



UNIVERSITAS  
GADJAH MADA

**Perencanaan Aplikasi Kanban Pada Proses Produksi Recorder di PT. YMMI**  
Nova Mukti Bawono, Dr.Eng. Ir. Muh Arif Wibisano S.T.,M.T.,IPM.,ASEAN Eng.  
Universitas Gadjah Mada, 2003 | Diunduh dari <http://etd.repository.ugm.ac.id/>

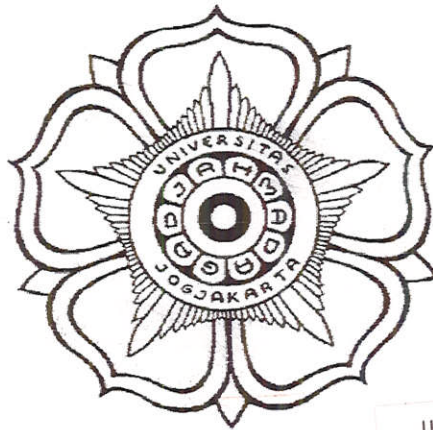
**SKRIPSI**

**TKI 532T/I/2002/2003/MAW/08.01/2003**

**PENGEMBANGAN SISTEM PENDUKUNG  
KEPUTUSAN PEMESANAN ULANG  
PADA USAHA RETAIL**

**(Studi Kasus Swalayan Mataram Mandiri)**

104



TGI00055

**Disusun oleh:**

**NOORFINA LUTHFIANY**

**98/121120/TK/22820**

**PROGRAM STUDI TEKNIK INDUSTRI**

**JURUSAN TEKNIK MESIN**

**FAKULTAS TEKNIK**

**UNIVERSITAS GADJAH MADA**

**YOGYAKARTA**

**2003**



## KATA PENGANTAR

Puji syukur ke hadirat Allah SWT, atas seluruh rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini dengan baik. Banyak hal yang harus dilalui untuk dapat menyelesaikan tugas akhir ini. Namun dengan seluruh petunjuk-Nya dan dukungan dari semua pihak maka tugas akhir ini dapat terselesaikan dengan baik.

Terima kasih kepada orang-orang terkasih yang telah membantu seluruh proses pembuatan tugas akhir ini baik yang terlibat secara langsung maupun tidak langsung, diantaranya:

1. Bapak Ir. M. Arif Wibisono, M.T. selaku dosen pembimbing atas bimbingan, arahan dan kebaikannya;
2. Bapak Ir. Alva Edy Tontowi M.Sc., Ph.D., selaku Ketua Program Studi Teknik Industri, atas dukungan dan bimbingannya;
3. Almarhum Bapak Ir. Tjahjana Adi, MSME, selaku pembimbing akademis, yang telah memberikan perhatiannya;
4. Bapak Sutarjo, selaku pengelola swalayan Mataram Mandiri, yang telah memberikan banyak masukan dan arahan;
5. Mama, Papa, beserta seluruh keluarga, atas semua dukungan doa, kasih, dan perhatiannya;
6. Teman-teman di Teknik Industri yang telah memberikan bantuan doa dan pemikiran;
7. Semua pihak yang tidak dapat penulis sebutkan satu persatu.

Dengan penulisan tugas akhir yang berjudul Pengembangan Sistem Pendukung Keputusan Pemesanan Ulang pada Usaha Retail ini, penulis berharap dapat memberikan hal yang berguna bagi perkembangan usaha retail maupun pembaca yang ingin mengembangkan topik ini lebih lanjut. Sehingga dapat menghasilkan hal yang lebih bermanfaat bagi perkembangan industri pada umumnya.



Melalui tulisan ini, penulis berharap dapat memberikan yang terbaik sehingga dapat bermanfaat bagi pembaca. Namun penulis yakin bahwa tulisan ini masih memiliki kekurangan. Oleh karena itu, penulis tetap mengharapkan kritik dan saran dari pembaca.

Yogyakarta, 29 Maret 2003

Penulis



## DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	ii
NASKAH SOAL	iii
LEMBAR PERSEMBAHAN	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiv
INTISARI	xvi
BAB I. PENDAHULUAN	1
I.1. LATAR BELAKANG MASALAH	1
I.2. PERUMUSAN MASALAH	4
I.3. TUJUAN PERANCANGAN	5
I.4. MANFAAT PERANCANGAN	5
I.5. BATASAN MASALAH	6
I.6. METODOLOGI PENELITIAN	6
I.7. SISTEMATIKA PENULISAN	7
BAB II. TINJAUAN PUSTAKA	9
II.1. PERANTARA PEMASARAN DALAM INDUSTRI	9
II.1.1. Fungsi dan Tanggung Jawab Perantara Pemasaran	10
II.1.2. Retail sebagai Perantara Pemasaran	11
II.2. PERSEDIAAN DAN PENGARUHNYA DALAM INDUSTRI	15



II.2.1.	Fungsi – Fungsi Persediaan	16
II.2.2.	Tujuan Pengendalian Persediaan	17
II.3.	SISTEM INFORMASI PADA PERUSAHAAN	18
II.3.1.	Sistem Informasi Berbasis Komputer	20
II.3.2.	Basis Data sebagai Komponen Utama Sistem Informasi Berbasis Komputer	22
II.3.3.	Sistem Manajemen Basis Data	25
II.3.4.	Microsoft Access sebagai Perangkat Lunak Basis Data	30
II.3.5.	Sistem Pendukung Keputusan	34
II.3.6.	Titik Pemesanan Kembali ( <i>Reorder Point/ROP</i> )	38
II.3.7.	Penentuan Jumlah Pemesanan Ulang	40
 <b>BAB III. METODOLOGI PENELITIAN</b>		 41
III.1.	GAMBARAN UMUM	41
III.2.	IDENTIFIKASI SISTEM INFORMASI	44
III. 2.1.	Bagian Pembelian	47
III. 2.2.	Bagian Penjualan	48
III. 2.3.	Bagian Gudang	50
III.3.	PENGEMBANGAN SISTEM PENDUKUNG KEPUTUSAN	50
III.3.1.	Pengolahan Data Persediaan	50
III.3.2.	Penentuan Kriteria Penjualan	52
III.3.3.	Penentuan Titik Pemesanan Ulang	55
III.3.4.	Penentuan Jumlah Pemesanan Ulang	57
III.3.5.	Model Penggunaan Sistem Pendukung Keputusan	58
III.3.6.	Model Matematis Penggunaan Sistem	



Pendukung Keputusan	61
<b>BAB IV. PEMBUATAN BASIS DATA DAN SISTEM</b>	
<b>PENDUKUNG KEPUTUSAN SERTA</b>	
<b>PENGGUNAANNYA</b>	65
<b>IV.1. PEMBUATAN DAN PENGGUNAAN BASIS DATA</b>	65
IV.1.1. Basis Data Pembelian	65
IV.1.1.1. Pengolahan Data Produk	66
IV.1.1.2. Pengolahan Data Supplier	69
IV.1.1.3. Pengolahan Data Transaksi Pembelian	72
IV.1.2. Basis Data Penjualan	82
<b>IV.2. PEMBUATAN SISTEM PENDUKUNG KEPUTUSAN</b>	95
IV.2.1. Pengolahan Data Persediaan	95
IV.2.2. Penentuan Laba Harian	102
IV.2.3. Sistem Pendukung Keputusan Kriteria Produk	104
IV.2.4. Sistem Pendukung Keputusan Titik Pemesanan	
Kembali	109
IV.2.5. DSS ROP	112
IV.2.6. Form Navigasi	114
<b>IV.3. PENGGUNAAN SISTEM PENDUKUNG KEPUTUSAN</b>	119
<b>BAB V. ANALISIS DAN PEMBAHASAN</b>	121
V.1. ANALISIS UMUM	121
V.2. ANALISIS BASIS DATA	121
V.3. ANALISIS SISTEM PENDUKUNG KEPUTUSAN	124
V.4. PENGUJIAN SISTEM PENDUKUNG KEPUTUSAN	127
V.4.1. Pengujian Waktu Penggunaan	127
V.4.2. Pengujian Hasil Peramalan	128



<b>BAB VI. KESIMPULAN DAN SARAN</b>	<b>131</b>
VI.1. KESIMPULAN	131
VI.2. SARAN	132
<b>DAFTAR PUSTAKA</b>	<b>xvii</b>



## DAFTAR GAMBAR

Gambar II.1.	Peran distributor sebagai saluran pemasaran	9
Gambar II.2.	Sistem informasi sederhana	19
Gambar II.3.	Sistem informasi berbasis komputer	21
Gambar II.4.	Arus perpindahan produk pada sistem <i>retail</i>	21
Gambar II.5.	Arus perpindahan uang pada sistem <i>retail</i>	21
Gambar II.6.	Sistem manajemen basis data fokus pada data	26
Gambar II.7.	<i>Database query processor</i>	28
Gambar II.8.	<i>Database report writer</i>	29
Gambar II.9.	<i>Database Form</i>	30
Gambar II.10.	Sistem Pendukung Keputusan pada Permasalahan Semi-Terstruktur	36
Gambar II.11.	Titik Pemesanan Kembali tanpa <i>Safety Stock</i>	38
Gambar II.12.	Titik Pemesanan Kembali dengan <i>Safety Stock</i>	39
Gambar III.1	Metodologi penelitian pembuatan sistem pendukung keputusan	41
Gambar III.2	Arus produk	45
Gambar III.3	Arus uang	46
Gambar III.4.	Arus informasi pada bagian pembelian	48
Gambar III.5.	Arus informasi pada bagian penjualan	49
Gambar III.6.	Arus informasi pada bagian gudang	49
Gambar III.7.	Jumlah persediaan dalam sistem	50
Gambar III.8.	Total laba harian penjualan	51
Gambar III.9.	Diagram alir penentuan kriteria penjualan produk	54
Gambar III.10	Diagram alir penentuan titik pemesanan ulang	56
Gambar III.11.	Diagram alir penentuan jumlah order	58
Gambar III.12.	Informasi yang dihasilkan oleh sistem pendukung	



keputusan	59
Gambar IV.1. Komponen basis data persediaan	65
Gambar IV.2. Komponen penyusun pengolah data produk	66
Gambar IV.3. Tabel Barang	66
Gambar IV.4. Form Input Barang	67
Gambar IV.5. Komponen penyusun pengolah data <i>supplier</i>	69
Gambar IV.6. Tabel <i>Supplier</i>	70
Gambar IV.7. Form Input <i>Supplier</i>	71
Gambar IV.8. Komponen penyusun pengolah data pembelian	72
Gambar IV.9. Komponen penyusun form <i>Purchasing</i>	72
Gambar IV.10. Tabel <i>Purchasing</i>	73
Gambar IV.11. Tabel Detail <i>Purchasing</i>	73
Gambar IV.12. Form <i>Purchasing</i>	75
Gambar IV.13. Digram Alir Penggunaan Bahasa VBA pada Form <i>Purchasing</i>	77
Gambar IV.14. Komponen <i>Report</i> Total Pembelian	78
Gambar IV.15. <i>Report</i> Total Pembelian	79
Gambar IV.16. Komponen Form Retur	80
Gambar IV.17. Form Retur	80
Gambar IV.18. Subsistem Basis Data Penjualan	82
Gambar IV.19. Komponen Form Transaksi	83
Gambar IV.20. Tabel Transaksi	83
Gambar IV.21. Tabel Detail Transaksi	84
Gambar IV.22. Form Transaksi	85
Gambar IV.23. Diagram Alir Penggunaan Bahasa VBA pada Form Detail Transaksi	86
Gambar IV.24. Diagram Alir Penggunaan Bahasa VBA pada Form Transaksi	87
Gambar IV.25. Komponen <i>Report</i> Print Kasir	89



UNIVERSITAS  
GADJAH MADA

**Perencanaan Aplikasi Kanban Pada Proses Produksi Recorder di PT. YMMI**  
Nova Mukti Bawono, Dr.Eng. Ir. Muh Arif Wibisano S.T.,M.T.,IPM.,ASEAN Eng.  
Universitas Gadjah Mada, 2003 | Diunduh dari <http://etd.repository.ugm.ac.id/>

## PENGESAHAN

Diajukan untuk memenuhi salah satu syarat  
guna memperoleh gelar Sarjana Teknik di Program Studi Teknik Industri  
Jurusan Teknik Mesin Fakultas Teknik Universitas Gadjah Mada

Disusun oleh:

**Noorfina Luthfiany**

98/121120/TK/22820

Telah disetujui untuk diuji

Dosen Pembimbing

Ir. M. Arif Wibisosno, M.T.



Gambar IV.26. <i>Report Print</i> Kasir	90
Gambar IV.27. Form Modal	92
Gambar IV.28. Komponen <i>Report</i> Total Penjualan	92
Gambar IV.29. <i>Report</i> Total Penjualan	93
Gambar IV.30. Komponen <i>Report</i> Grafik Total Penjualan	94
Gambar IV.31. Grafik Total Penjualan	94
Gambar IV.32. Komponen pengolah data persediaan	96
Gambar IV.33. Form Item	97
Gambar IV.34. Icon Filter	99
Gambar IV.35. Diagram alir prosedur <i>Apply Filter</i>	99
Gambar IV.36. Fasilitas pencarian data Microsoft Access	100
Gambar IV.37. Diagram alir prosedur <i>Find</i>	100
Gambar IV.38. Digram alir bahasa VBA pada form Item	101
Gambar IV.39. Komponen <i>report</i> Laba Harian	103
Gambar IV.40. <i>Report</i> Laba Harian	104
Gambar IV.41. Komponen Sistem Pendukung Keputusan Kriteria Produk	105
Gambar IV.42. <i>Report</i> DSS Kriteria Produk	107
Gambar IV.43. Komponen Penyusun <i>Report</i> Jumlah Pemesanan Ulang	107
Gambar IV.44. <i>Report</i> Jumlah Pemesanan Ulang	109
Gambar IV.45. Komponen sistem pendukung keputusan titik pemesanan ulang	110
Gambar IV.46. <i>Report</i> DSS ROP	111
Gambar IV.47. Report DSS Gabungan	112
Gambar IV.48. Komponen penyusun Form DSS ROP	112
Gambar IV.49. Form DSS ROP	114
Gambar IV.50. Form Menu Utama	117
Gambar IV.51. Form Menu DSS	117



## DAFTAR TABEL

Tabel	II.1.	Tipe data	33
Tabel	II.2.	Tipe data <i>number</i>	34
Tabel	III.1	Data penjualan bulan Juni	62
Tabel	III.2	Data penjualan bulan Juli	62
Tabel	III.3.	Data penjualan bulan Agustus	62
Tabel	III.4.	Data penjualan bulan September	62
Tabel	III.5.	Data penjualan bulan Oktober	63
Tabel	III.6.	Data penjualan bulan Nopember	63
Tabel	III.7.	Data persediaan akhir Nopember	63
Tabel	III.8.	Penentuan kriteria penjualan	63
Tabel	III.9.	Penentuan nilai <i>reorder point</i>	64
Tabel	III.10	Penentuan jumlah pemesanan ulang untuk bulan Desember	64
Tabel	IV.1.	<i>Query Purchasing1</i>	74
Tabel	IV.2.	<i>Query Query Detail Purchasing</i>	74
Tabel	IV.3.	<i>Query Update Stok</i>	78
Tabel	IV.4.	<i>Query Total Pembelian</i>	79
Tabel	IV.5.	<i>Query Detail Transaksi</i>	84
Tabel	IV.6.	<i>Query Kasir</i>	89
Tabel	IV.7.	<i>Query Print Kasir</i>	90
Tabel	IV.8.	<i>Query Total Penjualan</i>	93
Tabel.	IV.9.	<i>Query Purch-Dept</i>	97
Tabel	IV.10.	<i>Query Laba Harian</i>	103
Tabel	IV.11.	<i>Query Kriteria1</i>	105
Tabel	IV.12.	<i>Query Kriteria</i>	106
Tabel	IV.13.	<i>Query Krit-Produk</i>	106
Tabel	IV.14.	<i>Query Jumlah Pemesanan Ulang</i>	108



Tabel	IV.15. <i>Query ROP</i>	110
Tabel	IV.16. <i>Query ROP2</i>	111
Tabel	V.1. Pengujian aktu pemesanan ulang dengan dan tanpa Sistem pendukung keputusan	128
Tabel	V.2. Pengujain kesalahan pad kode produk 20805	129
Tabel	V.3. Perbandingan Hasil Peramalan LMA 3x3 dengan 2x2	129



## INTISARI

Sebagai salah satu perusahaan retail, produktivitas swalayan amat dipengaruhi oleh jumlah persediaan yang dimiliki. Jika swalayan mampu memenuhi permintaan konsumen dengan tepat, maka keuntungannya juga akan meningkat. Pengelolaan jumlah persediaan dipengaruhi oleh tingkat penjualan produk tersebut selama periode tertentu.

Agar dapat mengelola jumlah persediaan dengan tepat, maka diperlukan sebuah sistem yang mampu mengolah data persediaan dalam swalayan dengan baik. Sistem yang dapat mengolah data pembelian dan penjualan produk menjadi data jumlah persediaan aktual dalam swalayan. Untuk mengolah data tersebut digunakan basis data yang terintegrasi antara bagian pembelian dan penjualan.

Keuntungan dari basis data tersebut adalah dapat diperoleh informasi yang dapat digunakan dalam proses pengambilan keputusan pemesanan ulang. Berdasarkan jumlah penjualan selama periode tertentu, maka akan diperoleh tingkat penjualan setiap produk dalam swalayan tersebut.

Untuk menentukan kriteria penjualan produk, diperlukan data penjualan selama enam bulan agar dapat menghasilkan kriteria penjualan produk yang tepat. Tingkat persediaan minimal untuk melakukan pemesanan ulang ditentukan dengan menggunakan titik pemesanan kembali (*reorder point*). Sedangkan jumlah produk yang harus dipesan untuk periode berikutnya ditentukan dengan teknik *forecasting* yang dihitung dari data penjualan bulanan. Metode yang digunakan adalah rerata linear bergerak 3 x 3.

Berdasarkan informasi kriteria penjualan produk, maka dapat ditentukan produk mana yang harus dipesan ulang karena memiliki penjualan yang bagus selama periode tersebut. Batas minimal persediaan setiap produk berguna untuk menentukan waktu pemesanan ulang. Pemesanan dapat dilakukan jika persediaan telah mencapai jumlah yang sama dengan batas minimal persediaan. Jumlah produk yang harus dipesan dapat ditentukan dengan informasi dari sistem pendukung keputusan maupun kebijakan manajemen. Dari informasi tersebut, maka akan diperoleh keputusan pemesanan ulang yang baik.



## **BAB I**

### **PENDAHULUAN**

#### **1.1. LATAR BELAKANG MASALAH**

Agar mampu bersaing dengan berbagai perusahaan yang memiliki usaha yang sama, suatu perusahaan harus mampu meningkatkan prestasi kerjanya secara efektif dan efisien. Tuntutan ini akan semakin besar karena hanya perusahaan yang mampu bekerja secara efektif dan efisien yang mampu bertahan untuk merebut pangsa pasar. Diperlukan keunggulan kompetitif tertentu untuk dapat bertahan dalam persaingan bisnis antar perusahaan yang bergerak dalam usaha yang sama. Banyak hal yang harus diperbaiki untuk meningkatkan prestasi kerja tersebut. Apalagi hingga menjadikannya sebagai suatu keunggulan kompetitif dan mampu bersaing dengan perusahaan lain. Tidak mungkin suatu perusahaan tetap bertahan jika tidak mampu meningkatkan kinerjanya.

Satu diantaranya adalah dengan memperbaiki sistem persediaan. Pengaturan jumlah persediaan yang dimiliki oleh suatu perusahaan bukan lagi merupakan permasalahan administrasi biasa. Karena kini pengaturan jumlah persediaan secara efektif akan memberikan kontribusi bagi keberhasilan perusahaan secara keseluruhan (Smith, 1989). Dengan perhitungan dan perencanaan yang tepat, maka perusahaan dapat menentukan jumlah persediaan yang dibutuhkan demi kelangsungan usaha, peningkatan efektivitas gudang serta pengurangan biaya persediaan.

Hal ini akan sangat berpengaruh pada perusahaan terutama jika pengambilan keputusan mengenai permasalahan persediaan kurang tepat. Proses usaha akan terhambat dan jumlah biaya yang harus dikeluarkan perusahaan menjadi tidak efisien. Yaitu biaya untuk menutupi kerugian akibat proses usaha yang terhambat karena jumlah persediaan kurang atau bahkan untuk menyimpan persediaan yang berlebih. Banyak hal yang berpengaruh pada proses pengambilan keputusan tersebut. Diantaranya adalah ketersediaan data persediaan yang cukup dan akurat sehingga mampu membantu proses pengambilan keputusan dan menghasilkan keputusan yang tepat.



Ketersediaan data persediaan dapat dipenuhi oleh setiap bagian perusahaan yang terkait dengan permasalahan tersebut, namun terkadang data persediaan yang cukup lengkap tidak mampu memberikan dukungan pada proses pengambilan keputusan karena data tersebut tidak terintegrasi dengan baik serta bukan merupakan data persediaan yang terbaru. Kerancuan seringkali muncul ketika perusahaan harus mengambil keputusan dari sejumlah data yang disediakan oleh beberapa departemen secara terpisah. Data yang saling tumpang tindih serta tidak saling melengkapi tentunya akan menyulitkan kinerja perusahaan dalam upayanya mengambil keputusan bagi kelangsungan perusahaan. Hal ini dapat timbul akibat kesalahan dalam memasukkan data, dimana seharusnya tidak terjadi proses memasukkan data yang sama oleh departemen yang berbeda untuk kepentingan yang berbeda. Kesalahan ini sering terjadi pada sistem pemasukan data yang tidak terintegrasi. Hasilnya tentu saja adalah data keluaran yang tidak terintegrasi, tidak efektif dan tidak efisien.

Untuk itu diperlukan suatu sistem informasi yang baik untuk mendukung sistem pengambilan keputusan. Suatu sistem yang mampu memberikan data aktual terintegrasi dari setiap departemen yang berkaitan dalam perusahaan untuk membantu proses pengambilan keputusan. Dengan informasi tersebut, perusahaan diharapkan mampu menjawab kebutuhan pasar yang ada. Sehingga, kemampuan perusahaan untuk bersaing dalam bisnis dan merebut pasar yang lebih luas dapat berhasil. Kemampuan perusahaan menjawab tantangan pasar dalam waktu yang tepat, tentu saja akan meningkatkan keunggulan kompetitif perusahaan. Dan secara tidak langsung perusahaan akan diuntungkan melalui proses tersebut.

Pada dasarnya sistem pendukung keputusan persediaan perusahaan, tersusun atas sistem basis data terintegrasi yang mampu menerima masukan berupa data dari departemen yang secara langsung berhubungan dengan penambahan maupun pengurangan jumlah persediaan dan memberikan keluaran berupa informasi mengenai data aktual jumlah persediaan. Sistem ini diharapkan mampu memberikan dukungan kepada perusahaan dalam menentukan keputusan yang tepat sesuai kebutuhan pasar dan kemampuan perusahaan memenuhi

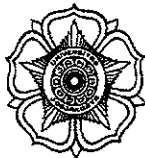


UNIVERSITAS  
GADJAH MADA

**Perencanaan Aplikasi Kanban Pada Proses Produksi Recorder di PT. YMMI**  
Nova Mukti Bawono, Dr.Eng. Ir. Muh Arif Wibisano S.T.,M.T.,IPM.,ASEAN Eng.  
Universitas Gadjah Mada, 2003 | Diunduh dari <http://etd.repository.ugm.ac.id/>

kebutuhan pasar. Dengan begitu, perusahaan tidak dirugikan karena proses produksi yang terhambat akibat sistem persediaan yang tidak efisien.

Pada perusahaan *retail* atau pengecer, sistem basis data persediaan menjadi hal penting karena bentuk usahanya adalah memberikan pelayanan pada konsumen dalam bentuk penjualan barang-barang siap pakai. Sebagai distributor



DEPARTEMEN PENDIDIKAN DAN KEBUDAYAAN  
UNIVERSITAS GADJAH MADA FAKULTAS TEKNIK  
**JURUSAN TEKNIK MESIN**  
Jl. Grafika 2 Telp. (0374) 90-2181, 2183, 2184 Fax (0274) 90-2180 Yogyakarta 55284

---

## NASKAH SOAL TUGAS AKHIR

Mata Kuliah : **Perancangan Sistem Informasi**

Nama Mahasiswa : **Noorfina Luthfiyany**

Nomor Mahasiswa: **98 / 121120 / TK / 22820**

Nomor persoalan : **TKI 532T/I/2002/2003/M.A.W./08.01/2003**

**Pengembangan Sistem Pendukung Keputusan  
Pemesanan Ulang pada Usaha Retail  
(Studi Kasus Swalayan Mataram Mandiri)**

Yogyakarta, 28 Maret 2003

Dosen Pembimbing,



## **I.2. PERUMUSAN MASALAH**

Permasalahan utama dalam penulisan ini adalah mengetahui produk yang harus tersedia untuk memenuhi kebutuhan konsumen. Untuk itu diperlukan informasi mengenai jenis, waktu, dan jumlah produk yang harus dipesan untuk memenuhi kebutuhan konsumen. Informasi tersebut diperoleh berdasarkan data penjualan selama periode tertentu dan data jumlah persediaan yang dimiliki. Kemudian dari permasalahan tersebut, dikembangkan suatu sistem basis data persediaan yang terintegrasi antar bagian yang terkait dengan masalah persediaan dalam perusahaan untuk mengetahui jumlah persediaan aktual sehingga dimungkinkan untuk membentuk sistem pendukung keputusan pemesanan ulang setiap produk dalam perusahaan.

Sistem basis data akan dikembangkan dengan menggunakan perangkat lunak Microsoft Access dan terdiri atas subsistem pembelian dan penjualan. Basis data dikembangkan untuk menyatukan kedua bagian tersebut hingga diperoleh suatu kesatuan data jumlah persediaan. Kemudian data tersebut diolah hingga menghasilkan informasi yang dapat mendukung proses pengambilan keputusan pemesanan ulang. Informasi tersebut berupa kriteria penjualan setiap produk pada periode tertentu. Dengan pengelompokan jenis produk berdasar pada kriteria penjualan selama periode tersebut, dapat diperoleh suatu informasi apakah suatu produk akan mengalami pemesanan ulang, tetap ditaruh di rak penjualan hingga periode berikutnya untuk dievaluasi kembali, atau tidak dibeli lagi pada periode-periode berikutnya, karena tidak memiliki tingkat penjualan yang diinginkan. Setelah itu dapat ditentukan jumlah produk yang harus dipesan untuk memenuhi kebutuhan konsumen periode berikutnya.

Permasalahan tersebut akan menjadi fokus utama pembahasan. Secara rinci, permasalahan tersebut dapat dijelaskan sebagai berikut:

1. Bagaimana mengembangkan suatu basis data terintegrasi antara bagian pembelian dan penjualan sehingga mampu memberikan informasi terbaru mengenai perubahan jumlah persediaan dari transaksi pembelian, penjualan, dan retur.



2. Bagaimana mengembangkan suatu sistem pendukung keputusan persediaan dari data penjualan dan persediaan yang mampu memberikan informasi waktu pemesanan ulang.
3. Bagaimana mengolah data hasil penjualan selama periode tertentu, menjadi informasi kriteria penjualan dan jumlah produk untuk pemesanan ulang periode berikutnya, sehingga dihasilkan keputusan yang tepat bagi perusahaan.

### **I.3. TUJUAN PERANCANGAN**

Berdasarkan perumusan masalah tersebut, maka tujuan perancangan ini dapat dijelaskan sebagai berikut:

1. Mengembangkan suatu sistem basis data persediaan yang terintegrasi sehingga diperoleh informasi jumlah persediaan terbaru pada perusahaan.
2. Mengembangkan sistem basis data persediaan yang mampu mengolah data bagian penjualan dan persediaan menjadi informasi jenis produk, estimasi batas persediaan yang harus tersedia, serta jumlah produk untuk pembelian periode berikutnya.

### **I.4. MANFAAT PERANCANGAN**

Manfaat yang dapat diperoleh dari hasil perancangan ini dapat dijelaskan sebagai berikut:

1. Memberikan informasi bagi perusahaan mengenai tingkat persediaan tiap produk setiap waktu.
2. Memberikan informasi pada perusahaan dalam menentukan keputusan mengenai kriteria penjualan setiap produk pada suatu periode.
3. Memberikan tambahan informasi pada perusahaan dalam menentukan keputusan mengenai jenis, waktu, dan jumlah produk yang harus dibeli untuk persediaan periode berikutnya.
4. Memberikan alternatif dalam meningkatkan kinerja perusahaan, terutama dengan meningkatkan efisiensi dalam menentukan keputusan persediaan berikutnya.



5. Memberikan gambaran mengenai pengembangan sistem pendukung keputusan pemesanan ulang yang didukung oleh sistem informasi persediaan pada swalayan.

#### **I.5. BATASAN MASALAH**

Agar perancangan ini lebih terarah dan agar pembahasan masalah yang telah ditetapkan dapat mencapai sasaran yang diinginkan, maka perlu adanya batasan masalah sebagai berikut:

1. Objek pembahasan adalah swalayan yang memiliki pangsa pasar dengan tingkat penjualan rata-rata 400 transaksi per hari.
2. Perancangan basis data persediaan dilakukan dengan menggunakan Microsoft Access 2000 dan Visual Basic for Application yang terdapat dalam Microsoft Access 2000.
3. Perancangan basis data dilakukan pada produk-produk dengan asumsi memiliki harga yang tetap selama periode basis data.
4. Pembahasan ditekankan pada pengembangan pengambilan keputusan pemesanan ulang yang didukung basis data persediaan terintegrasi dan tidak akan membahas bagian keuangan pada perusahaan.

#### **I.6. METODOLOGI PENELITIAN**

Untuk mengembangkan sistem pendukung keputusan pemesanan ulang, langkah awal yang harus dilakukan adalah melakukan identifikasi sistem informasi persediaan pada sumber data. Sumber data adalah swalayan Mataram Mandiri. Sumber data primer adalah dengan mengamati sistem yang berlaku pada swalayan tersebut dan melakukan informasi dari pihak manajemen swalayan. Sumber data sekunder merupakan informasi tambahan dari swalayan berupa data penjualan dan studi pustaka sebagai referensi. Dari data tersebut dikembangkan suatu model sistem informasi persediaan yang disesuaikan dengan sistem sebenarnya.

Model sistem informasi persediaan tersebut dikembangkan menjadi basis data persediaan yang mendukung pengembangan sistem pendukung keputusan



pemesanan ulang produk-produk yang dijual di swalayan tersebut. Model sistem informasi persediaan menjelaskan alur informasi produk dan uang yang berlaku di swalayan tersebut. Model tersebut menjelaskan proses input dan pengolahan data produk dan uang pada bagian pembelian dan penjualan yang berpengaruh pada perubahan jumlah persediaan dalam swalayan. Setelah model sistem informasi tersebut dikembangkan, langkah selanjutnya adalah mengembangkan basis data persediaan sesuai model sistem informasi tersebut. Basis data yang dikembangkan harus mendukung pengembangan sistem pendukung keputusan pemesanan ulang. Basis data dikembangkan dengan menggunakan Microsoft Access 2000 dan Visual Basic for Application yang telah terdapat di dalamnya.

Basis data tersebut mengolah data dari setiap sub sistem yang berpengaruh dalam perubahan jumlah persediaan secara tepat. Data dari sub sistem diolah sehingga menghasilkan data jumlah persediaan terakhir dalam swalayan. Dengan informasi tersebut dan dari sistem pendukung keputusan, maka akan diketahui produk yang menjadi kebutuhan konsumen sehingga swalayan dapat menentukan jumlah persediaan yang harus dimiliki. Informasi tersebut diolah lebih lanjut hingga menghasilkan informasi kriteria penjualan setiap produk, titik pemesanan kembali, serta estimasi jumlah pembelian produk berikutnya. Penentuan jumlah produk yang harus dipesan dilakukan dengan teknik prakiraan (*forecasting*).

Informasi tersebut merupakan bagian dari sistem pendukung keputusan yang kemudian dijadikan bahan evaluasi oleh bagian pembelian dalam menentukan pemesanan ulang. Dengan sistem pendukung keputusan tersebut, diharapkan bagian pembelian dapat menghasilkan keputusan yang tepat mengenai pemesanan ulang produk-produk pada periode berikutnya.

## **I.7. SISTEMATIKA PENULISAN**

Untuk memperoleh gambaran yang sistematis dari pengembangan sistem pendukung keputusan pemesanan ulang ini maka penulisan ini disusun sebagai berikut:



## **BAB I      Pendahuluan**

Bab ini menjelaskan mengenai latar belakang permasalahan, perumusan masalah, tujuan perancangan, manfaat perancangan, batasan masalah, metodologi penelitian, dan sistematika penulisan.

## **BAB II     Tinjauan Pustaka**

Bab ini menguraikan perkembangan *retail* dalam dunia industri, fungsi persediaan dan aplikasinya dalam dunia industri, sistem informasi, basis data, penggunaan Microsoft Access sebagai sarana perancangan, dan sistem pendukung keputusan secara umum.

## **BAB III    Metodologi Penelitian**

Bab ini memberikan langkah-langkah yang dilakukan dalam pengembangan basis data persediaan dan pengolahan data untuk sistem pendukung keputusan dengan menggunakan model sistem dan model matematis.

## **Bab IV     Pengembangan Basis Data dan Sistem Pendukung Keputusan serta Penggunaannya**

Bab ini menggambarkan pembuatan basis data dan sistem pendukung keputusan pemesanan ulang berdasarkan model sistem informasi serta penggunaannya hingga menghasilkan informasi yang dibutuhkan.

## **Bab V      Analisis dan Pembahasan**

Bab ini akan membahas penggunaan sistem pendukung keputusan ditinjau dari segi kelebihan dan kekurangannya jika diterapkan pada sistem nyata.

## **Bab VI     Kesimpulan dan Saran**

Sebagai bagian penutup dari pembahasan, bab ini akan memaparkan tentang kesimpulan dari perancangan yang telah dilakukan dan saran-saran bagi perancangan selanjutnya.

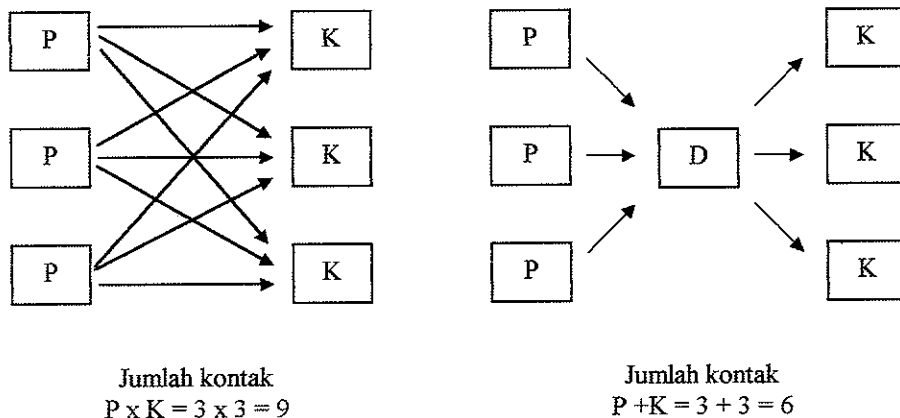


## BAB II

### TINJAUAN PUSTAKA

#### II.1. PERANTARA PEMASARAN DALAM INDUSTRI

Dalam dunia industri, dibutuhkan saluran-saluran pemasaran yang berfungsi untuk memasarkan produk agar sampai ke tangan konsumen. Saluran pemasaran adalah serangkaian organisasi yang saling tergantung yang terlibat dalam proses untuk menjadikan suatu produk atau jasa hingga siap untuk digunakan atau dikonsumsi (Kotler, 1997). Perantara pemasaran banyak digunakan oleh produsen karena efisiensi yang dimiliki oleh perantara pemasaran dalam membuat barang-barang agar dapat tersedia secara luas dan mudah diperoleh di pasaran. Hubungan, pengalaman, spesialisasi, dan skala operasi yang biasanya lebih banyak daripada yang dapat dilakukan oleh produsen itu sendiri merupakan alasan mengapa perantara pemasaran lebih banyak digunakan oleh produsen. Hal ini terutama untuk menghubungkan ketidaksesuaian antara berbagai barang dan jasa yang dihasilkan produsen dan berbagai macam produk yang diinginkan oleh konsumen. Ketidaksesuaian ini muncul karena produsen biasanya menghasilkan sejumlah besar produk dengan keragaman terbatas, sedang konsumen biasanya hanya menginginkan jumlah terbatas dari berbagai macam produk.



P: Produsen                      D: Distributor                      K: Konsumen

Gambar II.1. Peran distributor sebagai saluran pemasaran



### **II.1.1. Fungsi dan Tanggung Jawab Perantara Pemasaran**

Sebuah saluran pemasaran dalam melaksanakan fungsinya menyampaikan produk dari produsen ke konsumen, juga menjalankan fungsi lain yaitu:

- Informasi: pengumpulan dan penyebaran informasi riset pemasaran mengenai pelanggan, pesaing, pelaku, dan kekuatan lain yang ada saat ini maupun yang potensial dalam lingkungan pemasaran.
- Promosi: pengembangan dan penyebaran informasi persuasif yang dirancang untuk menarik pelanggan pada penawaran tersebut.
- Negosiasi: usaha untuk mencapai persetujuan akhir mengenai harga dan syarat lain sehingga transfer kepemilikan dapat dilakukan.
- Pemesanan: komunikasi dari anggota saluran pemasaran ke produsen mengenai minat untuk membeli.
- Pembiayaan: perolehan dan pengalokasian dana yang dibutuhkan untuk membiayai persediaan pada berbagai tingkat saluran pemasaran.
- Pengambilan Resiko: penanggungungan resiko yang berhubungan dengan pelaksanaan fungsi saluran pemasaran tersebut.
- Pemilikan Fisik: kesinambungan penyimpanan dan pergerakan produk fisik dari bahan mentah sampai ke pelanggan akhir.
- Pembayaran: pembeli membayar tagihannya ke penjual lewat bank dan institusi keuangan lainnya.
- Hak Milik: transfer kepemilikan sebenarnya dari satu organisasi atau orang ke organisasi atau orang lainnya (Kotler, 1997).

Fungsi-fungsi yang ditimbulkan oleh saluran pemasaran tersebut tentunya juga akan menimbulkan biaya tertentu yang pada akhirnya dapat meningkatkan harga produk. Jika produsen melaksanakan fungsi-fungsi tersebut tanpa menggunakan perantara pemasaran, bisa dipastikan harga produk hingga ke tangan konsumen akan lebih tinggi. Namun bila beberapa fungsi dialihkan ke perantara pemasaran, biaya dan harga yang harus ditanggung produsen akan menjadi lebih rendah. Tentu saja perantara pemasaran akan menimbulkan biaya tersendiri atas usaha yang telah dilakukan. Tetapi jika perantara pemasaran dapat



bekerja dengan lebih efisien daripada produsen, harga yang dibebankan pada konsumen seharusnya akan menjadi lebih rendah.

Perantara pemasaran dan produsen memiliki tanggung jawab dan persyaratan yang harus dipenuhi jika ingin membina suatu hubungan yang baik. Hal tersebut adalah: kebijakan harga, syarat penjualan, hak teritorial distributor, serta jasa dan tanggung jawab kedua pihak. Kebijakan harga mewajibkan produsen untuk menetapkan suatu daftar harga dan rencana diskon yang dianggap adil dan memadai oleh perantara. Syarat penjualan mengacu pada syarat pembayaran dan garansi produsen. Sebagian besar produsen memberikan diskon tunai pada distributor yang membayar lebih cepat. Produsen juga dapat memberikan jaminan pada distributor terhadap kerusakan barang atau penurunan harga. Garansi terhadap penurunan harga akan mendorong distributor untuk membeli produk dalam jumlah yang besar. Hak teritorial distributor yang berarti pengakuan penuh dari produsen untuk menjual produknya pada suatu wilayah dengan syarat tertentu. Sedang jasa dan tanggung jawab kedua pihak harus dinyatakan dengan cermat, terutama untuk saluran agen eksklusif dan waralaba. Biasanya pada bisnis waralaba terjadi dalam hal dukungan promosi, sistem pembukuan, teknis dari produsen, sedangkan dari sisi distributor, harus memenuhi standard perusahaan yang telah ditetapkan (Kotler, 1997).

### II.1.2. Retail sebagai Perantara Pemasaran

Salah satu diantara saluran-saluran pemasaran tersebut adalah *retail*. *Retailing* atau penjualan eceran didefinisikan sebagai semua kegiatan yang melibatkan penjualan barang atau jasa secara langsung kepada konsumen akhir untuk penggunaan pribadi, bukan bisnis. Sedang *retailer* atau pengecer didefinisikan sebagai usaha bisnis yang volume penjualan utamanya berasal dari penjualan eceran (Kotler, 1997).

Terdapat berbagai jenis *retailer*, diantaranya adalah pengecer toko. Pengecer toko dapat dikategorikan menjadi delapan yaitu: toko khusus, toko serba ada, swalayan, toko kelontong, toko diskon, pengecer potongan harga, toko super, dan ruang pameran katalog.



Toko khusus (*specialty stores*), menjual satu lini produk yang terbatas dengan macam barang yang cukup banyak dalam lini tersebut. Contohnya adalah toko pakaian, toko bunga, dan toko buku. Suatu toko sepatu merupakan contoh toko dengan satu lini sedang toko sepatu wanita merupakan toko dengan lini terbatas, toko sepatu wanita pesanan merupakan toko sangat khusus.

Toko serba ada (*department stores*), menjual beberapa lini produk, yang masing-masing dijalankan sebagai toko tersendiri yang dikelola oleh para spesialis. Toko ini biasanya menjual pakaian dan perlengkapan rumah tangga.

Swalayan (*supermarket*), menganut sistem yang menekankan pada operasi yang relatif besar, volume produk tinggi, margin rendah, biaya rendah. Swalayan memiliki ukuran yang relatif besar dan dirancang untuk memenuhi kebutuhan konsumen seluruhnya dengan menjual produk yang bermacam-macam dalam jumlah besar. Produk yang dijual umumnya adalah produk kebutuhan sehari-hari dan barang-barang konsumsi. Toko ini memperoleh keuntungan kira-kira 1 persen dari penjualan dan 10 persen dari modal. Dengan sistem swalayan, konsumen dapat melakukan proses memilih, membandingkan, dan menemukan harga sendiri untuk menghemat uang tanpa perlu dibantu wiraniaga.

Toko kelontong (*convenience store*), merupakan toko relatif kecil yang terletak di daerah pemukiman dan memiliki jam buka yang relatif panjang. Toko ini menjual lini produk *convenience* yang terbatas dan memiliki tingkat perputaran yang tinggi. Jam buka yang panjang, disebabkan oleh konsumen yang hanya membeli produk pelengkap. Toko ini memenuhi kebutuhan konsumen yang penting, dan konsumen bersedia membayar harga yang lebih tinggi untuk memperoleh kenyamanannya.

Toko diskon (*discount stores*), memberikan potongan harga pada produk-produk standar yang dijual dengan harga lebih murah daripada pedagang biasa dengan cara mengambil margin laba sedikit dan volume yang tinggi.

Pengecer potongan harga (*off-price retailers*), membeli dengan harga yang lebih rendah daripada harga grosir dan menetapkan harga pada konsumen lebih rendah daripada harga eceran. Toko ini cenderung menjual koleksi produk yang berkualitas tinggi yang berubah-ubah dan tidak stabil dan sering merupakan



barang sisa, berlebih, dan tidak reguler, yang diperoleh dengan harga yang lebih rendah daripada pengecer lain. Ada tiga jenis utama pengecer potongan harga, yaitu: toko pabrik (*factory outlets*) yang dimiliki dan dioperasikan oleh produsen, pengecer potongan harga independen (*independent off-price retailers*) yang dimiliki dan dijalankan oleh pengusaha atau divisi dari perusahaan pengecer besar, dan klub gudang/grosir (*warehouse/wholesale clubs*). Klub gudang/grosir, menjual pilihan terbatas dari produk makanan bermerk, peralatan rumah tangga, pakaian, dan produk lain dengan diskon besar bagi anggota yang membayar iuran tahunan tertentu. Klub ini melayani usaha kecil, lembaga pemerintah, organisasi nirlaba, dan perusahaan besar.

Toko super (*superstore*), ukurannya lebih besar dibanding swalayan biasa dan diarahkan untuk memenuhi kebutuhan konsumen seluruhnya. Variasi dari toko super adalah toko kombinasi (*combination stores*), dan pasar hiper (*hypermarkets*)

Ruang pameran katalog (*catalogue showrooms*) merupakan suatu bentuk toko yang menawarkan produk yang dijual dalam katalog dan menerapkan prinsip potongan harga pada produk bermerk yang cepat laku serta berharga tinggi (Kotler, 1997).

Dengan makin banyaknya bentuk pengecer toko, pengecer makin berusaha meningkatkan pelayanannya. Berdasarkan jasa yang ditawarkan, *retailer* dapat dibedakan sebagai berikut:

- Eceran swalayan (*self-service retailing*), konsumen melakukan proses memilih, membandingkan, dan menemukan harga sendiri untuk menghemat uang.
- Eceran swapilih (*self-selection retailing*), konsumen dapat mendapatkan produk yang diinginkan sendiri. Transaksi diselesaikan dengan mendapatkan seorang wiraniaga yang akan memberikan nota pembelian atas produk tersebut.
- Eceran dengan pelayanan terbatas (*limited-service retailing*), konsumen akan mendapatkan pelayanan lebih karena produk yang dijual lebih



banyak serta mendapatkan pelayanan tambahan dengan menerima kredit dan pengembalian produk dari konsumen.

- Eceran dengan pelayanan penuh (*full-service retailing*), konsumen mendapat pelayanan penuh dalam memilih-membandingkan-menemukan produk (Kotler, 1997).

Dalam meningkatkan pelayanannya pada konsumen serta untuk menghemat jumlah biaya yang harus dikeluarkan oleh *retailer*, diperlukan peran teknologi yang tepat. Teknologi berfungsi besar dalam perangkat persaingan. *Retailer* menggunakan komputer untuk proses peramalan yang lebih baik, pengendalian biaya persediaan, bahkan penjualan produk dengan *web*.

Pada sistem *retail*, bisnis yang dilakukan adalah sebagai distributor akhir berbagai macam produk. Untuk mendapatkan konsumen sebanyak-banyaknya diperlukan pemikiran khusus dalam menentukan jenis produk yang akan dijual. Jenis produk yang dijual harus sesuai dengan harapan pasar. *Retailer* harus menentukan volume dan jenis produk yang akan dijual. Baik berdasarkan kualitas, dan pilihan yang beredar di pasar.

Masalah yang akan dihadapi *retailer* adalah menentukan campuran produk dan tingkat mutunya. Sebab akan selalu ada pesaing yang memiliki campuran dan tingkat mutu yang sejenis. Hal ini penting terutama untuk menguasai pasar lokal karena konsumen amat sensitif atas pilihan produknya. Jika produk yang diinginkan tidak tersedia pada *retail* yang dituju, umumnya konsumen yang setia pada produk tersebut akan mencari produk tersebut pada *retail* lain yang menyediakannya.

Dalam menentukan pilihan produk, terdapat beberapa strategi campuran produk bagi *retailer*:

1. menyediakan sejumlah merk nasional eksklusif yang tidak disediakan oleh *retail* pesaing.
2. menjual produk-produk yang sebagian besar merupakan merk sendiri.
3. menyelenggarakan peragaan-peragaan produk yang terkenal.



4. mengikutsertakan produk sampingan atau mengganti sebagian barang-barang yang dijual secara periodis.
5. menjadi yang pertama dalam menjual produk terbaru.
6. menawarkan produk-produk yang spesifikasinya disesuaikan dengan selera konsumen.
7. menawarkan bauran yang memiliki konsumen yang sangat sempit (Kotler, 1997).

Hal lain yang harus diperhatikan pada usaha *retail* adalah keputusan harga produk. Harga yang diberikan oleh *retailer* merupakan gambaran kualitas produk yang dijual dan pelayanan yang diberikan. Biaya produk merupakan dasar penetapan harga. Setelah itu, harga ditetapkan oleh *retailer* setelah mendapatkan penambahan nilai atas pelayanan dari *retail* pada konsumen.

Berdasarkan batasan masalah, maka pembahasan selanjutnya akan menitikberatkan pada sistem pengecer toko bentuk swalayan dengan bentuk layanan *self-service retailing*.

## II.2. PERSEDIAAN DAN PENGARUHNYA DALAM INDUSTRI

Perkembangan industri yang makin pesat membutuhkan kesiapan khusus dari para pelaku industri agar tetap mampu bersaing dalam memperebutkan pangsa pasar yang tersedia. Terlebih lagi jika pangsa pasar yang tersedia diperebutkan oleh berbagai kalangan pelaku industri. Dibutuhkan kesiapan khusus dari masing-masing pelaku industri agar dapat tetap mampu mempertahankan pangsa pasar yang telah dikuasai.

Salah satu hal yang harus diperhatikan adalah penanganan permasalahan persediaan. Karena secara tidak langsung, persediaan akan mempengaruhi kelancaran produksi suatu perusahaan. Pengendalian jumlah persediaan merupakan hal utama yang harus diperhatikan. Kurangnya perhatian pada masalah pengendalian jumlah persediaan merupakan penyebab awal dari kegagalan dalam usaha (Smith, 1989).



Persediaan didefinisikan sebagai jumlah barang tersedia. Secara umum persediaan dapat diartikan sebagai sumber daya tidak bergerak yang memiliki nilai ekonomis. Sedang pengendalian persediaan didefinisikan sebagai kegiatan atau teknik-teknik untuk mempertahankan jumlah persediaan suatu produk pada tingkat tertentu (Smith, 1989). Peran utama persediaan adalah sebagai pendukung sistem produksi dan distribusi untuk mencapai tingkat efisiensi yang lebih tinggi. Peran lainnya adalah sebagai pelindung dalam menangani permasalahan peningkatan harga produk dan fluktuasi jumlah permintaan (Smith, 1989).

### II.2.1. Fungsi – Fungsi Persediaan

Persediaan dapat dikategorikan sesuai fungsinya sebagai berikut:

1. *Lot size inventories* – Fungsi persediaan ini timbul karena adanya fungsi pelengkap dalam ekonomi. Secara ekonomis akan lebih mudah jika memiliki jumlah persediaan yang besar atau paling tidak, melebihi jumlah permintaan. Karena akan menghemat biaya *setup*, biaya persiapan produksi atau biaya pesan, biaya transportasi, serta mempengaruhi besarnya potongan harga dari vendor.
2. *Safety stocks* – Pengendalian persediaan merupakan subjek dari berbagai ketidakpastian yang muncul selama proses produksi. Jumlah permintaan konsumen diperkirakan dengan suatu sistem perkiraan yang disertai dengan perkiraan potensi kesalahan. Contohnya adalah *lead time* yang lebih lama dibandingkan waktu yang telah diperkirakan sebelumnya. Bahkan jumlah persediaan yang menjadi pelengkap juga merupakan subjek dalam menghitung potensi kesalahan. *Safety stock* mencegah kemungkinan kegagalan dalam memenuhi jumlah permintaan konsumen atau menyelesaikan proses produksi sesuai jadwal yang telah ditentukan walaupun dipengaruhi oleh berbagai ketidakpastian.
3. *Anticipation inventories* – Persediaan dibangun untuk mengantisipasi pengurangan jumlah suplai, peningkatan jumlah permintaan, atau peningkatan harga. Untuk mempertahankan kelangsungan jumlah suplai



- pada konsumen, suatu perusahaan dapat membangun persediaan untuk menghadapi kemungkinan terjadinya pemogokan atau penutupan pabrik.
4. *Pipeline inventories* – Persediaan dapat diartikan sebagai serangkaian *stock points* (lokasi penyimpanan persediaan) dengan aliran persediaan di dalamnya. Pengendalian persediaan berupa pengendalian aliran persediaan dan oleh karenanya volume persediaan akan terakumulasi pada *stock points*. Jika aliran melibatkan suatu perubahan fisik pada produk seperti pemanasan suatu bagian atau perakitan beberapa bagian persediaan dalam aliran tersebut maka aliran itu disebut sebagai *work in process*. Jika produk tidak mengalami perubahan fisik dan hanya dipindahkan dari suatu *stock point* ke *stock point* lainnya, maka persediaan tersebut disebut sebagai *transportation inventory*. Jumlah *work in process* dan *transportation inventory* adalah *pipeline inventory*. *Pipeline inventory* adalah bagian dari investasi total perusahaan di bidang persediaan sehingga harus dapat dikendalikan.
  5. *Excess inventory* – Ketika kelangsungan persediaan tidak lagi disesuaikan dengan kondisi yang tidak sesuai dengan perubahan yang terjadi maupun kemunduran fisik maka persediaan berada dalam jumlah yang berlebih. Sumber yang mungkin adalah penjualan sisa produksi maupun produk yang mengalami pengerjaan ulang, sesuai spesifikasi beberapa produk aktif (Smith, 1989).

### II.2.2. Tujuan Pengendalian Persediaan

Pada beberapa departemen yang berbeda, fungsi dan tujuan pengendalian persediaan akan berbeda pula. Hal tersebut dapat dijelaskan sebagai berikut:

1. Pemasaran – berusaha untuk memenuhi kebutuhan konsumen sesegera mungkin sehingga jika dimungkinkan, persediaan harus tersedia dalam jumlah yang besar.
2. Produksi – berusaha menjalankan sistem yang efisien. Sehingga cenderung untuk memproduksi dalam skala besar untuk mengurangi *setup* mesin yang tentu saja memerlukan persediaan dalam jumlah yang besar. Selain itu,



bagian produksi juga menginginkan jumlah persediaan yang cukup agar tidak terjadi kemacetan produk akibat kurangnya persediaan.

3. Pembelian – berusaha menjalankan sistem yang efisien. Lebih memilih menjalankan beberapa produksi skala besar daripada banyak produksi skala kecil. Bagian pembelian juga menginginkan sejumlah persediaan yang mencukupi untuk menghindari kekurangan persediaan.
4. Keuangan – berusaha meminimasi jumlah investasi untuk persediaan, karena akan berpengaruh pada nilai pengembalian aset perusahaan.
5. Personalia – berusaha menciptakan sejumlah persediaan sebagai antisipasi agar jika suatu saat terjadi kemunduran ekonomi dapat lebih meningkatkan stabilisasi perusahaan dan menghindari pemecatan pegawai.
6. Teknik – lebih menyukai jumlah persediaan yang kecil karena pada pergantian teknik tidak akan berpengaruh besar pada perusahaan karena penundaan yang timbul akibat pengosongan sejumlah besar persediaan dari rancangan terdahulu (Smith, 1989).

Dari penjelasan di atas, jelas tampak adanya ketidakcocokan tujuan persediaan bagi tiap departemen. Kesalahan dalam penanganan masalah persediaan akan berpengaruh pada seluruh bagian perusahaan. Sehingga masalah ini harus dipertimbangkan dengan baik. Tujuan-tujuan tersebut digabungkan hingga diperoleh tujuan dari tiap departemen secara optimal.

Sesuai dengan perumusan masalah, pembahasan berikutnya akan menitikberatkan pada jumlah persediaan untuk memenuhi permintaan konsumen untuk mengantisipasi kekurangan persediaan yang mungkin timbul karena terlambatnya pesanan tiba, atau meningkatnya jumlah permintaan.

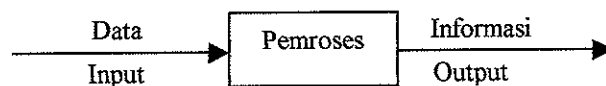
### **II.3. SISTEM INFORMASI PADA PERUSAHAAN**

Suatu perusahaan yang ingin bergerak maju agar tetap mampu bertahan dalam persaingan untuk memperebutkan pasar, perlu memiliki suatu sistem informasi yang tepat untuk membantu mencapai tujuan tersebut. Sistem informasi tersebut berfungsi sebagai sarana untuk mengolah setiap input data dari setiap



bagian perusahaan, dan menghasilkan output yang dapat digunakan untuk mengembangkan dan memajukan perusahaan.

Sistem secara terminologi didefinisikan sebagai himpunan komponen yang berbentuk interaksi atau kerjasama yang saling independen tetapi terintegrasi sebagai satu kesatuan untuk satu kegunaan pokok. Data adalah fakta kasar yang relatif sulit dimengerti oleh pengguna. Sedang informasi adalah data yang telah diinterpretasi dan diproses menjadi konteks yang berarti dan dapat dipahami penggunaannya. Sehingga sistem informasi adalah himpunan prosedur yang terorganisir dan berkaitan dengan peralatan yang menterjemahkan data menjadi informasi yang dapat digunakan untuk mengambil keputusan dan mengendalikan organisasi (McLeod, 1996).



Gambar II.2. Sistem informasi sederhana

Semua sistem informasi memiliki tiga komponen utama, yaitu: input berupa data; kemudian diproses dengan melakukan penghitungan, penggabungan unsur data, *update* data, dan lain-lain; dan diperoleh informasi sebagai outputnya. Prinsip tersebut berlaku bagi sistem informasi manual maupun komputer. Secara sederhana sistem informasi merupakan suatu metode untuk mengolah input data pada setiap organisasi hingga menjadi suatu informasi yang berguna dan dapat digunakan untuk mendukung proses pengambilan keputusan (Scott, 1999).

Tujuan sistem informasi adalah untuk memberikan informasi yang akurat pada pengguna pada waktu yang tepat. Sifat sistem informasi yang diperlukan oleh suatu organisasi terutama sangat bergantung pada jenis kegiatan yang dilaksanakan serta jenis keputusan yang dibuat oleh pengguna informasi.

Sistem informasi yang digunakan antara lain sebagai pendukung keputusan, merupakan suatu sistem informasi manajemen. Sistem informasi



manajemen didefinisikan sebagai serangkaian sub-sistem informasi yang menyeluruh dan terkoordinasi dan secara rasional terpadu dan mampu mentransformasikan data dengan serangkaian proses sehingga menjadi informasi yang berguna untuk meningkatkan produktivitas sesuai kriteria mutu yang telah ditetapkan (Scott, 1999).

Karena sifatnya yang menyeluruh, terkoordinasi, dan terintegrasi secara rasional, diperlukan suatu sistem yang mampu memudahkan proses pengolahan data tersebut hingga menjadi suatu informasi. Komputer merupakan perangkat yang mampu membantu proses tersebut. Komputer berfungsi sebagai perangkat keras yang menyediakan perangkat lunak untuk menerima input data, mengolah data dan menyimpannya dalam sistem basis data.

### II.3.1 Sistem Informasi Berbasis Komputer

Sistem informasi berbasis komputer memiliki sejumlah kemampuan di atas sistem non-komputer. Kemampuan ini telah merevolusikan proses manajemen yang menggunakan informasi yang dihasilkan sistem tersebut. Beberapa kemampuan teknis terpenting dari sistem komputer adalah:

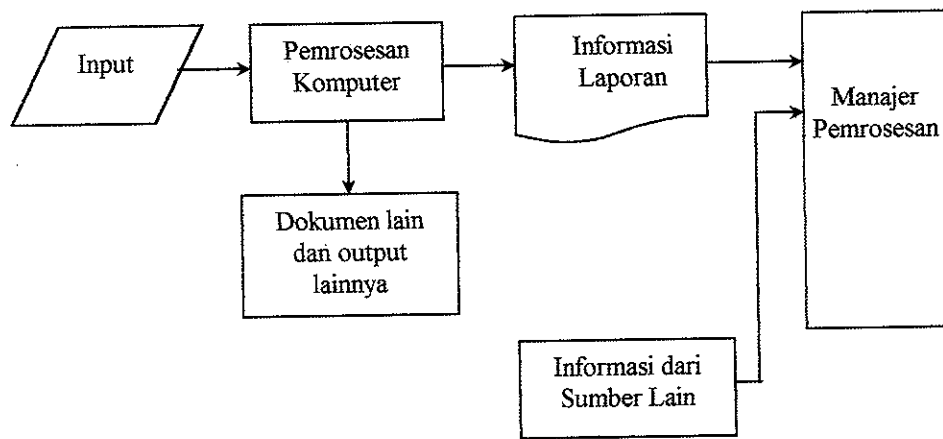
- pemrosesan transaksi secara *batch*
- pemrosesan transaksi tunggal
- pemrosesan transaksi secara *online*
- komunikasi data dan *switching* pesan
- memasukkan data jarak jauh dan *update* data
- pencarian *record* dan analisis
- pencarian *file*
- model dan algoritma keputusan
- otomasi perkantoran (Scott, 1999)

Sistem informasi berbasis komputer terdiri atas lima subsistem yaitu: sistem informasi akuntansi, sistem informasi manajemen, sistem pendukung keputusan, sistem otomasi kantor, dan sistem pakar. Semua subsistem tersebut menghasilkan informasi untuk pemecahan masalah (McLeod, 1996).

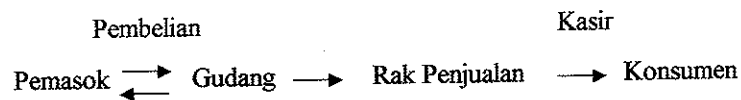


Gambar II.3. menunjukkan sistem informasi berbasis komputer yang menghasilkan informasi lebih dari sekedar yang mampu disiapkan oleh komputer. Gambar tersebut menunjukkan peran seorang manajer sebagai bagian dari sistem informasi, terutama untuk sumber informasi yang berasal dari sumber selain komputer.

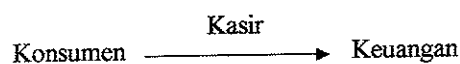
Pada sistem *retail*, arus informasi merupakan arus perpindahan produk dari pemasok, gudang, rak penjualan hingga konsumen. Digambarkan arus produk dari pemasok ke gudang merupakan arus dua arah. Hal ini karena dimungkinkan terjadinya retur atas produk cacat beli. Sedang arus uang dari konsumen ke perusahaan. Secara sederhana, sistem tersebut dijelaskan pada gambar II.4. dan II.5.



Gambar II.3. Sistem informasi berbasis komputer



Gambar II.4. Arus perpindahan produk pada sistem *retail*



Gambar II.5. Arus perpindahan uang pada sistem *retail*



### **II.3.2. Basis Data sebagai Komponen Utama Sistem Informasi Berbasis Komputer**

Tujuan utama dari sistem informasi adalah penyediaan informasi untuk berbagai tujuan kegiatan manajerial dan penyediaan informasi yang dibutuhkan dalam berbagai tingkat operasi dalam organisasi. Dalam penyediaan informasi tersebut, hal yang harus diperhatikan adalah:

1. Data harus dapat diakses dan dikeluarkan dengan mudah dari sistem komputer.
2. Setelah dikeluarkan, data hendaknya dapat diproses dengan cepat dan efisien (Scott, 1999).

Fokus permasalahan aktivitas pemrosesan data dan sumber data adalah meningkatkan kemampuan akses, pemanggilan, dan pemrosesan data dengan cepat. Faktor terpenting dalam hal pemanggilan, dan pemrosesan data adalah organisasi *file* data. Organisasi *file* data adalah struktur *file* yang berisi data dan sistem alamat yang memudahkan pengaksesan dan penggantian data dari *file* tersebut. Usaha untuk mengotomasi penggabungan *file* fungsional telah menciptakan konsep basis data. Dengan konsep basis data, proses akses data menjadi lebih mudah, karena penggabungan antar *file* dimungkinkan oleh program aplikasi dari berbagai bidang fungsi untuk memproses semua data yang terkandung dalam basis data (Scott, 1999).

Pada penggunaan sistem konvensional, banyak permasalahan yang muncul karena data didefinisikan dalam suatu program individu. Setiap program memiliki struktur *file* yang unik. Sehingga semua *file* operasi harus didefinisi ulang untuk setiap aplikasi. Berikut adalah beberapa permasalahan umum yang sering muncul pada penggunaan *file-file* tersebut:

- Duplikasi yang tidak terkendali
  - Penggunaan *space* yang berlebihan
  - Sulit melakukan *update* seluruh file
- Data yang tidak konsisten
- Tidak fleksibel



- Sulit mengubah data
- Sulit mengubah program
- *Data sharing* yang terbatas
- Kurangnya standardisasi
- Produktivitas pembuat program yang rendah
- Proses pemeliharaan program yang berlebihan (Post, 2002)

Sistem basis data modern merupakan suatu alat yang amat bermanfaat yang dapat digunakan untuk membangun suatu aplikasi bisnis. Basis data menyediakan berbagai tampilan yang menunjukkan kelebihanannya dibandingkan metode pembuatan program secara tradisional. Basis data didefinisikan sebagai kumpulan data yang disimpan dalam suatu format yang distandardisasi, dan dirancang untuk dapat digunakan bersama-sama oleh beberapa pengguna (Post, 2002).

Sebagian besar basis data memiliki kriteria sebagai berikut:

1. Struktur *filenya* memudahkan untuk mengkaitkan suatu *record* dengan *record* lainnya.
2. Dimungkinkan adanya penggabungan *file* secara menyilang, sehingga *record* yang sebelumnya bebas, kini dapat digabungkan dan diproses bersama secara otomatis.
3. *File* program/data bersifat bebas, sehingga memudahkan proses *update* dan perawatan basis data. Selain itu juga meningkatkan kualitas laporan.
4. Memiliki rumusan bersama (*common definition*) dalam kaitannya dengan definisi data, format *record*, dan berbagai jenis deskripsi data lainnya.
5. Memiliki sistem manajemen basis data untuk mengelola *file* data.
6. Memiliki kamus data untuk memuat informasi tentang data dan basis data.
7. Memiliki kases memori akses langsung yang besar untuk memuat data dan sistem manajemen basis data.
8. Memiliki program dan piranti komunikasi yang canggih, yang memungkinkan banyak pengguna mengakses data secara serempak.



9. Memiliki teknik-teknik penyalinan (*back-up*), penghidupan kembali (*restart*), dan pemulihan data (*recovery*) yang canggih yang memungkinkan untuk mengkonstruksikan kembali *file-file* basis data jika ada data yang rusak atau hilang.
10. Adanya *query language* yang memudahkan permintaan online serta memutakhirkan *record* transaksi demi transaksi (Scott, 1999).

Kelebihan menggunakan basis data adalah:

- Efisiensi pemrosesan data: berkurangnya jumlah data berlebih, kecuali pada sistem data berhubungan, sehingga mengurangi kemungkinan terjadinya kekacauan informasi dalam basis data.
- Waktu pembuatan program relatif lebih singkat: hal ini dikarenakan program yang dibuat dapat mengakses seluruh basis data sehingga jumlah program yang dibuat lebih sedikit.
- Program lebih sederhana: pada pemrosesan data yang relasional pada basis data, tidak perlu dilakukan perubahan pada program aplikasi karena format *record* diubah, karena program aplikasi hanya mengidentifikasi sistem manajemen basis data terhadap data yang diperlukan.
- Mudah berkembang: sejumlah informasi baru yang muncul tetap dapat diproses, karena terdapat berbagai jalur pengaksesan data.
- Proses sederhana: berkurangnya aktivitas input data, karena sebuah transaksi telah memperbaharui banyak *record* sekaligus (*simultaneous updating of related records*) (Scott, 1999).

Kelemahan penggunaan basis data adalah:

- Kurangnya ahli basis data (*insufficient database expertise*)
- Biaya pemrosesan data yang terlalu tinggi (*higher data processing cost*)
- Kebutuhan perangkat keras dan lunak yang semakin bertambah (*increased hardware and software needs*)
- Penggabungan dan pengamanan data. Karena kemampuannya untuk diakses oleh banyak pengguna, maka perlu dikembangkan suatu sistem



pengamanan untuk mencegah kemungkinan data diakses oleh pengguna yang tidak berkepentingan (Scott, 1999).

Masalah yang sering muncul dalam membangun aplikasi yaitu bagaimana menghemat *space*, menyelamatkan data dengan cepat, digunakan bersama oleh beberapa pengguna pada saat yang sama, dan menyediakan data *backup* dan *recovery* kini telah dapat diselesaikan dengan sistem manajemen basis data.

### II.3.3. Sistem Manajemen Basis Data

Sistem manajemen basis data didefinisikan sebagai suatu perangkat lunak yang membentuk basis data, menyimpan data, mendukung bahasa *query*, menghasilkan tampilan *report*, dan menampilkan form input data (Post, 2002).

Kelebihan sistem manajemen basis data adalah sebagai berikut:

- minimal redundansi data
- konsistensi data
- integrasi data
- penggunaan data bersama
- pemberlakuan standard
- pengembangan aplikasi mudah dilakukan
- tingkat keamanan, keleluasaan pribadi, dan integritas yang seragam
- data yang independen (Post, 2002)

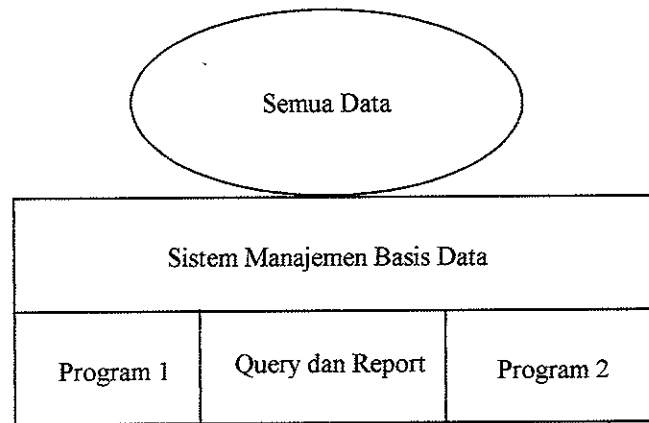
Sistem manajemen basis data mampu menyimpan data secara efisien karena data disimpan dengan *space* terbuang yang minimal. Selain itu, data dapat dengan cepat merespon *query*. Sistem manajemen basis data memiliki sistem yang mampu memelihara konsistensi dengan usaha kecil. Bahasa *query* yang ada pada sistem manajemen basis data memungkinkan terbentuknya data yang terintegrasi. Sistem ini memungkinkan data pada basis data apapun untuk memperoleh *recovery*, penggabungan, dan perbandingan (Post, 2002).

Tujuan sistem manajemen basis data adalah untuk memudahkan pembuatan struktur data, sehingga tidak akan menyulitkan pengguna saat harus



berhadapan dengan *file* yang rumit. Sistem tersebut memberikan pelayanan berupa kemudahan memasukkan, memperbaharui, memproses dan mengakses data.

Pendekatan basis data berfokus pada data. Sehingga komponen yang berubah hanyalah data, sedang strukturnya tidak banyak berubah. Sistem manajemen basis data bertanggung jawab atas pendefinisian, penyimpanan, dan *recovery* data. Seluruh permintaan data harus melalui *database engine*. Sekali struktur data terdefinisi dengan seksama, peralatan tambahan seperti *report writer*, *forms generator*, dan *query language*, akan membuat aplikasi bisnis menjadi lebih mudah dan lebih cepat dikembangkan (Post, 2002).



Gambar II.6. Sistem manajemen basis data fokus pada data

Selain itu, pendekatan basis data yang berfokus pada data, akan memisahkan definisi data dari program, yang disebut dengan *data independence*. *Data independence* memungkinkan perubahan pada definisi data tanpa mengubah program. Data dapat dipindahkan ke perangkat keras lain atau pada mesin lain yang berbeda. Sekali sistem manajemen basis data memahami cara mengakses data, maka semua form, *report*, atau program yang menggunakan data tersebut tidak perlu diubah. Kecuali jika terjadi penghapusan struktur basis data, maka beberapa aplikasi tidak akan berfungsi (Post, 2002). Gambar II.6 merupakan ilustrasi dari sistem manajemen basis data yang fokus pada data. Dalam sistem manajemen basis data yang fokus pada data, yang harus dilakukan pertama kali adalah menentukan data yang akan digunakan. Kemudian query, report dan



program, mengakses data tersebut dengan melalaui sistem manajemen basis data. Sistem manajemen basis data selalu menangani permasalahan keamanan dan hak akses. Sistem manajemen basis data dievaluasi berdasarkan *database engine*, *data dictionary*, *query processor*, *report writer*, *forms generator*, *application generator*, *communication and integration*, dan *security*.

*Database engine* merupakan inti dari sistem manajemen basis data. Bertanggung jawab atas proses penyimpanan, *recovery*, dan *update* data. Komponen ini paling mempengaruhi tampilan (kecepatan) dan kemampuan untuk menangani masalah besar. Komponen lain bergantung pada *database engine* untuk penyimpanan aplikasi data dan sistem internal data yang menentukan bagaimana cara kerja aplikasi tersebut (Post, 2002).

*Data dictionary* memegang seluruh ketentuan pada semua tabel data, menjelaskan tipe data yang disimpan, memungkinkan sistem manajemen basis data meneliti asal data, dan membantu pembuat program dan pengguna mencari data yang dibutuhkan. Sebagian besar sistem basis data modern memiliki *data dictionary* sebagai satu kesatuan dalam tabel sistem.

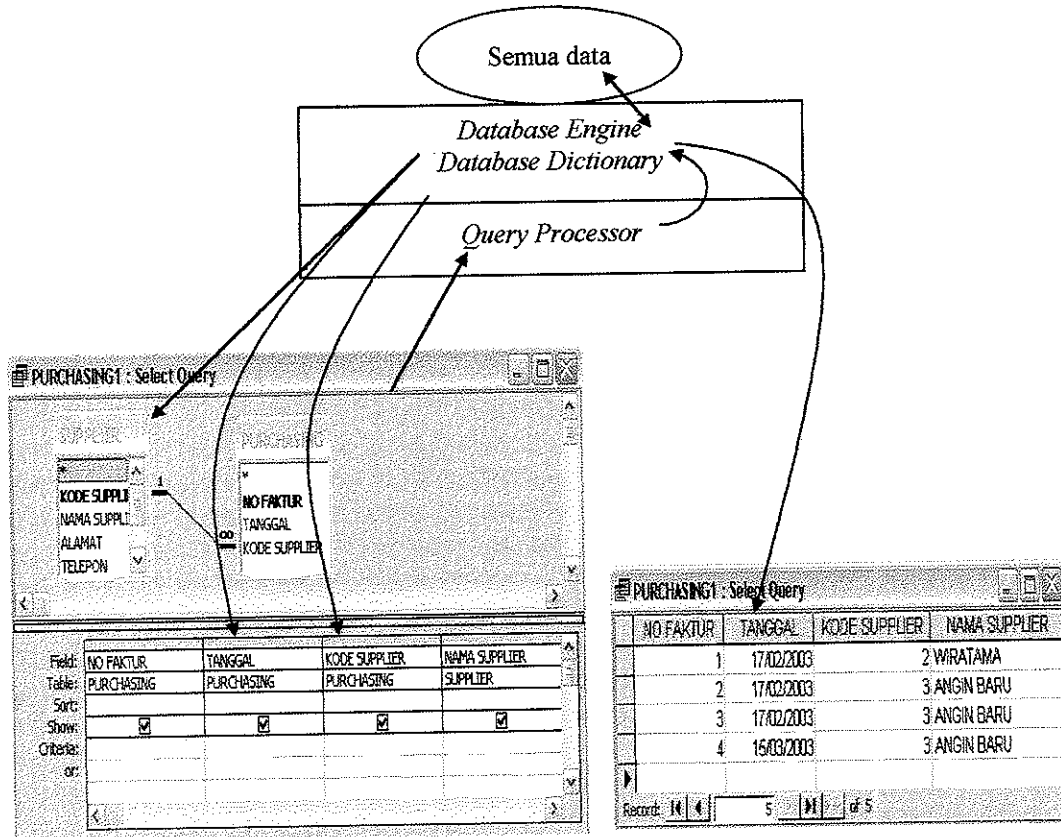
*Query processor* merupakan komponen dasar sistem manajemen basis data, karena memungkinkan pembuat program dan pengguna menyimpan dan menyelamatkan data. Pada beberapa kasus, *query processor* merupakan satu-satunya penghubung pada basis data. Semua operasi basis data dapat dijalankan pada bahasa *query*. *Query* dibentuk dari pertanyaan bisnis. Bahasa *query* sangat dibutuhkan karena bahasa normal seperti bahasa Inggris sulit diterjemahkan pada *query*.

*Report writer* modern memungkinkan pembentukan laporan pada layar untuk menentukan bagaimana suatu item akan ditampilkan atau dikalkulasi. *Report writer* profesional memungkinkan untuk membentuk laporan kompleks pada waktu yang singkat tanpa menulis satupun kode program.

*Report writer* dapat diintegrasikan pada sistem manajemen basis data atau dapat merupakan aplikasi yang berdiri sendiri dimana pembuat program dapat menuliskan kode program untuk menciptakan laporan yang diinginkan. Pada pembentukan desain *report* dasar, yang umumnya berdasarkan suatu *query*, saat



report dijalankan, *report writer* akan menyampaikan *query* pada *query processor* dan yang dihubungkan pada *database engine* untuk mendapatkan kembali baris data yang diinginkan. Kemudian *report writer* akan memformat data sesuai *report template* dan menciptakan laporan lengkap dengan nomor halaman, *header* dan *footer*.



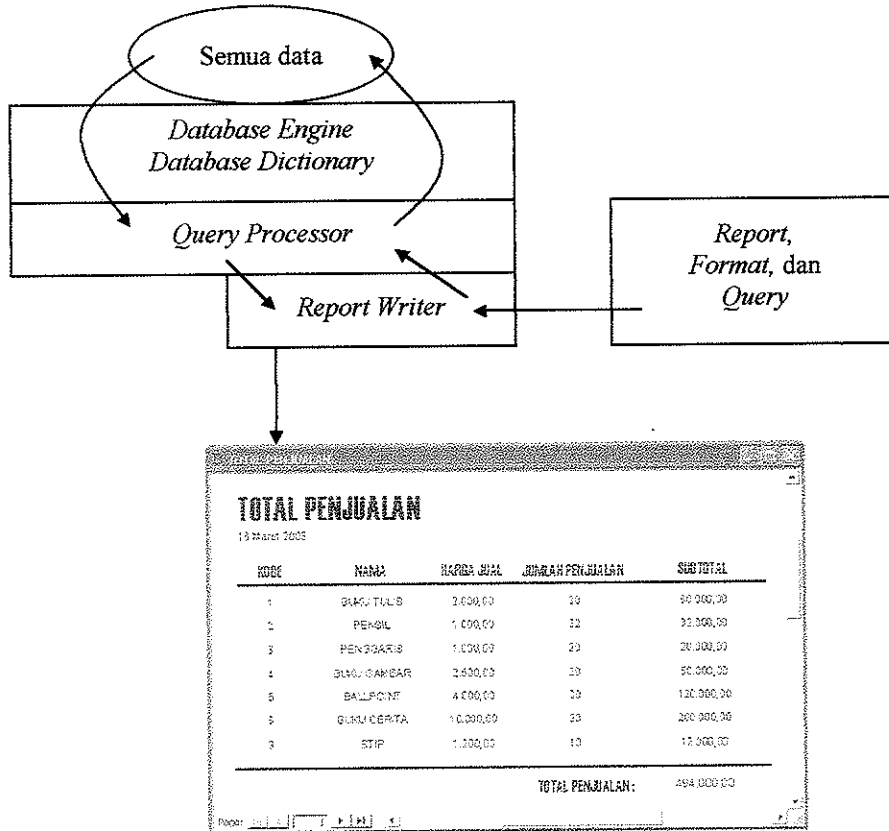
Gambar II.7. Database query processor

*Forms generator* atau input pada layar membantu pembuat program menciptakan *form* input. *Forms generator* memungkinkan pembuat program membentuk form dengan menarik dan menempatkan item pada layar.

Aplikasi merupakan kumpulan *form* dan *report* yang dirancang untuk pengguna tertentu. Aplikasi dapat berbentuk kecil dan terdiri atas beberapa *form* dan *report* input, atau sistem yang besar dan kompleks yang mengintegrasikan data dari beberapa basis data dengan ratusan *form* dan *report*. Sistem manajemen basis data yang baik akan memiliki application generator yang terdiri atas



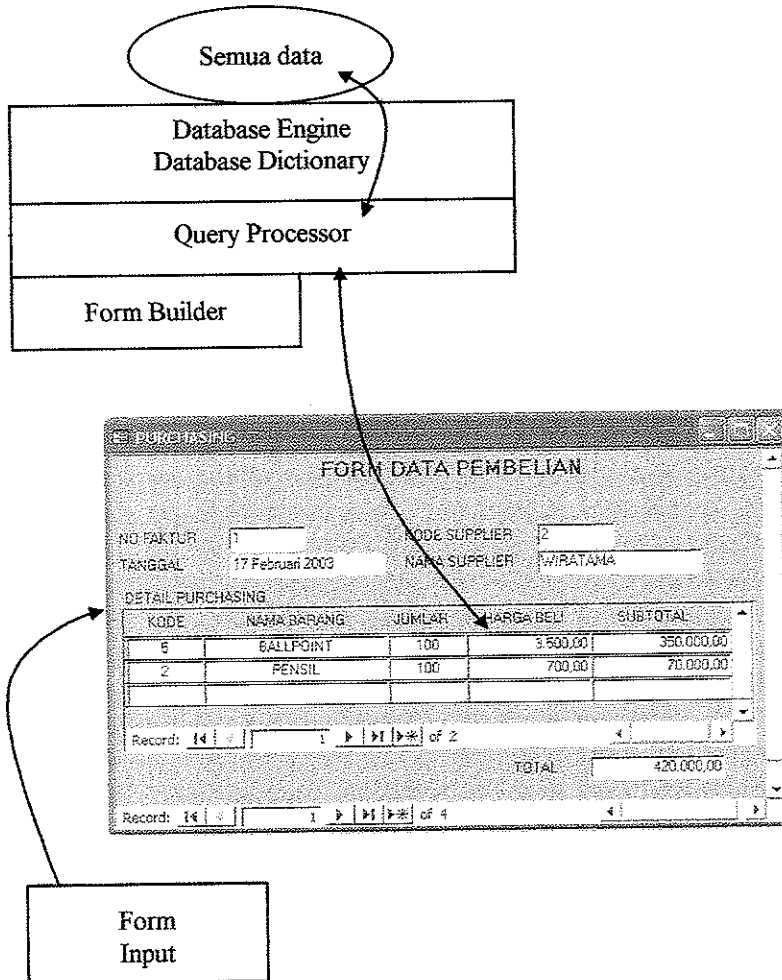
peralatan yang akan membantu pembuat program membentuk aplikasi yang lengkap.



Gambar II.8. Database report writer

Beberapa sistem basis data memiliki utiliti komunikasi dan integrasi yang dirancang untuk menyimpan dan menggunakan data pada beberapa basis data yang sedang dijalankan pada mesin atau lokasi yang berbeda.

Tujuan utama basis data adalah membagi data dengan beberapa pengguna, sistem manajemen basis data juga harus bertanggung jawab atas pembentukan dan pemeliharaan keamanan akses.



Gambar II.9. Database form

### II.3.4. Microsoft Access sebagai Perangkat Lunak Basis Data

Penggunaan Microsoft Access sebagai perangkat lunak untuk membuat basis data, dikarenakan fasilitas yang dimiliki oleh sistem tersebut. Kemudahan yang diperoleh adalah tersedianya sistem manajemen basis data yang mendukung pembuatan basis data. Pada penggunaan sebagai basis data, Microsoft Access dapat digunakan untuk sistem yang tidak terlalu besar. Dalam hal ini sistem yang menjadi acuan adalah sistem *retail* dengan tingkat penjualan rerata perhari sebesar 400 transaksi.

Microsoft Access yang digunakan dalam pembuatan basis data persediaan pada tulisan ini adalah Microsoft Access 2000. Sesuai perkembangannya,



program ini merupakan penyempurnaan dari versi sebelumnya. Hal baru yang terdapat dalam Microsoft Access 2000 antara lain adalah:

- *Database window toolbar* untuk mempercepat proses pembuatan, penataan, dan pengolahan basis data.
- Fasilitas grup untuk mengelompokkan objek dalam basis data
- Menampilkan sub data pada objek tabel, *query*, dan form (LPKBM, 2001).

Dalam pengoperasiannya, Microsoft Access hanya mampu digunakan untuk mengolah satu *file* basis data. Data dalam basis data disimpan dalam sebuah objek yang disebut tabel. Tabel adalah sekumpulan data sejenis yang terdiri atas beberapa informasi sebagai berikut:

- *Field* adalah data terkecil dari suatu tabel yang menempati bagian kolom.
- *Record* adalah kumpulan beberapa *field* yang saling berhubungan yang menempati bagian baris (LPKBM, 2001).

Selain itu, data pada tabel basis data akan didukung oleh objek basis data lainnya yaitu:

- *Query* – digunakan untuk menampilkan, menyunting, dan menganalisis data dengan cara tertentu.
- *Form* – digunakan untuk membuat kontrol-kontrol dalam proses input, memeriksa, dan memperbarui data.
- *Report* – digunakan untuk menampilkan data yang telah diformat sesuai ketentuan yang telah ditetapkan sebelumnya (LPKBM, 2001).

Table memiliki beberapa bagian yang berbeda, yaitu: *field name*, *data type*, dan *description*. Dalam tabel, tidak boleh terdapat nama *field* yang sama. Panjang sebuah nama *field* maksimal 64 karakter yang merupakan kombinasi huruf, angka, spasi, dan karakter khusus kecuali titik (.), tanda seru (!), tanda petik atas (^), dan kurung siku ([ ]). *Data type* digunakan untuk menentukan tipe data suatu *field* dalam *table*. Tipe data dijelaskan pada tabel II.1. *Field name* dengan tipe data *Number*, akan memiliki ukuran yang ditentukan berdasarkan pilihan



properti seperti pada tabel II.2. Bagian yang lain adalah *description*, digunakan untuk menampilkan teks tertentu pada bagian *status bar* dalam *datasheet window* sebuah *table* atau *query*. Jumlah karakter maksimal adalah 255. *Field Properties* merupakan bagian yang digunakan untuk mengatur beberapa ketentuan dari suatu *field* aktif, dan ketentuan yang ditampilkan disesuaikan dengan tipe data *field* tersebut (LPKBM, 2001).

Selain itu, Microsoft Access juga disertai dengan perangkat lunak Visual Basic for Application (VBA) untuk membantu pengembangan basis data tersebut. Visual Basic merupakan *event driven programming* (pemrograman pengendali kejadian). Program akan menunggu sampai respon dari pengguna berupa *event/kejadian* tertentu terdeteksi, maka kode yang berhubungan dengan *event* tersebut akan dijalankan (Suryo Kusumo, 2002). ✕

Aplikasi (proyek) terdiri atas:

- *Form – windows* tempat untuk membuat *user interface*.
- *Control* – tampilan berbasis grafis yang dimasukkan pada form untuk membuat interaksi dengan pengguna (*text box*, *label*, *scroll bar*)
- *Property*- nilai/karakteristik yang dimiliki oleh sebuah objek Visual Basic. Contohnya: *Name*, *Captions*, *Size*, *Color*, *Position*, dan *Text*. Visual Basic menerapkan properti *default/standard*.
- *Method* – serangkaian perintah yang tersedia pada suatu objek yang diminta untuk mengerjakan tugas khusus.
- *Event Procedures* – Kode yang berhubungan dengan suatu objek. Kode ini akan dieksekusi ketika ada respon dari pemakai berupa *event* tertentu.
- *General Procedures* – Kode yang tidak berhubungan dengan suatu objek. Kode ini harus diminta oleh aplikasi.
- *Module* – Kumpulan prosedur umum, deklarasi variabel dan definisi konstanta yang digunakan oleh aplikasi.



Tabel II.1. Tipe data

Tipe	Keterangan
<i>Text</i>	Untuk menerima data teks sampai 255 karakter yang terdiri atas huruf, angka, dan simbol grafik. Aturan <i>default</i> yang digunakan adalah 50 karakter
<i>Memo</i>	Untuk menerima data teks sampai 65.535 karakter yang terdiri atas huruf, bilangan, tanda baca, serta simbol grafik. Data tipe memo tidak dapat diindex.
<i>Number</i>	Untuk menerima data digit, tanda minus, dan titik desimal. Tipe data ini mempunyai 5 pilihan ukuran bilangan dan jumlah digit tertentu.
<i>Date/Time</i>	Untuk menerima data nilai tanggal dan waktu serta nilai tahun yang dimulai dari 100 sampai 9999.
<i>Currency</i>	Untuk menerima data digit, tanda minus, dan titik desimal dengan ketepatan 15 digit desimal; di sebelah kiri titik desimal dan 4 digit di sebelah kanan titik desimal.
<i>AutoNumber</i>	Untuk menampilkan secara otomatis data pencacah atau angka yang dimulai dari 1 dengan selang naik 1.
<i>Yes/No</i>	Untuk menerima salah satu data dari dua nilai yang ada, yaitu: Yes/No, True/False, On/Off.
<i>Ole Project</i>	Untuk menerima sebuah <i>image</i> grafik, <i>spreadsheet</i> , foto digital, rekaman suara, atau video yang dapat diambil dari program aplikasi lain. Ukuran maksimumnya 1 <i>gigabyte</i> . Tipe ini tidak dapat diindex
<i>Hyperlink</i>	Untuk menerima data berupa teks berwarna dan bergaris bawah serta grafik. Tipe ini berhubungan dengan jaringan
<i>Lookup Wizard</i>	Untuk menampilkan satu dari beberapa tipe data yang ada dalam suatu daftar. Data tersebut diambil dari <i>table</i> atau <i>query</i> yang ada.

Sumber: (LPKBM, 2001)



Tabel II.2. Tipe data *number*

Pilihan	Keterangan
<i>Byte</i>	Memberikan nilai integer (bilangan bulat) dari 0 hingga 255 tidak termasuk pecahan.
<i>Decimal</i>	Memberikan nilai integer dari $-10E28$ hingga $10E28$ dengan ketelitian hingga 28 digit dibelakang titik desimal.
<i>Integer</i>	Memberikan nilai integer dari $-32,768$ hingga $32,768$ tidak termasuk pecahan.
<i>Long Integer</i>	Memberikan nilai integer dari $-2,147483648$ hingga $2,147483647$ tidak termasuk pecahan.
<i>Single</i>	Memberikan nilai bilangan nyata (bilangan yang mungkin memiliki pecahan terkecil) dengan ketelitian hingga 7 digit di belakang titik desimal, dimulai dari $-3,402823E38$ hingga $-1,401298E-45$ untuk nilai negatif dan $1,401298E-45$ hingga $3,402823E38$ untuk nilai positif
<i>Double</i>	Memberikan nilai bilangan nyata dengan ketelitian sampai 15 digit di belakang titik desimal, dimulai dari $-1,7200069313486231E308$ hingga $-4,94065645841247E-324$ untuk nilai negatif dan dari $1,7200069313486231E308$ hingga $4,94065645841247E-324$ untuk nilai positif.

Sumber:(LPKBM, 2001)

### II.3.5. Sistem Pendukung Keputusan

Dalam sistem informasi manajemen, seringkali informasi yang dihasilkan tidak memadai untuk membuat keputusan yang spesifik untuk memecahkan permasalahan yang spesifik. Sistem pendukung keputusan (*decision support system*) dibuat sebagai suatu cara untuk memenuhi kebutuhan ini. Sistem pendukung keputusan adalah sistem penghasil informasi yang ditujukan pada suatu masalah tertentu yang harus dipecahkan oleh manajer dan keputusan yang harus dibuat manajer. Keputusan merupakan rangkaian tindakan yang perlu diikuti dalam memecahkan masalah untuk menghindari atau mengurangi dampak negatif, atau untuk memanfaatkan kesempatan (McLeod, 1996).

Menurut Simon dalam buku Sistem Informasi Manajemen (McLeod, 1996), keputusan terprogram di satu ujung dan keputusan tak terprogram di ujung lainnya sebagai satu rangkaian. Keputusan terprogram memiliki sifat berulang dan



rutin, sehingga suatu prosedur telah dibuat untuk menanganinya sehingga keputusan tersebut tidak perlu diperlakukan sebagai hal yang baru tiap kali terjadi. Sedang keputusan tak terprogram bersifat baru, tidak terstruktur, dan jarang konsekuen. Tidak ada metode yang pasti untuk menanganinya karena belum pernah ada sebelumnya, atau karena sifat dan strukturnya tidak terlihat atau rumit atau karena begitu pentingnya, sehingga perlu penanganan khusus (McLeod, 1990).

Dalam memecahkan masalah, menurut Simon ada empat tahap penting yang harus dilakukan, yaitu:

- Intelijen: mengamati lingkungan untuk mencari kondisi yang perlu diperbaiki.
- Perancangan: menemukan, mengembangkan, dan menganalisis berbagai alternatif tindakan yang mungkin.
- Pemilihan: memilih satu rangkaian tindakan tertentu dari beberapa yang tersedia.
- Evaluasi: menilai pilihan-pilihan yang telah lalu (McLeod, 1996).

Tahap-tahap dalam pengambilan keputusan tersebut digunakan untuk menentukan struktur masalah. Masalah terstruktur merupakan suatu masalah yang memiliki struktur pada tiga tahap pertama, yaitu intelijen, perancangan, pemilihan. Sehingga dapat dibuat algoritma, atau aturan yang memungkinkan masalah diidentifikasi dan dievaluasi serta ditentukan solusinya. Masalah tak terstruktur, sebaliknya merupakan masalah yang sama sekali tidak memiliki struktur pada proses pengambilan keputusan di atas. Masalah semi-terstruktur merupakan masalah yang memiliki struktur hanya satu atau dua tahap saja pada proses di atas. Sistem pendukung keputusan menyediakan informasi pemecahan permasalahan semi-terstruktur.

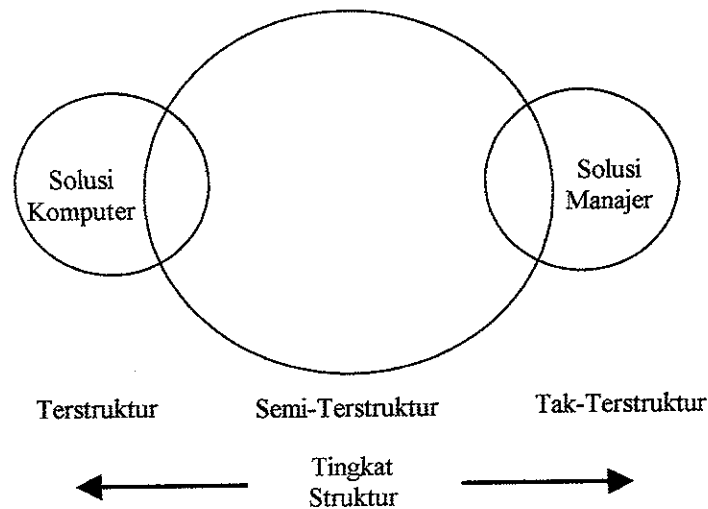
Tujuan sistem pendukung keputusan adalah:

- Membantu proses pengambilan keputusan untuk memecahkan permasalahan semi terstruktur.



- Mendukung penilaian yang telah dibuat, dan tidak berusaha untuk menggantikannya.
- Meningkatkan efektivitas pengambilan keputusan daripada efisiensinya.

Tujuan-tujuan tersebut berkaitan dengan tiga prinsip dasar dari konsep sistem pendukung keputusan, yaitu: struktur masalah, dukungan keputusan, dan efektivitas keputusan. Sulit untuk menentukan masalah yang sepenuhnya terstruktur atau tak terstruktur. Sebagian besar berada pada area semi-terstruktur. Hal ini berarti, sistem pendukung keputusan diarahkan pada area tempat sebagian permasalahan berada. Selain itu sistem pendukung keputusan tidak diarahkan untuk menggantikan posisi manajer sebagai pengambil keputusan. Seperti terlihat pada gambar II.12, komputer, dalam hal ini sistem pendukung keputusan, diterapkan pada bagian masalah terstruktur, tetapi manajer bertanggung jawab atas seluruh permasalahan yang tak terstruktur dengan menerapkan penilaian, intuisi dan analisis. Sedang komputer dan manajer bekerja sama dalam memecahkan masalah yang terletak di area semi terstruktur yang luas (McLeod, 1996).



Gambar II.10. Sistem pendukung keputusan pada permasalahan semi-terstruktur

Tujuan dari sistem pendukung keputusan bukan untuk membuat proses pengambilan keputusan seefisien mungkin. Waktu dalam proses pengambilan



keputusan memang sangat berharga, namun manfaat utama sistem ini adalah untuk mendapatkan menghasilkan keputusan yang lebih baik/efektif. Ketika keputusan dibuat dari sekian alternatif yang ada, hasil yang diperoleh akan sebanding dengan usaha yang telah dilakukan. Disini keputusan seorang manajer menjadi penting karena pertimbangannya dalam menentukan kapan suatu keputusan akan berkontribusi pada suatu solusi masalah (McLeod, 1996).

Secara khusus sistem pendukung keputusan menggunakan model komputer yang kecil dan sederhana, dalam pengertian bahwa itulah model aktivitas manajerial. Model adalah serangkaian program, biasanya berisi persamaan matematika, yang menggambarkan permasalahan atau tugas manajemen yang khas. Dengan sedikit mengubah model atau data yang dimasukkan, akan dihasilkan alternatif pemecahan bagi manajerial, untuk kemudian dievaluasi tiap alternatif tersebut.

Unsur dalam sistem pendukung keputusan termasuk model, yakni *file* data khusus dalam bentuk basis data, dan seorang manajer yang berinteraksi langsung dengan model melalui terminal komputer guna menguji pemecahan yang mungkin bagi suatu masalah manajerial. Data dalam basis data secara khusus merupakan kombinasi data yang diambil dari transaksi organisasi atau *master file* dan yang berasal dari sumber luar, walau mungkin seluruh data berasal hanya dari satu diantara sekian banyak sumber.

Pada sistem *retail*, tingkat persediaan merupakan keputusan utama logistik pasar yang mempengaruhi kepuasan pelanggan. Bagian pemasaran menginginkan perusahaan menyimpan cukup persediaan untuk dapat segera memenuhi semua pesanan pelanggan. Namun tidak efektif jika perusahaan harus menyimpan persediaan secara berlebihan. Hal ini akan berpengaruh langsung pada pembiayaan yang harus dikeluarkan oleh perusahaan dalam hal persediaan. Perusahaan harus mengetahui berapa jumlah penjualan dan laba yang akan diperoleh dengan memiliki sejumlah persediaan yang lebih besar dan menjanjikan pemenuhan pesanan dengan lebih cepat dan membuat keputusan berdasarkan hal tersebut (Kotler, 1997).

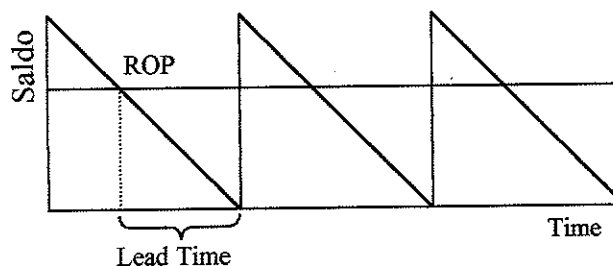


Pengambilan keputusan persediaan perlu mengetahui kapan waktu yang tepat ketika harus melakukan pemesanan, dan jumlah pesanan. Dengan berkurangnya jumlah persediaan, perusahaan harus mengetahui pada tingkat persediaan berapa, perusahaan perlu melakukan pesanan baru. Tingkat persediaan ini yang kemudian disebut sebagai titik pemesanan kembali (*reorder point*). Titik pemesanan kembali harus menyeimbangkan antara resiko kehabisan persediaan dan resiko memiliki jumlah persediaan yang berlebihan (Kotler, 1997).

Keputusan yang lain adalah jumlah produk yang dipesan. Semakin besar kuantitas pemesanan, semakin jarang pemesanan harus dilakukan. Hal ini akan berkaitan dengan biaya yang harus dikeluarkan perusahaan yang meliputi biaya pemrosesan pesanan dan penyimpanan persediaan. Semakin besar persediaan rata-rata disimpan, semakin tinggi biaya persediaan. Jumlah pesanan optimal dapat ditentukan dengan mengamati total biaya pemrosesan pesanan dan biaya penyimpanan persediaan untuk berbagai tingkat pesanan. Biaya pemrosesan pesanan akan menurun dengan semakin banyaknya unit yang dipesan, namun biaya persediaan akan semakin meningkat dengan semakin banyaknya unit yang dipesan (Kotler, 1997).

### II.3.6. Titik Pemesanan Kembali (Reorder Point/ROP)

Pengambilan keputusan persediaan meliputi informasi kapan harus melakukan pemesanan dan jumlah pesanan. Bila jumlah persediaan berkurang, perusahaan harus mengetahui bagaimana cara memasukkan pesanan baru. Titik ini yang disebut titik pemesanan (atau titik pemesanan kembali /ROP).



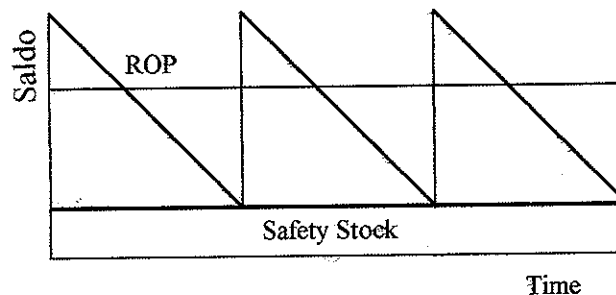
Gambar II.11. Titik pemesanan kembali tanpa *safety stock*



Gambar II.11, merupakan ilustrasi sistem persediaan dalam perusahaan. Bentuk gigi gergaji menggambarkan bagaimana persediaan berangsur-angsur berkurang dalam sistem *retail* dikarenakan proses penjualan, kemudian diisi kembali. Setelah persediaan habis, pesanan tiba dan persediaan kembali penuh. Siklus tersebut diulang berkali-kali (McLeod, 1996).

Sistem tersebut merupakan kondisi ideal dalam perusahaan. Namun dengan adanya berbagai kemungkinan seperti tingkat penjualan yang tinggi atau lambatnya pesanan sampai ke gudang dapat menyebabkan perusahaan kehabisan persediaan (*stockout*). Perusahaan harus mengantisipasinya dengan melakukan pemesanan kepada pemasok saat jumlah persediaan mencapai titik pemesanan kembali. Titik pemesanan kembali merupakan batas jumlah persediaan yang menentukan saat harus melakukan pemesanan pada pemasok. Jumlah persediaan tersebut ditentukan oleh jumlah rerata penjualan per hari, waktu yang diperlukan oleh pemasok untuk mengisi persediaan (*lead time*) dan sejumlah persediaan yang digunakan untuk berjaga-jaga agar jangan sampai perusahaan kehabisan persediaan karena berbagai hal yang disebut sebagai *safety stock* (McLeod, 1996).

Sistem tersebut membantu dalam menentukan keputusan kapan harus melakukan pemesanan kembali. Sehingga perusahaan tidak perlu menyimpan persediaan dalam jumlah yang begitu besar dan menghindarkan kemungkinan perusahaan kehilangan pasar karena kehabisan persediaan.



Gambar II.12. Titik pemesanan kembali dengan *safety stock*



### II.3.7. Penentuan Jumlah Pemesanan Ulang

Salah satu bagian dari sistem pendukung keputusan dalam hal persediaan adalah penentuan jumlah pemesanan ulang untuk periode berikutnya. Dalam menentukan jumlah pemesanan ulang, manajemen harus memutuskan berdasarkan berbagai faktor baik dari dalam maupun dari luar perusahaan yang berpengaruh pada keputusan tersebut.

Salah satu faktor yang umum digunakan dalam menentukan jumlah pemesanan berikutnya adalah menggunakan metode prakiraan (*forecasting*). Metode ini cukup efektif karena dihasilkan dari rerata penjualan di masa lalu. Dari metode ini kemudian akan dihasilkan nilai perkiraan jumlah penjualan untuk periode berikutnya yang kemudian dapat dijadikan acuan untuk menentukan jumlah pembelian periode berikutnya.

Metode prakiraan yang cukup efektif untuk menangani permasalahan tersebut adalah metode rerata bergerak linear. Metode ini mampu mengakomodasi pengaruh *trend* dalam tingkat penjualan. Dasar metode ini adalah menghitung rerata bergerak dari rerata bergerak tunggal. Rerata bergerak tunggal merupakan rerata yang dihitung dari sejumlah data masa lalu selama beberapa periode dibagi dengan jumlah periode tersebut. Dari rerata bergerak tunggal tersebut kemudian dihitung menjadi rerata bergerak kedua yang berasal dari sejumlah rerata bergerak tunggal selama beberapa periode dibagi jumlah periode. Fungsi penghitungan rerata bergerak kedua adalah untuk penyesuaian *trend*. Salah satu cara yang dapat mengurangi kesalahan sistematis yang ditimbulkan oleh rerata bergerak tunggal akibat *trend* adalah menggunakan perbedaan antara nilai rerata bergerak tunggal dan rerata bergerak kedua. Prosedur pada metode ini meliputi tiga aspek, yaitu:

1. penggunaan rerata bergerak tunggal pada waktu tertentu
2. penyesuaian, yang merupakan perbedaan antara rerata bergerak tunggal dan ganda pada waktu tertentu
3. penyesuaian untuk kecenderungan dari periode tertentu ke periode tertentu berikutnya (Makridakis, 1983).

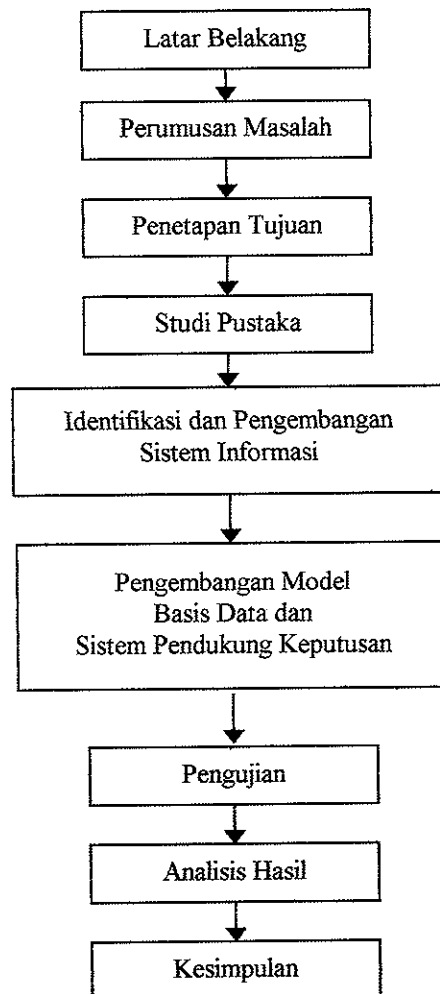


## BAB III

### METODOLOGI PENELITIAN

#### III.1. GAMBARAN UMUM

Secara umum, proses pembuatan sistem pendukung keputusan pemesanan ulang dapat dijelaskan seperti diagram alir berikut ini:



Gambar III.1 Metodologi penelitian pembuatan sistem pendukung keputusan

Setelah merumuskan masalah dan tujuan pembuatan program sistem pendukung keputusan, maka dilakukan penyusunan laporan tugas akhir sarjana dengan tahapan-tahapan sebagai berikut:



## 1. Latar Belakang Masalah

Latar belakang masalah merupakan dasar untuk melakukan suatu penelitian/perancangan. Dengan latar belakang masalah, akan terbentuk suatu pemecahan masalah yang dapat diangkat sebagai topik utama. Latar belakang masalah bersifat umum dan luas.

Pada bagian tersebut, dibahas mengenai perlunya suatu keunggulan kompetitif agar mampu bersaing dalam dunia industri. Diantaranya adalah manajemen jumlah persediaan dalam swalayan. Pada tulisan ini, latar belakang masalah telah dibahas pada bagian Pendahuluan, Bab I.

## 2. Perumusan Masalah dan Penetapan Tujuan.

Dalam memecahkan suatu permasalahan, diperlukan suatu perumusan masalah dan penetapan tujuan yang tepat. Tindakan-tindakan pemecahan masalah kemudian dilakukan sesuai dengan tujuan yang telah ditetapkan. Penetapan tujuan yang tepat akan membantu dalam menyelesaikan masalah. Hal ini dikarenakan penetapan tujuan akan menjadi suatu batasan yang akan dituju dengan permasalahan utama sebagai ruang lingkup, sehingga pembahasan masalah akan menjadi lebih terarah dan terstruktur.

Permasalahan utama dan tujuan yang ingin dicapai pada perancangan ini adalah bagaimana membuat sistem pendukung keputusan pemesanan ulang dari basis data persediaan yang terintegrasi. Perumusan masalah dan penetapan tujuan pada tulisan ini telah dijelaskan pada bagian Pendahuluan, Bab I.

## 3. Studi Pustaka.

Dalam memecahkan masalah yang telah dirumuskan dan mencapai tujuan penelitian, diperlukan referensi dengan melakukan studi pustaka. Dengan studi pustaka, akan diperoleh referensi secara teoretis mengenai permasalahan yang akan diselesaikan. Referensi inilah yang kemudian digunakan sebagai dasar penyelesaian masalah. Pada tulisan ini, tahap tersebut mencakup pengumpulan data mengenai basis data dan sistem pendukung keputusan sebagai bagian sistem informasi manajemen.



Studi pustaka yang dilakukan berhubungan dengan pembuatan sistem pendukung keputusan pada usaha *retail*. Termasuk di dalamnya adalah pengertian-pengertian dan fungsi mengenai *retail*, fungsi persediaan, komponen basis data, dan proses pemesanan ulang dengan sistem *reorder point* yang dilakukan pada usaha *retail*. Uraian tersebut terdapat pada bagian Tinjauan Pustaka, Bab II.

#### 4. Identifikasi dan Pengembangan Sistem Informasi

Perancangan ini diawali dengan merumuskan metodologi penelitian. Metodologi penelitian diawali dengan identifikasi sistem informasi yang digunakan untuk mengolah data persediaan pada swalayan Mataram Mandiri. Dengan sistem informasi tersebut, dikembangkan suatu model arus informasi persediaan yang menyatukan data dari bagian pembelian, kasir, dan gudang.

Informasi tersebut kemudian diolah dalam sistem pendukung keputusan pemesanan ulang. Sistem tersebut terdiri atas penentuan kriteria penjualan produk, penentuan titik pemesanan ulang, dan penentuan jumlah pemesanan ulang. Pengembangan model tersebut dibahas pada bagian Metodologi Penelitian, Bab III.

#### 5. Pengembangan Basis Data dan Sistem Pendukung Keputusan

Pengembangan dimaksudkan untuk membangun suatu basis data yang mampu mendukung sistem pendukung keputusan pemesanan ulang berdasarkan model yang telah ditentukan sebelumnya. Basis data tersebut dikembangkan dari model arus informasi pada swalayan dan menyatukan data persediaan dari bagian pembelian, dan penjualan. Dengan sejumlah data penjualan selama periode tertentu dan persediaan, diperoleh data setiap produk yang kemudian dapat diolah menjadi suatu pendukung keputusan.

Sistem basis data yang dikembangkan adalah bentuk pengembangan dari sistem yang berlaku pada swalayan Mataram Mandiri. Sistem basis data yang menyatukan data dari bagian pembelian, gudang,



dan penjualan ini kemudian dikembangkan menjadi sistem pendukung keputusan yang menjadi tujuan permasalahan.

Kemudian dari pengembangan aplikasi tersebut, dijelaskan cara penggunaan sistem pendukung keputusan hingga diperoleh informasi yang akan membantu proses penentuan keputusan pemesanan ulang. Pembahasan tersebut ada pada bagian Pembuatan Basis Data dan Sistem Pendukung Keputusan serta Penggunaannya, Bab IV.

#### 6. Pembahasan dan Analisis Hasil

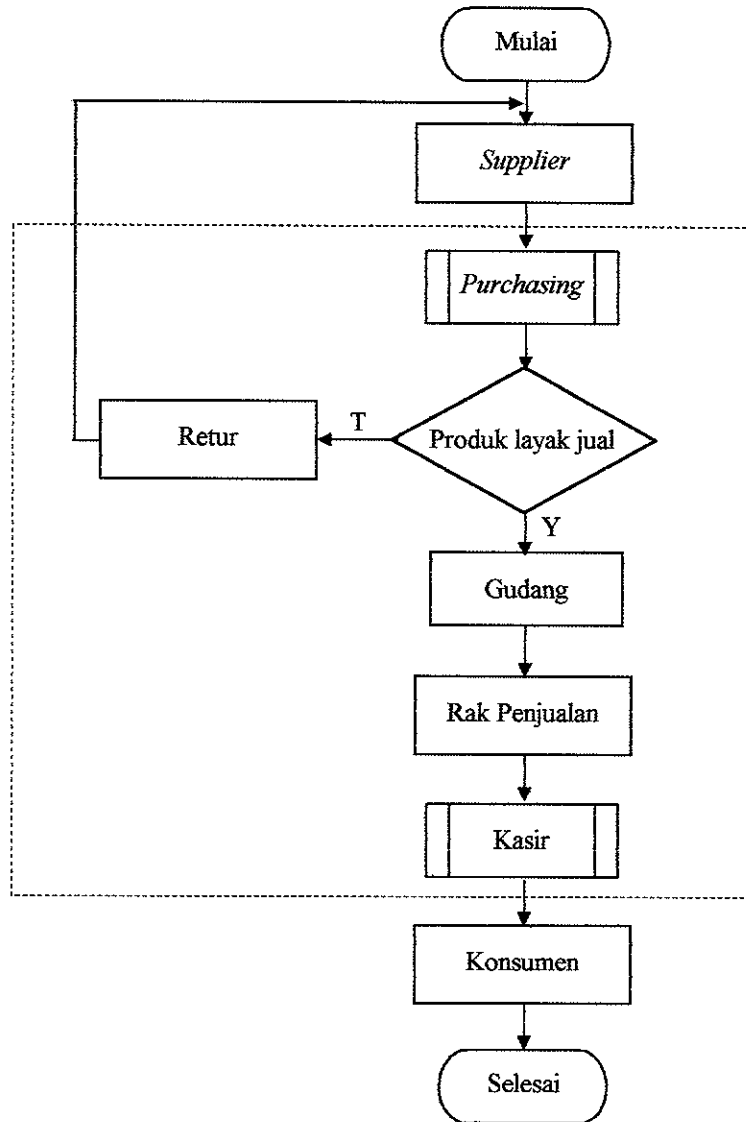
Setelah dilakukan suatu pengembangan basis data yang berasal dari sistem yang telah ada, kemudian dilakukan suatu pengujian untuk membuktikan bahwa basis data tersebut dapat menghasilkan sistem pendukung keputusan yang baik. Pengujian dilakukan dengan membandingkan waktu yang diperlukan untuk memperoleh informasi pemesanan ulang periode berikutnya tanpa sistem pendukung keputusan dengan jika menggunakan sistem pendukung keputusan. Analisis hasil pengujian ini diuraikan pada bagian Analisis dan Pembahasan, Bab V.

#### 7. Kesimpulan

Kesimpulan disusun berdasarkan hasil pengembangan sistem keputusan dan hasil pengujian yang dilakukan. Disamping itu juga diberikan saran-saran yang berkaitan dengan sistem pendukung keputusan yang dihasilkan. Kesimpulan dan saran dapat dilihat pada bagian Kesimpulan dan Saran, Bab VI.

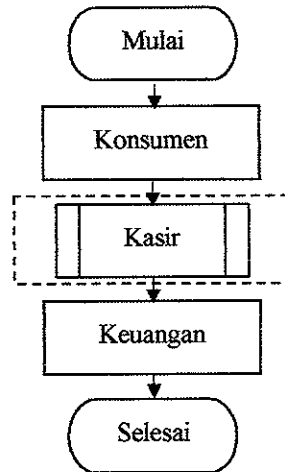
### **III.2. IDENTIFIKASI SISTEM INFORMASI**

Pada tahap ini, sistem informasi yang akan diamati meliputi dua hal pokok. Yaitu arus produk dari *supplier* hingga ke tangan konsumen melalui kasir dan arus uang dari konsumen ke swalayan. Proses tersebut diuraikan seperti terlihat pada gambar III.2. dan III.3. Pada arus informasi produk, proses pengolahan data ditekankan pada kebutuhan informasi akan jumlah persediaan terakhir yang dimiliki oleh swalayan. Sedangkan arus informasi uang menekankan pada jumlah laba yang didapat dari total penjualan pada hari itu.



Gambar III.2 Arus produk

Pada gambar III.2. dan III.3. bagian yang berada dalam kotak bergaris putus-putus merupakan bagian yang bertanggung jawab atas perubahan data dalam swalayan. Proses pengolahan data jumlah persediaan terakhir memerlukan dukungan data dari bagian pembelian dan penjualan. Sedang proses pengolahan data total penjualan memerlukan dukungan data dari bagian penjualan. Seluruh bagian tersebut, memiliki tugas untuk melakukan input data atas setiap transaksi yang dilakukan pada sistem basis data yang terintegrasi.



Gambar III.3 Arus uang

Setiap produk yang telah dibeli akan mendapatkan perubahan harga sesuai kebijakan swalayan. Harga jual setiap produk berubah dari harga belinya secara bervariasi. Kebijakan swalayan dalam menentukan harga jual amat dipengaruhi oleh kondisi ekonomi yang ditentukan oleh pasar. Jika harga beli meningkat, tentu saja harga jual turut meningkat.

Bagian pembelian juga bertanggung jawab untuk melakukan pemeriksaan fisik atas produk-produk tersebut. Produk yang layak jual, dapat segera disimpan di gudang, sedang produk-produk yang cacat fisik segera mengalami retur pada *supplier* dikeluarkan ke rak penjualan sesuai kebutuhan bagian penjualan. Selain itu setiap produk yang mengalami retur harus segera didata. Bagian gudang bertanggung jawab atas penyimpanan seluruh produk yang telah dibeli dan telah mengalami proses pendataan yang lengkap.

Untuk memenuhi permintaan konsumen, maka produk yang diinginkan harus tersedia di bagian penjualan. Dari bagian penyimpanan di gudang, produk yang telah dibeli dari *supplier* dan telah memiliki harga jual dapat segera diletakkan di rak penjualan.

Proses berikutnya adalah transaksi penjualan produk pada konsumen. Konsumen yang telah menemukan produk sesuai yang dibutuhkan dapat segera membelinya di bagian kasir. Di sini terjadi dua arus data, yaitu data penjualan dan data uang masuk ke swalayan. Kasir bertanggung jawab atas seluruh penjualan



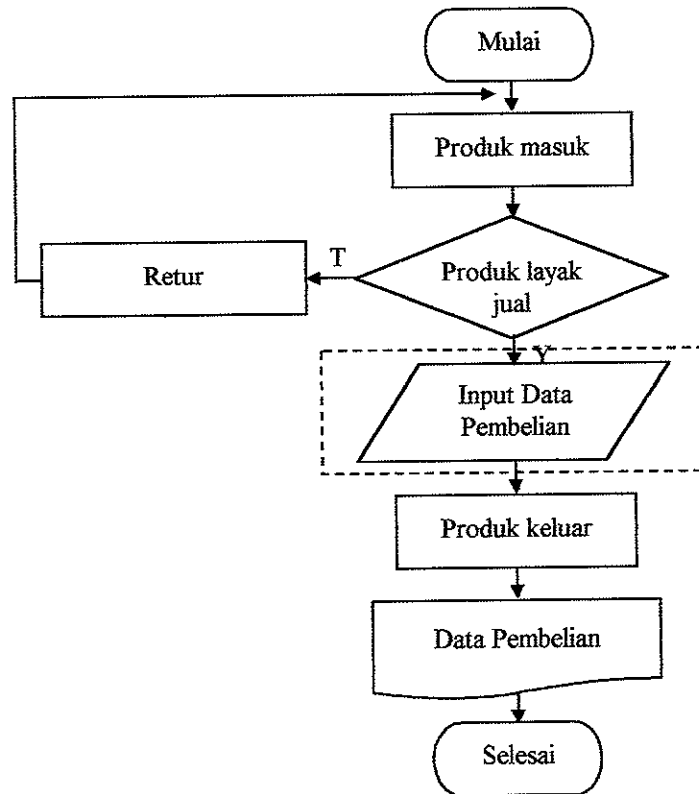
produk dan pencatatan data penjualan berikut seluruh uang masuk. Dari bagian penjualan, uang dikirim ke bagian keuangan. Seluruh pengolahan uang beserta data-datanya menjadi tanggung jawab bagian keuangan.

Berdasarkan sistem yang ada tersebut kemudian dikembangkan model sistem informasi yang sesuai dengan sistem yang ada di swalayan Mataram Mandiri. Model tersebut diperlukan untuk mengembangkan basis data. Berikut adalah penjelasan bentuk model arus informasi pada setiap bagian yang mempengaruhi terjadinya perubahan jumlah persediaan dan jumlah laba harian pada swalayan Mataram Mandiri.

### **III.2.1. Bagian Pembelian**

Arus informasi dimulai pada bagian pembelian dengan melakukan input data atas seluruh pembelian produk yang dilakukan pada sistem basis data. Input data dilakukan setelah proses seleksi produk cacat dilakukan. Hal ini penting dilakukan agar setiap produk yang diterima benar-benar produk layak jual. Input data meliputi seluruh keterangan yang berkaitan dengan produk yang dibeli, seperti nama produk, harga beli, tanggal pembelian, jumlah pembelian, dan keterangan mengenai *supplier*. Basis data yang berpengaruh meliputi pengolahan data produk, data *supplier*, dan data pembelian. Setiap produk yang telah dibeli akan mengalami pengolahan data. Input data pembelian produk yang dilakukan oleh bagian pembelian juga disertai dengan pencantuman harga jual produk sesuai kebijakan dari bagian keuangan.

Setelah proses pembelian, produk mengalami seleksi atas cacat produksi yang mungkin dimiliki. Produk cacat akan mengalami retur. Data retur juga menjadi tanggung jawab bagian pembelian. Data retur meliputi nama produk, nama *supplier*, tanggal pembelian dan jumlah retur. Retur akan mengurangi jumlah persediaan. Input data yang dilakukan bagian pembelian akan berpengaruh pada pengolahan jumlah persediaan dalam gudang. Jumlah persediaan di gudang akan meningkat sesuai pembelian yang dilakukan dikurangi dengan jumlah retur yang terjadi. Proses pencatatan pada bagian pembelian dilakukan segera setelah produk diperiksa. Proses tersebut dijelaskan pada gambar III.4.



Gambar III.4. Arus informasi pada bagian pembelian

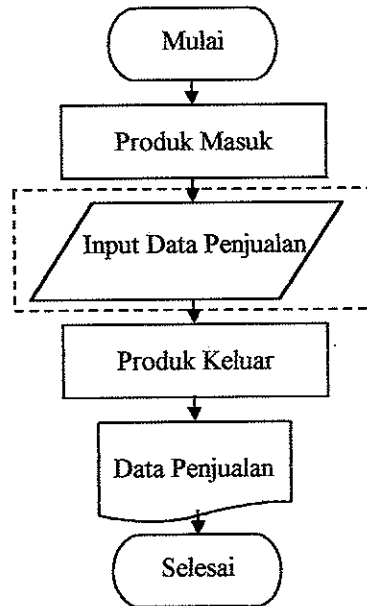
### III.2.2. Bagian Penjualan

Bagian penjualan bertanggung jawab atas setiap transaksi penjualan. Bagian penjualan memiliki dua arus informasi, produk dan uang. Basis data yang berpengaruh meliputi pengolahan data penjualan dan produk. Input data penjualan meliputi tanggal penjualan, kode produk, jumlah penjualan. Sedangkan input data uang meliputi tanggal transaksi, nomor nota, dan jumlah uang masuk. Setiap produk yang dijual akan mengalami proses pencatatan. Dan fungsi kasir adalah melakukan input data penjualan bertepatan dengan proses penjualan.

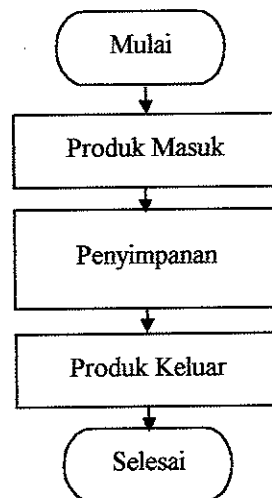
Data penjualan merupakan bahan utama pada pengolahan pada sistem pendukung keputusan. Fungsi bagian kasir dalam proses input data amat menentukan hasil yang akan diperoleh dari pengolahan data pada sistem pendukung keputusan. Data penjualan akan menentukan kriteria penjualan produk dan digunakan untuk memprediksi kebutuhan produk pada periode yang akan



datang, sehingga akan menghasilkan peramalan jumlah produk yang harus dibeli untuk memenuhi kebutuhan produk tersebut.



Gambar III.5. Arus Informasi pada Bagian Penjualan



Gambar III.6. Arus informasi pada bagian gudang



### III.2.3. Bagian Gudang

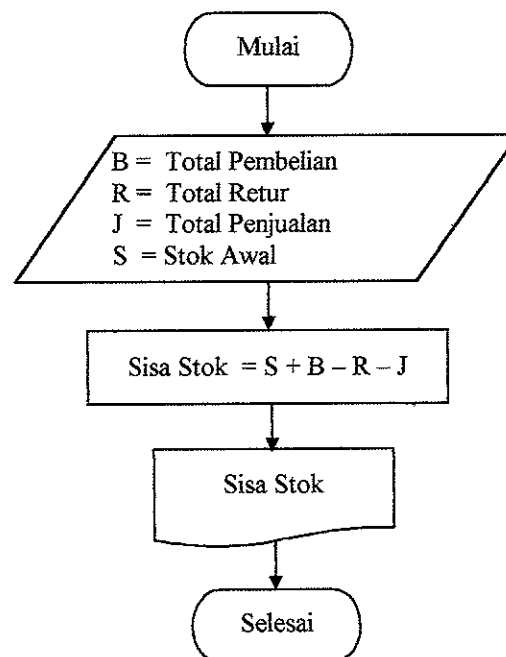
Bagian gudang pada intinya melakukan proses penyimpanan produk yang telah dibeli sebelum produk tersebut dibawa ke rak penjualan. Bagian gudang tidak melakukan input data.

## III.3. PENGEMBANGAN SISTEM PENDUKUNG KEPUTUSAN

Setelah sistem informasi pada swalayan tersebut telah dapat diidentifikasi, proses selanjutnya adalah membuat basis data. Basis data dikembangkan agar dapat mengolah data persediaan menjadi informasi. Informasi tersebut diolah untuk kemudian dapat menjadi sistem pendukung keputusan.

### III.3.1. Pengolahan Data Persediaan

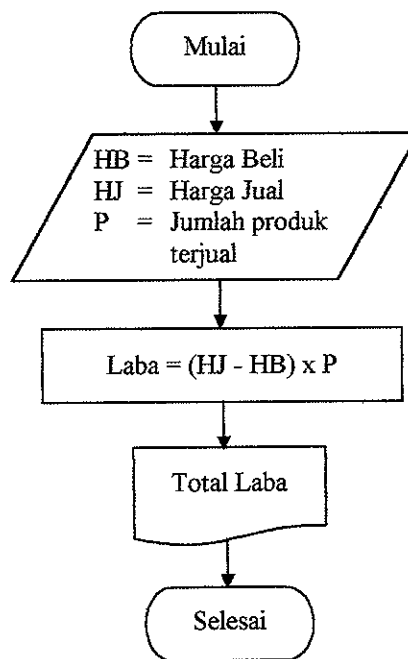
Pengembangan sistem pendukung keputusan diawali dengan pengolahan data persediaan secara menyeluruh. Tujuannya adalah basis data harus mampu menyediakan informasi jumlah persediaan untuk setiap produk setiap saat dibutuhkan. Data yang diolah berasal dari bagian pembelian dan penjualan.



Gambar III.7. Jumlah persediaan dalam sistem



Basis data akan mengolah jumlah persediaan terakhir yang dimiliki oleh swalayan berdasarkan total persediaan sebelumnya, total pembelian, total retur dan total penjualan. Sisa stok pada gambar III.7., merupakan informasi yang didapat dari pengolahan data-data tersebut. Perhitungannya dilakukan setiap kali terjadi input transaksi pembelian dan penjualan selesai dilakukan. Informasi tersebut merupakan hasil olahan terakhir atas seluruh transaksi. Proses tersebut berlaku untuk setiap produk. Sehingga manajemen swalayan dapat memeriksa jumlah persediaan yang dimiliki sewaktu-waktu dan memperoleh informasi yang terbaru.



Gambar III.8. Total Laba Harian Penjualan

Proses berikutnya adalah menentukan laba harian yang dimiliki oleh swalayan. Tujuannya adalah agar dapat diketahui jumlah laba yang dihasilkan oleh swalayan dari hasil transaksi yang dilakukan pada hari tersebut. Hal ini perlu dilakukan agar dapat dengan mudah memantau tingkat laba yang diperoleh swalayan setiap hari.

Seperti halnya pengolahan data persediaan, pengolahan data laba meliputi data yang dimasukkan oleh bagian penjualan dan pembelian. Hitungan pada



gambar III.8. merupakan proses pengolahan data yang berasal dari total penjualan. Dari seluruh transaksi penjualan pada hari itu kemudian dihitung laba yang dihasilkan oleh setiap produk dengan mengurangi harga penjualan dengan harga beli dikalikan dengan jumlah produk terjual. Kemudian seluruh laba dijumlahkan hingga diperoleh laba atas seluruh penjualan pada hari itu. Proses pengolahan data tersebut juga berlangsung setelah terjadi input transaksi pada basis data.

### III.3.2. Penentuan Kriteria Penjualan

Setelah menentukan laba harian, langkah berikutnya adalah menentukan kriteria penjualan setiap produk. Proses penentuan kriteria produk ini dilakukan berdasarkan jumlah penjualan masing-masing produk pada suatu periode yang sama. Hasil penjualan tersebut diolah sehingga diperoleh data tingkat penjualan masing-masing produk pada suatu periode tertentu.

Pada model ini, periode basis data adalah enam bulan. Penentuan waktu enam bulan, berdasarkan data swalayan yaitu produk yang dijual adalah produk-produk siap pakai dan umumnya adalah kebutuhan sehari-hari. Sehingga waktu enam bulan cukup menggambarkan apakah suatu produk laku di pasar atau tidak.

Penentuan kriteria penjualan bertujuan untuk mengklasifikasikan produk-produk sesuai dengan tingkat penjualan yang dihasilkan selama enam bulan. Dari hasil tersebut, akan diperoleh data penjualan produk sesuai dengan kriteria penjualan yang dimiliki. Pentingnya proses tersebut adalah karena hal itu dapat membantu proses penentuan jumlah persediaan yang harus dimiliki swalayan untuk memenuhi kebutuhan konsumen. Proses ini belum dimiliki oleh swalayan, sehingga pada penentuan kriteria, penulis menggunakan asumsi yang didasarkan atas informasi dari swalayan.

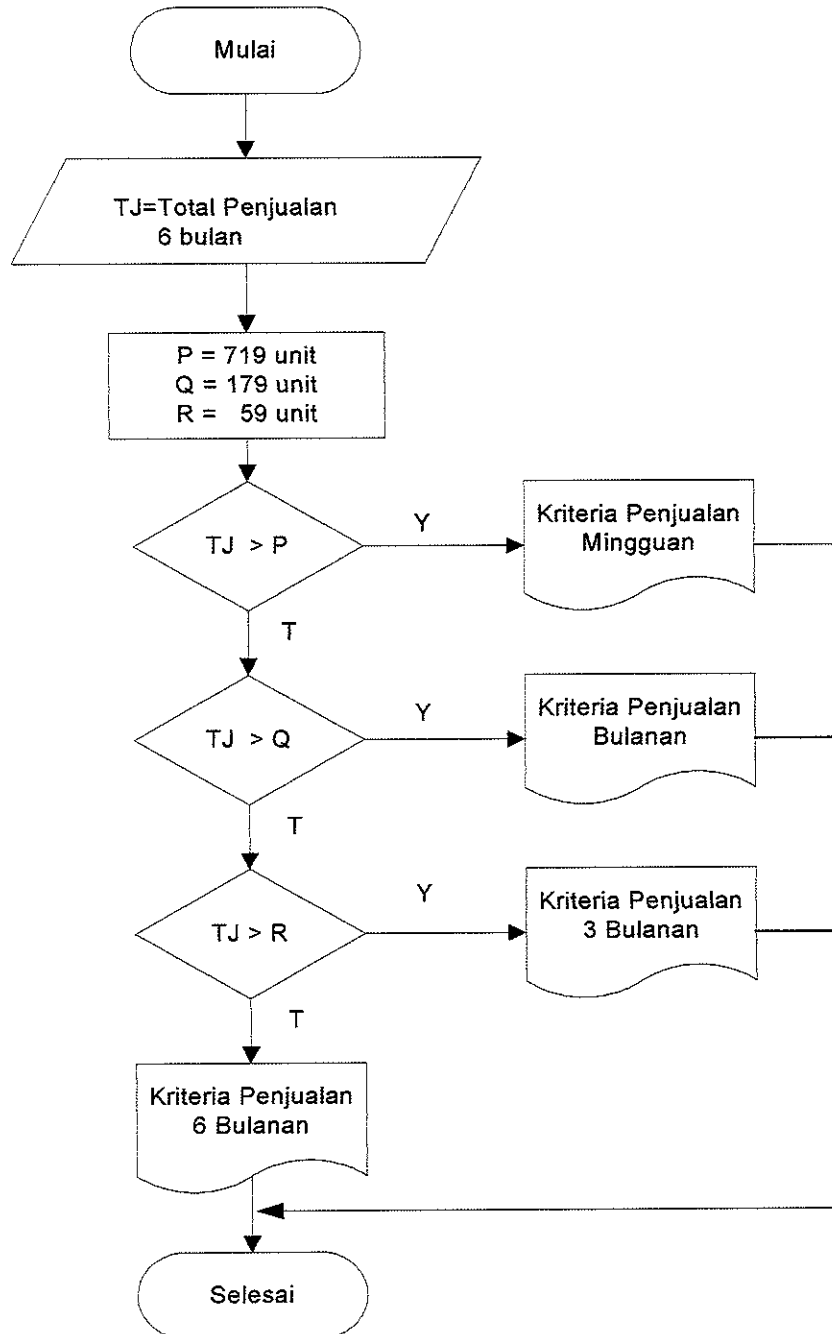
Proses penentuan kriteria penjualan didasarkan atas informasi dari bagian penjualan atas produk-produk yang secara umum memiliki kategori cepat laku dengan rata-rata penjualan 30-40 unit per minggu. Dari informasi tersebut kemudian disusun kriteria penjualan produk yang mungkin terjadi dan batas dalam penentuan kriteria sebagai berikut:



1. Kriteria penjualan produk mingguan – merupakan kriteria penjualan untuk produk-produk dengan rata-rata penjualan 30-40 unit per minggu, sehingga kemudian digunakan batas angka penjualan terendah sama dengan 30 unit per minggu untuk kriteria penjualan tersebut. Alasan penggunaan mingguan sebagai acuan penentuan kriteria adalah, dalam waktu tersebut dapat dilihat data penjualan nyata atas produk-produk yang memiliki kategori cepat laku.
2. Kriteria penjualan produk bulanan – penentuan kriteria penjualan bulanan merupakan analogi dari kriteria mingguan karena swalayan tidak memiliki data rinci mengenai kriteria ini, kemudian diasumsikan bahwa rata-rata penjualan bulanan adalah sama, dengan periode yang lebih lama atau 30-40 unit per bulan. Sehingga kemudian dapat ditentukan batas angka penjualan terendah adalah 30 unit per bulan untuk kriteria penjualan bulanan. Kriteria penjualan bulanan dipilih sebagai acuan penentuan kriteria karena pada periode tersebut akan diperoleh data penjualan produk-produk dengan tingkat penjualan di bawah kriteria penjualan mingguan dengan lebih lengkap;
3. Kriteria penjualan produk 3 bulanan – seperti halnya kriteria bulanan, kriteria ini juga tidak memiliki data yang rinci, hingga kemudian ditentukan analogi yang sama untuk kriteria tersebut berupa batas terendah untuk kriteria penjualan 3 bulanan adalah 30 unit per 3 bulan atau 10 unit per bulan. Penentuan kriteria 3 bulanan karena dengan waktu 3 bulan dapat diperoleh penjualan produk-produk yang memiliki kriteria penjualan di bawah kriteria penjualan bulanan;
4. Kriteria penjualan produk 6 bulanan – kriteria ini akan menunjukkan produk-produk yang memiliki tingkat penjualan terendah dengan tingkat penjualan di bawah kriteria penjualan 3 bulanan. Penentuannya dilakukan dengan acuan bahwa waktu enam bulan merupakan waktu yang cukup untuk mengevaluasi suatu produk konsumsi laku di pasar atau tidak.



Karena proses evaluasi sistem pendukung keputusan dilakukan dengan data selama enam bulan maka seluruh proses penghitungan dilakukan dalam waktu enam bulan. Hingga kemudian didapat batas rerata dalam satuan waktu yang sama yaitu enam bulan.



Gambar III.9. Diagram alir penentuan kriteria penjualan produk



Perhitungannya adalah sebagai berikut:

- kriteria mingguan:  $30 \text{ unit/minggu} \times (4\text{minggu/bulan} \times 6 \text{ bulan}) = 720$  unit,  
untuk perhitungan digunakan  $\geq 720$  atau  $> 719$  unit per enam bulan;
- kriteria bulanan:  $30 \text{ unit/bulan} \times 6 \text{ bulan} = 180$  unit,  
untuk perhitungan digunakan  $\geq 180$  atau  $> 179$  unit per enam bulan;
- kriteria 3 bulanan:  $10 \text{ unit/bulan} \times 6 \text{ bulan} = 60$  unit,  
untuk perhitungan digunakan  $\geq 60$  atau  $> 59$  unit per enam bulan;
- kriteria 6 bulanan:  $< 59$  unit per enam bulan.

### III.3.3. Penentuan Titik Pemesanan Ulang

Proses selanjutnya adalah menentukan titik pemesanan kembali (*reorder point*) pada setiap produk yang dimiliki oleh swalayan tersebut. Titik pemesanan kembali yang dihasilkan dari proses tersebut dapat dijadikan acuan bagi swalayan dalam menentukan batas persediaan minimum sebelum melakukan pemesanan ulang atas produk yang bersangkutan. Batas tersebut yang kemudian dijadikan acuan dalam penentuan waktu pemesanan ulang. Jika persediaan suatu produk telah berada pada titik tersebut, maka produk tersebut harus segera dipesan untuk memenuhi kebutuhan penjualan periode berikutnya. Untuk menentukan nilai titik pemesanan kembali digunakan rumus *reorder point* sebagai berikut (McLeod, 1996):

$$R = LU + S \quad (\text{III.1})$$

Dengan  $R$  = Titik Pemesanan Kembali (ROP)

$L$  = *Lead Time* Pemasok (dalam satuan hari)

$U$  = Rerata Penjualan (dalam satuan hari)

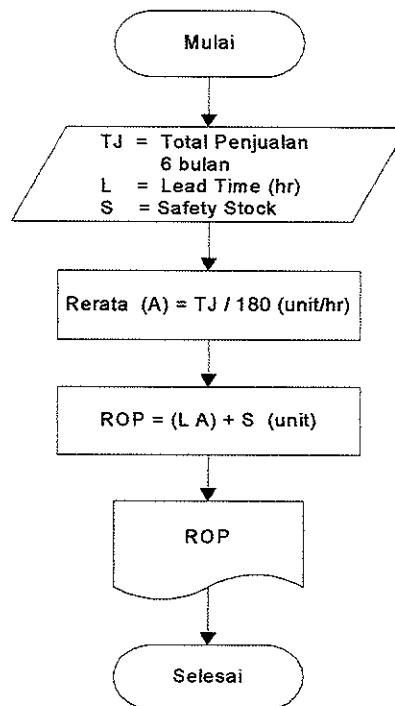
$S$  = *Safety Stock* (dalam satuan unit)



Proses ini meliputi data dari bagian penjualan berupa jumlah penjualan setiap produk selama periode waktu tertentu yang kemudian diolah bersama dengan waktu tenggang pemesanan (*lead time*) dari *supplier* dan keputusan manajemen swalayan atas jumlah cadangan persediaan (*safety stock*) yang dimungkinkan.

*Lead time supplier* dapat ditentukan berdasarkan waktu yang diperlukan dari saat pesan hingga produk datang. Sedangkan *safety stock* sebagai jumlah cadangan persediaan, amat ditentukan oleh kebijakan swalayan. Namun pada kenyataannya swalayan hanya memiliki *safety stock* untuk produk-produk yang cepat laku, sebesar 1 lusin (12 unit). Hingga kemudian, penulis menggunakan asumsi dalam menentukan *safety stock* untuk produk-produk dengan kriteria penjualan lainnya. Perhitungannya adalah sebagai berikut:

- kriteria penjualan mingguan: *safety stock* = 12 unit;
- kriteria penjualan bulanan: *safety stock* = 9 unit;
- kriteria penjualan 3 bulanan: *safety stock* = 6 unit;
- kriteria penjualan 6 bulanan: *safety stock* = 3 unit.



Gambar III.10. Diagram Alir Sistem Pendukung Keputusan Titik Pemesanan Kembali



### III.3.4. Penentuan Jumlah Pemesanan Ulang

Langkah berikutnya adalah menentukan jumlah pemesanan ulang yang harus dilakukan untuk memenuhi penjualan periode berikutnya. Seperti telah dijelaskan pada bab sebelumnya, prosedur ini menggunakan rerata linear bergerak. Alasan penggunaan rerata linear bergerak adalah karena peramalan yang dilakukan dipengaruhi oleh *trend*, dan data yang tersedia hanyalah data penjualan. Peramalan dilakukan untuk dua bulan ke depan karena pada saat evaluasi, belum diperoleh data penjualan untuk bulan terakhir secara lengkap. Sehingga digunakan peramalan dua bulan ke muka.

Untuk menentukan jumlah pemesanan ulang untuk periode berikutnya digunakan rumus rerata linear bergerak sebagai berikut (Makridakis, 1983):

$$S'_t = \frac{X_t + X_{t-1} + X_{t-2} + \dots + X_{t-N+1}}{N} \quad (\text{III.2})$$

$$S''_t = \frac{S'_t + S'_{t-1} + S'_{t-2} + \dots + S'_{t-N+1}}{N} \quad (\text{III.3})$$

$$a_t = 2S'_t - S''_t \quad (\text{III.4})$$

$$bt = \frac{2}{N-1}(S'_t - S''_t) \quad (\text{III.5})$$

$$F_{t+m} = a_t + b_t m \quad (\text{III.6})$$

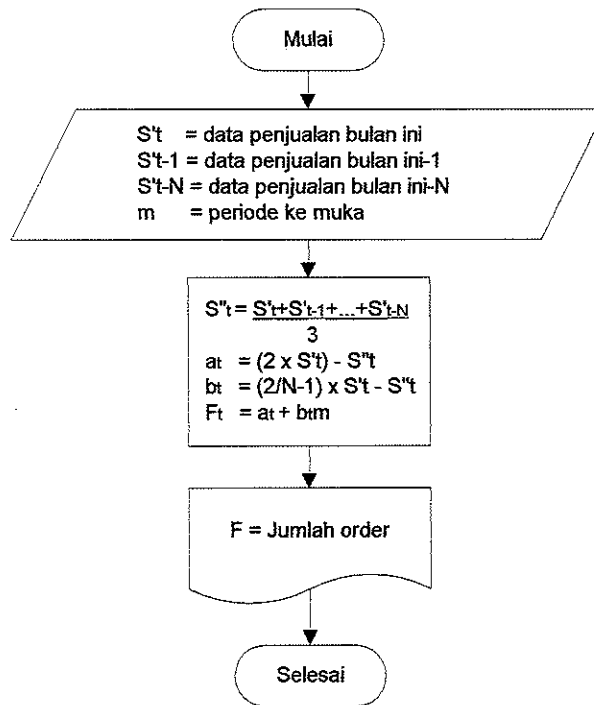
- Dengan
- X = data penjualan selama satu periode
  - t = periode
  - N = jumlah periode
  - S' = rerata bergerak tunggal
  - S'' = rerata bergerak kedua
  - a = penyesuaian terhadap rerata bergerak tunggal
  - b = estimasi *trend* dari periode satu ke periode berikutnya
  - F = prakiraan untuk periode ke muka dari periode t



prakiraan untuk  $m$  periode ke muka adalah  $a_t$ , yang merupakan nilai rerata yang disesuaikan untuk periode  $t$  ditambah  $m$  kali komponen trend  $b_t$ .

$m$  = periode ke muka  
(Makridakis, 1983)

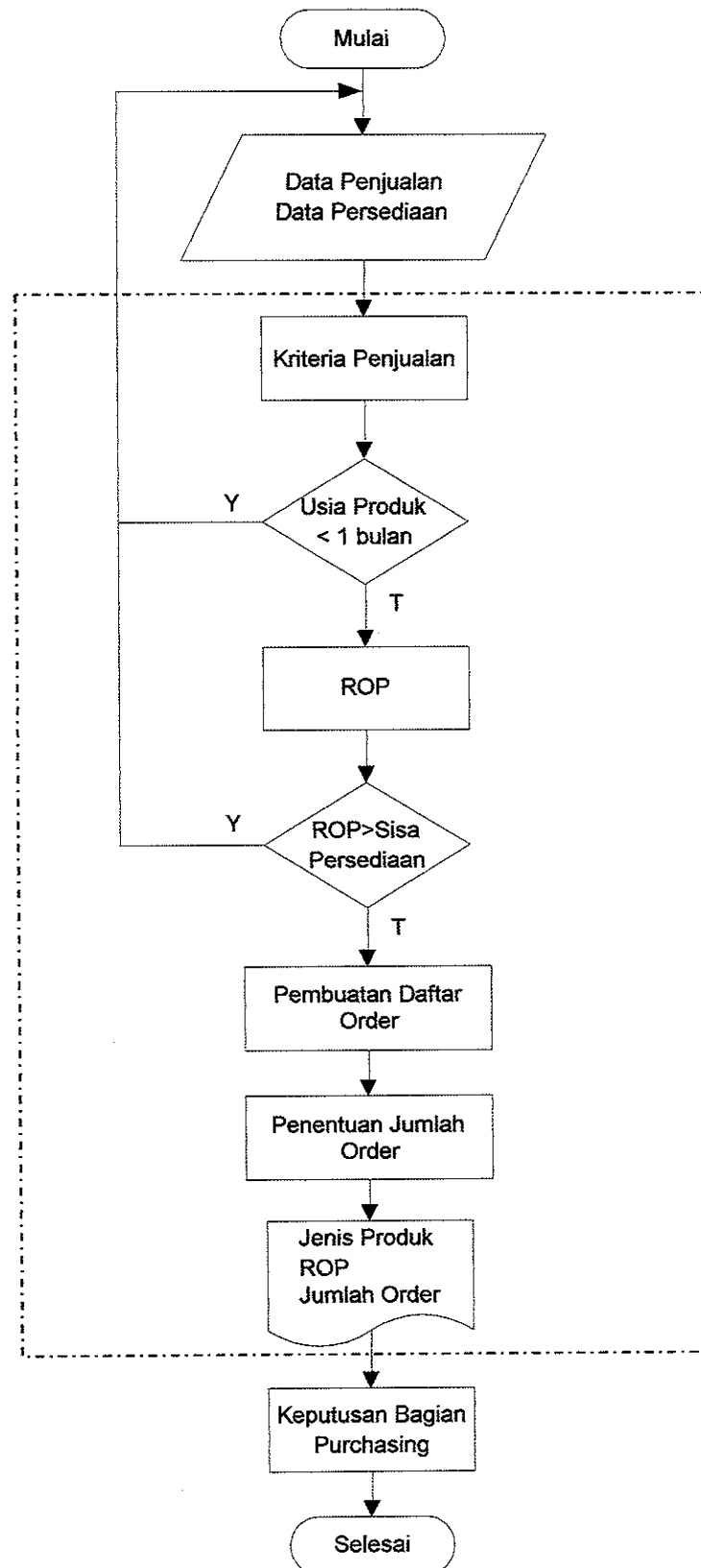
Sistem pendukung keputusan menggunakan umur basis data selama enam bulan. Dengan umur tersebut, rerata bergerak linear yang digunakan adalah rerata linear bergerak  $3 \times 3$ . Alasan penggunaan metode tersebut, karena umur basis data yang relatif singkat dan perhitungan yang didapat akan lebih baik dibanding metode rerata linear bergerak dengan periode yang lebih kecil.



Gambar III.11. Diagram alir penentuan jumlah order

### III.3.5. Model Penggunaan Sistem Pendukung Keputusan

Sistem pendukung keputusan ini akan berujung pada informasi jumlah sisa persediaan yang masih ada, batas sebelum melakukan pemesanan ulang, kriteria penjualan, dan jumlah pemesanan yang harus dilakukan untuk setiap produk.



Gambar III.12. Informasi yang dihasilkan oleh sistem pendukung keputusan



Gambar III.12 menjelaskan langkah-langkah pembentukan informasi dengan sistem pendukung keputusan yang berasal dari data-data penjualan. Data tersebut diolah dengan sistem pendukung keputusan, yang berada dalam kotak, hingga menghasilkan informasi jenis, waktu, dan jumlah produk yang dibutuhkan oleh pihak manajemen khususnya bagian pembelian untuk melakukan pemesanan ulang. Secara rinci langkah-langkah tersebut dijelaskan sebagai berikut:

1. data penjualan sebagai bahan utama pengolahan dalam sistem pendukung keputusan;
2. sistem pendukung keputusan dimulai dengan mengolah data penjualan selama periode basis data menjadi informasi kriteria penjualan produk;
3. setiap produk yang dievaluasi harus memiliki data penjualan minimal 1 bulan, sehingga untuk produk baru yang belum memiliki data penjualan selama 1 bulan, tidak diikuti dalam evaluasi produk berikutnya;
4. produk-produk yang telah memiliki data penjualan minimal 1 bulan dan memiliki kriteria penjualan minimal, maka produk-produk tersebut langsung berada dalam evaluasi manajemen, apakah produk-produk tersebut akan mengalami pemesanan ulang untuk memenuhi kelengkapan swalayan atau tidak akan dipesan lagi;
5. langkah berikutnya dalam sistem pendukung keputusan adalah menentukan nilai *reorder point*;
6. jika suatu produk memiliki sisa persediaan yang sama dengan nilai *reorder point*, maka produk tersebut harus segera dipesan untuk memenuhi kebutuhan konsumen periode berikutnya, namun jika sisa persediaan masih mencukupi, maka produk tersebut akan menunggu hingga data persediaan menunjukkan nilai yang sama dengan *reorder point* untuk dapat dipesan ulang;
7. jumlah pemesanan ulang produk-produk ditentukan dengan proses *forecasting* berdasarkan rerata penjualan selama beberapa bulan sebelumnya;
8. hasil dari sistem pendukung keputusan ini merupakan informasi berupa jenis produk mana yang mengalami pemesanan ulang, waktu pemesanan



ulang berikutnya, dan jumlah produk yang harus dipesan untuk periode berikutnya;

9. dari *report* tersebut, manajemen dapat memiliki informasi untuk memutuskan permasalahan pemesanan ulang tiap produk, namun tetap merupakan tugas manajemen memutuskan solusi terbaik sesuai kondisi swalayan.

Tipe permasalahan yang dihadapi oleh manajemen tersebut dikategorikan sebagai permasalahan semi terstruktur yang memerlukan dukungan komputer dalam menentukan solusi terbaik. Sebagai permasalahan semi terstruktur, pemesanan ulang merupakan permasalahan rutin yang dipengaruhi oleh berbagai macam variabel. Informasi yang diberikan oleh sistem pendukung keputusan hanyalah merupakan salah satu alternatif yang kemudian masih memerlukan evaluasi lanjut dari pihak manajemen. Evaluasi manajemen tersebut yang kemudian menghasilkan keputusan.

Dari informasi yang diberikan oleh sistem pendukung keputusan, manajemen harus melakukan evaluasi terhadap pelaksanaan alternatif tersebut sebagai suatu keputusan. Banyak hal yang mempengaruhi suatu keputusan, namun dengan informasi yang diberikan oleh sistem pendukung keputusan, manajemen dapat lebih mudah dalam melakukan evaluasi hingga menghasilkan keputusan. Karena tujuan sistem pendukung keputusan hanyalah membantu proses pengambilan keputusan dalam permasalahan semi terstruktur dan tidak berusaha untuk menggantikan fungsi manajemen dalam menentukan keputusan.

### **III.3.6. Model Matematis Penggunaan Sistem Pendukung Keputusan**

Pada bagian ini akan diberikan contoh penggunaan sistem pendukung keputusan yang ditampilkan dengan model matematis. Disini dapat dengan jelas terlihat proses perhitungan yang terjadi dari data mentah berupa data penjualan hingga menjadi informasi yang dibutuhkan oleh bagian pembelian untuk melakukan pemesanan ulang. Berikut adalah contoh data-data penjualan selama



enam bulan dari bulan Juni hingga minggu ketiga bulan Nopember, disajikan dalam bentuk tabel:

Tabel III.1. Data penjualan bulan Juni

Kode	Nama	Minggu 1	Minggu 2	Minggu 3	Minggu 4	Jumlah
41209	BBF Pouch AS 250ml	0	0	0	0	0
20805	Bayclean R-500ml	4	0	1	1	6
10705	Sunlight Refill 400ml	4	1	2	3	10
20105	Softener Molto Sachet	16	17	20	34	87
270101	Indomie Goreng	0	0	0	142	142

Tabel III.2. Data penjualan bulan Juli

Kode	Nama	Minggu 1	Minggu 2	Minggu 3	Minggu 4	Jumlah
41209	BBF Pouch AS 250ml	0	0	0	0	0
20805	Bayclean R-500ml	2	2	1	1	6
10705	Sunlight Refill 400ml	4	2	3	3	12
20105	Softener Molto Sachet	24	40	27	54	145
270101	Indomie Goreng	89	84	88	99	360

Tabel III.3. Data penjualan bulan Agustus

Kode	Nama	Minggu 1	Minggu 2	Minggu 3	Minggu 4	Jumlah
41209	BBF Pouch AS 250ml	0	0	0	0	0
20805	Bayclean R-500ml	0	0	0	2	2
10705	Sunlight Refill 400ml	4	2	4	0	10
20105	Softener Molto Sachet	12	17	10	33	72
270101	Indomie Goreng	115	51	58	61	285

Tabel III.4. Data penjualan bulan September

Kode	Nama	Minggu 1	Minggu 2	Minggu 3	Minggu 4	Jumlah
41209	BBF Pouch AS 250ml	0	0	0	0	0
20805	Bayclean R-500ml	2	0	0	3	5
10705	Sunlight Refill 400ml	10	6	3	2	21
20105	Softener Molto Sachet	33	21	18	20	92
270101	Indomie Goreng	107	118	106	87	418



**Tabel III.5. Data penjualan bulan Oktober**

Kode	Nama	Minggu 1	Minggu 2	Minggu 3	Minggu 4	Jumlah
41209	BBF Pouch AS 250ml	0	0	0	0	0
20805	Bayclean R-500ml	2	0	0	1	3
10705	Sunlight Refill 400ml	7	8	5	6	26
20105	Softener Molto Sachet	17	4	8	16	45
270101	Indomie Goreng	118	62	98	172	450

**Tabel III.6. Data penjualan bulan Nopember**

Kode	Nama	Minggu 1	Minggu 2	Minggu 3	Minggu 4	Jumlah
41209	BBF Pouch AS 250ml	0	2	0	0	2
20805	Bayclean R-500ml	1	1	0	0	2
10705	Sunlight Refill 400ml	2	2	2	0	6
20105	Softener Molto Sachet	22	10	16	0	48
270101	Indomie Goreng	71	70	80	0	221

**Tabel III.7. Data persediaan akhir Nopember**

Kode	Nama	Sisa Stok (SS)	Lead Time Supplier (L)
41209	BBF Pouch AS 250ml	3	5 hari
20805	Bayclean R-500ml	5	9 hari
10705	Sunlight Refill 400ml	11	9 hari
20105	Softener Molto Sachet	544	5 hari
270101	Indomie Goreng	291	7 hari

**Tabel III.8. Penentuan kriteria penjualan**

Kode	Nama	Total Penjualan (T)	Kriteria	Usia Data Penjualan
41209	BBF Pouch AS 250ml	3	D	2 minggu
20805	Bayclean R-500ml	24	D	> 1 bulan
10705	Sunlight Refill 400ml	90	C	> 1 bulan
20105	Softener Molto Sachet	493	B	> 1 bulan
270101	Indomie Goreng	1898	A	> 1 bulan



Tabel III.9. Penentuan nilai *reorder point*

Kode	Nama	K	L	$U = T/180$	S	$ROP = LU+S$	Ket.	Order
20805	Bayclean R-500ml	D	9	0,13	3	$4,20 \approx 5$	SS=ROP	Ya
10705	Sunlight Refill 400ml	C	9	0,47	6	$10,25 \approx 11$	SS=ROP	Ya
20105	Softener Molto Sachet	B	5	2,72	9	$22,58 \approx 23$	SS>ROP	Tidak
270101	Indomie Goreng	A	7	10,42	12	$84,96 \approx 85$	SS>ROP	Tidak

Penentuan jumlah order:

ST1 = rerata penjualan tiga bulanan dari data penjualan bulan terakhir -1, bulan terakhir -2, dan bulan terakhir -3;

ST2 = rerata penjualan tiga bulanan dari data penjualan bulan terakhir-2, bulan terakhir-3, dan bulan terakhir-4

ST3 = rerata penjualan tiga bulanan dari data penjualan bulan terakhir-3, bulan terakhir-4, dan bulan terakhir-5

SST = rerata dari ST1, ST2, dan ST3

FT = prakiraan jumlah penjualan untuk periode berikutnya = jumlah pemesanan periode berikutnya

Tabel III.10. Penentuan jumlah pemesanan ulang untuk bulan Desember

Kode	ST1	ST2	ST3	SST	$AT=2ST1-SST$	$BT=(2/(N-1))(ST1-SST)$	$F=A+2B$
20805	3,33	4,33	4,67	4,11	2,56	-0,78	$2,67 \approx 3$
10705	19,00	14,33	10,67	14,67	23,33	4,33	32
20105	69,67	103,00	101,33	91,33	48,00	-21,67	$4,67 \approx 7$
270101	384,33	354,33	262,33	333,67	435,00	50,67	$536,33 \approx 537$

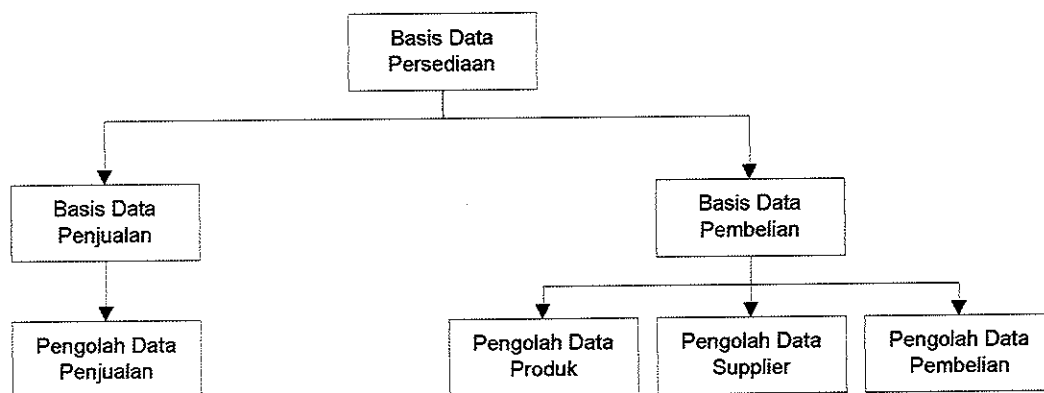
## BAB IV

### PEMBUATAN BASIS DATA DAN SISTEM PENDUKUNG KEPUTUSAN SERTA PENGGUNAANNYA

#### IV.1. PEMBUATAN DAN PENGGUNAAN BASIS DATA

Pada perancangan ini, basis data dikembangkan dengan menggunakan Microsoft Access 2000. Sistem basis data dibangun dengan menggunakan tabel, *query*, form, dan *report*. Berdasarkan model sistem informasi yang telah dibahas sebelumnya, maka basis data persediaan akan dibangun dari tiga subsistem utama, yaitu bagian pembelian dan penjualan.

Setiap subsistem tersebut akan bertanggung jawab atas input seluruh data transaksi yang terdapat pada bagiannya masing-masing. Input dilakukan segera setelah transaksi terjadi, sehingga basis data mampu mengolah data dan menghasilkan informasi terbaru. Penjelasan mengenai pembuatan basis data pada setiap bagian akan dijelaskan di bawah ini.



Gambar IV.1. Komponen basis data persediaan

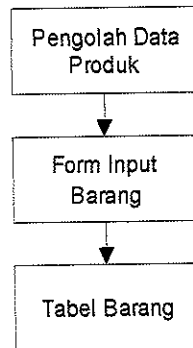
##### IV.1.1. Basis Data Pembelian

Basis data yang ada pada bagian pembelian, tersusun atas basis data pengolahan data produk, data *supplier*, dan data pembelian. Ketiga bagian tersebut akan mengolah seluruh data produk yang berada dalam swalayan. Baik berupa penambahan maupun pengurangan jumlah produk. Basis data tersebut akan dibahas satu per satu berikut ini.



#### IV.1.1.1 Pengolahan Data Produk

Fungsi basis data pengolahan data produk adalah sebagai pusat data seluruh produk yang dimiliki oleh swalayan. Basis data ini digunakan untuk menyimpan data-data produk yang menjadi persediaan dalam swalayan maupun untuk memasukkan data produk baru yang sebelumnya belum pernah dimiliki oleh swalayan.



Gambar IV.2. Komponen penyusun pengolah data produk

Field Name	Data Type	Description
KODE	Number	
NAMA	Text	
HARGA BELI	Number	
HARGA JUAL	Number	
STOK	Number	
KODE SUPPLIER	Number	

Field Properties	
General	Lookup
Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	0
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (No Duplicates)

Gambar IV.3. Tabel Barang

Basis data ini dibangun dari tabel Barang dan form Input Barang. *Field* tabel Barang terdiri atas: kode untuk kode produk, nama untuk nama produk,



harga beli untuk harga beli produk, harga jual untuk harga jual produk, stok untuk jumlah persediaan awal yang dimiliki swalayan, kode *supplier*.

*Field* kode menjadi *primary key*. *Primary key* adalah satu atau lebih kolom *field* yang memiliki nilai yang secara unik mengidentifikasi suatu *record* pada tabel. *Primary key* tidak boleh memiliki nilai *Null* dan harus selalu memiliki indeks unik. *Primary key* digunakan untuk menghubungkan suatu tabel dengan *foreign key* yang terdapat pada tabel lain. *Foreign key* adalah kolom *field* suatu tabel yang mengacu pada *primary key* tabel lain. *Foreign key* akan mengindikasikan bagaimana bentuk relasi antar tabel (Microsoft Access Help).

FORM DATA BARANG	
KODE	<input type="text"/>
NAMA BARANG	<input type="text" value="BUKU TULIS"/>
HARGA BELI	<input type="text" value="1.500,00"/>
HARGA JUAL	<input type="text" value="2.000,00"/>
STOK	<input type="text" value="500"/>
KODE SUPPLIER	<input type="text" value="1"/>

Record: 1 of 3

Gambar IV.4. Form Input Barang

Form Input Barang, dibuat dari tabel Barang, dengan bentuk kolumnar. Pada form ini, ditambahkan tombol *Add* untuk memasukkan data produk baru dan *Close* untuk menutup form dengan cepat. Pada form ini, *navigation button* ditampilkan untuk mempermudah pengguna menuju *record* sesudah maupun sebelumnya. Begitu juga dengan *tab order* yang disusun sedemikian rupa agar pengguna dapat memfungsikan form ini dengan *keyboard*.

Pada penggunaannya, form ini juga menggunakan aplikasi Visual Basic for Application (VBA) agar pengguna dapat lebih mudah memfungsikan form ini. Penerapan VBA ada pada *field* kode *supplier*, tombol *Add* dan *Close*. Bahasa VBA tersebut dituliskan pada *event procedure* sebuah *field*, properti *event*.



Pada *field* kode *supplier*, *event procedure* diberikan pada *event: on lost focus*. Bahasa VBA tersebut berfungsi untuk memindahkan *tab order* dari bagian detail ke bagian *footer* yaitu tombol *Add*.

Pada tombol *Add*, *event procedure* diberikan pada *event: on click*. Prosedur tersebut untuk menjalankan fungsi tombol *Add*, yaitu untuk memindahkan *tab order* ke halaman form *record* baru jika pengguna menekan *enter* pada *keyboard* dan memindahkan *tab order* dari tombol *Add* halaman sebelumnya ke *field* kode halaman form *record* baru tersebut. Sehingga pengguna dapat langsung mengisi form tersebut tanpa harus menggunakan *mouse*.

Untuk tombol *Close*, *event procedure* diberikan pada *event: on lost focus* dan *on click*. Prosedur pada *on lost focus*, berkaitan dengan perpindahan *tab order*. Prosedur tersebut memungkinkan *tab* untuk dapat berpindah ke *field* kode setelah fokus pada tombol *Close*.

Prosedur pada *field* kode *supplier*:

```
Private Sub KODE_SUPPLIER_LostFocus()  
Me.ADD_REC.SetFocus  
End Sub
```

Berikut adalah prosedur pada tombol *Add*:

```
Private Sub ADD_REC_Click()  
On Error GoTo Err_ADD_REC_Click  
  
DoCmd.GoToRecord , , acNewRec  
Me.KODE.SetFocus  
Exit_ADD_REC_Click:  
Exit Sub  
  
Err_ADD_REC_Click:  
MsgBox Err.Description  
Resume Exit_ADD_REC_Click  
  
End Sub
```

Prosedur tombol *Close*:

```
Private Sub Close_LostFocus()  
Me.KODE.SetFocus  
End Sub
```



Field Name	Data Type	Description
KODE SUPPLIER	Number	
NAMA SUPPLIER	Text	
ALAMAT	Text	
TELEPON	Text	
LEAD TIME	Number	

Field Properties	
General	Lookup
Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	0
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (No Duplicates)

Gambar IV.6. Tabel *Supplier*

Form Input *Supplier*, dibuat dari tabel *Supplier*, dengan bentuk kolumnar. Pada form ini, juga ditambahkan tombol *Add* dan *Close*. Dan untuk memudahkan pengguna, form ini juga dikembangkan dengan menggunakan VBA. Sehingga pengguna dapat bekerja dengan lebih cepat.

Bahasa VBA terletak pada *field* telepon, tombol *Add*, dan *Close*. Fungsi *event procedure* pada *field* telepon adalah memindahkan *tab order* dari *field* telepon pada bagian detail ke tombol *Add* di bagian *footer*. Prosedur tersebut dilakukan pada *event on lost focus*. Berikut adalah prosedurnya:

```
Private Sub TELEPON_LostFocus()  
Me.add.SetFocus  
End Sub
```

Pada tombol *Add*, prosedur berfungsi untuk memindahkan *tab order* ke halaman form *record* baru pada *field* kode *supplier*. Sehingga pengguna tidak perlu menggunakan *mouse* untuk memindahkan *tab order* untuk mengisi form tersebut. Prosedur tersebut diberikan pada *event on click*, yang berarti prosedur akan berfungsi jika tombol tersebut diklik, baik dengan *enter* pada *keyboard* maupun dengan *mouse*. Prosedurnya adalah sebagai berikut:

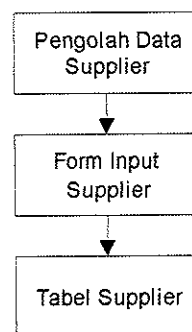


Pada *event on click*, prosedur tersebut untuk menjalankan fungsi tombol *Close*, yaitu menutup halaman form. Prosedur ini juga untuk memudahkan pengguna, sehingga saat pengguna telah selesai melakukan input data produk, dapat segera menutup form tersebut dengan menekan *enter* pada *keyboard*. Prosedur tersebut adalah sebagai berikut:

```
Private Sub Close_Click()  
On Error GoTo Err_Close_Click  
  
DoCmd.close  
  
Exit_Close_Click:  
Exit Sub  
  
Err_Close_Click:  
MsgBox Err.Description  
Resume Exit_Close_Click  
  
End Sub
```

#### IV.1.1.2. Pengolahan Data Supplier

Basis data pengolahan data *supplier* merupakan pusat data seluruh *supplier* yang bekerja sama dengan swalayan. Basis data ini digunakan untuk menyimpan data-data *supplier* yang telah menjadi rekan kerja swalayan maupun untuk memasukkan data *supplier* baru yang sebelumnya belum pernah bekerja sama dengan swalayan. Basis data ini dibangun dari tabel *Supplier* dan form Input *Supplier*. *Field* kode *supplier* pada tabel *Supplier* menjadi *primary key*.



Gambar IV.5. Komponen penyusun pengolah data *supplier*



```
Private Sub add_Click()  
On Error GoTo Err_add_Click  
  
DoCmd.GoToRecord , , acNewRec  
Me.KODE_SUPPLIER.SetFocus  
Exit_add_Click:  
Exit Sub  
  
Err_add_Click:  
MsgBox Err.Description  
Resume Exit_add_Click  
  
End Sub
```

Tombol *Close* memiliki prosedur pada *event on lost focus* dan *on click*. Pada *event on lost focus*, prosedur berfungsi untuk memindahkan *tab order* dari tombol *Close* ke *field* kode *supplier*. Prosedurnya adalah sebagai berikut:

```
Private Sub close_LostFocus()  
Me.KODE_SUPPLIER.SetFocus  
End Sub
```

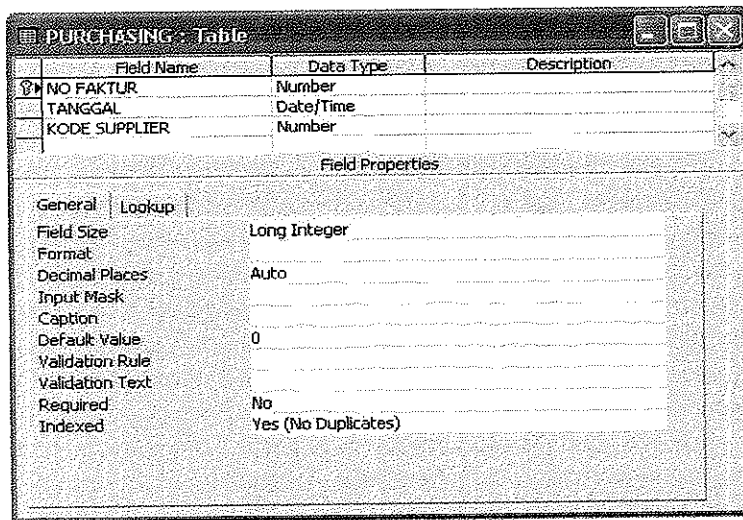
Gambar IV.7. Form Input *Supplier*

Sedangkan prosedur pada *event on click*, digunakan untuk menjalankan fungsi tombol *Close* yaitu menutup form. Prosedur ini dijalankan dengan mengklik *mouse* pada tombol tersebut atau menekan *enter* pada *keyboard*. Prosedur tombol *Close* adalah sebagai berikut:

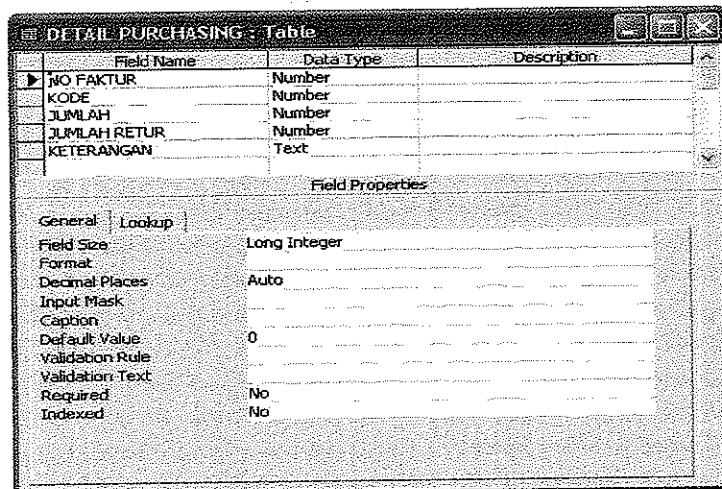
```
Private Sub Close_Click()  
On Error GoTo Err_Close_Click  
  
DoCmd.close
```



Form *Purchasing*, disusun dari tabel *Supplier*, tabel *Purchasing* dan form Detail *Purchasing*. Tabel *Supplier* dan tabel *Purchasing* diolah dengan menggunakan *query Purchasing1*. Bentuk relasi kedua tabel tersebut adalah *one to many*. *One to many relationship* adalah hubungan antara dua tabel dimana nilai *primary key* setiap *record* pada *primary table*, berhubungan dengan nilai berbagai *record* yang sesuai pada *field*, tabel terhubung. *Primary table* adalah sebuah tabel dalam relasi *one to many* yang memiliki *primary key* dan setiap *record* di dalamnya harus memiliki nilai yang unik (Microsoft Access Help). *Query Purchasing1* merupakan *query* yang berbentuk *select query*.



Gambar IV.10. Tabel *Purchasing*



Gambar IV.11. Tabel Detail *Purchasing*



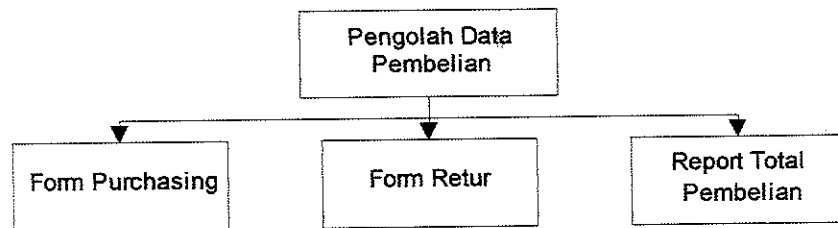
```
Exit_Close_Click:  
Exit Sub
```

```
Err_Close_Click:  
MsgBox Err.Description  
Resume Exit_Close_Click
```

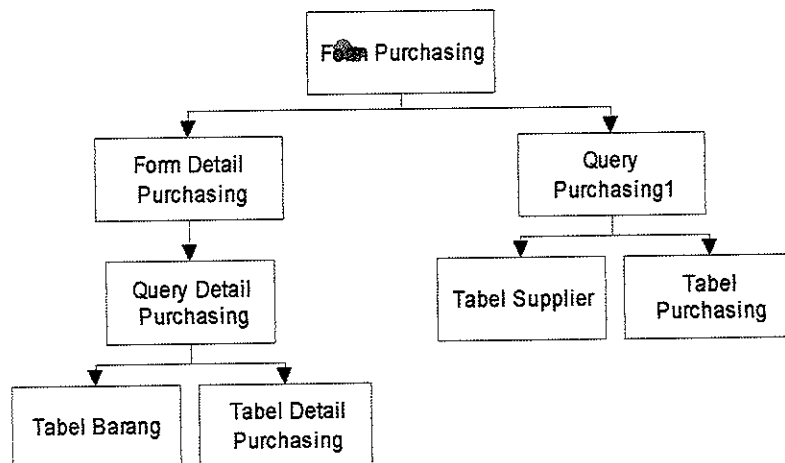
```
End Sub
```

#### IV.1.1.3. Pengolahan Data Transaksi Pembelian

Basis data pengolahan data transaksi pembelian berfungsi untuk menyimpan seluruh transaksi pembelian produk dan memasukkan data transaksi pembelian baru di swalayan. Basis data ini dibangun dari tabel *Barang*, tabel *Supplier*, tabel *Detail Purchasing*, tabel *Purchasing*, form *Detail Purchasing*, form *Purchasing*, form *Detail Retur*, dan form *Retur*. Pada tabel *Detail Purchasing*, tidak ada *field* yang menjadi *primary key*. Tabel *Purchasing* menggunakan *field no faktur* sebagai *primary key*.



Gambar IV.8. Komponen penyusun pengolah data pembelian



Gambar IV.9. Komponen penyusun form *Purchasing*



Form Detail *Purchasing* disusun dari tabel Barang dan tabel Detail *Purchasing*. Data pada kedua tabel tersebut diolah dengan menggunakan *query Query Detail Purchasing*. Bentuk relasi kedua tabel tersebut adalah *one to many*.

Tabel IV.1. *Query Purchasing1*

<i>Field</i>	Tabel	<i>Sort</i>	<i>Show</i>
No Faktur	<i>Purchasing</i>		√
Tanggal	<i>Purchasing</i>		√
Kode <i>Supplier</i>	<i>Purchasing</i>		√
Nama <i>Supplier</i>	<i>Supplier</i>		√

Tabel IV.2. *Query Query Detail Purchasing*

<i>Field</i>	Tabel	<i>Sort</i>	<i>Show</i>
No Faktur	Detail <i>Purchasing</i>		√
Kode	Detail <i>Purchasing</i>		√
Nama	Barang		√
Jumlah	Detail <i>Purchasing</i>		√
Harga Beli	Barang		√
Total: (IV.1)			√
Jumlah Retur	Detail <i>Purchasing</i>		√

*Query Query Detail Purchasing* merupakan *query* yang berbentuk *select query*. Pada *query* ini, selain berisi *field-field* yang berasal dari tabel Barang dan Tabel Detail *Purchasing*, juga ditambahkan sebuah *field* untuk menghitung total nilai pembelian, yaitu *field* total. Karena *query* ini nantinya juga akan digunakan dalam mengolah data untuk form Retur, maka perhitungan yang dilakukan juga melibatkan retur. *Field* tersebut memiliki perhitungan yang dapat dilihat dengan jelas pada *expression builder* dengan menekan tombol *Build*. Lebih jelasnya perhitungan total pada *query Query Detail Purchasing* adalah sebagai berikut:

$$\text{TOTAL: [BARANG].[HARGA BELI]*([DETAIL PURCHASING].[JUMLAH]-$$
$$[\text{DETAIL PURCHASING].[JUMLAH RETUR]}) \quad (\text{IV.1})$$



Pada form ini, ditambahkan *field* subtotal pada bagian *footer* yang berfungsi untuk menjumlahkan seluruh nilai dari *field* total. Perhitungan yang digunakan pada *field* subtotal adalah sebagai berikut:

$$\text{Subtotal} = \text{Sum}([\text{TOTAL}]) \quad (\text{IV.2})$$

The screenshot shows a software window titled 'PURCHASING' with a sub-header 'DATA PEMBELIAN'. It contains several input fields for 'NO FAKTUR', 'TANGGAL', 'KODE SUPPLIER', and 'NAMA SUPPLIER'. Below these is a table with columns 'KODE', 'NAMA BARANG', 'JUMLAH', 'HARGA BELI', and 'SUBTOTAL'. The table lists 'BALLPOINT' and 'PENSIL'. At the bottom right, there is a 'TOTAL' field showing '420.000,00'. Navigation controls for records are visible at the bottom.

KODE	NAMA BARANG	JUMLAH	HARGA BELI	SUBTOTAL
5	BALLPOINT	100	3.500,00	350.000,00
2	PENSIL	100	700,00	70.000,00
*				

Gambar IV.12. Form *Purchasing*

Bentuk desain Form *Purchasing* adalah kolumnar. Pada form ini, form *Detail Purchasing* ditampilkan sebagai subform pada bagian detail. Di bagian *footer* form ini ditambahkan *field* total. Fungsinya adalah menampilkan nilai pada subtotal form *Detail Purchasing*. Namun karena nilai yang hendak ditampilkan berasal dari subform, maka nilai tersebut tidak dapat langsung terlihat pada form *Purchasing*. Sehingga dibuat *field* total pada form *Purchasing* yang nilainya berasal dari subtotal pada subform. Untuk menghindari kerancuan nilai yang sama pada dua *field* yang berbeda, maka *field* subtotal pada subform *Detail Purchasing* disembunyikan. Perhitungan yang digunakan pada *field* total untuk mengambil nilai pada subtotal adalah sebagai berikut:

$$\text{Total} = [\text{DETAIL PURCHASING}].\text{Form!SUBTOTAL} \quad (\text{IV.3})$$

Input pada form *Purchasing*, dilakukan pada kode *supplier*, kode produk, dan jumlah pembelian. Sementara pada *field* no faktur, tanggal, dan total pada



form *Purchasing*, nama *supplier*, nama produk, harga beli, subtotal, pada subform *Detail Purchasing*, tidak bisa diakses, agar tidak terjadi kerancuan data. Untuk *field* no faktur, pada properti *Data*, nilai pada *default value* dituliskan sebagai berikut:

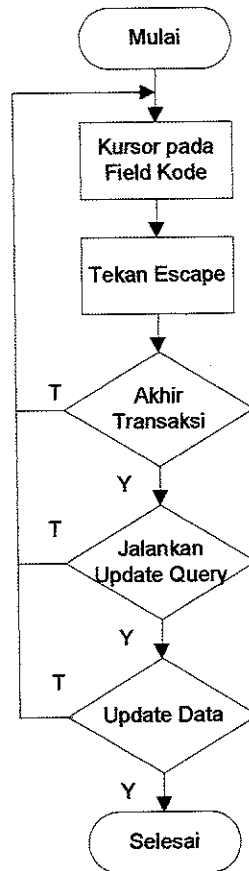
Default Value : DMax("[NO FAKTUR]";"PURCHASING")+1 (IV.4)

Fungsinya agar pengguna tidak perlu lagi maengisikan no faktur. Data pada *field* tersebut hanya bisa dibaca dan tidak bisa diubah, dengan cara mengunci *field* tersebut pada propertinya. *Field* tanggal memiliki properti *default value: date()*. Sehingga form akan memunculkan tanggal pada saat transaksi dan data itu tidak akan berubah karena perubahan tanggal, setelah disimpan.

Pada penggunaannya, form *Purchasing* juga menggunakan bahasa VBA. Namun penerapannya lebih pada subform *Detail Purchasing*. Pada subform, pengguna dapat menggunakan *tab order* pada *keyboard*, yang akan memindahkan letak kursor pada *field* yang harus diisi. Namun untuk berpindah dari form ke subform, kursor harus dipindah dengan *mouse*.

Bahasa VBA pada form *Detail Purchasing* adalah pada akhir input, kursor akan berada pada *field* kode. Untuk mengakhiri input dan memperbarui *record* jumlah stok, maka pada *field* kode, *event key press*, diberikan bahasa VBA. Prosedur tersebut dijalankan dengan menekan *escape* pada *keyboard*, saat kursor berada pada *field* kode. Prosedur tersebut akan memanggil sebuah *message box* yang berisi pilihan untuk mengakhiri proses input. Jika pengguna menekan *enter* atau mengklik *mouse* pada tombol OK, maka proses akan berlanjut dengan pilihan untuk memperbarui *record*. Jika pengguna setuju maka proses akan berlanjut dengan *update record* jumlah stok. Prosedur tersebut dapat ditulis sebagai berikut:

```
Private Sub KODE_KeyPress(KeyAscii As Integer)
  If KODE.Text = "" Then
    If KeyAscii = 27 Then
      pesan1 = MsgBox("Data terakhir dari transaksi ini? Tekan OK
        untuk selesai.", vbOKCancel, "Swalayan")
      If pesan1 = vbOK Then
        DoCmd.OpenQuery "UPDATE STOK", acViewNormal, acEdit
      End If
    End If
  End If
End Sub
```



Gambar IV.13. Digram alir penggunaan bahasa VBA pada form *Purchasing*

Form ini juga menyediakan tombol yang berfungsi untuk menampilkan halaman untuk *record* baru jika pengguna ingin menambahkan data pembelian baru. Prosedur pada tombol ini terdapat pada properti *event on click* yang dapat dituliskan sebagai berikut:

```
Private Sub add_Click()
On Error GoTo Err_add_Click

    DoCmd.GoToRecord , , acNewRec
    Me.NO_FAKTUR.SetFocus
Exit_add_Click:
    Exit Sub

Err_add_Click:
    MsgBox Err.Description
    Resume Exit_add_Click

End Sub
```



Proses *update* data pada form *Purchasing* berkaitan dengan *query Update Stok*. *Query* tersebut digunakan untuk mengolah perubahan data persediaan pada form *Purchasing*. *Query* tersebut merupakan *query update*, yang tersusun dari tabel *Barang* dan *Detail Purchasing*. Relasi yang menghubungkan kedua tabel tersebut adalah *one to many*. *Field* pada *query* tersebut adalah *stok* dari tabel *Barang* dan *no faktur* dari tabel *Detail Purchasing*.

Tabel IV.3. *Query Update Stok*

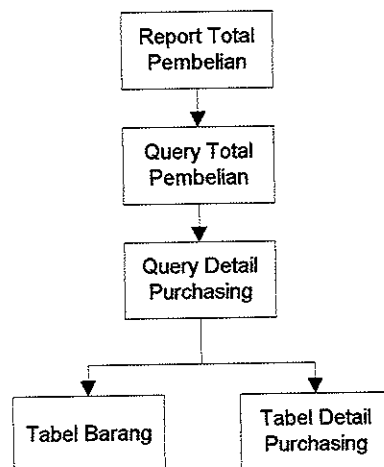
<i>Field</i>	Tabel	<i>Update to</i>	<i>Criteria</i>
Stok	Barang	(IV.5)	
No Faktur	<i>Detail Purchasing</i>		(IV.6)

Pada *field* *stok*, terdapat perhitungan pada *field update to*, yang berkenaan dengan *update* data jumlah *stok* pada tabel *Barang* sebagai berikut:

Update to : [BARANG].[STOK]+[DETAIL PURCHASING].[JUMLAH] (IV.5)

Sedangkan *field* *no faktur*, pada bagian *criteria* menggunakan perhitungan sebagai berikut:

Criteria : DMax("[NO FAKTUR]";"DETAIL PURCHASING") (IV.6)



Gambar IV.14. Komponen *report* Total Pembelian



Hasil dari seluruh pengolahan data tersebut, adalah *report* total Pembelian. *Report* tersebut disusun dari tabel Barang dan tabel Detail *Purchasing*. Kedua tabel tersebut setelah diolah dengan *query* Detail *Purchasing* diolah lagi dengan menggunakan *query* Total Pembelian. Seluruh pembelian produk ditampilkan dalam *report*, sehingga pihak manajemen akan lebih mudah memahami.

Tabel IV.4. *Query* Total Pembelian

<i>Field</i>	Tabel	Total	<i>Sort</i>	<i>Show</i>
Kode	<i>Query</i> Detail <i>Purchasing</i>	<i>Group By</i>		√
Nama	<i>Query</i> Detail <i>Purchasing</i>	<i>Group By</i>		√
Harga Beli	<i>Query</i> Detail <i>Purchasing</i>	<i>Group By</i>		√
Jumlah	<i>Query</i> Detail <i>Purchasing</i>	<i>Sum</i>		√
Subtotal : (IV.7)		<i>Expression</i>		√

Untuk menampilkan *field* subtotal pembelian berdasarkan satu nomor kode produk selama periode tertentu pada *report*, maka ditambahkan sebuah *field* pada *query* yaitu:

Subtotal:  $\text{Sum}([\text{HARGA BELI}] * [\text{JUMLAH}])$  (IV.7)

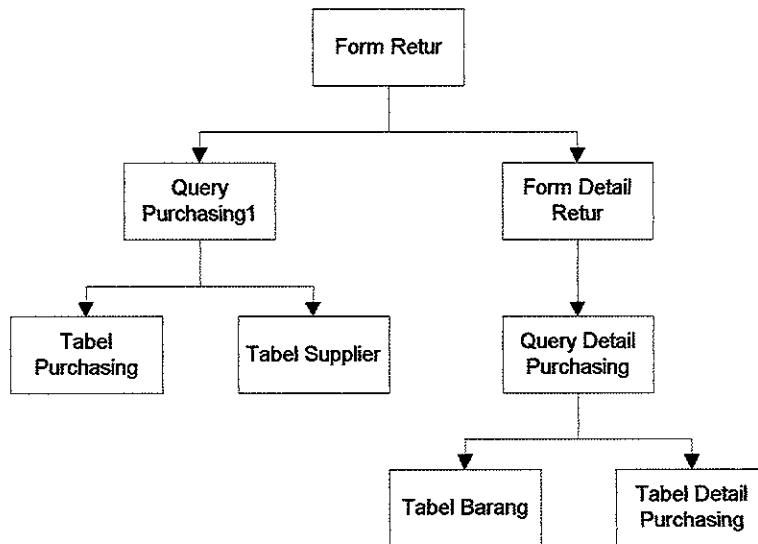
KODE	NAMA	HARGA BELI	JUMLAH PEMBELIAN	SUBTOTAL
1	BUKU TULIS	1.500,00	100	150.000,00
2	PENSIL	700,00	100	70.000,00
3	PENGGARIS	600,00	100	60.000,00
4	BUKU GAMBAR	2.000,00	100	200.000,00
5	BALLPOINT	3.500,00	100	350.000,00
6	BUKU CERITA	6.500,00	100	650.000,00
TOTAL PEMBELIAN:				1.770.000,00

Gambar IV.15. *Report* Total Pembelian



Desain *report* pada bagian *header* terdiri atas *field* nama *report* dan tanggal. *Report* ini berfungsi untuk menunjukkan data seluruh pembelian yang dilakukan hingga hari ini. Sehingga *field* tanggal pada *report* tersebut diatur dengan perhitungan:  $Now ( )$ . Pada bagian detail terdapat *field-field* yang menampilkan jumlah seluruh pembelian dari setiap produk. Pada bagian *footer* terdapat *field* halaman *report*. Selain itu juga ditambahkan total seluruh pembelian yang dilakukan pada periode tersebut, dengan menggunakan perhitungan sebagai berikut:

$$\text{Total Pembelian} = \text{Sum}([\text{SUBTOTAL}]) \quad (\text{IV.8})$$



Gambar IV.16. Komponen form Retur

KODE	NAMA BARANG	JUMLAH	H. BELI	SUBTOTAL	JML RETUR	H. RETUR
5	BALLPOINT	100	3.500,00	350.000,00	0	0,00
2	PENSIL	100	700,00	70.000,00	0	0,00

Record: 1 of 2

TOTAL PEMBELIAN: 420.000,00  
TOTAL RETUR: 0,00  
TOTAL BAYAR: 420.000,00

Record: 1 of 4

Gambar IV.17. Form Retur



Adakalanya produk yang telah dibeli ternyata memiliki cacat produksi. Jika hal ini terjadi, maka swalayan dapat mengembalikan produk tersebut pada *supplier* yang bersangkutan. Untuk mengantisipasi perubahan jumlah stok karena retur, maka disusun form Retur untuk memasukkan data tersebut pada basis data.

Form Retur disusun dari tabel *Purchasing*, tabel *Supplier* dan form Detail Retur. Data dari tabel *Purchasing* dan tabel *Supplier* diolah dengan *query Purchasing1*. Form Retur dirancang dengan bentuk kolomnar. Sama seperti form *Purchasing*, form ini juga menyertakan sebuah subform yaitu form Detail Retur.

Form Detail Retur, disusun dari tabel Barang dan tabel Detail *Purchasing*. Kedua tabel tersebut diolah dengan *query Query Detail Purchasing*. Form ini dirancang dengan bentuk tabular. Form ini mirip dengan form Detail *Purchasing*, hanya pada *field*nya ditambah jumlah retur dan harga retur. Form ini juga menyertakan *field* subtotal beli dan subtotal retur ada form ini. *Field* subtotal beli untuk menjumlahkan seluruh nilai dari *field* total, sedang *field* subtotal retur untuk menjumlahkan seluruh nilai pada *field* harga retur. Kedua *field* tersebut dirancang agar tidak tampak pada saat form Detail Retur ditampilkan sebagai subform. Seperti halnya pada form Detail *Purchasing*. Perhitungan *field* tersebut adalah sebagai berikut:

$$\text{Harga Retur} = [\text{HARGA BELI}] * [\text{JUMLAH RETUR}] \quad (\text{IV.9})$$

$$\text{Subtotal Beli} = \text{Sum}([\text{TOTAL}]) \quad (\text{IV.10})$$

$$\text{Subtotal Retur} = \text{Sum}([\text{HARGA BELI}] * [\text{JUMLAH RETUR}]) \quad (\text{IV.11})$$

Selain itu, pada form Retur juga ditambahkan tiga buah *field* untuk menampilkan total pembelian yang telah dilakukan pada no faktur tersebut, total harga produk yang mengalami retur, dan total pembayaran. Pada form ini input dilakukan hanya pada *field* jumlah retur, dan *field* lainnya dikunci pada propertinya agar tidak terjadi perubahan data pada *field* tersebut saat input dilakukan. Pada *field* no faktur, pengguna tidak perlu lagi melakukan input data. Karena data pada form Retur berasal dari form *Purchasing*. Sehingga pengguna hanya perlu melakukan input jumlah item yang mengalami retur pada form Retur



sesuai no faktur pembelian produk tersebut. *Field* tanggal merupakan tanggal saat melakukan retur. Pengaturan tanggal dilakukan pada properti *default value:date()*.

Pada form ini perhitungan khusus diberikan pada *field-field* tambahan. perhitungan tersebut adalah sebagai berikut:

$$\text{Total Pembelian} = [\text{DETAIL RETUR}].\text{Form!}[\text{SUBTOTAL BELI}] \quad (\text{IV.12})$$

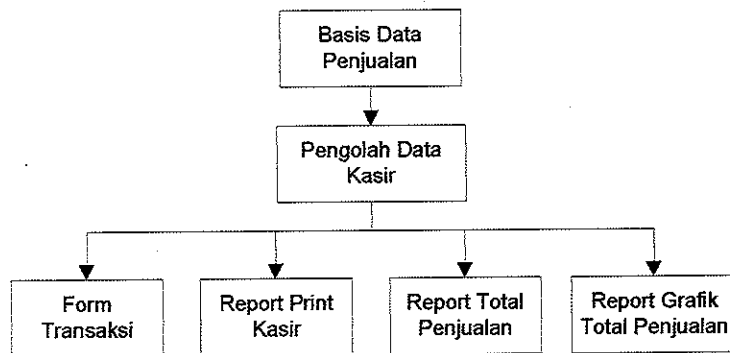
$$\text{Total Retur} = [\text{DETAIL RETUR}].\text{Form!}[\text{SUBTOTAL RETUR}] \quad (\text{IV.13})$$

$$\text{Total Bayar} = ([\text{TOTAL PEMBELIAN}] - [\text{RETUR}]) \quad (\text{IV.14})$$

#### IV.1.2. Basis Data Penjualan

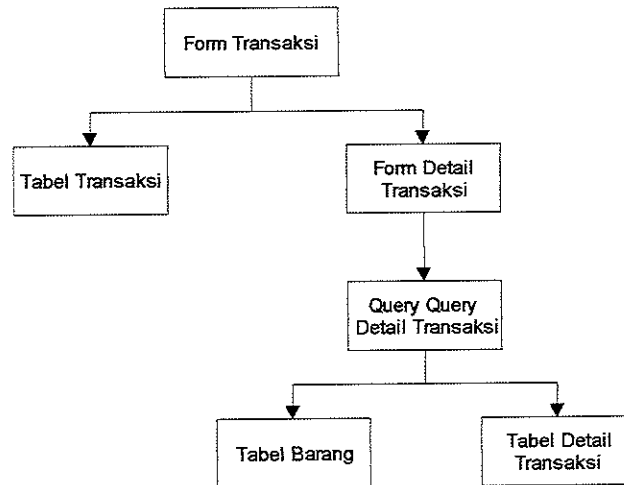
Basis data yang ada pada bagian penjualan, hanya terdiri atas basis data pengolahan data kasir. Basis data bagian penjualan digunakan untuk menyimpan data seluruh transaksi penjualan dan untuk memasukkan data transaksi penjualan baru.

Pengolah data kasir terdiri atas form Transaksi, *report Print Kasir*, *report Laba Harian*, *report Total Penjualan*, dan *report Grafik Total Penjualan*. Tabel pembentuk pengolah data tersebut adalah tabel Barang, tabel Transaksi, dan tabel Detail Transaksi.



Gambar IV.18. Subsistem Basis Data Penjualan

Form Transaksi, dirancang dengan bentuk kolumnar. Form Transaksi disusun dari tabel Transaksi dan subform Detail Transaksi. Tabel Transaksi memiliki *primary key* pada field no nota. Subform Detail Transaksi, disusun dari tabel Barang dan tabel Detail Transaksi. Tabel Detail Transaksi tidak memiliki *primary key*.



Gambar IV.19. Komponen form Transaksi

Field Name	Data Type	Description
NO NOTA	Number	
TANGGAL	Date/Time	
BAYAR	Number	

Field Properties	
General	Lookup
Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (No Duplicates)

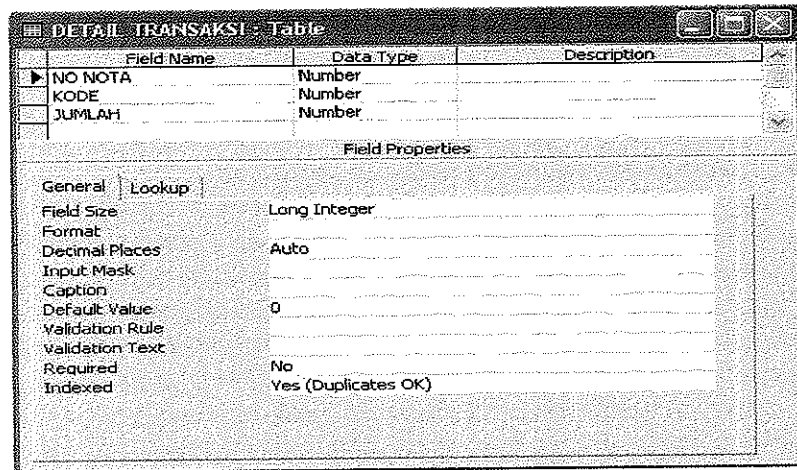
Gambar IV.20. Tabel Transaksi

Subform Detail Transaksi dirancang dengan bentuk tabular. Untuk menampilkan jumlah seluruh nilai pada *field* total, pada bagian *footer* ditambahkan *field* subtotal seperti pada form Detail *Purchasing*. *Field* tersebut tidak ditampilkan pada saat form Detail Transaksi menjadi subform dalam form Transaksi karena *field* ini hanya untuk mengolah data. Sedang tampilan hasil olahannya terdapat pada *field* Total form Transaksi. Perhitungan pada *field* subtotal tersebut sama dengan perhitungan IV.2.



Untuk menghasilkan data pada subform Detail Transaksi, kedua tabel penyusun subform diolah pada *query Query Detail Transaksi*. Kedua tabel tersebut dihubungkan dengan bentuk relasi *one to many*. Pada *query* tersebut ditambahkan *field* total untuk menghitung total pembelian dalam satu transaksi penjualan. Perhitungan yang diberikan adalah:

$$\text{TOTAL: [BARANG].[HARGA JUAL]*[DETAIL TRANSAKSI].[JUMLAH]} \quad (\text{IV.15})$$



Gambar IV.21. Tabel Detail Transaksi

Tabel IV.5. *Query* Detail Transaksi

<i>Field</i>	<i>Tabel</i>	<i>Sort</i>	<i>Show</i>
No Nota	Detail Transaksi		√
Kode	Detail Transaksi		√
Nama	Barang		√
Harga Jual	Barang		√
Jumlah	Detail Transaksi		√
Total: (IV.15)			√

Pada bagian *footer* form Transaksi ditambahkan dengan dua *field* tambahan yaitu *field* total dan *field* kembali. Fungsi *field* total adalah untuk menampilkan total nilai pada *field* total dari subform Detail Transaksi. *Field* kembali digunakan untuk menampilkan jumlah uang kembali atas pembayaran



yang telah dilakukan pada produk yang dibeli konsumen. Perhitungan pada kedua *field* tersebut dapat dituliskan sebagai berikut:

$$\text{Total} = [\text{DETAIL TRANSAKSI}].\text{Form}!\text{SUBTOTAL} \quad (\text{IV.16})$$

$$\text{Kembali} = [\text{BAYAR}]-[\text{SUBTOTAL}] \quad (\text{IV.17})$$

Dalam penggunaannya, input pada form transaksi dilakukan pada *field* kode, jumlah, dan bayar. Pada *field* no nota, nomor telah disikan oleh komputer. Pengguna yang ingin melakukan input transaksi baru dapat langsung menuju pada form baru dengan nomor nota yang telah diurutkan. Prosedur tersebut diatur dalam properti.

$$\text{Default Value: DMax}([\text{NO NOTA}];\text{"TRANSAKSI"})+1 \quad (\text{IV.18})$$

KODE	NAMA	HARGA	JUMLAH ITEM	TOTAL
1	BUKU TULIS	2.000.00	10	20.000.00
2	PENSIL	1.000.00	10	10.000.00
3	BUKU CERITA	10.000.00	5	50.000.00

TOTAL: 50.000.00  
BAYAR: 100.000.00  
KEMBALI: 20.000.00

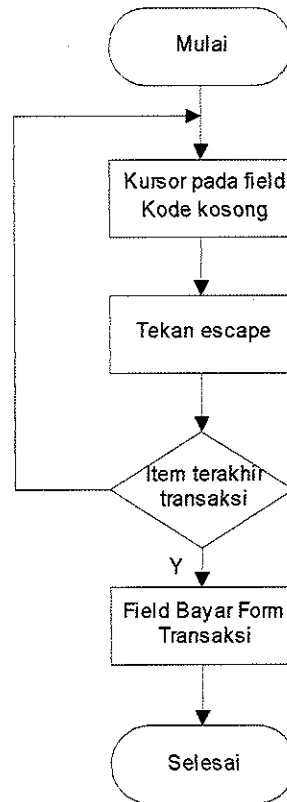
Gambar IV.22. Form Transaksi

*Field* tanggal diatur untuk selalu menampilkan tanggal hari ini, namun pada penyimpanan data transaksi tanggal tersebut akan tetap. Sehingga tanggal data transaksi tidak mengalami perubahan. Pengaturannya sama seperti pada form Retur maupun *Purchasing*. Yaitu dengan menuliskan pada properti *default value: date( )*. *Field* ini dikunci agar tidak terjadi kesalahan penanggalan karena kesalahan input.

Pada waktu pengguna melakukan input data pada form baru, kursor akan langsung berada pada *field* kode. Setelah kode diisikan, maka pada *field* nama



akan muncul nama produk yang bersangkutan. Begitu juga dengan *field* harga yang secara otomatis akan menampilkan harga produk tersebut. Input berikutnya adalah pada *field* jumlah.



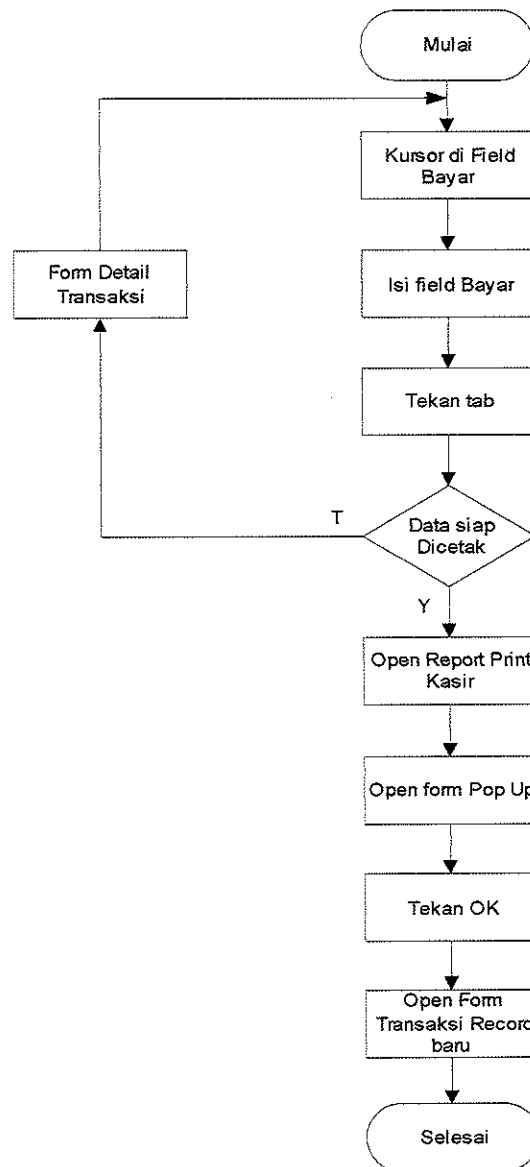
Gambar IV.23. Diagram Alir Penggunaan Bahasa VBA pada Form Detail Transaksi

Jika pengguna telah selesai mengisi seluruh data pada *field-field* tersebut dan ingin mengakhiri input, maka kursor diletakkan pada *field* kode dengan *tab order*. Proses input dapat diakhiri jika pada *field* kode, pengguna menekan *escape* pada *keyboard* yang kemudian akan memunculkan *message box* berupa pilihan untuk mengakhiri input.

Proses berikutnya jika pengguna menekan tombol OK, maka kursor akan berpindah pada *field* bayar. Jika pengguna menekan *Cancel*, kursor akan kembali pada *field* kode. Setelah *field* tersebut diisi, proses diakhiri dengan menekan *tab order* pada *keyboard*. Proses tersebut akan menampilkan sebuah *message box* yang memberikan pilihan untuk melanjutkan proses berikutnya yaitu mencetak



hasil transaksi tersebut. Jika pengguna menekan tombol OK, maka setelah prosedur cetak dilakukan, akan muncul form modal yang berisi tampilan total pembelian, jumlah pembayaran, dan sisa uang pembayaran atas transaksi yang dilakukan oleh konsumen. Pada form tersebut terdapat tombol OK untuk mengakhiri seluruh transaksi tersebut. Jika tombol tersebut ditekan, maka form akan berpindah ke halaman form baru.



Gambar IV.24. Diagram alir penggunaan bahasa VBA pada form Transaksi



### Prosedur pada form Detail Transaksi

```
Private Sub KODE_KeyPress(KeyAscii As Integer)
On Error GoTo gagal
If KODE.Text = "" Then
If KeyAscii = 27 Then
    pesan1 = MsgBox("Item Terakhir dari Transaksi ini? Tekan OK
    untuk selesai.", vbOKCancel, "Swalayan")
    If pesan1 = vbOK Then
        Form_TRANSAKSI.BAYAR.SetFocus
    End If
End If
End If
Exit Sub

gagal:
Exit Sub
End Sub
```

### Prosedur pada *field* bayar form Transaksi:

```
Private Sub BAYAR_LostFocus()
pesan1 = MsgBox("Data siap ditransfer ke printer. Lanjutkan
    Proses?", vbOKCancel, "Swalayan")
If pesan1 = vbOK Then
    DoCmd.OpenReport "print kasir", acViewNormal
    DoCmd.OpenForm "POP UP", acNormal
End If
End Sub
```

### Prosedur pada form modal Pop Up

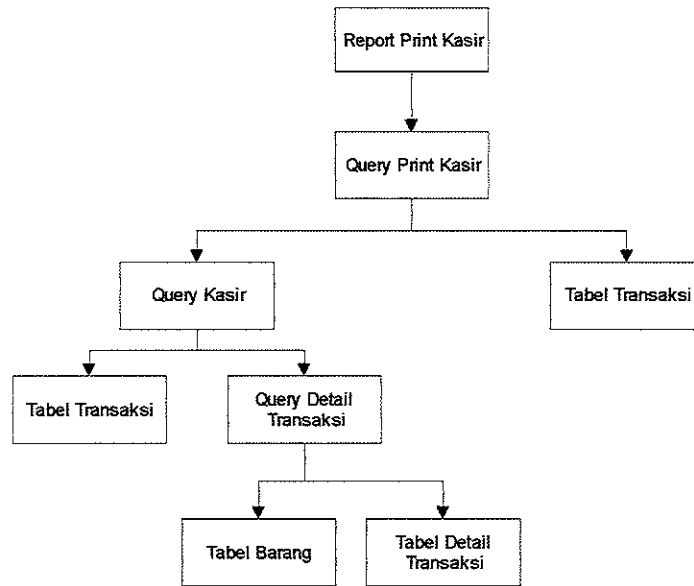
```
Private Sub Command18_Click()
On Error GoTo Err_Command18_Click
    DoCmd.GoToRecord acDataForm, "TRANSAKSI", acNewRec
    DoCmd.close

Exit_Command18_Click:
Exit Sub

Err_Command18_Click:
    MsgBox Err.Description
    Resume Exit_Command18_Click

End Sub

Private Sub Form_Load()
    DoCmd.GoToRecord , , acLast
End Sub
```



Gambar.IV.25. Komponen *report Print Kasir*

Tabel IV.6. *Query Kasir*

<i>Field</i>	<i>Tabel</i>	<i>Sort</i>	<i>Show</i>
No Nota	Transaksi		√
Tanggal	Transaksi		√
Kode	<i>Query Detail Transaksi</i>		√
Nama	<i>Query Detail Transaksi</i>		√
Harga Jual	<i>Query Detail Transaksi</i>		√
Jumlah	<i>Query Detail Transaksi</i>		√
Total	<i>Query Detail Transaksi</i>		√
Bayar	Transaksi		√

Untuk mendukung kegiatan di bagian kasir, setelah konsumen selesai melakukan transaksi, sebagai bukti transaksi, kasir akan mengeluarkan hasil transaksi tersebut dalam bentuk struk pembelian pada konsumen. Pada form Transaksi tersebut, bahasa VBA berfungsi untuk menjalankan perintah *print*. rosedur tersebut akan membuka *report Print Kasir*.



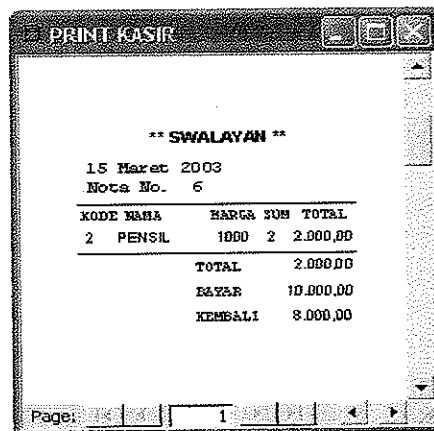
Tabel IV.7. *Query Print Kasir*

Field	Tabel	Sort	Show	Criteria
No Nota	<i>Query Kasir</i>			(IV.19)
Tanggal	<i>Query Kasir</i>			
Kode	<i>Query Kasir</i>			
Nama	<i>Query Kasir</i>			
Harga Jual	<i>Query Kasir</i>			
Jumlah	<i>Query Kasir</i>			
Total	<i>Query Kasir</i>			
Kembali: (IV.20)				

*Report Print Kasir* disusun dari tabel Transaksi, tabel Barang dan tabel Detail Transaksi. Tabel Barang dan tabel Detail Transaksi diolah dengan menggunakan *query* Detail Transaksi. Kemudian bersama dengan tabel Transaksi, *query* tersebut diolah dalam *query* Kasir. Dan kemudian diolah lagi dengan *query Print Kasir* untuk menampilkan data tersebut. Fungsi *report* ini adalah mencetak struk penjualan yang diberikan pada konsumen setelah transaksi selesai dilakukan. Agar prosedur mencetak dapat tepat pada halaman *report* yang terakhir, maka pada *field criteria query Print Kasir* ditambahkan perhitungan sebagai berikut:

Criteria : DMax("[NO NOTA]";"TRANSAKSI") (IV.19)

KEMBALI : KASIR.BAYAR-[TOTAL] (IV.20)



Gambar IV.26. *Report Print Kasir*



Desain *report* di bagian *page header*, terdiri atas label untuk menampilkan nama swalayan, *field* tanggal, dan *field* no nota. *Field* no nota ditampilkan beserta dengan keterangannya. Untuk keperluan tersebut, pada *field* tersebut ditambahkan sebuah perhitungan berikut ini:

$$\text{No Nota} = \text{"Nota No. " \& [NO NOTA]} \quad (\text{IV.21})$$

Pada bagian detail terdapat *field* kode, nama, harga jual, jumlah, dan total. Sedangkan bagian *footer* terdiri atas *field* bayar dan dua *field* tambahan yaitu: *field* total dan kembali. *Field* total bayar memiliki perhitungan sebagai berikut:

$$\text{Total} = \text{Sum}([\text{TOTAL PEMBELIAN}]) \quad (\text{IV.22})$$

Bahasa VBA pada form Transaksi juga menampilkan sebuah form modal. Form modal tersebut dibentuk dari form *Pop Up* yang berfungsi sebagai tampilan pada layar, agar pada saat transaksi pada kasir selesai dilakukan dan terhubung pada *report Print Kasir*, konsumen dapat melihat dengan lebih jelas tampilan pada layar komputer kasir berapa total belanja yang telah dilakukan, jumlah uang yang telah dibayarkan serta jumlah kembalian yang berhak diterima.

Form ini serupa dengan form Transaksi, hanya saja *field* no nota, tanggal dan subform pada bagian detail tidak ditampilkan. Hal tersebut dapat dilakukan dengan cara mengubah format *visible* pada properties pada masing-masing *field* menjadi No. Kemudian ditambahkan empat buah *field* pada bagian detail tersebut. *Field* untuk menampilkan nama swalayan, *field* total belanja, *field* total bayar, dan *field* kembali.

*Field* nama swalayan dibentuk dari label. Sedangkan ketiga *field* lainnya dari text box. Perhitungan pada *field-field* tersebut adalah sebagai berikut:

$$\begin{aligned} \text{Jumlah Belanja} &= \text{"TOTAL BELANJA = Rp " \& [DETAIL TRANSAKSI].Form!SUBTOTAL} \end{aligned} \quad (\text{IV.23})$$

$$\text{Bayar} = \text{"BAYAR = Rp " \& [BAYAR]} \quad (\text{IV.24})$$

$$\begin{aligned} \text{Kembali} &= \text{"KEMBALI = Rp " \& [BAYAR]-([DETAIL TRANSAKSI].Form!} \\ &\quad \text{SUBTOTAL)} \end{aligned} \quad (\text{IV.25})$$



SWALAYAN

TOTAL BELANJA = Rp 2000

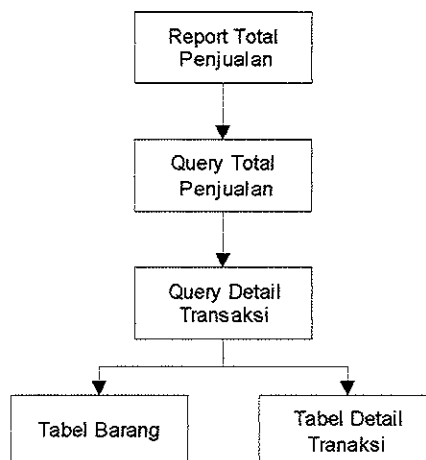
BAYAR = Rp 10000

KEMBALI = Rp 8000

OK

Gambar IV.27. Form Modal

Untuk menampilkan form ini, pada properti *other*, *field* modal diubah menjadi Yes. Modal adalah *window* atau kotak dialog yang membutuhkan beberapa tindakan dari operator sebelum fokus berpindah pada kotak dialog atau form lainnya. Fokus adalah kemampuan untuk menerima input dari operator melalui *mouse*, *keyboard*, atau metode *SetFocus*. Fokus dapat diatur oleh operator oleh aplikasi. Jika modal properti suatu form adalah Yes, maka form akan dibuka dalam bentuk *formview*. (Microsoft Access Help).



Gambar IV.28. Komponen *report* Total Penjualan



Tabel IV.8. *Query* Total Penjualan

<i>Field</i>	<i>Tabel</i>	<i>Total</i>	<i>Sort</i>	<i>Show</i>	<i>Criteria</i>
Kode	<i>Query</i> Detail Transaksi	<i>Group By</i>		√	
Nama	<i>Query</i> Detail Transaksi	<i>Group By</i>		√	
Harga Jual	<i>Query</i> Detail Transaksi	<i>Group By</i>		√	
Jumlah	<i>Query</i> Detail Transaksi	<i>Sum</i>		√	
Subtotal: (IV.26)		<i>Expression</i>		√	
Kiri: (IV.27)		<i>Group By</i>			(IV.28)

Untuk menampilkan seluruh penjualan yang telah dilakukan, diperlukan *report* Total Penjualan. *Report* ini disusun dari tabel Barang dan tabel Detail Transaksi. Kedua tabel tersebut diolah dengan menggunakan *query* Detail Transaksi dan *query* Total Penjualan. Seperti *report* Total Pembelian, *report* ini juga menampilkan data seluruh penjualan seluruh produk, kecuali jika tidak ada penjualan sama sekali, hingga hari ini. Pada *query* Total Penjualan, ditambahkan sebuah *field* untuk menghitung nilai subtotal. Perhitungannya adalah sebagai berikut:

SUBTOTAL: (Sum([HARGA JUAL]\*[JUMLAH])) (IV.26)

KODE	NAMA	HARGA JUAL	JUMLAH PENJUALAN	SUBTOTAL
1	BUKU TULIS	2.000,00	30	60.000,00
2	PENSIL	1.000,00	32	32.000,00
3	PENGGARIS	1.000,00	20	20.000,00
4	BUKU GAMBAR	2.500,00	20	50.000,00
5	BALLPOINT	4.000,00	30	120.000,00
6	BUKU CERITA	10.000,00	20	200.000,00
8	STIP	1.200,00	10	12.000,00
TOTAL PENJUALAN:				494.000,00

Gambar IV.29. *Report* Total Penjualan



Data ini juga ditampilkan dalam bentuk grafik. Grafik Total Penjualan akan membantu menganalisa hasil penjualan selama periode tertentu. Grafik ini akan menampilkan tingkat penjualan setiap produk dalam satu kelompok. Sehingga akan mudah mengamati tingkat penjualan suatu produk dibanding produk lainnya. Grafik Total Penjualan juga berasal dari data olahan *query* Total Penjualan yang ditampilkan dengan *report Chart Wizard*. Untuk mempermudah pengamatan, maka grafik dapat dilihat berdasarkan kode produk. Pengguna dapat memasukkan 2 karakter awal kode produk, sehingga menghasilkan grafik penjualan seluruh produk yang memiliki kode dengan karakter awal seperti yang diinginkan oleh pengguna. Data yang diambil adalah berdasarkan tingkat penjualan total pada sumbu Y dan jenis produk pada sumbu X.

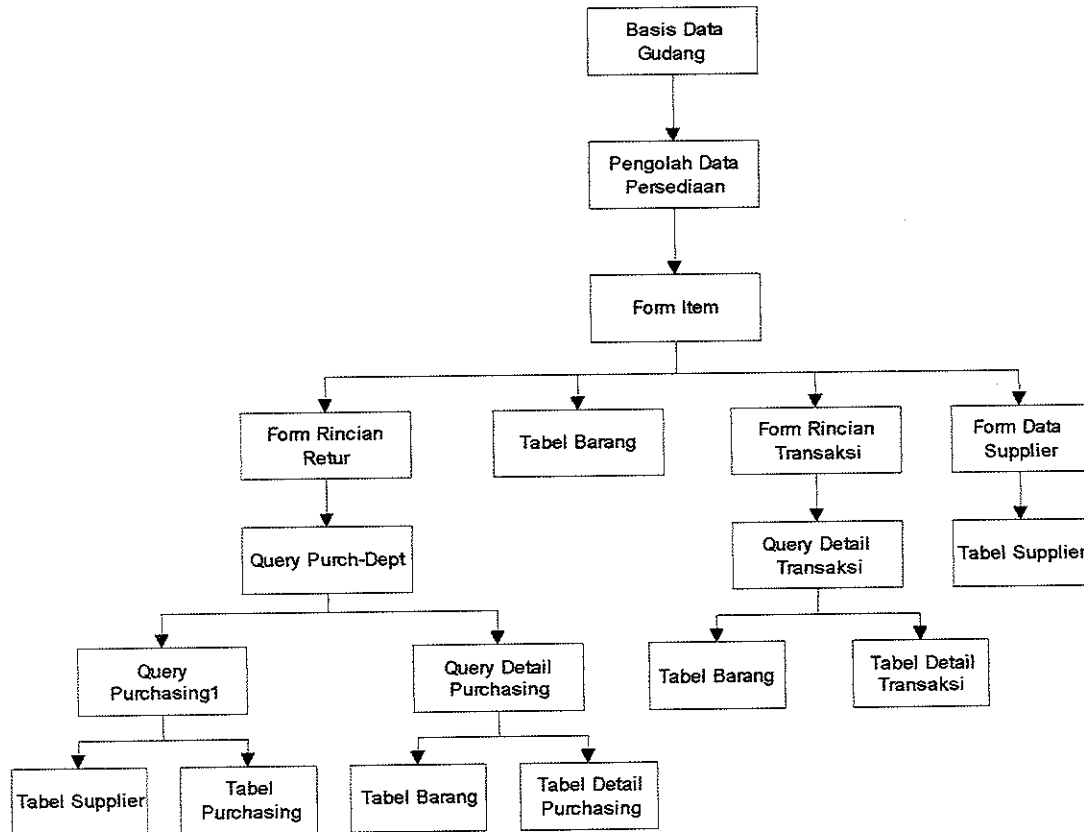
## IV.2. PEMBUATAN SISTEM PENDUKUNG KEPUTUSAN

Tujuan utama pembuatan basis data adalah untuk membangun sistem pendukung keputusan yang diperlukan oleh manajemen untuk memberikan informasi dalam memecahkan permasalahan semi terstruktur dalam swalayan. Sistem pendukung keputusan terdiri atas pengolahan data persediaan, laba harian, penentuan kriteria penjualan produk, penentuan titik pemesanan ulang, dan jumlah pemesanan.

### IV.2.1. Pengolahan Data Persediaan

Pengolahan data persediaan merupakan bagian dari sistem pendukung keputusan yang berfungsi mengolah data penjualan dan pembelian hingga menghasilkan informasi aktual persediaan.

Basis data ini disusun dari tabel Barang, tabel *Supplier*, tabel Detail Transaksi, tabel Detail *Purchasing*, tabel *Purchasing*, dan form Item. Form ini terdiri atas tiga buah subform yaitu subform Data *Supplier*, subform Rincian Transaksi, dan subform Rincian Retur. Form Item disusun dari tabel barang, subform Data *Supplier* disusun dari tabel *Supplier*, subform Rincian Transaksi dari tabel tabel Barang dan tabel Detail Transaksi, dan subform Rincian Retur dari tabel Barang dan tabel Detail *Purchasing*.



Gambar IV.32. Komponen penyusun pengolah data persediaan

Subform *Data Supplier* menampilkan data *supplier*. Form ini akan secara otomatis menampilkan data *supplier* jika input data produk telah dilakukan. Tampilan subform ini dikunci, untuk menghindari kesalahan pada waktu input.

Subform *Rincian Transaksi* menampilkan seluruh transaksi yang berkenaan pada produk tersebut berdasarkan nomor notanya. Pada subform ini ditambahkan *field* subjumlah yang tidak dimunculkan nilainya pada tampilan form. *Field* ini berfungsi untuk menjumlahkan nilai pada *field* jumlah agar dihasilkan data jumlah penjualan seluruh produk. Perhitungannya adalah sebagai berikut:

$$\text{Subjumlah} = \text{Sum}([\text{JUMLAH}]) \quad (\text{IV.29})$$

Seperti juga subform *Rincian Transaksi*, subform *Rincian Retur* menampilkan seluruh data *Retur* berdasarkan no fakturnya. Form ini



menampilkan data retur yang berkaitan dengan produk tersebut. Data pada retur tersebut diurutkan berdasarkan nomor faktur dan tanggal pembelian. Data diolah pada *query Purch-Dept*.

Tabel IV.9. *Query Purch-Dept*

Field	Tabel	Sort	Show
No Faktur	<i>Query Purchasing1</i>		√
Tanggal	<i>Query Purchasing1</i>		√
Kode Supplier	<i>Query Purchasing1</i>		√
Nama Supplier	<i>Query Purchasing1</i>		√
Kode	<i>Query Detail Purchasing</i>		√
Nama	<i>Query Detail Purchasing</i>		√
Jumlah	<i>Query Detail Purchasing</i>		√
Harga Beli	<i>Query Detail Purchasing</i>		√
Total	<i>Query Detail Purchasing</i>		√
Jumlah Retur	<i>Query Detail Purchasing</i>		√

The screenshot shows a software interface titled "FORM DATA INVENTORI" dated "16 Maret 2003". It features several sections:

- CARI DATA** and **EXIT FILTER** buttons.
- KODE** field with value "11".
- NAMA** field with value "BUKU TULIS".
- KODE SUPPLIER** field with value "1".
- HARGA BELI** field with value "1.500.00".
- HARGA JUAL** field with value "2.000.00".
- STOK** field with value "900".
- DATA SUPPLIER** section with fields for **KODE SUPPLIER** (1), **NAMA SUPPLIER** (ADI JAYA), **ALAMAT** (YOGYAKARTA), and **TELEPON** (855213).
- RINCIAN TRANSAKSI** table with columns: NO NOTA, KODE, JUMLAH. It shows three rows with values (1, 1, 10), (3, 1, 10), and (5, 1, 10).
- RINCIAN RETUR** table with columns: NO FAKTUR, TANGGAL, KODE, JML RETUR. It shows one row with values (3, 17/02/2003, 1, 0).
- TOTAL PENJUALAN** field with value "30".
- TOTAL RETUR** field with value "0".
- SISA STOK** field with value "870".
- Record navigation controls at the bottom.

Gambar IV.33. Form Item

Form Item dirancang dengan bentuk kolombar. Form ini juga memiliki *field* tambahan untuk menampung data jumlah total penjualan, total retur, dan sisa stok. *Field* utama pada form ini adalah *field* sisa stok. Karena pada *field* ini ditampilkan data jumlah stok terakhir yang dimiliki oleh swalayan. Form ini



menggabungkan data penjualan dan persediaan, sehingga data persediaan dapat diolah hingga menghasilkan informasi jumlah sisa stok. Perhitungan yang dilakukan pada *field-field* tambahan adalah sebagai berikut:

$$\text{Total Penjualan} = [\text{RINCIAN TRANSAKSI}].\text{Form!SUBJUMLAH} \quad (\text{IV.30})$$

$$\text{Total Retur} = [\text{RINCIAN RETUR}].\text{Form!SUBJUMLAH} \quad (\text{IV.31})$$

$$\text{Sisa Stok} = [\text{STOK}] - [\text{TOTAL JUAL}] - [\text{TOTAL RETUR}] \quad (\text{IV.32})$$

Pada perhitungan IV.32., terlihat bahwa perhitungan tidak dilakukan pada total pembelian. Hal ini dikarenakan pada proses sebelumnya, total pembelian telah dijumlahkan terlebih dahulu pada data jumlah stok awal. Prosedur ini telah dibahas pada pembuatan basis data pembelian. Sehingga proses berikutnya hanya tinggal melakukan perhitungan atas jumlah stok akhir yang dipengaruhi oleh tingkat penjualan dan retur.

Form ini hanya merupakan tampilan, karena seluruh data pada *field* tersebut merupakan data olahan dari form lainnya. Input hanya dilakukan pada kode produk yang dimaksud. Proses berikutnya adalah menampilkan data keseluruhan dari produk tersebut.

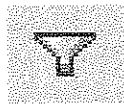
Form ini sebenarnya merupakan bagian dari sistem pendukung keputusan yang dibuat berdasarkan basis data yang telah dibangun. Sebagai sistem pendukung keputusan, form ini dilengkapi dengan bahasa VBA yang memungkinkan pihak manajemen untuk melakukan evaluasi lanjut atas produk yang berkaitan.

Prosedur yang pertama berkaitan dengan input nomor kode produk yang diinginkan. Prosedur ini cukup rumit, karena diperlukan beberapa langkah untuk melakukan input atas karakteristik produk yang bersangkutan. Hal ini bisa berupa no kode, nama, kode *supplier*, harga beli, harga jual, maupun jumlah stok.

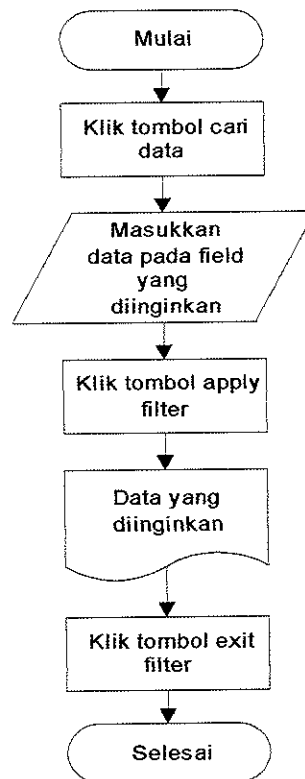
Ada dua cara untuk mencari produk yang diinginkan. Yang pertama adalah dengan fasilitas *filter*. Pada form ini telah disediakan dua buah tombol untuk membantu proses pencarian data. Tombol cari data berfungsi untuk menjalankan fungsi pencarian data. Dengan mengklik tombol tersebut, form akan berada dalam bentuk *disable*, yaitu bentuk dimana form tidak memunculkan datanya sama



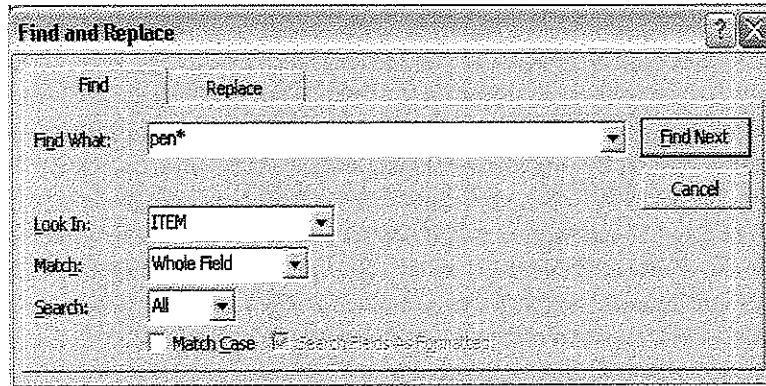
sekali. Pengguna dapat melakukan input pada salah satu *field* kode, nama, kode *supplier*, harga, maupun jumlah stok. Bahkan input bisa dilakukan pada lebih dari satu *field*. Setelah input dilakukan, klik tombol *apply filter* pada *formview bar*, atau klik kanan pada form kemudian klik pada menu *apply filter/sort*. Maka akan muncul seluruh data yang berkaitan dengan data tersebut pada subform-subform. Untuk menutup menu pencarian, klik tombol *exit filter*, maka form akan kembali ke bentuk semula.



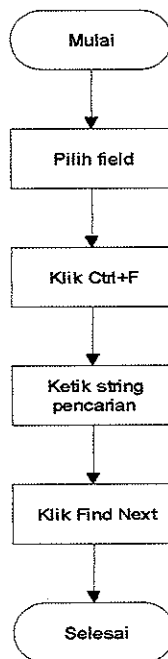
Gambar IV.34. *Icon Filter*



Gambar IV.35. Diagram alir prosedur *Apply Filter*



Gambar IV.36. Fasilitas pencarian data Microsoft Access



Gambar IV.37. Diagram alir prosedur *find*

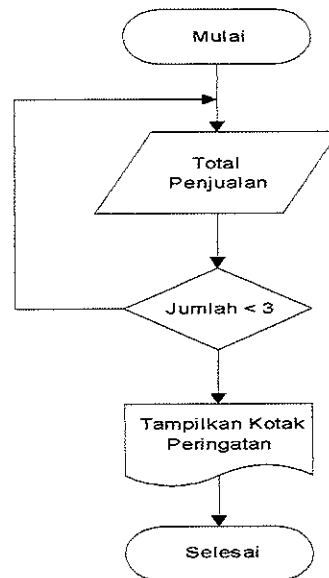
Cara yang kedua adalah menggunakan fasilitas Find yang telah disediakan oleh Microsoft Access. Prosedur ini dimulai dengan meletakkan kursor pada *field* kriteria yang diinginkan. Kemudian tekan Ctrl+F, yang akan memunculkan form pencarian pada *field* yang diinginkan. Ketik data pada *field Find What.*, kemudian klik *Find*. Jika data yang dimasukkan memiliki jumlah karakter yang banyak, maka input dapat dilakukan dengan menuliskan sedikit karakter maka akan



muncul data-data produk yang diinginkan pada subform-subform. Klik *find next* jika data yang keluar bukan yang diinginkan hingga data tersebut ditemukan.

Prosedur berikutnya adalah untuk membantu proses evaluasi atas produk yang memiliki penjualan yang sangat kurang maupun belum terjual sama sekali. Hal ini penting karena dengan demikian manajemen akan lebih memperhatikan produk mana yang memiliki tingkat penjualan rendah.

Prosedur untuk mengetahui produk yang memiliki tingkat penjualan rendah diberikan pada *event lost focus field* kode. Prosedur ini akan berfungsi ketika kursor meninggalkan fokus dari *field* kode. Untuk mengetahui suatu produk memiliki tingkat penjualan yang rendah atau belum terjual, pengguna hanya perlu menggunakan *tab order* pada *field* kode. Pada saat fokus meninggalkan *field* kode yang dimaksud, jika produk memiliki tingkat penjualan rendah, pada prosedur ini rendah berarti kurang dari 3, maka akan muncul *message box* yang akan meminta perhatian khusus dari pengguna untuk melakukan evaluasi pada tanggal pembelian produk dan kriteria produk.



Gambar IV.38. Diagram alir bahasa VBA pada form Item

Tujuannya untuk mengetahui apakah produk yang dimaksud benar-benar memiliki tingkat penjualan rendah dalam periode tertentu sesuai kriteria produk tersebut ataukah karena evaluasi manajemen yang terlalu dini pada produk



tersebut. Evaluasi tersebut terjadi pada waktu yang amat berdekatan dengan pembelian produk, sehingga belum terlihat tingkat penjualannya.

Jika produk sama sekali tidak terjual pada saat evaluasi dilakukan, maka pada saat fokus *order tab* meninggalkan *field* kode produk tersebut, maka akan muncul *message box* yang bertujuan agar pengguna melakukan evaluasi terhadap tanggal pembelian. Jika ternyata selama periode tertentu produk tersebut tidak terjual, maka manajemen perlu mengevaluasi perlu tidaknya produk tersebut dibeli lagi pada periode penjualan berikutnya.

Prosedur pada form Item:

```
Private Sub KODE_LostFocus()  
Dim JUMLAH As Long  
On Error GoTo Cancel  
JUMLAH = [TOTAL JUAL].Value  
If JUMLAH < 3 Then  
MsgBox "Penjualan produk ini sangat kurang. Mohon periksa  
tanggal pembelian dan kriteria produk ini pada DSS  
Kriteria Produk.", , "Jumlah Penjualan"  
End If  
Exit Sub  
  
Cancel:  
MsgBox "Produk belum terjual sampai saat ini. Mohon periksa  
tanggal pembelian.", , "Data Penjualan"  
End Sub
```

#### IV.2.2. Penentuan Laba Harian

Komponen sistem pendukung keputusan berikutnya adalah *report* Laba Harian. *Report* ini memungkinkan manajemen mengetahui jumlah laba yang dimiliki oleh swalayan setiap hari. *Report* Laba Harian disusun dari tabel Barang, tabel transaksi dan tabel Detail Transaksi yang disusun dari *query* Laba Harian. Pada tampilan *report*, ditambahkan *field* laba harian yang memiliki perhitungan sebagai berikut:

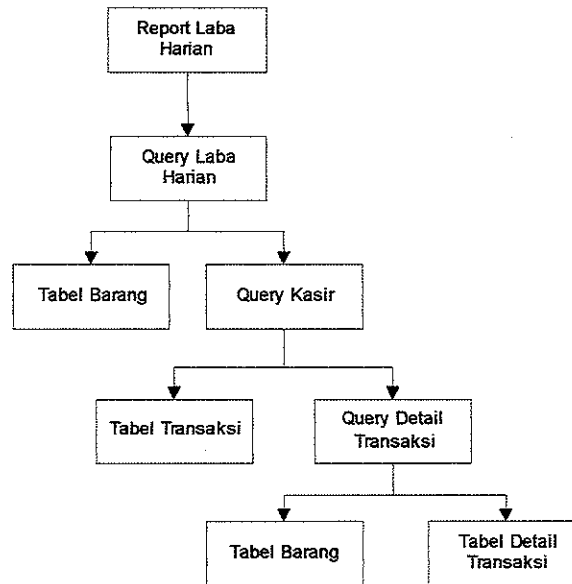
$$\text{Laba Harian} = ([\text{HARGA JUAL}] - [\text{HARGA BELI}]) * [\text{SUMOFJUMLAH}] \quad (\text{IV.33})$$

Pada bagian *footer* terdapat *field* jumlah total penjualan pada hari itu dan total laba harian yang menjumlahkan laba swalayan yang didapat dari seluruh penjualan produk pada hari itu. Perhitungan pada *field* tersebut adalah:



$$\text{Total Penjualan} = \text{Sum}(\{\text{SumOfJUMLAH}\}) \quad (\text{IV.34})$$

$$\text{Total Laba} = \text{Sum}(\{([\text{HARGA JUAL}]-[\text{HARGA BELI}]) * [\text{SUMOFJUMLAH}]\}) \quad (\text{IV.35})$$



Gambar IV.39. Komponen *Report Laba Harian*

Karena *report* ini merupakan *report* harian, maka pada saat membuka, pengguna harus mengisikan tanggal hari yang diinginkan pada kotak dialog. Untuk memunculkan kotak tersebut, prosedurnya ada pada *query* Laba Harian. Pada *criteria field* tanggal diberikan perhitungan sebagai berikut:

$$\text{Criteria: [MASUKKAN TANGGAL TRANSAKSI ex:04-01-03 :]} \quad (\text{IV.36})$$

Tabel.IV.10. *Query Laba Harian*

<i>Field</i>	Tabel	Total	Sort	Show	Criteria
Kode	<i>Query Kasir</i>	<i>Group By</i>		√	
Nama	Barang	<i>Group By</i>		√	
Harga Beli	Barang	<i>Group By</i>		√	
Harga Jual	Barang	<i>Group By</i>		√	
Tanggal	<i>Query Kasir</i>	<i>Group By</i>		√	(IV.36)
Jumah	<i>Query Kasir</i>	<i>Sum</i>		√	



KODE	NAMA	H. BELI	H. JUAL	TANGGAL	JUMLAH	LABA
1	BUKU TULIS	1.500,00	2.000,00	17/02/2003	20	10.000,00
2	PENSIL	700,00	1.000,00	17/02/2003	20	6.000,00
3	PENGGARIS	500,00	1.000,00	17/02/2003	20	10.000,00
4	BUKU GAMBAR	2.000,00	2.500,00	17/02/2003	20	10.000,00
5	BALLPOINT	3.500,00	4.000,00	17/02/2003	20	10.000,00
6	BUKU CERTA	9.500,00	10.000,00	17/02/2003	15	7.500,00
					115	53.500,00

Gambar IV.40. Report Laba Harian

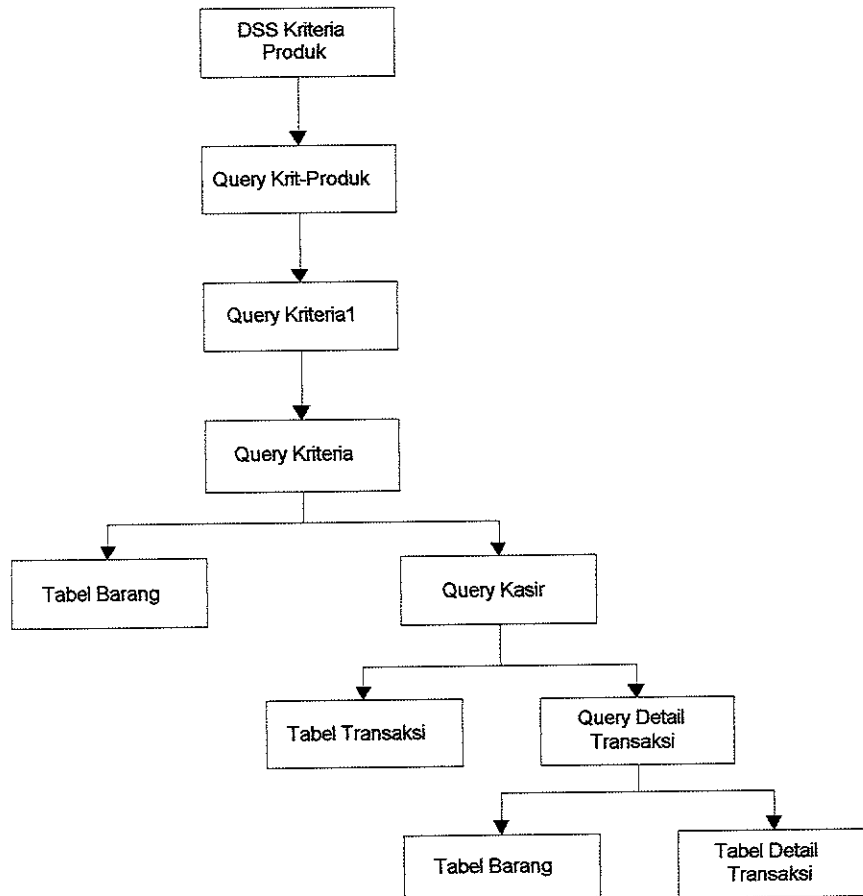
### IV.2.3. Sistem Pendukung Keputusan Kriteria Produk

Sistem ini dibangun untuk membantu proses evaluasi pada tingkat penjualan setiap produk. Sistem ini dibangun dari basis data penjualan. Pada basis data tersebut, tersimpan seluruh data penjualan selama periode tertentu. Dari data penjualan tersebut, kemudian dihasilkan tingkat penjualan yang berbeda-beda pada tiap produk.

Report DSS Kriteria Produk disusun dari komponen basis data penjualan yaitu: tabel Barang, tabel Detail Transaksi, dan tabel Transaksi. Ketiga tabel tersebut diolah dengan *query* Kriteria1, Kriteria, dan Krit-Produk untuk menghasilkan rerata penjualan mingguan. Diperlukan tiga tingkat *query* untuk memperoleh rerata penjualan selama enam bulan dilihat pada hari ini selama enam bulan yang lalu. Sehingga pada *report* akan tampak rerata penjualan aktual dari setiap produk berdasarkan penjualan selama enam bulan yang lalu hingga hari ini. Dari rerata penjualan tersebut, kemudian setiap produk akan dikelompokkan sesuai tingkat penjualannya selama satu bulan tersebut. Perhitungan pada *query* Kriteria1 untuk menghasilkan rerata enam bulan hingga hari ini tersebut dilakukan pada *criteria field* tanggal:

Criteria: >Date()-181

(IV.37)



Gambar IV.41. Komponen sistem pendukung keputusan kriteria produk

Tabel IV.11. *Query Kriteria1*

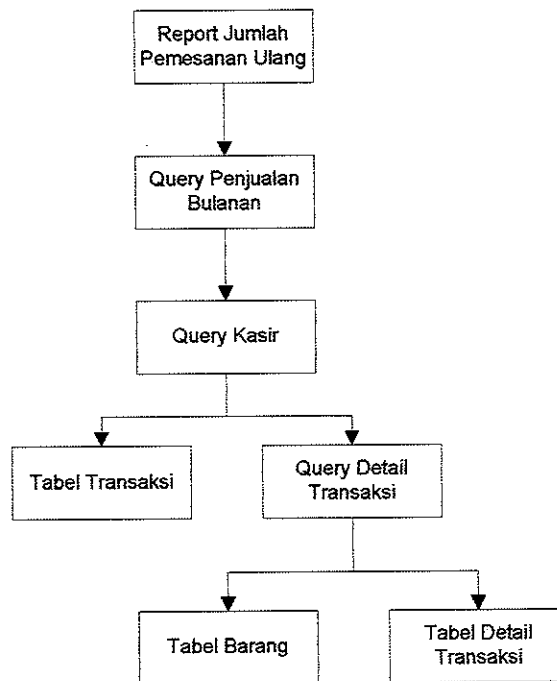
<i>Field</i>	<i>Tabel</i>	<i>Sort</i>	<i>Show</i>	<i>Criteria</i>
Kode	Barang	<i>Ascending</i>	√	
Nama	Barang		√	
Harga Jual	Kasir		√	
Jumlah	Kasir		√	
Tanggal	Kasir		√	(IV.37)

Kemudian data diolah lagi dengan menggunakan *query Kriteria* untuk memperoleh data yang dikelompokkan untuk setiap produk. Data tersebut diolah lagi dalam *query Krit-Produk* untuk memperoleh nilai *sumofjumlah* (nilai total *field* jumlah) dan untuk menuliskan perhitungan untuk tiap kriteria produk.



KODE	NAMA	HARGA JUAL	JUMLAH	PERHARI	KRITERIA
1	BUKU TULIS	2.000,00	30	0 unit/hari	6 BULANAN
2	PENSIL	1.000,00	32	0 unit/hari	6 BULANAN
3	PENGGARIS	1.000,00	22	0 unit/hari	6 BULANAN
4	BUKU GAMBAR	2.500,00	20	0 unit/hari	6 BULANAN
5	BALLPOINT	4.000,00	30	0 unit/hari	6 BULANAN
6	BUKU CERITA	10.000,00	20	0 unit/hari	6 BULANAN
9	STIP	1.200,00	10	0 unit/hari	6 BULANAN

Gambar IV.42. Report DSS Kriteria Produk



Gambar IV.43. Komponen penyusun *report* Jumlah Pemesanan Ulang

Untuk mendukung proses evaluasi produk, selain menentukan kriteria penjualan produk selama enam bulan, sistem ini juga menyediakan *report* yang menampilkan jumlah pemesanan ulang berdasarkan penjualan setiap bulan selama satu periode enam bulan tersebut. Tampilan tersebut dapat dilihat pada *report* Jumlah Pemesanan Ulang. *Report* Jumlah Pemesanan Ulang tersusun dari tabel



Transaksi, tabel Barang dan tabel Detail Transaksi. Untuk menghasilkan data jumlah penjualan setiap produk setiap bulan, maka data dari tabel-tabel tersebut diolah dengan *query* Penjualan Bulanan. Pada *query* ditambahkan *field-field* untuk menampilkan jumlah penjualan produk setiap bulan.

Berikut adalah perhitungan untuk setiap *field-field* tambahan:

- ENAM :  $\text{Sum}(\text{If}(\text{Month}([\text{TANGGAL}])=\text{Month}(\text{Date()});[\text{JUMLAH}];0))$  (IV.40)  
 LIMA :  $\text{Sum}(\text{If}(\text{Month}([\text{TANGGAL}])=(\text{Month}(\text{Date()})-1);[\text{JUMLAH}];0))$  (IV.41)  
 EMPAT :  $\text{Sum}(\text{If}(\text{Month}([\text{TANGGAL}])=(\text{Month}(\text{Date()})-2);[\text{JUMLAH}];0))$  (IV.42)  
 TIGA :  $\text{Sum}(\text{If}(\text{Month}([\text{TANGGAL}])=(\text{Month}(\text{Date()})-3);[\text{JUMLAH}];0))$  (IV.43)  
 DUA :  $\text{Sum}(\text{If}(\text{Month}([\text{TANGGAL}])=(\text{Month}(\text{Date()})-4);[\text{JUMLAH}];0))$  (IV.44)  
 SATU :  $\text{Sum}(\text{If}(\text{Month}([\text{TANGGAL}])=(\text{Month}(\text{Date()})-5);[\text{JUMLAH}];0))$  (IV.45)

Tabel IV.14. *Query* Jumlah Pemesanan Ulang

<i>Field</i>	<i>Tabel</i>	<i>Total</i>	<i>Sort</i>	<i>Show</i>
Kode	Kasir	<i>Group By</i>		√
Nama	Kasir	<i>Group By</i>		√
Enam : (IV.40)		<i>Expression</i>		√
Lima : (IV.41)		<i>Expression</i>		√
Empat : (IV.42)		<i>Expression</i>		√
Tiga : (IV.43)		<i>Expression</i>		√
Dua : (IV.44)		<i>Expression</i>		√
Satu : (IV.45)		<i>Expression</i>		√
ST1 : (IV.46)		<i>Expression</i>		√
ST2 : (IV.47)		<i>Expression</i>		√
ST3 : (IV.48)		<i>Expression</i>		√
SST : (IV.49)		<i>Expression</i>		√
F : (IV.50)		<i>Expression</i>		√

Dari hasil perhitungan di atas, maka akan dihasilkan data jumlah penjualan produk setiap bulan dalam satu periode enam bulan. Kemudian untuk menghitung estimasi jumlah pemesanan ulang periode berikutnya yang berasal dari prakiraan jumlah penjualan periode berikutnya, dipakai perhitungan sebagai berikut:

$$\text{ST1} : ([\text{LIMA}] + [\text{EMPAT}] + [\text{TIGA}]) / 3 \quad (\text{IV.46})$$



$$ST2 : ([EMPAT]+[TIGA]+[DUA])/3 \quad (IV.47)$$

$$ST3 : ([TIGA]+[DUA]+[SATU])/3 \quad (IV.48)$$

$$SST : ([ST1]+[ST2]+[ST3])/3 \quad (IV.49)$$

$$F : \text{Int}(\text{If}(((2*[ST1])-[SST])+( [ST1]-[SST] ))<0;0; \\ ((2*[ST1])-[SST])+( [ST1]-[SST] ))) \quad (IV.50)$$

KODE	NAMA	E-5	E-4	E-3	E-2	E-1	E-0	EST. JUM. ORDER
10.101	Dico 1 kg	0	0	2	1	0	1	1
10.102	Dico 500	4	3	1	3	1	1	0
10.201	Seliga 1	0	1	0	0	0	2	0
10.202	Seliga 0	0	0	0	0	2	0	1
10.203	Riso 100	20	1	0	0	0	0	0
10.204	Riso 1kg	11	11	10	7	12	5	5
10.205	Risowat	8	3	5	10	4	3	5
10.206	Risowat	0	0	1	11	2	1	8
10.208	Riso Sati	100	90	86	147	91	104	92
10.209	Riso 1 kg	4	58	32	55	41	25	47
10.210	Riso 1 kg	14	19	22	12	21	6	18

Gambar IV.44. Report Jumlah Pemesanan Ulang

#### IV.2.4. Sistem Pendukung Keputusan Titik Pemesanan Kembali

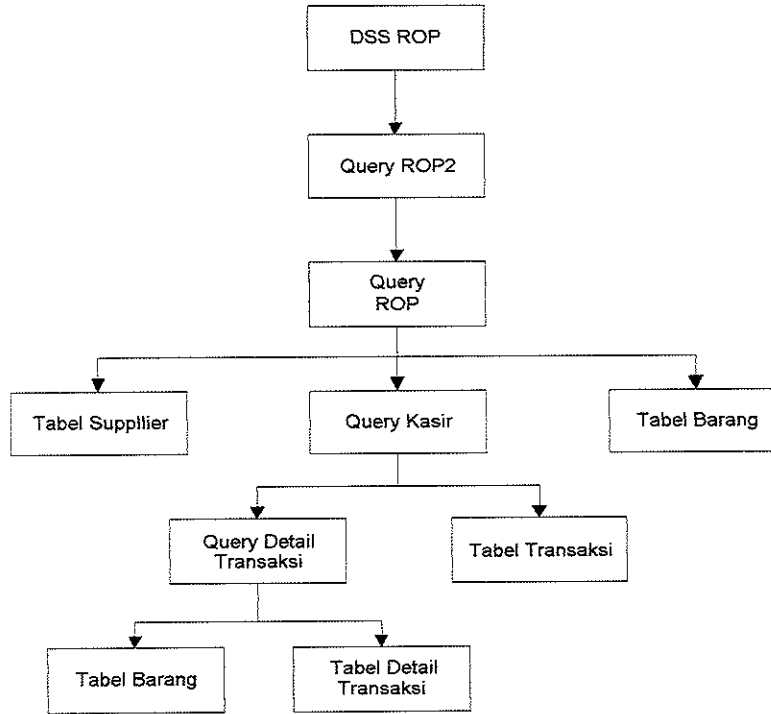
Sistem pendukung keputusan titik pemesanan kembali disusun dari tabel Barang, tabel *Supplier*, tabel Transaksi, dan tabel Detail Transaksi. Tabel-tabel tersebut diolah dalam *query* Kasir, ROP dan ROP2 untuk menghasilkan data pada *report*. *Query* ROP mengolah data untuk menghasilkan nilai penjualan bulanan setiap produk. Perhitungannya sama dengan perhitungan (IV.36). *Query* ROP digunakan untuk mengumpulkan data dari *query* Kasir dalam satu kelompok bulanan. Kemudian *query* ROP2 digunakan untuk mengelompokkan data dari *query* ROP2 untuk setiap produk.

Untuk menentukan keputusan kapan waktu untuk melakukan pemesanan ulang, diperlukan batasan untuk menentukan titik minimal jumlah persediaan sebelum melakukan pemesanan ulang. Titik ini yang disebut sebagai titik pemesanan kembali atau *reorder point*. Sesuai rumus III.1, maka untuk



menentukan titik pemesanan kembali pada pembuatan sistem pendukung keputusan dibatasi oleh:

- *lead time* dari *supplier* (hari)
- rerata penjualan (unit/hari)
- *safety stock* tiap produk (unit)



Gambar IV.45. Komponen sistem pendukung keputusan titik pemesanan ulang

Tabel IV.15. *Query ROP*

<i>Field</i>	<i>Tabel</i>	<i>Sort</i>	<i>Show</i>	<i>Criteria</i>
Kode	Kasir		√	
Nama	Barang		√	
Kode <i>Supplier</i>	<i>Supplier</i>		√	
Nama <i>Supplier</i>	Barang		√	
Stok	Barang		√	
Tanggal	Kasir		√	(IV.37)
Jumlah	Kasir		√	



Tabel IV.16. Query ROP2

Field	Tabel	Total	Sort	Show
Kode	QROP	Group By		√
Nama	QROP	Group By		√
Kode Supplier	QROP	Group By		√
Nama Supplier	QROP	Group By		√
Stok	QROP	Group By		√
Jumlah	QROP	Sum		√

*Safety stock* untuk tiap produk berbeda-beda tergantung pada kategori penjualannya. Pada sistem ini, tingkat *safety stock* dibuat = 3 unit untuk produk dengan kriteria penjualan enam bulanan, 6 unit untuk kriteria produk tiga bulanan, 9 unit untuk kriteria produk bulanan, dan 12 unit untuk kriteria produk mingguan. Pemberian nilai *safety stock* untuk setiap kriteria merupakan kebijakan perusahaan. Pada desain *report*, ditambahkan *field* untuk menampilkan nilai rerata mingguan dan *field* titik pemesanan kembali setiap produk. Perhitungan pada *field* titik pemesanan kembali perhitungannya menurut (III.1).

RODP	NAMA	K. SUPL	NAMA SUPPLIER	TUC PERI	AVG STOK	ROP
1	10-8 TULS	1	ASLIANA	II	0 IN PERI	5 Unit
2	PERUL	2	URATANA	II	0 IN PERI	5 Unit
3	PASIS-ANG	2	URATANA	II	0 IN PERI	5 Unit
4	10-8 SAMBAR	2	ANGUS BAHU	II	0 IN PERI	5 Unit
5	8-10-80 NE	2	URATANA	II	0 IN PERI	5 Unit
6	SUNUCERNA	3	ANGUS BAHU	II	0 IN PERI	5 Unit
7	STP	3	ANGUS BAHU	II	0 IN PERI	5 Unit

Gambar IV.46. Report DSS ROP

Langkah berikutnya adalah menyusun sebuah *report* gabungan yang mampu menampilkan sekaligus data waktu pesan ulang yang ditentukan oleh titik pemesanan kembali, jumlah pemesanan ulang untuk setiap produk.

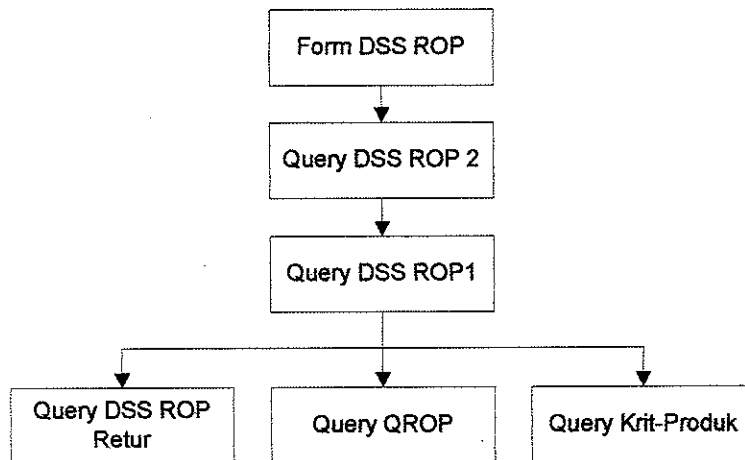


KODE	NAMA	TOT. PENJUALAN	EST. JML. ORDER	ROP
10201	Duo 1kg	9	1	5 Unit
10202	Duo 500gr	13	3	5 Unit
10203	Sata 1kg	3	3	5 Unit
10204	Sata 0.5	2	1	5 Unit
10205	Raso 100gr	24	3	5 Unit
10206	S.20 1kg	57	3	5 Unit
10207	Raso Warna 1kg	12	3	5 Unit
10208	Raso Warna 400gr	15	3	5 Unit

Gambar IV.47. Report DSS Gabungan

#### IV.2.5. DSS ROP

Komponen DSS ini berfungsi untuk langsung menghasilkan keputusan perlu atau tidaknya pemesanan dilakukan pada suatu hari tertentu. Keputusan ini diambil dengan melakukan perbandingan antara jumlah stok yang tersisa dengan batas minimal jumlah produk dimana harus dilakukan pemesanan ulang (ROP). Komponen yang menyusun Form DSS Re-Order Point ini adalah seperti ditunjukkan pada Gambar IV.48. di bawah ini.



Gambar IV.48. Komponen Penyusun Form DSS ROP



Yang membedakan antara Form DSS ROP dengan Report DSS ROP adalah informasi yang ditampilkan. Pada Report DSS ROP, keputusan belum dapat dihasilkan secara otomatis, jadi pengguna administrator harus membuka form DSS ROP dan melihat stok melalui Form Item, lalu mengambil keputusan secara manual apakah harus melakukan pemesanan atau tidak.

Di sisi lain, Form DSS ROP akan memberikan informasi secara otomatis mengenai produk-produk apa saja yang harus segera dipesan pada hari tertentu. Dengan menggunakan form, fungsi filter dalam Microsoft Access juga dapat dimanfaatkan, sehingga administrator bukan hanya dapat mengetahui produk apa saja yang harus dipesan, namun juga yang berasal dari supplier tertentu misalnya. Form DSS ROP akan meninjau kondisi stok aktual terlebih dulu. Pengecekan ini dilakukan dengan memanfaatkan hubungan antara Query QROP, Query DSS ROP Retur, dan Query Krit-Produk. Persamaan yang digunakan untuk menghitung sisa stok aktual adalah :

$$\text{SISA:} \quad \text{QROP.STOK-[KRIT-PRODUK].SUMOFJUMLAH-[DSS ROP} \\ \text{RETUR].[SUMOFJUMLAH RETUR]} \quad (\text{IV.51})$$

Persamaan yang digunakan di atas akan mengurangi jumlah stok awal sebuah produk pada tabel order yang terdapat pada Query QROP dengan total jumlah penjualan pada Tabel Detail Transaksi yang terdapat pada Query Krit-Produk, dan dikurangi lagi dengan total retur yang diambil dari Tabel Detail Purchasing. Hasil sisa ini akan dibandingkan dengan nilai ROP untuk produk yang bersangkutan. Perhitungan nilai ROP pada ini menggunakan rumus yang sama dengan yang digunakan pada Report DSS ROP. Dengan membandingkan kedua angka tersebut, maka basis data dapat secara langsung menghasilkan keputusan apakah harus memesan atau tidak sebuah produk dari supplier produk yang bersangkutan.

Form DSS ROP hanya akan menampilkan produk yang berstatus “harus pesan” saja. Pembuatan status ini merupakan hasil perbandingan dari dua buah angka di atas. Kode program yang digunakan adalah :



STATUS: IIf((QROP.STOK-[KRIT-PRODUK].SUMOFJUMLAH) <=(Int((QROP.[LEAD TIME]\*([KRIT-PRODUK].SUMOFJUMLAH/180)))+(IIf([KRITERIA]="A: 'MINGGUAN";12;IIf([KRITERIA]="B: 'BULANAN";9;IIf([KRITERIA]="C: '3 BULANAN";6;3)))));"ORDER";"STOK AMAN") (IV.52)

Kode program diatas akan menghasilkan status “STOK AMAN” jika jumlah stok lebih besar dibandingkan nilai ROP, sebaliknya jika stok lebih kecil dari nilai ROP, maka status produk tersebut adalah “ORDER”. Untuk dapat menampilkan hanya produk-produk dengan status “ORDER” saja, maka pada Query DSS ROP 2 ditambahkan kriteria “ORDER” pada baris kriteria kolom STATUS. Dengan kriteria ini maka produk-produk dengan jumlah stok aman tidak akan ditampilkan pada form tersebut.

DAFTAR PRODUK

DAFTAR PRODUK YANG HARUS DIORDER

25 November 2008

KODE	NAMA	KD SUPP	NAMA SUPPLIER	SISA	KRITERIA	ROP
48800	Coffesor	2	WIRATAMA	800	A: 'MINGGUAN'	82
270101	Indonsi Gung	2	WIRATAMA	898	A: 'MINGGUAN'	85
270704	ABC Semu / Gula P	2	WIRATAMA	44	A: 'MINGGUAN'	52
230101	Energen T.Bh	1	ADI JAYA	34	A: 'MINGGUAN'	40
303304	Yakui	3	ANGIN BARU	944	A: 'MINGGUAN'	105
JR201	Aqua gelas	3	ANGIN BARU	73	A: 'MINGGUAN'	85

Record: 14 | Page: 1 of 6

Gambar IV.49. Form DSS ROP

#### IV.2.6. Form Navigasi

Form navigasi merupakan form yang berfungsi untuk mempermudah pengguna dalam memilih form yang diinginkan. Form navigasi terdiri atas dua macam, yaitu: form Menu Utama dan form Menu DSS. Tombol-tombol tersebut dapat difungsikan dengan menggunakan *mouse* maupun *keyboard* dengan menggunakan tab order untuk memilih dan menekan *enter* untuk membuka form yang diinginkan. Penggunaan bahasa VBA pada kedua form tersebut adalah sebagai berikut:



## Menu Utama

```
Private Sub supplier_Click()
On Error GoTo Err_supplier_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "INPUT SUPPLIER"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
    DoCmd.GoToRecord acDataForm, stDocName, acLast
Exit_supplier_Click:
    Exit Sub
Err_supplier_Click:
    MsgBox Err.Description
    Resume Exit_supplier_Click
End Sub

Private Sub barang_Click()
On Error GoTo Err_barang_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "INPUT BARANG"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
    DoCmd.GoToRecord acDataForm, stDocName, acLast
Exit_barang_Click:
    Exit Sub
Err_barang_Click:
    MsgBox Err.Description
    Resume Exit_barang_Click
End Sub

Private Sub purchasing_Click()
On Error GoTo Err_purchasing_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "PURCHASING"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
    DoCmd.GoToRecord acDataForm, stDocName, acLast
Exit_purchasing_Click:
    Exit Sub
Err_purchasing_Click:
    MsgBox Err.Description
    Resume Exit_purchasing_Click
End Sub

Private Sub total_pembelian_Click()
On Error GoTo Err_total_pembelian_Click
    Dim stDocName As String
    stDocName = "TOTAL PEMBELIAN"
    DoCmd.OpenReport stDocName, acPreview
Exit_total_pembelian_Click:
    Exit Sub
Err_total_pembelian_Click:
    MsgBox Err.Description
    Resume Exit_total_pembelian_Click
End Sub
```



```
Private Sub retur_Click()
On Error GoTo Err_retur_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "RETUR"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_retur_Click:
    Exit Sub
Err_retur_Click:
    MsgBox Err.Description
    Resume Exit_retur_Click
End Sub

Private Sub transaksi_Click()
On Error GoTo Err_transaksi_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TRANSAKSI"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
    DoCmd.GoToRecord acDataForm, stDocName, acNewRec
Exit_transaksi_Click:
    Exit Sub
Err_transaksi_Click:
    MsgBox Err.Description
    Resume Exit_transaksi_Click
End Sub

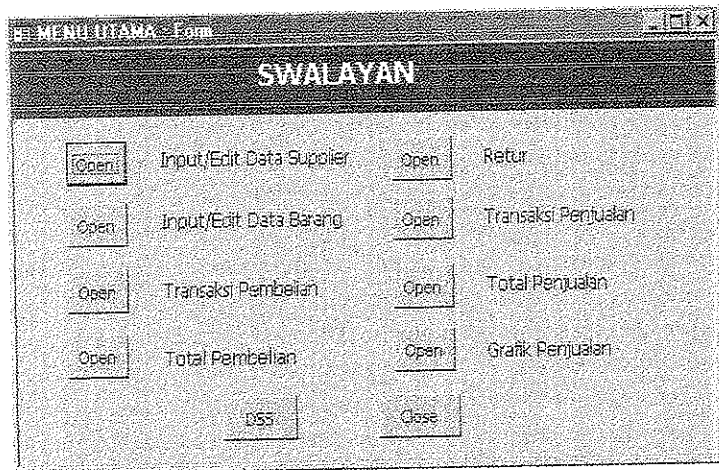
Private Sub Total_Penjualan_Click()
On Error GoTo Err_Total_Penjualan_Click
    Dim stDocName As String
    stDocName = "TOTAL PENJUALAN"
    DoCmd.OpenReport stDocName, acPreview
Exit_Total_Penjualan_Click:
    Exit Sub
Err_Total_Penjualan_Click:
    MsgBox Err.Description
    Resume Exit_Total_Penjualan_Click
End Sub

Private Sub grafik_penjualan_Click()
On Error GoTo Err_grafik_penjualan_Click
    Dim stDocName As String
    stDocName = "GRAFIK PENJUALAN"
    DoCmd.OpenReport stDocName, acPreview
Exit_grafik_penjualan_Click:
    Exit Sub
Err_grafik_penjualan_Click:
    MsgBox Err.Description
    Resume Exit_grafik_penjualan_Click
End Sub

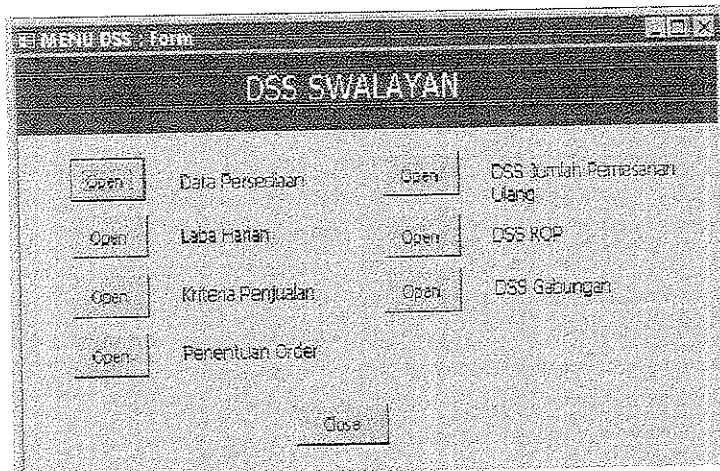
Private Sub dss_Click()
On Error GoTo Err_dss_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
```



```
stDocName = "MENU DSS"  
DoCmd.OpenForm stDocName, , , stLinkCriteria  
Exit_dss_Click:  
Exit Sub  
Err_dss_Click:  
MsgBox Err.Description  
Resume Exit_dss_Click  
End Sub  
  
Private Sub close_Click()  
On Error GoTo Err_close_Click  
DoCmd.close  
Exit_close_Click:  
Exit Sub  
Err_close_Click:  
MsgBox Err.Description  
Resume Exit_close_Click  
End Sub
```



Gambar IV.50. Form Menu Utama



Gambar IV.51. Form Menu DSS



## Menu DSS

```
Private Sub item_Click()  
On Error GoTo Err_item_Click  
    Dim stDocName As String  
    Dim stLinkCriteria As String  
    stDocName = "ITEM"  
    DoCmd.OpenForm stDocName, , , stLinkCriteria  
Exit_item_Click:  
    Exit Sub  
Err_item_Click:  
    MsgBox Err.Description  
    Resume Exit_item_Click  
End Sub
```

```
Private Sub laba_Click()  
On Error GoTo Err_laba_Click  
    Dim stDocName As String  
    stDocName = "LABA HARIAN"  
    DoCmd.OpenReport stDocName, acPreview  
Exit_laba_Click:  
    Exit Sub  
Err_laba_Click:  
    MsgBox Err.Description  
    Resume Exit_laba_Click  
End Sub
```

```
Private Sub dss_jual_Click()  
On Error GoTo Err_dss_jual_Click  
    Dim stDocName As String  
    stDocName = "DSS KRITERIA PRODUK"  
    DoCmd.OpenReport stDocName, acPreview  
Exit_dss_jual_Click:  
    Exit Sub  
Err_dss_jual_Click:  
    MsgBox Err.Description  
    Resume Exit_dss_jual_Click  
End Sub
```

```
Private Sub jumlah_reorder_Click()  
On Error GoTo Err_jumlah_reorder_Click  
    Dim stDocName As String  
    stDocName = "JUMLAH PEMESANAN ULANG"  
    DoCmd.OpenReport stDocName, acPreview  
Exit_jumlah_reorder_Click:  
    Exit Sub  
Err_jumlah_reorder_Click:  
    MsgBox Err.Description  
    Resume Exit_jumlah_reorder_Click  
End Sub
```

```
Private Sub dss_rop_Click()  
On Error GoTo Err_dss_rop_Click  
    Dim stDocName As String  
    stDocName = "DSS ROP"  
    DoCmd.OpenReport stDocName, acPreview
```



```
Exit_dss_rop_Click:
    Exit Sub
Err_dss_rop_Click:
    MsgBox Err.Description
    Resume Exit_dss_rop_Click
End Sub

Private Sub dss_gabungan_Click()
On Error GoTo Err_dss_gabungan_Click
    Dim stDocName As String
    stDocName = "DSS GABUNGAN"
    DoCmd.OpenReport stDocName, acPreview
Exit_dss_gabungan_Click:
    Exit Sub

Err_dss_gabungan_Click:
    MsgBox Err.Description
    Resume Exit_dss_gabungan_Click
End Sub

Private Sub DSS_ROP_FORM_Click()
On Error GoTo Err_DSS_ROP_FORM_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "DSS ROP"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_DSS_ROP_FORM_Click:
    Exit Sub

Err_DSS_ROP_FORM_Click:
    MsgBox Err.Description
    Resume Exit_DSS_ROP_FORM_Click
End Sub

Private Sub close_Click()
On Error GoTo Err_close_Click
    DoCmd.close
    stDocName = "MENU UTAMA"
    DoCmd.OpenForm stDocName, acNormal, , , stLinkCriteria
Exit_close_Click:
    Exit Sub

Err_close_Click:
    MsgBox Err.Description
    Resume Exit_close_Click
End Sub
```

### **IV.3. PENGGUNAAN SISTEM PENDUKUNG KEPUTUSAN**

Sistem pendukung keputusan pemesanan ulang dibentuk untuk membantu kerja bagian pembelian. Sistem ini berguna dalam memberikan informasi produk-produk yang mengalami pemesanan ulang. Sistem ini semula belum ada, dan dirancang untuk membantu proses pemesanan ulang agar proses pemesanan ulang



dapat dilakukan dengan lebih baik dan lebih cepat. Penggunaan sistem pendukung keputusan ini dapat dijelaskan sebagai berikut:

1. Pada form navigasi menu utama, klik tombol DSS;
2. Klik tombol *Open* Penentuan Order;
3. Evaluasi produk yang harus dipesan pada hari ini pada tampilan *report* tersebut, kemudian tutup *report* jika evaluasi telah selesai;
4. Jika ingin mengetahui produk mana yang harus dipesan berdasarkan nama *supplier*, lakukan *filter* berdasarkan nama *supplier* yang diinginkan;
5. Evaluasi kriteria produk yang harus dipesan sesuai nama *supplier*; *remove filter* jika evaluasi telah selesai;
6. Klik tombol *Open* Jumlah Pemesanan Ulang pada form navigasi menu DSS;
7. Lakukan pemesanan ulang sesuai dengan jumlah yang tertera pada *report* tersebut, atau sesuaikan dengan kebijakan swalayan. Tutup *report* jika evaluasi telah selesai dilakukan.



## BAB V

### ANALISIS DAN PEMBAHASAN

#### V.1. ANALISIS UMUM

Tujuan utama perancangan ini adalah menciptakan sistem pendukung keputusan pemesanan ulang setiap jenis produk dalam swalayan. Pembahasan akan difokuskan pada penerapan basis data tersebut pada sampel data nyata dari swalayan. Dari data penjualan tersebut akan dibahas kelebihan dan kekurangan penggunaan basis data ini untuk menentukan efektivitas pengambilan keputusan pemesanan ulang setiap produk.

Sistem pendukung keputusan pemesanan ulang ini disusun dari basis data persediaan yang telah dibentuk sebelumnya. Nilai-nilai yang digunakan untuk menentukan keputusan didapatkan dari informasi manajemen swalayan. Pengembangan nilai-nilai yang digunakan dalam sistem pendukung keputusan merupakan asumsi penulis berdasarkan informasi tersebut. Informasi dari manajemen swalayan yang digunakan sebagai acuan penentuan nilai dalam basis data adalah sebagai berikut:

- penjualan terhadap produk baru diberi tenggang waktu hingga satu bulan untuk evaluasi;
- *lead time supplier* produk bervariasi;
- *safety stock* hanya diberikan pada produk yang memiliki kategori cepat laku;
- produk-produk akan mengalami retur jika mengalami cacat produk;
- rerata penjualan produk cepat laku adalah 30-40 unit per minggu.

#### V.2. ANALISIS BASIS DATA

Berdasarkan data-data tersebut kemudian penulis menentukan nilai-nilai pada basis data. Pada basis data, nilai-nilai tersebut dijadikan dasar dalam menentukan tabel pembentuk form maupun *report*.



Seperti dijelaskan sebelumnya pada bab IV, mengenai prosedur penggunaan basis data, maka akan dibahas mengenai asumsi-asumsi dalam penggunaan basis data tersebut. Berikut akan dibahas kelebihan dan kekurangan basis data tersebut.

Basis data harus memiliki *record* penjualan minimal enam bulan, karena untuk menentukan nilai rerata penjualan pada sistem pendukung keputusan titik pemesanan kembali maupun sistem pendukung keputusan kriteria produk diperlukan data penjualan selama enam bulan. Penggunaan batas waktu periode enam bulan berdasarkan atas kriteria penjualan terlama. Enam bulan merupakan periode yang dijadikan acuan dalam menentukan batas waktu produk dalam rak penjualan. Swalayan tersebut menjual produk-produk konsumsui, sehingga periode enam bulan dapat dijadikan acuan dalam menentukan apakah suatu produk laku di pasaran. Sehingga dimungkinkan untuk dapat memperhatikan tingkat penjualan produk tersebut selama enam bulan. Selain itu, pada *report* Jumlah Pemesanan Ulang akan nampak jumlah penjualan produk tersebut setiap bulannya selama enam bulan. Sehingga proses evaluasi dapat menjadi lebih mudah. Kelemahan basis data ini jika diterapkan pada sistem yang benar-benar baru, maka sistem pendukung keputusan tidak akan berfungsi dengan baik karena belum memiliki *record* penjualan selama enam bulan. Sedang jika diterapkan pada sistem yang sudah berjalan, dan memiliki kesamaan bentuk dasar, maka data dari sistem lama dapat diubah ke dalam basis data ini.

Keputusan perubahan harga timbul dari *supplier*. Jika terjadi perubahan harga sebelum periode dalam basis data berakhir, maka produk yang sama akan didata dengan kode yang berbeda. Kelemahan basis data ini adalah tidak dimungkinkan terjadi perubahan harga produk pada pertengahan periode basis data. Sehingga agar data penjualan produk dengan harga lama tidak terhapus, maka produk tersebut didata dengan kode yang berbeda. Namun kelebihanannya, data dapat terorganisir dengan baik karena tidak dimungkinkan terjadi kerancuan atas satu jenis produk dengan dua identitas atau lebih. Sehingga untuk proses *update* data dapat dilakukan dengan cepat. Basis data dapat melakukan proses pengolahan data tanpa kesalahan identitas.



*Supplier* memiliki *lead time* yang berbeda-beda, tergantung pada tingkat penjualan produknya. Basis data menyediakan data *lead time* untuk setiap *supplier*. Data tersebut kemudian digunakan untuk menghitung nilai titik pemesanan kembali. Sehingga dihasilkan data yang akurat pada perhitungan tersebut.

Pada form input barang dan *supplier* nomor kode diisikan oleh pihak manajemen. Proses penomoran suatu produk didasarkan pada kelompok produk tersebut. Begitu pula dengan penomoran kode *supplier*. Sehingga pada proses input, nomor kode ini harus diisikan sendiri oleh manajemen agar sesuai ketentuan.

Penggunaan form Item sebagai alat untuk mengetahui jumlah persediaan suatu produk membantu pihak manajemen dalam menentukan keputusan lebih lanjut dalam hal pemesanan ulang produk tersebut pada periode berikutnya. Penggunaannya sedikit rumit tapi akan lebih cepat dibandingkan dengan proses manual.

Proses penambahan jumlah persediaan yang disebabkan oleh pembelian produk baru dilakukan secara otomatis dengan *update query*. Proses penambahan jumlah persediaan tersebut tidak ditampilkan secara rinci dikarenakan proses pembelian merupakan hal yang rutin. Sehingga rincian pembelian bukanlah hal yang perlu ditampilkan secara detail.

Dalam proses pengolahan data jumlah persediaan, data jumlah stok ditampilkan dalam *field* stok dan menjadi dasar proses berikutnya, yaitu pengurangan jumlah persediaan karena penjualan dan retur. Rincian penjualan dan retur perlu ditampilkan karena proses tersebut merupakan hal yang perlu selalu dievaluasi. Karena proses pada form ini memungkinkan manajemen segera mengetahui tingkat penjualan suatu produk. Sehingga jika terjadi penjualan yang sangat kurang dapat segera dievaluasi. Kelemahannya adalah jika manajemen tidak secara rutin mengevaluasi penjualan seluruh produk maka dimungkinkan suatu produk mengalami penjualan yang sangat kurang atau bahkan tidak terjual sampai periode tertentu dan tidak mendapat evaluasi lebih lanjut. Hal ini dikarenakan pada report total penjualan, produk yang tidak memiliki penjualan



sama sekali tidak akan muncul dalam report tersebut. Sehingga seakan-akan produk tersebut tidak ada.

### V.3. ANALISIS SISTEM PENDUKUNG KEPUTUSAN

Pembahasan pada sistem pendukung keputusan ditekankan pada penentuan nilai-nilai untuk menentukan kriteria suatu produk dan menentukan titik pemesanan kembali.

Pada sistem pendukung keputusan kriteria produk, rerata penjualan adalah rerata bergerak, yang berarti nilai pada rerata penjualan akan selalu berubah sesuai dengan tingkat penjualan produk tersebut setiap harinya. Rerata penjualan dihitung berdasarkan periode enam bulan terhitung mulai enam bulan sebelum hari ini. Sehingga dimungkinkan terjadi perubahan nilai rerata suatu produk, jika diamati pada hari yang berbeda pada bulan yang sama. Tujuannya adalah untuk mendapatkan informasi rerata penjualan suatu produk yang cukup mewakili dan informatif. Rerata penjualan dihitung selama enam bulan, sebab dalam enam bulan akan terlihat dengan jelas pola penjualan suatu produk. Selain itu, untuk kriteria penjualan produk terlama, periode penjualannya adalah enam bulan. Sehingga batas enam bulan dijadikan acuan untuk mengamati tingkat penjualan suatu produk. Hal ini juga menjadi acuan untuk dapat mengamati perubahan tingkat penjualan produk tersebut selama enam bulan. Sehingga dapat dijadikan bahan pertimbangan untuk menentukan pembelian produk pada periode berikutnya.

Pada penjualan produk baru, jika selama satu bulan produk tersebut diterima konsumen, dalam arti memiliki tingkat penjualan yang cukup, maka produk akan dibeli kembali pada periode berikutnya. Namun jika ternyata penjualannya di bawah standard, maka produk tersebut akan dievaluasi kembali.

Pada sistem pendukung keputusan kriteria produk, nilai-nilai yang digunakan berasal dari pengembangan informasi dari manajemen swalayan. Nilai-nilai tersebut digunakan dalam menentukan batas kriteria penjualan suatu produk. Dalam penggunaannya, nilai-nilai pada penentuan kriteria produk dapat berubah jika swalayan mengalami perubahan yang cukup berpengaruh pada tingkat



penjualan produk secara umum. Penggunaan kriteria tersebut adalah sebagai acuan evaluasi penjualan produk. Jika manajemen menemukan produk dengan kriteria penjualan mingguan. Namun kemudian ditemukan penjualannya tiba-tiba menurun, maka dapat segera dilakukan evaluasi penyebab penurunan penjualan maupun usaha untuk meningkatkan penjualan produk tersebut kembali.

Dari informasi yang dihasilkan dari sistem pendukung keputusan, maka keputusan selanjutnya adalah wewenang manajemen. Informasi tersebut hanya untuk membantu dalam menentukan produk mana yang harus segera dipesan berdasarkan informasi dari sistem pendukung keputusan titik pemesanan kembali dan data jumlah persediaan dari gudang. Selain itu juga mengenai informasi kriteria penjualan produk tersebut jika

Mengenai sistem pendukung keputusan titik pemesanan kembali, masalah utama yang perlu diperhatikan adalah *safety stock*. Permasalahan *safety stock* juga menjadi keputusan manajemen. Sistem pendukung keputusan memberikan data minimal *safety stock* sebagai variabel yang bisa diubah oleh pihak swalayan sewaktu-waktu. Penghitungan *safety stock* didasarkan atas kebijakan manajemen pada kriteria penjualan produk tersebut. Sehingga produk-produk dengan tingkat penjualan berbeda akan memiliki *safety stock* yang berbeda pula. Sistem ini sedikit rumit karena setiap kali suatu produk mengalami perubahan kriteria, maka pihak manajemen harus menentukan lagi nilai *safety stoknya*. Namun kelebihanannya, sistem ini akan selalau dapat memberikan nilai *safety stock* yang memadai bagi setiap kriteria.

Sebelumnya, untuk menentukan pembelian produk untuk periode berikutnya bagian penjualan hanya mengambil keputusan berdasarkan kebijakan kriteria produk, *safety stock* dengan dasar keputusan manajemen semata. Namun dengan sistem pendukung keputusan ini dapat dihasilkan informasi mengenai kriteria dan *safety stock* secara otomatis untuk setiap produk. Sehingga keputusan yang salah dapat diminimalisasi.

Dalam penentuan sistem pendukung keputusan, data-data dari bagian penjualan merupakan sumber dari seluruh pengolahan dalam sistem pendukung keputusan sehingga diperoleh informasi yang diperlukan oleh manajemen. Dalam



pengolahannya, sistem pendukung keputusan menggunakan beberapa metode untuk menghasilkan informasi yang dibutuhkan oleh manajemen.

Penentuan waktu pemesanan ulang ditentukan dengan titik pemesanan ulang. Secara nyata sistem ini hanya memberikan informasi batas jumlah persediaan yang harus tersedia sebelum melakukan pemesanan ulang. Namun dalam proses evaluasi, batas tersebut dapat dijadikan acuan dalam penentuan waktu pemesanan ulang yang tepat untuk periode berikutnya.

Penentuan jenis produk yang mengalami pemesanan ulang didasarkan pada kriteria penjualan tiap produk. Jika produk memiliki kriteria penjualan yang sangat kecil, manajemen perlu mengevaluasi lebih lanjut mengenai perlu tidaknya melakukan pemesanan ulang terhadap produk tersebut. Ada beberapa faktor yang harus diperhatikan oleh pihak manajemen dalam menentukan keputusan tersebut. Tidak hanya berdasarkan jumlah penjualan selama periode tertentu, namun juga perlu tidaknya swalayan memiliki produk tersebut sebagai pelengkap kebutuhan konsumen.

Yang terakhir adalah jumlah pemesanan yang harus dilakukan pada produk yang mengalami pemesanan ulang. Jumlah ini ditentukan dengan metode rerata bergerak linear yang mampu mengakomodasi *trend*. Walaupun metode ini cukup efektif untuk mengatasi *trend*, ternyata masih memiliki kelemahan yaitu tidak mampu memberikan data yang tepat untuk *trend* yang random.

Secara menyeluruh sistem pendukung keputusan ini diterapkan pada sampel data swalayan periode di atas, maka akan menghasilkan data yang cukup akurat. Bahkan dari informasi yang dihasilkan oleh sistem ini, ternyata diperoleh hasil yang cukup baik. Dimana sistem pendukung keputusan ini mampu menghasilkan informasi yang dibutuhkan, sehingga jika sistem ini benar-benar diterapkan pada sistem nyata, akan memberikan informasi yang membantu proses pengambilan keputusan pemesanan ulang. Sehingga jumlah persediaan yang dimiliki oleh swalayan akan sesuai dengan kebutuhan konsumen, tidak berlebih dan tidak kekurangan.



Mean square error: 
$$MSE = \sum_{i=1}^n \frac{e_i^2}{n} \quad (V.3)$$

Percentage error: 
$$PE_t = \left( \frac{X_t - F_t}{X_t} \right) * (100) \quad (V.4)$$

Mean absolute percentage error: 
$$MAPE = \sum_{i=1}^n \frac{|PE_i|}{n} \quad (V.5)$$

Tabel V.2. Pengujian kesalahan pada kode produk 20805

Bulan	X <sub>t</sub>	S' <sub>t</sub>	S'' <sub>t</sub>	a	b	F2	e	e <sup>2</sup>	PE	PE
Juni	6									
Juli	6									
Agustus	2	4.67								
September	5	4.33								
Oktober	3	3.33	4.11	2.56	-0.78					
Nopember	2	3.33	3.67	3.00	-0.33					
Desember	3	2.67	3.11	2.22	-0.44	1.00	2.00	4.00	66.67	66.67
Januari	3	2.67	2.89	2.44	-0.22	2.33	0.67	0.44	22.22	22.22
Pebruari	0	2.00	2.44	1.56	-0.44	1.33	-1.33	1.78	0.00	0.00
Maret	1	1.33	2.00	0.67	-0.67	2.00	-1.00	1.00	-100.00	100.00
April	3	1.33	1.56	1.11	-0.22	0.67	2.33	5.44	77.78	77.78
Mei	0	1.33	1.33	1.33	0.00	0.00	0.00	0.00	0.00	0.00
Juni	1	1.33	1.33	1.33	0.00	0.67	0.33	0.11	33.33	33.33
Juli	1	0.67	1.11	0.22	-0.44	1.33	-0.33	0.11	-33.33	33.33
Agustus	0	0.67	0.89	0.44	-0.22	1.33	-1.33	1.78	0.00	0.00
							ME	MSE	MPE	MAPE
							0.15	1.63	7.41	37.04

Tabel V. 3. Perbandingan Hasil Peramalan LMA 3X3 dengan 2X2

Kode Produk	LMA 3X3		LMA 2X2	
	MSE	MAPE	MSE	MAPE
323201	316.9753	88.57711	855.2938	104.744
270816	5207.494	56.71643	6137.069	66.89674
10601	17.74074	25.92593	25.4875	26.66667

Dari hasil perhitungan dengan kedua metode tersebut pada 3 produk yang diambil secara acak, dihasilkan nilai MSE dan MAPE pada *linear moving average* 3x3 lebih baik dibandingkan *linear moving average* 2x2.



Data penjualan yang digunakan adalah data penjualan swalayan Mataram Mandiri selama enam bulan periode Mei – November 2001. Diperoleh data sebagai berikut:

- terdapat 4108 jenis produk yang dijual dalam swalayan pada periode tersebut.
- produk yang dijual bervariasi dari produk makanan, kosmetik, alat-alat kebutuhan rumah tangga, dan alat tulis.
- produk terjual selama enam bulan bervariasi mulai dari tingkat penjualan 0 (nol) hingga ribuan.



## BAB VI

### KESIMPULAN DAN SARAN

#### VII. KESIMPULAN

Berdasarkan pembahasan pada bagian sebelumnya, dapat disimpulkan bahwa:

1. Basis data sangat membantu proses *update* data jumlah persediaan swalayan.
2. Pembuatan basis data persediaan terdiri atas dua subsistem yaitu:
  - Subsistem pembelian
  - Subsistem penjualan
3. Subsistem pembelian menangani seluruh input data pembelian, penyimpanan dan pelaporan total pembelian atas seluruh produk.
4. Subsistem penjualan bertanggung jawab atas proses input data penjualan, penyimpanan data penjualan, dan pelaporan total penjualan seluruh produk.
5. Pengolah data persediaan bertanggung jawab mengolah data dari bagian penjualan dan pembelian sehingga selalu diperoleh data persediaan yang terakhir.
6. Pada penggunaannya, basis data tersebut harus memiliki data penjualan minimal enam bulan agar mampu menghasilkan sistem pendukung keputusan yang baik.
7. Sistem pendukung keputusan terdiri atas penentuan jenis, waktu, dan jumlah pemesanan ulang untuk periode berikutnya.
8. Penentuan jenis produk yang mengalami pemesanan ulang dilakukan dengan DSS Kriteria Produk.
9. Penentuan waktu pemesanan ulang dilakukan dengan DSS ROP.
10. Penentuan jumlah pemesanan ulang untuk setiap produk dilakukan dengan DSS Jumlah Pemesanan Ulang
11. Nilai-nilai pada penentuan *safety stock*, dan kriteria penjualan produk merupakan variabel yang bisa diubah sesuai kebutuhan manajemen.



## **VI.2. SARAN**

1. Perlu dikembangkan basis data yang mampu mengakomodasi perubahan harga dengan lebih baik.
2. Perlu dikembangkan basis data yang mampu mengakomodasi subsistem keuangan. Sehingga dapat mengolah data dari sisi keuangan.
3. Perlu dikembangkan sistem pendukung keputusan yang mampu mengolah data keuangan menjadi faktor penentu dalam memutuskan pemesanan ulang berikutnya.
4. Perlu dikembangkan sistem pendukung keputusan yang mampu mengolah data untuk memprediksi jumlah pesanan berikutnya yang dipengaruhi oleh tidak hanya tingkat penjualan sebelumnya, namun juga kondisi yang lebih kompleks seperti kondisi ekonomi perusahaan, kebijakan negara, dan lain-lain.



## DAFTAR PUSTAKA

- Kotler, P., Teguh, H., Rusli, R.A., 1997, *Manajemen Pemasaran, Analisis, Perencanaan, Implementasi dan Kontrol*, Jilid I, Prenhallindo, Jakarta.
- Kotler, P., Teguh, H., Rusli, R.A., 1997, *Manajemen Pemasaran, Analisis, Perencanaan, Implementasi dan Kontrol*, Jilid II, Prenhallindo, Jakarta.
- LPKBM MADCOMS, 2000, *Seri Panduan Lengkap Microsoft Access 2000*, edisi I, cetakan ke-2, Andi, Yogyakarta.
- Makridakis, S., Wheelwright, S. C., McGee, V. E., Andriyanto, U. S., Basith, A., 1983, *Metode dan Aplikasi Peramalan*, Jilid 1, edisi 2, cetakan ke-5, Erlangga, Jakarta.
- McLeod Jr., R., Teguh, H., 1996, *Sistem Informasi Manajemen*, Jilid I, PT Bhuana Ilmu Populer, Jakarta.
- McLeod Jr., R., Teguh, H., 1996, *Sistem Informasi Manajemen*, Jilid II, PT Bhuana Ilmu Populer, Jakarta.
- Post, G. V., 2002, *Database Management System, Designing and Building Business Application*, 2<sup>nd</sup> ed, McGraw-Hill, New York.
- Scott, G. M., Budiman, A. N., 1999, *Prinsip-Prinsip Sistem Informasi Manajemen*, cetakan ke-5, PT Raja Grafindo Persada, Jakarta.
- Smith, S. B., 1989, *Computer Based Production and Inventory Control*, Prentice Hall International, Inc., New Jersey.