stok digudang habis.
  - Jumlah permintaan melebihi stok
  - Time last event
- Penentuan laporan
  - Average total cost : total ordering cost per bulan /  $n$
  - Average ordering cost : total ordering cost per bulan /  $n$
  - Average holding cost : total holding cost per bulan /  $n$
  - Average shortage cost : total shortage cost per bulan /  $n$

### 3. Analisis Sistem

Secara umum, proses yang ada pada supermarket dibedakan menjadi :

- Penitipan barang
- Mencari barang di rak/lorong supermarket
- Pembayaran di kasir
- Pengambilan titipan barang di kasir
- Pengambilan hadiah (jika ada)

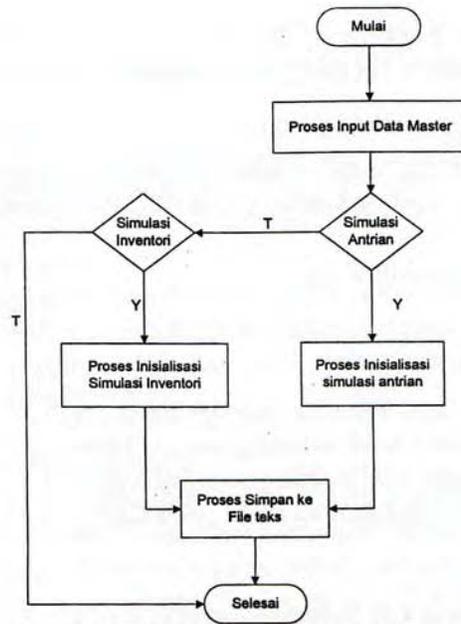
Pada aplikasi simulasi supermarket yang sudah ada, pola distribusi yang ada pada aplikasi tetap, sehingga ketika pihak supermarket ingin melihat pola kedatangan dengan distribusi yang lain, ia harus mengubah *source code* aplikasi. Oleh karena itu, hasil analisa menunjukkan bahwa sistem membutuhkan aplikasi yang dapat membantu mensimulasikan supermarket tanpa harus mengulang dari awal lagi dalam hal ini diperlukan adanya perangkat lunak yang dapat menyimpan hasil simulasi supermarket.

Selain itu, dibutuhkan pula sistem yang dapat menentukan distribusi secara fleksibel dimana pola yang ada dapat diatur sesuai dengan keadaan yang dimiliki. Dan aplikasi ini tidak dapat menentukan distribusi yang sesuai namun hanya dapat membantu pengguna program untuk memilih distribusi dan keputusan untuk menggunakan distribusi tertentu berada ditangan pengguna program itu sendiri. Dan tentunya sistem harus mampu mensimulasikan satu atau lebih stok dari barang yang ada.

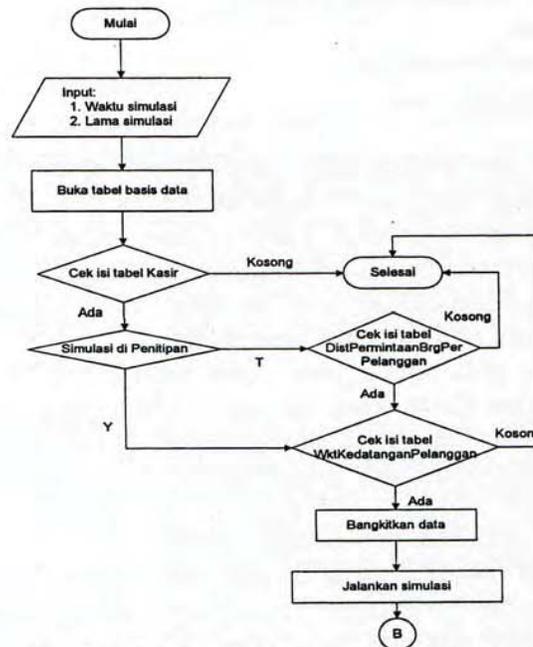
Semua proses simulasi dibuat dengan kondisi sama dimana hasilnya berupa data-data selama simulasi dijalankan dan perhitungan laporan berdasarkan inputan dari user. Keputusan untuk membuka berapa kasir dan stok yang harus tersedia dapat tentukan oleh pengguna program berdasarkan simulasi yang dilakukan. Dan, untuk mempermudah proses simulasi dan pembacaan data, proses simulasi dapat dilaksanakan untuk lebih dari satu barang, dimana barang-barang tersebut dikelompokkan berdasarkan golongan tertentu.

### 4. Desain Proses

Proses-proses yang ada meliputi proses penyimpanan hasil simulasi ke dalam file, proses run simulasi antrian dan proses untuk membangkitkan data. Desain proses (*flowchart*) untuk proses-proses yang ada dalam aplikasi ini dapat dilihat pada Gambar 1. Desain proses (*flowchart*) untuk proses simulasi pada antrian, dapat dilihat pada Gambar 2.



Gambar 1. Flowchart proses utama



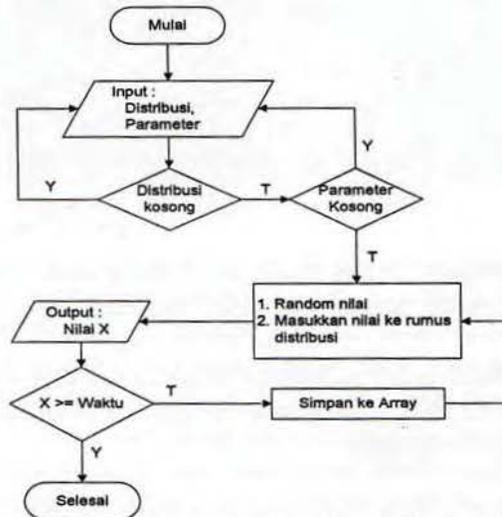
Gambar 2. Flowchart proses simulasi antrian

Pada proses simulasi antrian, simClock diawali dengan 0 untuk menandakan waktu simulasi dimulai. simClock ini menunjukkan nilai simulasi saat ini atau sedang berlangsung. Selama waktu simulasi sekarang kurang dari lama waktu simulasi yang diinputkan pada proses inisialisasi maka dilakukan pengecekan kedatangan. Jika kedatangan lebih kecil dari pelayanan terakhir (ServisTerakhir) maka simClock sekarang sama dengan kedatangan dan tipeServer diberi nilai true (berarti sistem sekarang akan menjadwalkan event kedatangan). Selain itu maka simClock sama dengan servisTerakhir dan tipeServer diberi nilai false (berarti sistem sekarang akan menjadwalkan event kepergian).

Setelah itu jika nilai dari tipeServer menjadwalkan event kedatangan maka kedatangan yang sekarang disimpan kedalam variabel kedatanganTerakhir dan sistem akan mengambil nilai kedatangan yang baru berdasarkan hasil generate data kedatangan pelanggan. Jika

serverStatus (digunakan untuk menandakan sibuk atau tidaknya suatu server) sibuk maka kedatangan terakhir dimasukkan kedalam antrian. Jika serverStatus idle maka ditampilkan servisMulai dan servisSelesai. Selain itu jika nilai dari tipeServer menjadwalkan event kepergian maka dilakukan pengecekan apakah terdapat antrian atau tidak. Jika terdapat antrian maka ditampilkan nilai servisMulai dan servisSelesai berdasarkan orang pertama dalam antrian. Jika antrian kosong maka servisTerakhir diberi nilai tak terhingga dan statusServer idle. Lakukan proses ini berulang sampai simClock lebih besar dari lama simulasi yang diinginkan.

Sedangkan flowchart untuk proses membangkitkan data berdasarkan parameter tertentu, dapat dilihat pada Gambar 3.



Gambar 3. Flocwchart proses membangkitkan data

## 5. Implementasi

Berikut ini akan dijabarkan bagian-bagian yang ada dalam simulasi antrian.

### 5.1 Barang

Barang dalam aplikasi ini merupakan item yang akan disimulasikan. Jenis barang yang disimpan dapat disesuaikan dengan tipe barang yang dijual oleh pengguna program.

Form barang digunakan untuk menambah/mengubah data barang terdapat dalam basis data. Data yang harus diisi adalah [Nama], [Stok], [Harga], [Lama Pelayanan], [Lama Cari], [Pro Subtitusi], [Kode Group], [Maksimum Jum Jual per pelanggan], [Maksimum Jum Jual per hari]. Data [Pro Subtitusi] digunakan untuk menghitung kemungkinan customer membeli barang lain jika barang yang diinginkan tidak ada. [Kode Group] digunakan untuk menggabungkan barang ini ke group tertentu. Sehingga jika terjadi substitusi/pergantian barang, barang-barang yang termasuk di group yang sama lah yang akan dipilih. Data [Maksimum Jum Jual per pelanggan], [Maksimum Jum Jual per hari] diinputkan jika ada batasan tertentu dalam penjualan. Misalnya barang A hanya boleh dijual 3 pcs per pelanggan. Implementasi form master barang dapat dilihat pada Gambar 4.

Selain barang, juga terdapat master group barang, master setting waktu, master kondisi dan master variabel. Master group barang digunakan untuk menambah/mengubah data kelompok barang, dimana kelompok barang ini dapat disimulasikan secara bersamaan.

Master setting waktu digunakan untuk mengatur waktu-waktu yang nanti akan digunakan dalam pengaturan data-data simulasi. Waktu ini digunakan ketika pola yang diinginkan berbeda antara waktu yang satu dengan waktu yang lain. User diberi kesempatan untuk memilih dari waktu-waktu yang disediakan.

Gambar 4. Form Master Barang

Master kondisi digunakan untuk menambah/mengubah data kondisi-kondisi yang mempengaruhi penjualan barang, seperti hari raya, *weekend*, akhir tahun, awal bulan, dan lain sebagainya.

Master variabel digunakan untuk mengatur variabel-variabel yang akan digunakan ketika proses simulasi antrian berlangsung. Data-data variabel ini akan digunakan untuk menghitung kepadatan yang terjadi selama simulasi.

## 5.2 Simulasi Antrian

Berikut ini akan dijabarkan tahapan yang harus dikerjakan untuk menjalankan simulasi antrian pada penjualan supermarket.

### 5.2.1 Input Data Kasir

User dapat menentukan pola dan parameter yang digunakan untuk setiap bagian tipe kasir. Khusus untuk bagian pembayaran dikasir, user terlebih dahulu harus menentukan jumlah kasir yang ada. Sewaktu memilih jumlah kasir, maka jumlah yang terpilih akan muncul secara otomatis pada kolom sebelah kanan. Setelah itu user dapat menentukan pola serta parameter yang sesuai untuk bagian tersebut. Form untuk menginputkan / mengubah data kasir dapat dilihat pada Gambar 5.

Gambar 5. Form input data kasir

### 5.2.2 Input Pola Kedatangan

Bagian kedua yang dapat dilakukan user dalam form input data ini yakni menginputkan pola kedatangan dari pelanggan dalam supermarket. Pola ini bisa diatur untuk semua waktu maupun untuk waktu-waktu tertentu. Implementasi user interface pada form input data

kedatangan pelanggan dapat dilihat pada Gambar 6. Tombol (Grafik) digunakan ketika user ingin melihat grafik dari distribusi dan parameter yang telah diinputkan.

Gambar 6. Form input data kedatangan pelanggan

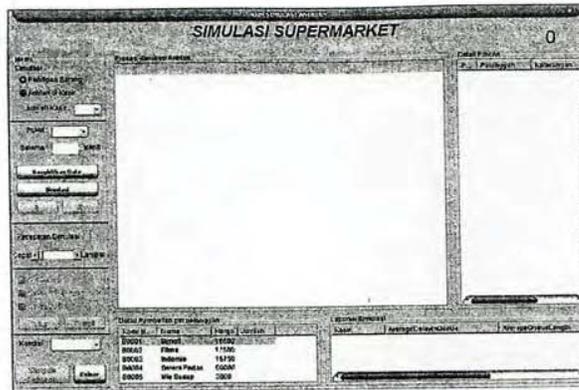
### 5.2.3 Input Pola Permintaan Barang

Bagian terakhir yang dapat dilakukan user dalam form input data ini adalah dengan menginputkan pola permintaan barang oleh pelanggan di supermarket. Form ini hampir sama dengan form input data kedatangan pelanggan, dimana user bisa mengatur pola permintaan yang ada untuk semua waktu maupun untuk waktu-waktu tertentu. Ada dua pilihan yang dapat dilakukan yakni dengan menginputkan permintaan berdasarkan kode barang maupun dengan kode group. Kode barang dan kode group memiliki arti yang sama, dimana input berdasarkan kode group dimaksudkan agar user tidak perlu menginputkan barang satu per satu. Form input data permintaan pelanggan dapat dilihat pada Gambar 7.

Gambar 7. Form input Pola Permintaan

### 5.2.4 Simulasi Antrian

Dalam form ini akan ditunjukkan proses-proses yang terjadi selama simulasi antrian diijalankan. Sebelum menjalankan simulasi, user terlebih dahulu memilih tipe simulasi yang diinginkan. Jika user memilih mensimulasikan antrian dipenitipan barang, maka sistem akan menunjukkan proses yang terjadi di penitipan barang. Sebaliknya jika user memilih simulasi antrian di kasir maka sistem akan menunjukkan proses simulasi dari pertama kali user datang ke supermarket sampai user keluar dari supermarket. Implementasi form simulasi antrian dapat dilihat pada Gambar 8.



Gambar 8. Form simulasi antrian

### Uji Coba

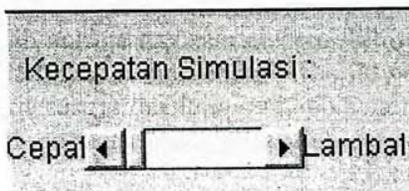
Tidak ada cara pasti untuk menentukan kebenaran dalam tahap uji coba validasi terhadap simulasi dari aplikasi ini. Uji coba dilakukan dalam tahap verifikasi. Verifikasi tampilan dan hasil simulasi antrian dapat dilihat pada Gambar 9.

Bagian-bagian yang ada dalam form simulasi antrian akan dijelaskan berikut ini. Sebelum menjalankan proses simulasi, user harus memilih proses simulasi mana yang diinginkan, apakah simulasi penitipan barang atau simulasi antrian di kasir. Jika ia memilih simulasi antrian di kasir, maka ia harus mengisi jumlah kasir yang dibuka. Data waktu mulai dan lama simulasi juga harus diisi. Untuk membangkitkan data sesuai inputan data sebelumnya, user dapat mengklik tombol "Bangkitkan Data". Dan untuk menjalankan simulasi, user dapat mengklik tombol "Simulasi". Gambar data awal ini dapat dilihat pada Gambar 9.



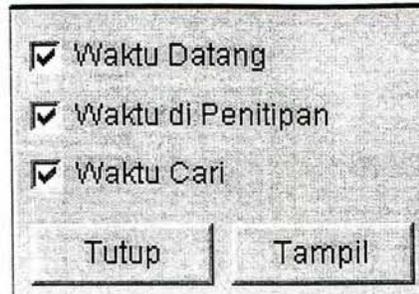
Gambar 9. Data awal simulasi

Kecepatan jalannya simulasi dapat diset melalui *scroll bar* yang ada pada form run simulasi ini. Dapat dilihat pada Gambar 10.



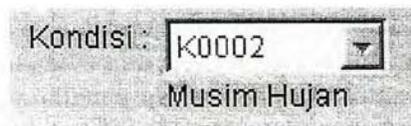
Gambar 10. Pengaturan kecepatan simulasi

Data yang ditampilkan pada layar dapat diatur sesuai keinginan user. Dapat dilihat pada Gambar 11.



Gambar 11. Pengaturan tampilan data

Kondisi simulasi dapat diseting berdasarkan keinginan user. Dapat dilihat pada Gambar 12.



Gambar 12. Seting kondisi simulasi

Daftar barang yang dibeli oleh pelanggan selama proses simulasi dapat dilihat dalam daftar pada Gambar 13.

Kode B...	Nama	Harga	Jumlah
B0001	Bimoli	16500	
B0002	Filma	17500	
B0003	Indomie	16750	
B0004	Selera Pedas	50000	
B0005	Mie Sedap	3000	

Gambar 13. Daftar barang yang dibeli pelanggan

Detail kedatangan per pelanggan dapat dilihat pada Gambar 14, dan detail kejadian per waktu dapat dilihat pada Gambar 15.

Pelanggan	W.Datang	W.DPenitipan	W.CariBarang	W.MasukKasir	Keterangan
1	1.22	0.61	1.6	3.43	Keluar supri
2	1.27	14.06	5.00	20.33	Lagi cari bs
3	1.33	2.22	0.50	4.05	Keluar supri
4	1.56	2.97	2.60	7.13	Keluar supri
5	1.88	2.21	0.50	4.59	Keluar supri
6	2.35	0.99	5.60	8.94	6
7	3.48	2.06	0.50	6.04	Keluar supri
8	3.84	7.61	1.00	12.45	Masuk Antri
9	6.45	4.18	2.60	13.23	Masuk Antri
10	6.9	0.03	1.50	8.43	Selesai dilz
11	7.96	4.67	0.10	12.73	Masuk Antri
12	8.29	1.27	5.10	14.66	Masuk Antri
13	12	1.14	1.10	14.24	Masuk Antri
14	15.78	2.38			Nilip Baran
15	16.12	0.81			Nilip Baran
16	17.36	2.62			
17	17.93	2.87			
18	18.07	1.42			
19	18.09	1.14			
20	19.35	1.48			
21	20.25	4.92			
22	22.24	3.78			
23	24.59	1.13			
24	25.37	0.3			
25	25.9	1.28			

Gambar 14. Detail simulasi per pelanggan

Pukul	Pelang	Keterangan
14.68	12	Masuk Antrian
14.58	4	Keluar superm
14.48	6	Dilayani
14.48	10	Selesai Dilaya
14.48	10	Ambil tllipan
14.24	13	Masuk Antrian
13.23	9	Masuk Antrian
12.73	11	Masuk Antrian
12.45	8	Masuk Antrian
12.29	10	Dilayani
12.29	4	Selesai Dilaya
12.29	4	Ambil tllipan
10.65	5	Keluar superm
10.34	4	Dilayani
10.34	5	Selesai Dilaya
10.34	5	Ambil tllipan
8.94	6	Masuk Antrian
8.68	3	Keluar superm
8.43	10	Masuk Antrian
7.9	7	Keluar superm
7.13	4	Masuk Antrian
6.91	5	Dilayani
6.91	3	Selesai Dilaya
6.91	3	Ambil tllipan
6.7	1	Keluar superm

Gambar 15. Detail simulasi per kejadian

## 7. Kesimpulan dan Saran

Kesimpulan yang dapat diambil dari implementasi simulasi ini antara lain adalah pencarian data dapat dilakukan dengan lebih cepat dibandingkan dengan pencarian data secara manual, dan adanya pengecekan dalam setiap penginputan data sesuai aturan-aturan dalam simulasi dapat membantu dalam mengurangi terjadinya kesalahan pada proses simulasi.

Dalam template simulasi supermarket ini user dapat menginputkan pola sesuai dengan yang diinginkan. Jika pola yang dipilih oleh user terasa kurang yakin, sistem menyediakan bantuan berupa grafik untuk membantu user dalam memilih distribusi.

Saran yang dapat digunakan untuk pengembangan penelitian ini adalah dengan menyediakan fasilitas yang dapat menentukan distribusi secara pasti dari data mentah yang diinputkan oleh user atau adanya penggunaan metode markov untuk memprediksi kejadian yang akan datang.

## DAFTAR PUSTAKA

- [1] Handoko, T.H. 1991. *Dasar-dasar Manajemen Produksi dan Operasi*. BPFE-Yogyakarta. pp. 33, 336-338
- [2] Law, A.M ; Kelton, W.D. 2007. *Simulation Modeling and Analysis(4<sup>th</sup> ed)*. McGraw-Hill, New York. pp.6-80.