

**PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN  
SISTEM KONSINYASI PADA CV. MITRA MANDIRI PASURUAN**

**SKRIPSI**



Disusun Oleh :  
**Donny Prasetyo**  
NIM.0612524



**JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2011**

UNIVERSITY OF CALIFORNIA LIBRARY  
AND ARCHIVES CENTER AND ARCHIVES CENTER

UNIVERSITY OF CALIFORNIA LIBRARY  
AND ARCHIVES CENTER

UNIVERSITY OF CALIFORNIA LIBRARY  
AND ARCHIVES CENTER

UNIVERSITY OF CALIFORNIA LIBRARY  
AND ARCHIVES CENTER  
UNIVERSITY OF CALIFORNIA LIBRARY  
AND ARCHIVES CENTER  
UNIVERSITY OF CALIFORNIA LIBRARY  
AND ARCHIVES CENTER  
UNIVERSITY OF CALIFORNIA LIBRARY  
AND ARCHIVES CENTER

## LEMBAR PERSETUJUAN

### PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM KONSINYASI PADA CV.MITRA MANDIRI PASURUAN

#### SKRIPSI

Disusun dan diajukan untuk melengkapi dan memenuhi persyaratan guna  
mencapai gelar Sarjana Teknik

**Disusun oleh :**

**Donny Prasetyo**

**NIM. 0612524**

Mengetahui,  
Ketua Program Studi Teknik Elektro S-1

Diperiksa dan Disetujui,  
Dosen Pembimbing



**Ir. Yusuf Ismail Nakhoda, MT**  
NIP. Y.1018800189

**Joseph Dedy Irawan, ST.MT**  
NIP.19740416 200501 1 002

**JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG**

## **ABSTRAK**

### **PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM KONSINYASI PADA CV. MITRA MANDIRI PASURUAN**

**Donny Prasetyo, NIM 0612524**

**Dosen Pembimbing : Joseph Dedy Irawan, ST. MT**

Sistem penjualan dengan sistem konsinyasi yang sudah ada di CV.Mitra Mandiri Pasuruan saat ini adalah dengan menggunakan sistem manual. Karena adanya beberapa kekurangan dari sistem yang sudah ada, maka diharapkan adanya suatu sistem baru yang dapat mendukung dan memperbaiki sistem yang ada. Teknologi informasi hadir untuk menjawab tantangan zaman saat ini, dimana dituntut keefisienan dan ketepatan dalam penyajian informasi. Sistem informasi ini diintegrasikan kedalam suatu perangkat komputer. mengingat komputer pada zaman ini bukan merupakan sesuatu yang baru, dan bahkan hampir setiap orang dapat mengoperasikanya. Tujuan tugas akhir ini adalah untuk menerapkan ilmu dan teknologi informasi pada dunia kerja nyata pada umumnya, dan khususnya pada CV.Mitra Mandiri Pasuruan dengan menggunakan aplikasi database yang berbasis windows. Dan diharapkan aplikasi yang dibuat dapat membantu dan mendukung sistem penjualan yang sudah ada. Hasil dari tugas akhir ini adalah suatu aplikasi database yang digunakan untuk melakukan penjualan dengan sistem piutang atau konsinyasi, pembelian, pemeliharaan stok barang, dan pembuatan laporan.

**Kata kunci :** sistem penjualan kosinyasi, sistem informasi.

## KATA PENGANTAR

Dengan mengucapkan syukur kehadiran Tuhan Yang Maha Esa yang dengan segala Kasih dan Anugerah-Nya, telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul: Pengembangan Sistem Informasi Penjualan Dengan Sistem Konsinyasi Pada CV. Mitra Mandiri Pasuruan.

Pembuatan skripsi ini dibuat untuk memenuhi salah satu syarat dalam memperoleh gelar sarjana teknik di Institut Teknologi Nasional Malang. Ucapan terima kasih yang sebesar-besarnya kami sampaikan pada:

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku rektor ITN Malang
2. Bapak Ir. Sidik Noertjahjono, MT selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. Yusuf Ismail Nakhoda, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
4. Bapak Joseph Dedy Irawan, ST, MT selaku Dosen Pembimbing
5. Bapak dan ibuku tercinta yang selalu memberikan semangat untuk selalu meraih cita-cita dengan ilmu dan amal soleh.
6. Teman – teman Elite FM yang memberikan support dan nasihat untuk segera menyelesaikan skripsi
7. Dan semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan. Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, Maret 2011

Penulis

## DAFTAR ISI

Lembar Persetujuan.....	i
Abstrak .....	ii
Kata Pengantar .....	iii
Daftar Isi .....	iv
Daftar Gambar .....	vii
Daftar Tabel .....	x
<b>BAB I : PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Tujuan .....	2
1.4 Batasan Masalah .....	3
1.5 Metodologi Penelitian.....	3
1.5.1 Studi literatur .....	3
1.5.2 Pengumpulan Data.....	3
1.5.3 Perencanaan Aplikasi.....	3
1.5.4 Pembuatan Aplikasi.....	4
1.5.5 Pengujian Aplikasi.....	4
1.5.6 Kesimpulan.....	4
1.6 Sistematika Penulisan.....	4
<b>BAB II : DASAR TEORI .....</b>	<b>6</b>
2.1 Teori Konsinyasi.....	6
2.1.1 Pengertian Konsinyasi .....	6
2.1.2 Perbedaan Perlakuan Akuntansi Penjualan Reguler dengan penjualan konsinyasi .....	6
2.1.3 Hubungan Konsinyor dan konsinyi dalam konsinyasi.	8
2.2 Borland Delphi 7.0 .....	10
2.3 Appserv.....	11
2.4 Php MyAdmin .....	11
2.5 MySql .....	12
2.6 Database Managemen Sistem.....	13
2.7 Entity Relationship Diagram(ERD).....	14
2.8 Data Flow Diagram(DFD).....	15

<b>BAB III</b>	<b>: PERANCANGAN DAN DESAIN SISTEM.....</b>	<b>19</b>
3.1	Perancangan Basisdata.....	19
3.1.1	Data Flow Diagram .....	19
3.1.2	Relasi Antar Tabel .....	23
3.2	Perancangan Struktur Data .....	27
3.3	Desain Antarmuka Aplikasi .....	32
3.3.1	Desain Form Utama.....	32
3.3.2	Desain Menu Aplikasi Form Utama.....	33
3.3.3	Desain Form Entri Data Barang .....	32
3.3.4	Desain Form Entri Data Supplier .....	34
3.3.5	Desain Form Entri Data Customer .....	34
3.3.6	Desain Form Entri Data Pembelian Barang.....	35
3.3.7	Desain Form Entri Data Retur Pembelian .....	36
3.3.8	Desain Form Entri Data Penjualan Barang.....	36
3.3.9	Desain Form Entri Data Retur Penjualan .....	37
3.3.10	Desain Form Pembayaran Piutang .....	38
3.3.11	Desain Laporan.....	38
<b>BAB IV</b>	<b>: IMPLEMENTASI DAN PENGUJIAN .....</b>	<b>42</b>
4.1	Implementasi Sistem .....	42
4.1.1	Koneksi Basis Data.....	43
4.1.2	Pengoprasian Basis data .....	43
4.2	Pengujian Hasil .....	43
4.2.1.	Form Login.....	43
4.2.2	Form Menu Utama.....	44
4.2.3	Form Ganti Password.....	44
4.2.4	Form Tambah User .....	45
4.2.5	Form Data Customer .....	46
4.2.6	Form Data Supplier .....	48
4.2.7	Form Data Barang .....	49
4.2.8	Form Transaksi Pembelian.....	51
4.2.9	Form Transaksi Penjualan .....	52
4.2.10	Form Laporan Pembelian .....	55
4.2.11	Form Laporan Penjualan .....	56
4.2.12	Form Retur Pembelian.....	57

4.2.13	Form Retur Penjualan.....	60
4.2.14	Form Pembayaran Piutang.....	62
BAB V	: PENUTUP .....	66
5.1	Kesimpulan .....	66
5.2	Saran .....	66
DAFTAR PUSTAKA	.....	67
LAMPIRAN	.....	68



## DAFTAR GAMBAR

Gambar 2.1	External Entity .....	16
Gambar 2.2	Alur Data .....	16
Gambar 2.3	Proses .....	17
Gambar 2.4	Penyimpanan Data.....	17
Gambar 3.1	DFD Sistem Penjualan Konsinyasi Level 0 .....	20
Gambar 3.2	DFD Sistem Penjualan Konsinyasi Level 1 .....	21
Gambar 3.3	DFD Level 2 Proses Penjualan.....	22
Gambar 3.4	DFD Level 2 Proses Pembelian.....	22
Gambar 3.5	CDM .....	24
Gambar 3.6	PDM .....	25
Gambar 3.7	Desain Form Utama .....	32
Gambar 3.8	Desain Menu Aplikasi Form Utama .....	33
Gambar 3.9	Desain Form Entri Data Barang .....	33
Gambar 3.10	Desain Form Entri Data Supplier .....	34
Gambar 3.11	Desain Form Entri Data Costomer .....	34
Gambar 3.12	Desain Form Entri Data Pembelian Barang.....	35
Gambar 3.13	Desain Form Entri Data Retur Pembelian.....	36
Gambar 3.14	Desain Form Entri Data Penjualan .....	36
Gambar 3.15	Desain Form Entri Data Retur Penjualan .....	37
Gambar 3.16	Desain Form Pembayaran Piutang .....	38
Gambar 3.17	Desain Laporan Data Barang.....	38
Gambar 3.18	Desain Laporan Data Supplier.....	39
Gambar 3.19	Desain Laporan Data Customer.....	39
Gambar 3.20	Desain Laporan Data Pembelian .....	39
Gambar 3.21	Desain Laporan Data Retur Pembelian .....	40
Gambar 3.22	Desain Laporan Data Penjualan .....	40
Gambar 3.23	Desain Laporan Data Retur Penjualan .....	41
Gambar 3.24	Desain Laporan Data Pembayaran piutang .....	41
Gambar 4.1	Form Login Sebelum Entri Data .....	43
Gambar 4.2	Form Login Setelah Entri Data.....	44
Gambar 4.3	Menu Utama .....	44

Gambar 4.4	Form Ganti Password Sebelum Entri Data.....	45
Gambar 4.5	Form Ganti Password Sesudah Entri Data .....	45
Gambar 4.6	Form Tambah User Sebelum Entri Data .....	46
Gambar 4.7	Form Tambah User Sesudah Entri Data.....	46
Gambar 4.8	Tampilan Form Data Customer Sebelum Entri Data.....	47
Gambar 4.9	Tampilan Form Data Customer Sesudah Entri Data .....	47
Gambar 4.10	Tampilan Form Tabel Data Customer .....	48
Gambar 4.11	Tampilan Form Data Supplier Sebelum Entri Data .....	48
Gambar 4.12	Tampilan Form Data Supplier Sesudah Entri Data .....	49
Gambar 4.13	Tampilan Form Tabel Data Supplier .....	49
Gambar 4.14	Tampilan Form Data Barang Sebelum Entri Data.....	50
Gambar 4.15	Tampilan Form Data Barang Sesudah Entri Data .....	50
Gambar 4.16	Tampilan Form Tabel Data Barang .....	51
Gambar 4.17	Tampilan Form Transaksi Pembelian Sebelum Entri Data .....	51
Gambar 4.18	Tampilan Form Transaksi Pembelian Setelah Entri Data .....	52
Gambar 4.19	Tampilan Tabel Transaksi Pembelian .....	52
Gambar 4.20	Tampilan Form Transaksi Penjualan Sebelum Entri Data.....	53
Gambar 4.21	Tampilan Form Transaksi Penjualan Setelah Entri Data.....	53
Gambar 4.22	Tampilan Daftar Customer Yang Melakukan Transaksi Penjualan .....	54
Gambar 4.23	Tampilan Laporan Faktur Penjualan Percustomer.....	54
Gambar 4.24	Tampilan Daftar Barang Yang Telah Dijual.....	55
Gambar 4.25	Tampilan Form Laporan Pembelian .....	55
Gambar 4.26	Tampilan Laporan Pembelian.....	56
Gambar 4.27	Tampilan Form Laporan Penjualan .....	56
Gambar 4.28	Tampilan Laporan Penjualan.....	57
Gambar 4.29	Tampilan Form Retur Pembelian Sebelum Entri Data.....	57
Gambar 4.30	Tampilan Form Retur Pembelian Setelah Entri Data .....	58
Gambar 4.31	Tampilan Form Daftar Retur Pembelian .....	58
Gambar 4.32	Tampilan Laporan Daftar Produsen Untuk Barang Yang Diretur .....	59
Gambar 4.33	Tampilan Laporan Daftar Barang Yang Diretur.....	59
Gambar 4.34	Tampilan Form Daftar Retur Pembelian Barang Percustomer .....	60
Gambar 4.35	Tampilan Form Retur Penjualan Sebelum Entri Data .....	60

Gambar 4.36	Tampilan Form Retur Penjualan Setelah Entri Data .....	61
Gambar 4.37	Tampilan Form Daftar Customer Yang Melakukan Retur Penjualan .....	61
Gambar 4.38	Tampilan Laporan Retur Penjualan Barang.....	62
Gambar 4.39	Tampilan Form Daftar Retur Penjualan Barang Percustomer.....	62
Gambar 4.40	Tampilan Form Daftar Piutang Semua Customer.....	63
Gambar 4.41	Tampilan Form Daftar Piutang Percustomer .....	63
Gambar 4.42	Tampilan Form Data Pembayaran Piutang Percustomer.....	64
Gambar 4.43	Tampilan Form Jatuh Tempo Pembayaran Piutang Customer .....	64
Gambar 4.44	Tampilan Form Retur Penjualan Setelah Entri Data .....	65

## DAFTAR TABEL

Gambar 3.2.1	Tabel Barang.....	27
Gambar 3.2.2	Tabel Customer.....	27
Gambar 3.2.3	Tabel Produsen .....	28
Gambar 3.2.4	Tabel Master Beli .....	28
Gambar 3.2.5	Tabel Detail Beli.....	29
Gambar 3.2.6	Tabel Master Jual.....	29
Gambar 3.2.7	Tabel Detail Jual.....	30
Gambar 3.2.8	Tabel Master Retur Beli.....	30
Gambar 3.2.9	Tabel Detail Retur Beli.....	31
Gambar 3.2.10	Tabel Retur Jual.....	31
Gambar 3.2.11	Tabel Detail Retur Jual.....	31
Gambar 3.2.12	Tabel User Name.....	32
Gambar 4.1	Spesifikasi Perlengkapan Implementasi.....	42

# **BAB I**

## **PENDAHULUAN**

### **1.1 LATAR BELAKANG**

CV. Mitra Mandiri Pasuruan merupakan distributor makanan ringan, disana masih menggunakan sistem penjualan konsinyasi dengan sistem manual. Menurut penelitian yang saya kerjakan disana, dengan sistem ini untuk pembuatan laporan penjualan dan laporan pembelian harus melakukan pencatatan ulang pada buku besar, karyawan harus melakukan perhitungan lagi apabila penambahan atau pengurangan stok barang, untuk mengetahui jumlah pembayaran transaksi penjualan dan transaksi pembelian masih melakukan perhitungan lagi, karyawan harus melakukan pengecekan lagi pada buku besar apabila ingin mengetahui jatuh tempo pembayaran piutang.

Dari permasalahan diatas maka diperlukan sebuah sistem baru yang bisa membantu karyawan dalam menyelesaikan semua pekerjaan dengan cepat dan efisien. Dengan adanya permasalahan ini maka saya membuat sebuah sistem baru yaitu sistem informasi penjualan dengan sistem konsinyasi.

Pada sistem informasi penjualan dengan sistem konsinyasi ini tidak perlu melakukan pencatatan ulang jika ingin membuat laporan penjualan dan laporan pembelian karena setiap melakukan order barang maka secara otomatis sistem akan membuat laporan, penambahan dan pengurangan stok barang tidak perlu melakukan perhitungan ulang karena apabila ada pembelian barang secara otomatis sistem akan menambahkan pada stok barang demikian juga apabila ada transaksi penjualan maka secara otomatis sistem akan melakukan pengurangan stok barang, dan untuk jumlah pembayaran tidak perlu melakukan perhitungan lagi karena dengan sistem ini jumlah pembayaran akan dijumlah secara otomatis.

Dilihat dari latar belakang tersebut penulis mengambil kesimpulan bagaimana memberikan yang terbaik bagi CV.Mitra Mandiri Pasuruan dalam pembuatan laporan pembelian dan laporan penjualan, perhitungan stok barang, dan transaksi pembayaran maka saya perlu mengadakan penelitian dengan judul Pengembangan

Sistem Informasi Penjualan Dengan Sistem Konsinyasi Pada CV.Mitra Mandiri Pasuruan.

## 1.2 RUMUSAN MASALAH

Berdasarkan kebutuhan yang ada maka akan dirancang suatu aplikasi Database berupa sistem penjualan dengan sistem piutang atau Konsinyasi pada CV. Mitra Mandiri Pasuruan. Yang secara garis besar meliputi :

- Pendataan stok barang yang secara otomatis ditambahkan pada saat melakukan transaksi pembelian dari *supplier*.
- Pendataan dari *supplier* dan pelanggan beserta entity dan klasifikasi yang ada.
- Pencatatan transaksi penjualan yang dalam memakai sistem piutang atau konsinyasi.
- Pencatatan transaksi *retur* barang dari pelanggan.
- Disertai juga laporan – laporan penunjang seperti :
  - Laporan Data Utama, meliputi:
    - a. Laporan data barang
    - b. Laporan data customer
    - c. Laporan data supplier
  - Laporan pembelian, meliputi :
    - a. Laporan pembelian barang
    - b. Laporan Master beli
    - c. Laporan retur pembelian
  - Laporan penjualan, meliputi:
    - a. Laporan penjualan barang
    - b. Laporan master jual
    - c. Laporan retur penjualan
  - Laporan pembayaran piutang

Sehingga dengan aplikasi ini, perusahaan dapat berjalan dengan *optimal*.

## 1.3 TUJUAN

Tujuan dari dilaksanakannya Skripsi ini adalah menghasilkan aplikasi sistem informasi penjualan dengan sistem konsinyasi yang diharapkan dapat membantu

CV.Mitra Mandiri Pasuruan untuk mengelolah data transaksi sehingga dapat dihasilkan proses kerja yang lebih baik dan efisien.

#### **1.4 BATASAN MASALAH**

Agar permasalahan mengarah sesuai dengan tujuan maka pembahasan dibatasi oleh hal – hal sebagai berikut :

1. Pengembangan aplikasi ini menggunakan Borland Delphi 7 dan databasenya menggunakan MySQL.
2. Implementasi Sistem Informasi Penjualan Konsinyasi ini adalah pada CV. Mitra Mandiri Pasuruan.
3. Hanya membahas sistem informasi penjualan konsinyasi.
4. Transaksi penjualan dan pembelian barang secara konsinyasi.
5. Pembayaran dilaksanakan setelah jatuh tempo pembayaran.
6. Penggunaan aplikasi ini untuk kalangan tertentu yang memiliki hak akses terhadap program.

#### **1.5 METODE PENELITIAN**

##### **1.5.1 Studi Literatur, meliputi :**

- Mengetahui teori dasar dari sistem informasi manajemen pada sistem penjualan dengan sistem konsinyasi atau piutang, beserta segala kebutuhannya.
- Mempelajari dan mendalami konsep dasar pembuatan sistem informasi manajemen pada bidang perdagangan pada umumnya.

##### **1.5.2 Pengumpulan Data**

Pengumpulan data dilakukan dengan cara bertanya dan pencatatan secara langsung semua data dan keterangan dari perusahaan serta pengambilan sampel data yang selanjutnya mempengaruhi desain *interface* dan penyusunan laporan pada aplikasi tersebut.

##### **1.5.3 Perencanaan Aplikasi**

- Merencanakan dan mendesain DFD (*Data Flow Diagram*) dari aplikasi tersebut.

- Merencanakan dan mendesain ERD (*Entity Relationship Diagram*) serta Database yang digunakan untuk aplikasi ini.
- Merencanakan fasilitas – fasilitas atau menu – menu yang ada pada aplikasi.

#### **1.5.4 Pembuatan Aplikasi**

Membuat suatu program aplikasi untuk menggantikan sistem manual yang telah ada yang meliputi :

- Sistem pembelian barang dari *supplier*.
- Sistem penjualan dengan piutang atau konsinyasi.
- Sistem stok barang.
- Sistem *retur* barang.
- Pembuatan laporan – laporan.

#### **1.5.5 Pengujian Aplikasi**

Menguji aplikasi sistem informasi manajemen penjualan dengan sistem konsinyasi dan database yang digunakan. Program yang dihasilkan mampu melakukan pengolahan dan penyimpanan data secara tepat dan baik serta memberikan hasil atau *output* yang diharapkan.

#### **1.5.6 Kesimpulan**

Mengambil kesimpulan berdasarkan pada hasil pengujian aplikasi yang dilakukan pada seluruh fasilitas dan menu yang terdapat pada aplikasi tersebut.

### **1.6. SISTEMATIKA PENULISAN**

#### **BAB I : PENDAHULUAN**

Bab ini berisi tentang latar belakang masalah, perumusan masalah, tujuan skripsi, metode penelitian yang digunakan untuk pembuatan skripsi dan sistematika penulisan.

#### **BAB II : DASAR TEORI**

Bab ini membahas tentang teori – teori pendukung dan prinsip – prinsip yang mendasari perencanaan dan pembuatan sistem.



### **BAB III : PERENCANAAN DAN DESAIN SISTEM**

Bab ini membahas tentang perencanaan dan pembuatan sistem secara keseluruhan.

### **BAB IV : IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab ini membahas tentang cara pengujian dan hasil pengujian yang dilakukan pada sistem.

### **BAB V : PENUTUP**

Bab ini membahas tentang kesimpulan yang didapat dari keseluruhan kegiatan perencanaan dan pembuatan sistem disertai dengan pengujian yang didapatkan, termasuk saran dan kritik untuk kemungkinan pengembangan sistem dilain waktu.

## **BAB II**

### **DASAR TEORI**

#### **2.1 TEORI KONSINYASI**

##### **2.1.1 Pengertian Konsinyasi**

Penjualan konsinyasi disebut juga dengan penjualan titipan, pihak yang menyarankan barang (pemilik) disebut consignor (konsinyor) atau pengamat, sedang pihak yang menerima titipan barang tersebut disebut konsinyi, komisioner. Adapun pengertian penjualan menurut Hadori Yunus Harnanto adalah: “Konsinyasi merupakan suatu perjanjian dimana pihak yang memiliki barang menyerahkan sejumlah barang kepada pihak tertentu untuk dijualkan dengan memberikan komisi”

Pada dasarnya semua penjualan konsinyasi tersebut adalah:

- Unsur perjanjian
- Unsur pemilik barang
- Unsur pihak yang dititipi barang
- Unsur barang yang dititipkan
- Unsur penjualan
- Unsur komisi

Mengabaikan salah satu unsur tersebut akan membuat transaksi tidak dapat disebut penjualan konsinyasi, oleh karena itu seluruh unsur tersebut harus ada pada saat penjualan konsinyasi.

##### **2.1.2 Perbedaan Perlakuan Akuntansi Penjualan Reguler dan Penjualan Konsinyasi**

Dalam pengertian penjualan reguler menurut M. Ichwan dan Arifin adalah:

“Peningkatan jumlah aktiva atau penurunan jumlah utang suatu badan usaha yang timbul dalam penyerahan barang/jasa/aktiva usaha lainnya dalam suatu periode”

Penjualan konsinyasi mempunyai perbedaan yang khusus dengan penjualan reguler. Menurut Hadori Yunus Harnanto memberikan kriteria yang merupakan perbedaan perlakuan akuntansi konsinyasi dengan transaksi penjualan reguler. Keempat kriteria itu adalah sebagai berikut:

1. Barang-barang masih menjadi hak milik konsinyor dan harus dilaporkan sebagai persediaan konsinyor boleh mengakui barang-barang konsinyasi sebagai persediaannya.
2. Pendapatan diakui konsinyor pada saat barang-barang konsinyasi dapat dijual pada pihak ke-3.
3. Pihak pengamat (consignor) sebagai pemilik tetap bertanggung jawab sepenuhnya terhadap semua biaya yang berhubungan dengan barang-barang konsinyasi. Sejak pengiriman barang sampai dengan saat komisioner berhasil menjual barang kepada pihak ke-3. Kecuali jika ditentukan lain dalam perjanjian diantara kedua pihak yang bersangkutan.
4. Komisioner dalam batas kemampuannya mempunyai kewajiban menjaga keamanan dan keselamatan barang-barang yang diterimanya. Oleh karena itu administrasi yang tertib harus diselenggarakan sampai dengan saat ia berhasil menjual barang-barang tersebut kepada pihak ketiga.

Jadi perbedaan prinsip antara transaksi penjualan reguler dengan penjualan konsinyasi adalah:

1) Perpindahan hak milik atas barang yang bersangkutan

Dalam transaksi penjualan reguler hak milik barang berpindah kepada pembeli pada saat barang diserahkan, kemudian keadaan demikian dipakai sebagai dasar pengakuan terhadap timbulnya pendapatan. Sedangkan penjualan pada penjualan konsinyasi tidak berarti adanya penyerahan hak milik atas barang yang bersangkutan.

2) Pengakuan pendapatan

perbedaan pengakuan pendapatan antara penjualan reguler dan penjualan konsinyasi akan berdampak pada laporan rugi laba. Menurut Ikatan Akuntansi Indonesia dalam bukunya Menurut Ikatan

Akuntansi Indonesia dalam bukunya yang berjudul Standar Akuntansi Keuangan Pengertian pendapatan adalah: “Pendapatan adalah arus masuk bruto dari manfaat ekonomi yang timbul dari aktivitas normal perusahaan selama periode bila arus masuk itu mengakibatkan kenaikan ekuitas yang tidak berasal dari kontribusi modal”

Theodoras M. Tuana Kotta menyebutkan kriteria yang lebih tepat bagi pengakuan pendapatan adalah:

- a. Adanya bukti yang kuat bahwa pembeli mempunyai maksud membeli dan penjual bermaksud menjual.
- b. Penentuan mengenai barang tertentu yang akan dijual dan sudah dalam keadaan siap untuk dijual.
- c. Perjanjian antara pembeli dan penjual mengenai barang jual formula untuk mencapai harga jual.

Jadi pendapatan suatu perusahaan dapat diakui oleh perusahaan apabila memenuhi kriteria diatas.

### **2.1.3 Hubungan Konsinyor dan Konsinyi dalam Konsinyasi**

Hubungan antara pihak konsinyor dan pihak konsinyi menyangkut antara pihak pemilik dan agen penjual.

Undang-undang keagenan mengatur penetapan perjanjian atau ketentuan-ketentuan yang biasanya mencakup hal-hal sebagai berikut:

- Komisi penjualan atau laba penjualan yang harus diberikan kepada pihak konsinyi.
- Syarat pembayaran dan penyerahan barang.
- Pemeliharaan dan penyimpanan serta penangganan persediaan barang konsinyasi dan hasil penjualannya.
- Syarat kredit yang harus diberikan untuk pihak konsinyi pada para pelanggan.
- Penyelesaian keuangan oleh pihak konsinyi kepada konsinyor dan bentuk serta jangka waktu (periode).

- Laporan yang harus dikirim oleh konsinyi.

Selain beberapa ketentuan yang terdapat dalam perjanjian di antara kedua pihak tersebut, undang-undang keagenan juga mengatur ketetapan hak dan kewajiban kedua pihak, yaitu:

- 1) Hak-hak komisioner
  - a. komisioner berhak untuk mendapatkan komisi dan pengantian biaya yang dikeluarkan sesuai dengan jumlah yang diatur dalam perjanjian kedua pihak.
  - b. Pada batas tertentu biasanya komisioner berhak memberikan jaminan terhadap barang-barang yang dijual.
  - c. Untuk menjamin pemasaran barang komisioner berhak memberikan syarat pembayaran kepada pelanggan, meskipun pengamat memberikan batasan – batasan yang dinyatakan dalam perjanjian.
- 2) Kewajiban komisioner
  - a. melindungi keamanan dan keselamatan barang yang dikirimkannya dari pihak pengamat (konsinyor).
  - b. Mematuhi dan berusaha menjualsemaksimal mungkin barang-barang milik pengamat sesuai dengan ketentuan yang diatur dalam perjanjian.
  - c. Mengelolah secara terpisah baik dari segi fisik maupun administrasi terhadap barang milik pengamat, sehingga identitas barang tersebut tetap dapat diketahui setiap saat.
  - d. Membuat laporan periodik tentang barang yang diterima, barang yang berhasil dijual dan barang yang masih dalam persediaan serta mengadakan penyelesaian keuangan seperti dinyatakan dalam perjanjian.

Adapun keuntungan-keuntungan dari penjualan konsinyasi dari kedua pihak sebagai berikut:

- 1) Bagi pihak konsinyor
  - a. untuk memperluas daerah pemasaran, terutama jika :

- barang itu merupakan barang diintroduksi dan permintaan akan produk tidak diketahui.
  - Penjualan tahun lalu tidak menguntungkan bagi agen penjualan.
  - Fluktuasi harga atau produk tidak tahan lama.
- b. konsinyor dapat memperoleh spesialis penjualan.  
Imbalan ini untuk jasa seperti ini hanya komisi, yang dapat persentase harga jual atau dapat juga berupa jumlah yang tetap untuk setiap unit yang terjual.
- 2) Bagi pihak konsinyi
- a. Pihak konsinyin terlepas dari resiko kegagalan dari barang itu atau resiko penjualan dengan rugi, factor ini sangat penting bagi produk baru.
  - b. Resiko kerusakan fisik dan fluktuasi harga dapat dihindari.
  - c. Kebutuhan modal kerja berkurang, karena penerapan harga pokok persediaan barang konsinyasi dilakukan oleh pihak konsinyor.

Karena keuntungan yang diperoleh sangat bermanfaat bagi kedua pihak maka keuntungan tersebut dijadikan alasan untuk mengadakan atau mengembangkan kebijaksanaan penjualan konsinyasi.

## **2.2 BORLAND DELPHI 7.0**

Borland Delphi 7 adalah paket bahasa pemrograman yang bekerja dalam system operasi Windows, yang mempunyai cakupan kemampuan yang luas dan sangat canggih. Beberapa jenis aplikasi yang dapat kita buat dengan Delphi, termasuk aplikasi untuk mengolah teks, grafik, angka, database dan aplikasi web.

Secara umum kemampuan Delphi adalah menyediakan komponen-komponen dan bahasa pemrograman yang handal, sehingga memungkinkan kita untuk membuat aplikasi sesuai dengan keinginan, dengan tampilan dan kemampuan yang canggih.

Untuk mempermudah pemrogram dalam membuat program aplikasi, Delphi menyediakan fasilitas pemrograman yang sangat lengkap. Fasilitas pemrograman tersebut dibagi dalam dua kelompok yaitu object dan bahasa

pemrograman. Secara ringkas, object adalah suatu komponen yang mempunyai bentuk fisik dan biasanya dapat dilihat (*visual*). Object biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman secara singkat dapat disebut sebagai sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu. Delphi menggunakan struktur bahasa Object Pascal yang sudah sangat dikenal dikalangan pemrograman profesional. Gabungan dari object dan bahasa pemrograman ini disebut sebagai bahasa pemrograman berorientasi object atau Object Oriented Programming (OOP).

Khusus untuk pemrograman database, Delphi menyediakan object yang sangat kuat, canggih dan lengkap, sehingga memudahkan pemrogram dalam merancang, membuat dan menyelesaikan aplikasi database yang diinginkan. Selain itu juga Delphi dapat menangani data dalam berbagai format database, misalnya format MS-Acces, Sybase, Oracle, FoxPro, Informix, DB2 dan lain-lain. Format database yang dianggap asli dari Delphi adalah Paradox dan dBase.

### **2.3 APPSERV**

Appserv adalah sebuah paket program yang terdiri PHP, MySQL, Apache dan phpMyAdmin yang dapat dijadikan web server di komputer lokal. Dengan menggunakan Appserv kita tidak perlu lagi susah-susah melakukan konfigurasi server lokal, karena semuanya sudah diotomatisasikan.

### **2.4 PHPMYADMIN**

PhpMyAdmin adalah suatu program open source yang berbasis web yang di buat menggunakan aplikasi PHP. Program ini digunakan untuk mengakses database MySQL. Program ini mempermudah dan mempersingkat kerja suatu database. Untuk membuat tabel dan field serta dapat menggunakan perintah SQL atau SQL query ataupun dapat memilih langsung pada menu yang tersedia.

### **2.5 MYSQL**

MySQL merupakan Relational Database Management Sistem (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public

Licensi). Di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structure Query Language).

SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu system database (DBMS) dapat diketahui dari cara kerja optimizernya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan dengan database server yang lainnya dalam query data.

Sebagai *database* yang memiliki konsep *database* modern, MySQL memiliki banyak sekali keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh MySQL :

- *Portability*. MySQL dapat berjalan stabil pada berbagai sitem operasi di antaranya adalah seperti Windows, Linux, FreeBSD, Mac OS X server, Solaris, Amiga, HP-UX dan masih banyak lagi.
- *Open Source*. MySQL didistribusikan secara *open source* (gratis), di bawah lisensi GPL.
- *Multiuser*. MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini memungkinkan sebuah *database server* MySQL dapat diakses *client* secara bersamaan.
- *Performance Tuning*. MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
- *Column Types*. MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, year, set* serta *enum*.
- *Command dan Function*. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam *query*.
- *Security*. MySQL memiliki beberapa lapisan sekuritas seperti *level*



*subnetmask*, nama *host*, dan *user* dengan sistem perizinan yang mendetail serta *password* ter-*encripsi*. *Stability dan Limits*. MySQL mampu menangani *database* dalam skala besar, dengan jumlah *records* lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu, batas indeks yang dapat di tampung mencapai 32 indeks pada tiap tabelnya.

- *Connectivity*. MySQL dapat melakukan koneksi dengan *client* menggunakan *protocol* TCP/IP, *Unix socket* (Unix), atau *Named Pipes* (NT).
- *Localisation*. MySQL dapat mendeteksi pesan kesalahan (*error code*) pada *client* dengan menggunakan lebih dari dua puluh bahasa. Meski demikian, bahasa Indonesia belum termasuk di dalamnya.
- *Interface*. MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
- *Client dan Tools*. MySQL dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi *database*, dan pada setiap *tool* yang ada disertai petunjuk *online*.
- *Struktur Tabel*. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan *database* lainnya semacam PostgreSQL ataupun Oracle.

## 2.6 DATABASE MANAGEMEN SISTEM

DBMS merupakan perangkat lunak atau program computer yang dirancang secara khusus untuk memudahkan pengelolaan database. DBMS terdiri dari beberapa basis data dan satu atau lebih program pengelola yang berguna pengolahan data yang ada seperti untuk menambah data , menghapus data, mengambil data dan memperbaharui data – data yang ada.

Basis data sendiri merupakan sekelompok kumpulan data yang saling berelasi yang mana dapat disimpan, diproses, dimanipulasi dan digunakan oleh pihak – pihak yang berkepentingan. Relasi tersebut biasanya ditunjukkan dengan bagian – bagian penting dari tiap – tiap file ( kumpulan data ) yang ada. Untuk mengenal *database* , secara lebih men-detail, perlu diketahui komponen – komponen dasar pembentuk *database*, yaitu :

1. *Entity*

*Entity* adalah suatu bentuk dari orang, tempat, kejadian atau konsep dari suatu informasi yang dijadikan suatu data bagi basis data yang digunakan.

2. *Atribut*

*Atribut* adalah sebutan untuk mewakili suatu *Entity*

3. *Data value.*

*Data value* adalah *data actual* atau informasi yang disimpan dalam tiap – tiap data elemen atau *atribut*.

4. *Record/tuple*

*Record/tuple* adalah kumpulan elemen – elemen yang saling berkaitan yang berguna untuk menginformasikan tentang suatu *Entity* secara lengkap.

5. *File*

*File* adalah sekumpulan record – record sejenis yang mempunyai panjang elemen sama, *atribut* yang sama namun berbeda data value-nya.

## 2.7 ENTITY RELATIONSHIP DIAGRAM (ERD)

Metode ini dipakai pada jumlah atribut yang banyak. Metode menggambarkan hubungan antar *table* yang ada pada *database*. Pertama kali yang dilakukan adalah membagi *database* menjadi beberapa *table* yang disebut object atau *entity*. Masing – masing *entity* ini mempunyai beberapa *atribut*, yang merupakan bagian dari *entity* tersebut. Antar beberapa *entity* tersebut dapat digambarkan suatu hubungan (*relationship*) yang biasanya dipakai dalam ERD. ERD (*Entity Relationship Diagram*) digunakan untuk menggambarkan hubungan antara *Entity*. Untuk menghubungkan satu *entity* satu dengan *entity* yang lain digunakan *entity key*, yaitu suatu *atribut* tertentu atau sekelompok *atribut* tertentu yang bersifat unik, sehingga dapat digunakan untuk membedakan suatu anggota *entity* dengan anggota yang lain pada *entity* yang sama. Selain itu harus ada pula yang disebut *relationship key*, yaitu suatu *relationship* yang digunakan untuk menyatakan suatu hubungan antar *Entity key*.

Macam – macam hubungan antar *Entity* :

➤ *One to one relationship*

Merupakan suatu hubungan dimana satu anggota *entity* mempunyai hubungan dengan satu anggota *entity* pada *entity* yang berbeda. Ada 2 macam hubungan yaitu, *obligatory* dan *non-obligatory*. *Obligatory* adalah semua anggota dari suatu *Entity* harus berpartisipasi atau mempunyai hubungan dengan *entity* yang lain. *Non -obligatory* adalah tidak semua anggota harus mempunyai *entity* yang lain. Contoh satu mahasiswa harus mempunyai satu nomor induk mahasiswa (nrp) dan satu nomor induk mahasiswa (nrp) hanya dimiliki oleh satu orang mahasiswa.

➤ *One To Many Relationship*

Merupakan suatu hubungan antara satu anggota *entity* yang satu dengan beberapa anggota *Entity* pada *Entity* yang berbeda. Hubungan ini juga bisa dibagi menjadi 2 macam yaitu *obligatory* dan *non-obligatory*. Contoh : seorang mahasiswa dapat mempunyai beberapa nomor telepon tetapi setiap nomor telepon hanya dapat dipunyai seorang mahasiswa.

➤ *Many To Many Relationship*

Merupakan suatu hubungan antara beberapa anggota *entity* yang satu dengan beberapa anggota *entity* pada *entity* yang lain. Jadi kedua pihak bisa mempunyai hubungan lebih dari satu dengan beberapa anggota *entity*. Hubungan *entity* juga bisa 2 macam yaitu *obligatory* dan *non-obligatory*. Contoh : satu orang mahasiswa mempunyai mata kuliah yang lebih dari satu dan matakuliah dapat diikuti oleh lebih dari satu orang mahasiswa.

## 2.8 DATA FLOW DIAGRAM (DFD)

Adalah suatu model diagram yang menggambarkan system sebagai jaringan kerja antar fungsi yang berhubungan satu sama lain dengan proses serta penyimpanan dan aliran data. DFD sering digunakan untuk menggambarkan system yang telah ada atau system yang baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik

dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan. Ada 4 komponen dalam model DFD, yaitu (Fatansyah, 1999):

1. Entitas luar (*external entity*)

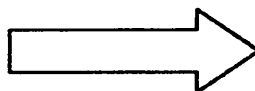
Setiap sistem pasti mempunyai batas sistem yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem kan menerima input dan menghasilkan output kepada lingkungan luarnya. Entitas luar merupakan kesatuan diluar lingkungan sistem yang dapat berupa orang, organisasi atau sistem yang lainnya berada dilingkungan luar yang akan memberikan input atau menerima output dari sistem.



Gambar 2.1 *External Entity*

2. Arus data

Arus data diberi symbol suatu panah. Arus data ini mengalir diantara proses, simpanan dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk system atau hasil dari proses system.



Gambar 2.2 Alur Data

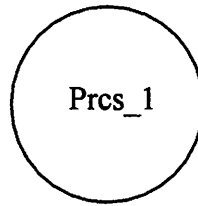
3. Proses

Proses menunjukkan transformasi dari masukan sampai keluaran. Suatu proses terjadi karena adanya arus data yang masuk dan hasil dari proses merupakan arus data yang mengalir.

Beberapa hal yang harus dihindari dalam menggambarkan suatu proses adalah :

- a. Proses yang mempunyai masukan tapi tidak memiliki keluaran, kesalahan ini disebut black hole.
- b. Proses yang menghasilkan keluaran tetapi tidak mendapatkan masukan, kesalahan ini disebut miracle.

- c. Proses yang menghasilkan keluaran lengkap tetapi masukan dengan data terbatas atau masukan yang kurang lengkap, kesalahan ini disebut *gray hole*.



Gambar 2.3 Proses

#### 4. Penyimpanan data (data storage)

Digunakan untuk menyimpan data hasil proses maupun menyediakan data untuk di proses pada waktu lain.



Gambar 2.4 Penyimpanan Data

*Context diagram* adalah DFD yang menunjukkan batas – batas dari sebuah sistem informasi, yaitu *top level view* dari sebuah sistem. Untuk menggambar *context diagram*, hanya dibuat satu proses saja yang menggambarkan keseluruhan proses dari sistem tersebut dan beberapa kesatuan luar disekelilingnya yang berhubungan. Contoh *Context Diagram* untuk DFD diatas adalah sebagai berikut :

- ❖ Dari DFD *level 0*, kadang – kadang ada proses yang dapat dijabarkan lebih *detail* lagi.
- ❖ Dari DFD *level 0* dapat dibuat sejumlah DFD *level 1*, demikian pula dari setiap *level 1* tersebut jika masih dapat dijabarkan lagi menjadi DFD *level 2*, 3 dan seterusnya. Sebuah DFD minimal digambarkan sampai *level 0* saja, namun dalam kenyataan sebuah DFD umumnya pasti terdiri dari beberapa *level*.

Hal – hal yang perlu diperhatikan dalam penyusunan DFD :

- 1) Dalam DFD tidak dibedakan antara data dan informasi, semua dianggap data.
- 2) Nama proses dalam Context Diagram harus sama dengan nama sistem tersebut.
- 3) Setiap DFD harus muat dalam satu halaman.
- 4) Setiap simbol harus diberi nama yang unik, namun harus konstan atau tidak boleh berubah jika berpindah ke level berikutnya.
- 5) Hindari garis yang berpotongan jika mungkin.
- 6) DFD pasti selalu mempunyai output.
- 7) Setiap Proses dalam DFD harus mempunyai input dan output.
- 8) Suatu kesatuan luar hanya boleh mempunyai input dan output tetapi tidak boleh kedua – duanya.
- 9) Setiap simpanan data hanya boleh menerima input dari proses dan juga memberikan output ke proses saja.

## **BAB III**

### **PERENCANAAN DAN DESAIN SISTEM**

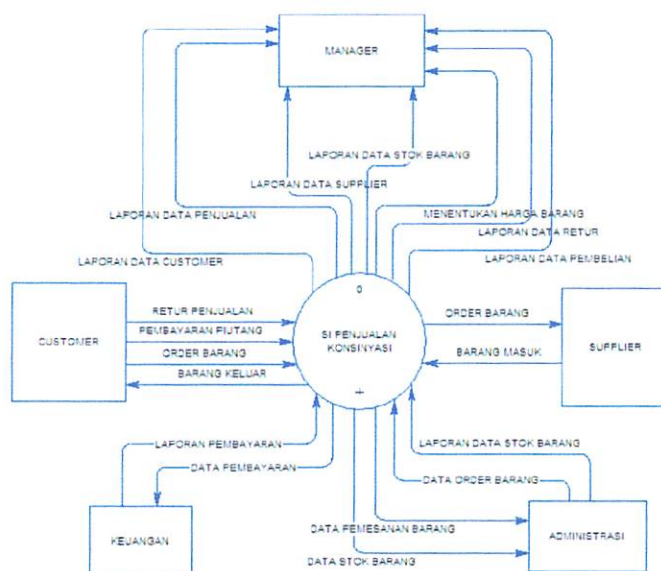
Perancangan program sistem penjualan ini menggunakan sistem informasi berbasis komputer untuk melakukan perhitungan agar lebih mudah dan cepat dalam penggunaannya. Perancangan sistem program dimulai dengan merancang basis data, termasuk didalamnya merancang DFD(*Data Flow Diagram*) dan relasi antar tabel, lalu merancang struktur data setelah itu baru akan dilakukan perancangan program.

#### **3.1 PERANCANGAN BASIS DATA**

Pada tahap perancangan basis data, perancangan sistem lebih diutamakan. Desain basis data lebih menjurus pada perancangan tabel – tabel yang akan dibuat, field – field dari tiap – tiap tabel dan hubungan atau relasi antar tabel yang satu dengan yang lain. Pada tahap ini diperlukan pengetahuan tentang basis data sehingga tabel – tabel yang dirancang lebih efisien dan relasi antar tabel – tabel tersebut lebih sederhana.

##### **3.1.1 Data Flow Diagram (DFD)**

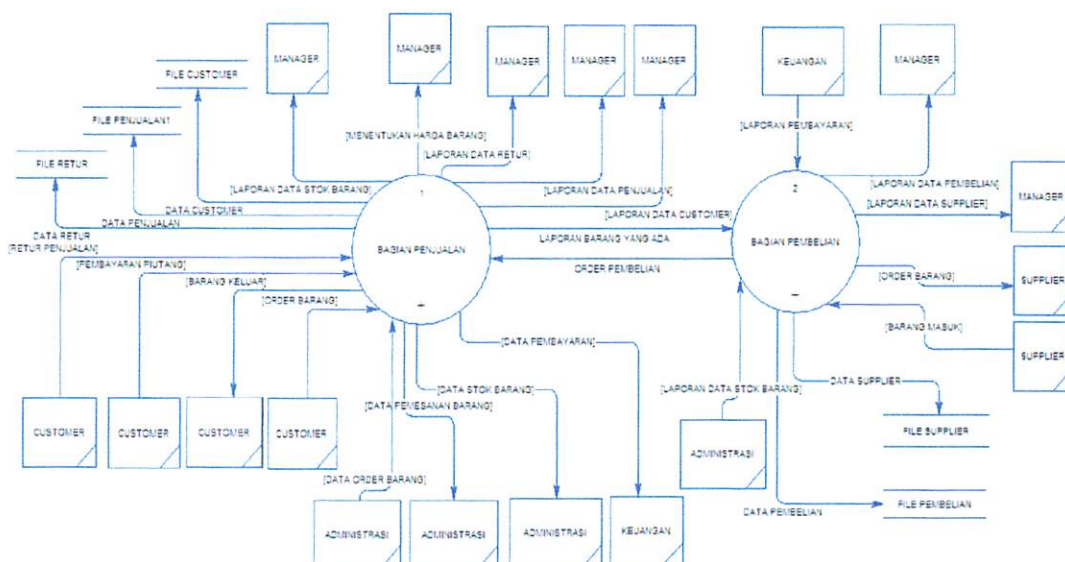
Untuk perancangan arus data dalam sistem penjualan dengan sistem konsinyasi atau piutang ini, digunakan metode Data Flow Diagram (DFD). DFD merupakan alat pengembangan sistem yang terstruktur dan jelas. Gambar DFD system penjualan ini dimulai dari DFD level 0 sampai level 2 dapat dilihat pada gambar.



Gambar 3.1 DFD Sistem Penjualan Konsinyasi level 0

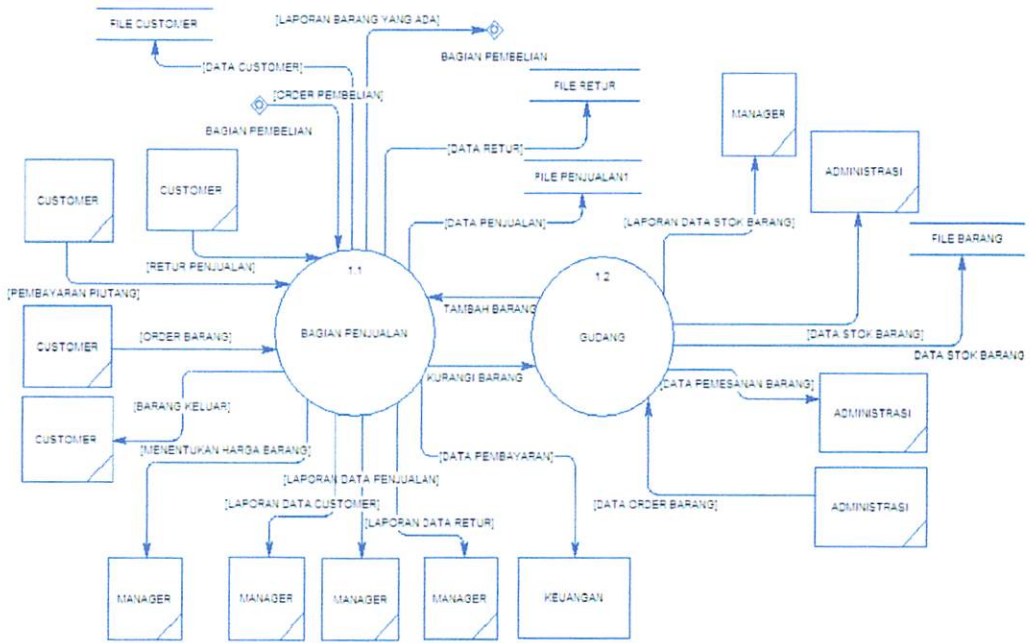
Gambar diatas menunjukan DFD sistem penjualan level 0. Customer pertama kali melakukan order atau pemesanan barang, setelah itu sistem akan melakukan order atau pemesanan barang, setelah itu sistem akan melakukan order atau pemesanan barang ke supplier. Setelah barang dari supplier diterima maka sistem akan membuat nota pembelian. Barang yang telah diterima tadi dikirimkan ke Customer sesuai pesanan yang ada, beserta nota penjualan yang sifatnya piutang. Untuk retur sistem akan membuat nota retur untuk barang yang dikembalikan dan menambah barang yang dikembalikan dan menambah barang tersebut pada stok barang. Bagian keuangan disini melakukan tugas membuat laporan pembayaran setelah menerima data pembayaran, Setelah itu sistem membuat laporan penjualan dan laporan pembelian maupun retur ke manager





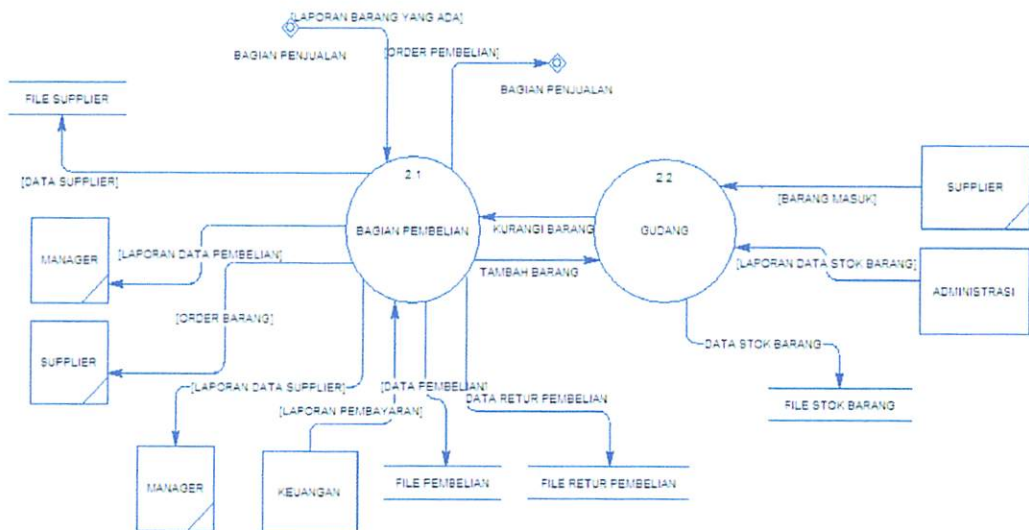
Gambar 3.2 DFD Sistem Penjualan Konsinyasi Level 1

Gambar diatas menunjukkan sistem penjualan level 1. Pada tahap ini sistem dipecah menjadi 2 bagian yaitu bagian pembelian dan penjualan. Dimana bagian penjualan bertugas untuk menerima order dari konsumen, mengirim barang sesuai order atau pesanan ke konsumen, menerima retur barang dari konsumen, membuat nota retur, serta memberikan laporan penjualan dan retur ke manajemen serta penyimpanan data penjualan dan retur. Sedangkan bagian pembelian bertugas untuk melakukan pembelian barang yang dipesan oleh bagian penjual ke supplier, dan melakukan pembayaran ke supplier serta memberikan laporan pembelian ke pihak manajemen dan melakukan penyimpanan data pembelian.



Gambar 3.3. DFD Level 2 Proses Penjualan

Gambar diatas menunjukan DFD sistem penjualan level 2 proses penjualan. Pada tahap ini terdapat penambahan bagian yaitu bagian gudang yang fungsinya untuk melakukan pencatatan dan penambahan stok barang yang dihasilkan dari barang yang diretur atau dikembalikan, serta membuat laporan stok barang ke manager dan melakukan penambahan stok ke file barang.

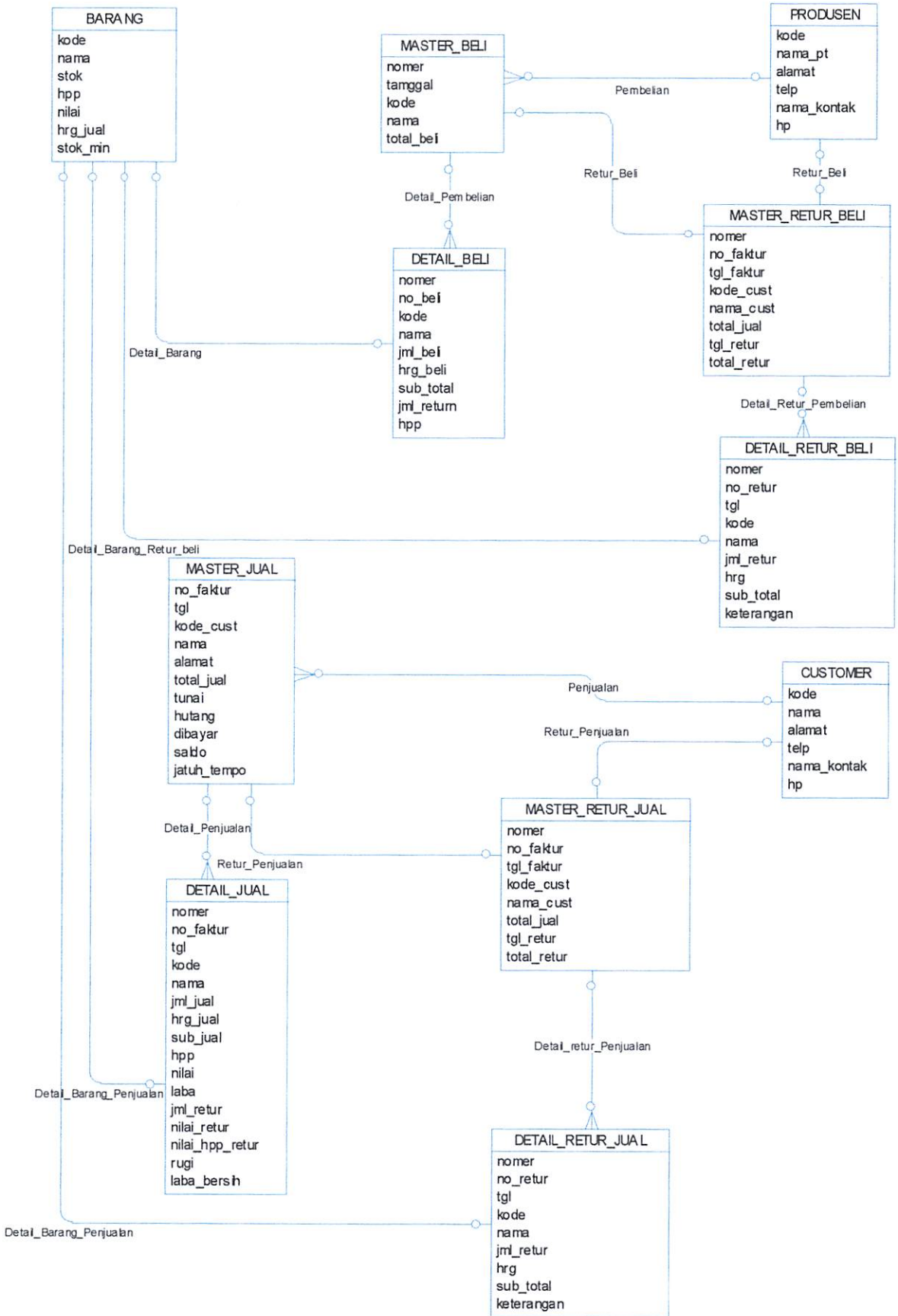


Gambar 3.4 DFD Level 2 Proses Pembelian

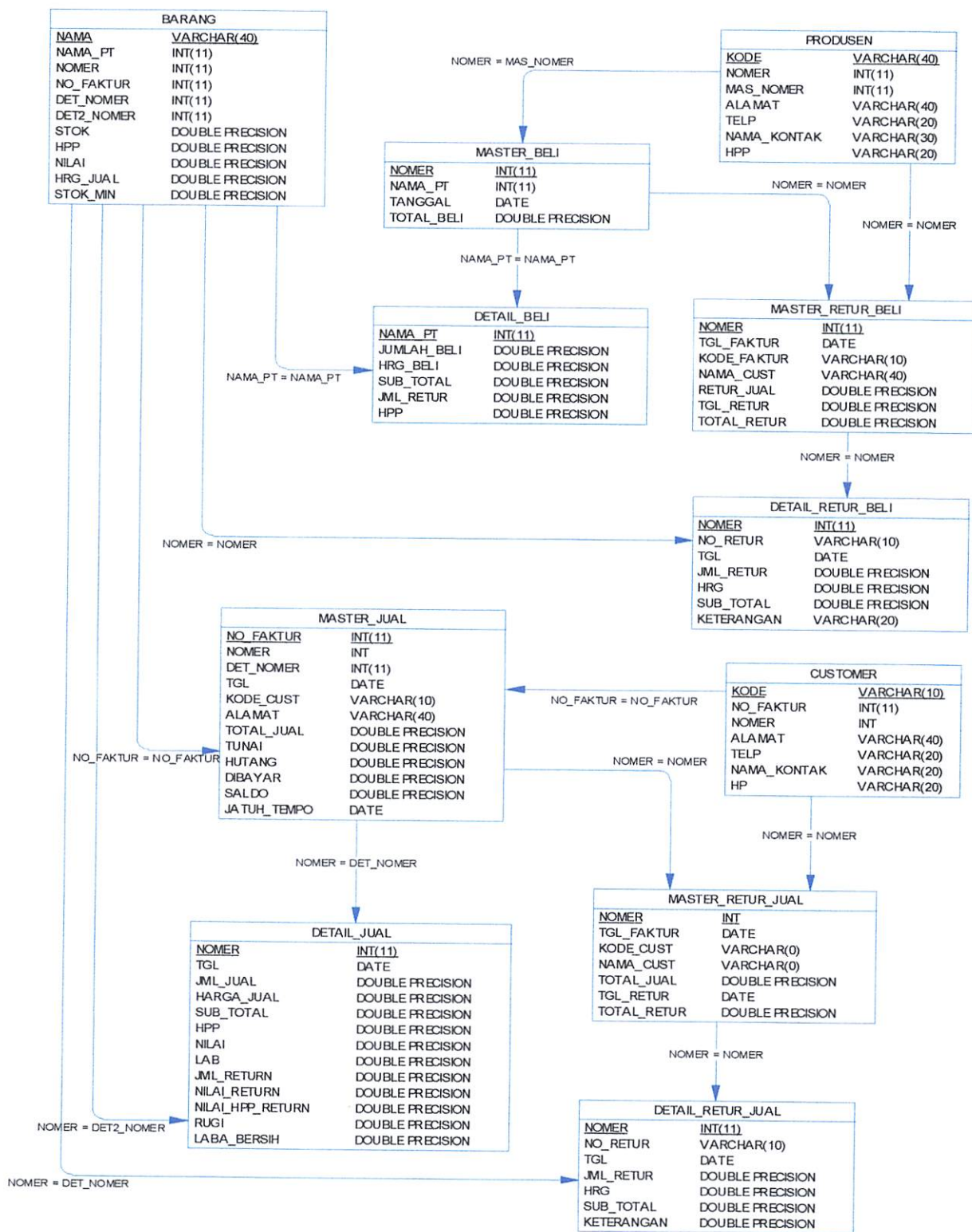
Gambar diatas menunjukkan DFD sistem penjualan level 2 proses penjualan. Pada tahap ini terdapat penambahan bagian yaitu bagian gudang yang fungsinya untuk melakukan pencatatan dan pengurangan stok barang yang dihasilkan dari barang yang diretur atau dikembalikan, serta membuat laporan stok barang ke manager dan melakukan pengurangan stok ke file barang.

### **3.1.2 Relasi Antar Tabel**

Untuk perancangan basis data sistem penjualan ini, digambarkan dalam bentuk *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM). Gambar dari CDM dan PDM sistem penjualan ini dapat dilihat pada gambar dibawah ini.



Gambar 3.5 Conceptual Data Model



Gambar 3.6 Physical Data Model

Dari PDM dan CDM diatas, dapat diperoleh sebanyak 12 tabel yang dibutuhkan. Adapun penjelasan dari masing – masing tabel Database yang dibuat adalah :

- Barang.db (kode, Nama, Stok, Hpp, Nilai, Harga\_jual, Stok\_Min). Berfungsi untuk melakukan penyimpanan data barang.
- Customer.db (kode, Nama, Alamat, telp, nama\_kontak, hp). Menyimpan data – data customer seperti alamat, telepon, dan sebagainya. Yang merupakan informasi penting.
- Produsen.db (kode, Nama\_pt, Alamat, Telp, Nama\_kontak, Hp). menyimpan data supplier seperti nama supplier, alamat dan sebagainya.
- Master\_beli.db (Nomer, Tanggal, kode, nama, total\_beli). Menyimpan data master pembelian.
- Detail\_beli.db (Nomer, No\_mbeli, kode, nama, jml\_beli, hrg\_beli, sub\_total, Jml\_retur, hpp). Menyimpan isi dari detail pembelian.
- Master\_jual.db (no\_faktur, Tgl, kode\_cust, nama, alamat, total\_jual, tunai, hutang, dibayar, saldo, jatuh\_tempo). Menyimpan data master penjualan.
- Detail\_jual.db (Nomer, No\_faktur, Tgl, Kode, Nama, Jml\_jual, Harga\_jual, Sub\_total, Hpp, Nilai, Laba, Jml\_retur, Nilai\_retur, Nilai\_hpp\_retur, Rugi, Laba\_bersih). Menyimpan isi dari item penjualan.
- Master\_retur\_beli.db (nomer, no\_faktur, tgl\_faktur, kode\_cust, nama\_cust, total\_jual, tgl\_retur, total\_retur). Menyimpan data Faktur pembelian dari supplier.
- Detail\_retur\_beli.db(nomer, no\_retur, tgl, kode, nama, jml\_retur, harga, sub\_total, keterangan). Menyimpan isi dari detail retur beli.
- Master\_retur\_jual (nomer, no\_faktur, tgl\_faktur, kode\_cust, nama\_cust, total\_jual, tgl\_retur, total\_retur ). Menyimpan data master retur jual.
- Detail retur jual.db (nomer, no\_retur, tgl, kode, nama, jml\_retur, hrg, sub\_total, keterangan). Menyimpan isi dari detail retur jual.

Sebagai tambahan dibuat satu tabel lagi sebagai keamanan sistem program, yaitu :

- User name.db ( user\_name, password, statuse). Menyimpan data user name.

### 3.2 PERANCANGAN STUKTUR DATA

Rancangan stuktur data pada sistem ini menggunakan *Paradox*, dan rancangan untuk tiap – tiap field adalah :

#### 1. Barang.db

No	Nama	Data Type	Keterangan
1	Kode	Varchar(10)	Kode barang
2	Nama	Varchar(40)	Nama barang
3	Stok	Double	Stok
4	Hpp	Double	Harga pokok pembelian
5	Nilai	Double	Nilai
6	Hrg_jual	Double	Harga jual
7	Stok_min	Double	Stok minimal

Tabel 3.2.1 Tabel Barang

#### 2. Customer.db

No	Nama	Data Type	Keterangan
1	Kode	Varchar(10)	Kode customer
2	Nama	Varchar(40)	Nama Customer
3	Alamat	Varchar(40)	Alamat
4	Telp	Varchar(20)	Telepon
5	Nama_kontak	Varchar(20)	Contak persen
6	Hp	Varchar(20)	Nomor handphone kontak persen

Tabel 3.2.2 Tabel Customer

## 3. Produsen.db

No	Nama	Data Type	Size
1	Kode	Varchar	Kode supplier
2	Nama_pt	Varchar	Nama supplier
3	Alamat	Varchar	Alamat
4	Telp	Varchar	Telephone
5	Nama_kontak	Varchar	Contak persen
6	Hp	Varchar	Nomer hp kontak persen

Tabel 3.2.3 Tabel Produsen

## 4. Master\_Beli.db

No	Nama	Data Type	Keterangan
1	Nomer	Int(11)	Nomor urut
2	Tanggal	Date	Tanggal
3	Kode	Varchar(10)	Kode barang
4	Nama	Varchar(40)	Nama barang
5	Total Beli	Double	Total beli

Tabel 3.2.4 Tabel Master Beli

## 5. Detail\_Beli.db

No	Nama	Data Type	Keterangan
1	Nomer	Int(11)	Nomor urut
2	No_mbeli	Int(11)	Nomor master beli
3	Kode	Varchar(10)	Kode barang
4	Nama	Varchar(40)	Nama barang
5	Jml_beli	Double	Jumlah pembelian
6	Hrg_beli	Double	Harga barang yang dibeli
7	Sub_total	Double	Jumlah total pembelian
8	Jml_return	Double	Jumlah retur



9	Hpp	Double	Harga pokok pembelian
---	-----	--------	-----------------------

Tabel 3.2.5 Tabel Detail Beli

## 6. Master\_Jual.db

No	Nama	Data Type	Size
1	Nomer	Int(11)	Nomorurut
2	Tgl	Date	Tanggal
3	Kode_cust	Varchar(10)	Kode customer
4	Nama	Varchar(40)	Nama customer
5	Alamat	Varchar(40)	Alamat
6	Total_jual	Double	Total penjualan
7	Tunai	Double	Tunai
8	Hutang	Double	Hutang
9	Dibayar	Double	Pembayaran hutang
10	Saldo	Double	Sisa hutang
11	Jatuh_tempo	Date	Tanggal pembayaran hutang

Tabel 3.2.6 Tabel Master Jual

## 7. Detail jual.db

No	Nama	Data Type	Keterangan
1	Nomer	Int(11)	Nomorurut
2	No_faktur	Varchar(10)	Nomor faktur
3	Tgl	Date	Tanggal
4	Kode	Varchar(10)	Kode barang
5	Nama	Char(40)	Nama barang
6	Jml_jual	Double	Jumlah barang yang dijual
7	Harga_jual	Double	Harga barang
8	Sub_total	Double	Jumlah total
9	Hpp	Double	Harga pokok pembelian
10	Nilai	Double	Nilai

11	Laba	Double	Keuntungan
12	Jml_return	Double	Jumlah retur
13	Nilai_return	Double	Nilai retur
14	Nilai_hpp_return	Double	Nilai harga pokok pembelian retur
15	Rugi	Double	Kerugian
16	Laba_bersih	Double	Laba bersih

Tabel 3.2.7 Tabel Detail Jual

## 8. Master\_Retur\_Beli.db

No	Nama	Data type	Size
1	Nomer	Int(11)	Nomorurut
2	No_faktur	Int(11)	Nomor faktur
3	Tgl_faktur	Date	Tanggal faktur
4	Kode_cust	Varchar(10)	Kode customer
5	Nama_cust	Varchar(40)	Nama customer
6	Total_jual	Double	Total jual
7	Tgl_retur	Date	Tanggal retur
8	Total_retur	Double	Total retur

Tabel 3.2.8 Tabel Master Retur Beli

## 9. Detail\_Retur\_Beli.db

No	Nama	Data Type	Keterangan
1	Nomer	Int(11)	Nomorurut
2	No_retur	Varchar(10)	Nomor retur
3	Tgl	Date	Tanggal
4	Kode	Varchar(10)	Kode barang
5	Nama	Varchar(40)	Nama barang
6	Jml_retur	Double	Jumlah retur
7	Hrg	Double	Harga barang retur
8	Sub_total	Double	Jumlah total

9	Keterangan	Varchar(20)	Keterangan
---	------------	-------------	------------

Tabel 3.2.9 Tabel Detail Retur Beli

## 10. Master\_Retur\_Jual.db

No	Nama	Data type	Keterangan
1	Nomer	Int(11)	Nomor urut
2	No_faktur	Int(11)	Nomor faktur
3	Tgl_faktur	Date	Tanggal faktur
4	Kode_cust	Varchar(10)	Kode customer
5	Nama_cust	Varchar(40)	Nama customer
6	Total_jual	Double	Total jual
7	Tgl_retur	Date	Tanggal retur
8	Total_retur	Double	Total retur

Tabel 3.2.10 Tabel Retur Jual

## 11. Detail\_Retur\_Jual.db

No	Nama	Data Type	Size
1	Nomer	Int(11)	Nomor urut
2	No_retur	Varchar(10)	Nomor retur
3	Tgl	Date	Tanggal retur
4	Kode	Varchar(10)	Kode barang
5	Nama	Varchar(40)	Nama barang
6	Jml_retur	Double	Jumlah retur
7	Hrg	Double	Harga
8	Sub_total	Double	Jumlah total
9	Keterangan	Varchar(20)	keterangan

Tabel 3.2.11 Tabel Detail Retur Jual

## 12. User name

No	Nama	Data type	Keterangan
1	User_name	Varchar(20)	Nama pegawai
2	Passwords	Varchar(20)	Passwords
3	Status	Varchar(15)	Pekerjaan / Bagian

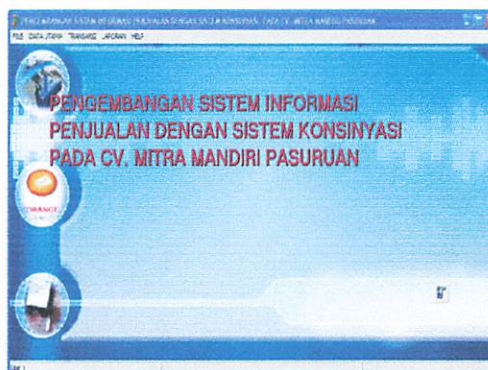
Tabel 3.2.12 Tabel User Name

### 3.3 DESAIN ANTARMUKA APLIKASI

Sesuai dengan spesifikasi sistem di atas, sistem informasi ini diharapkan dapat dengan mudah dipakai oleh operator yang mengoperasikan sistem ini. Untuk itu harus dibuat desain antarmuka yang mudah dipahami dan tidak terlalu rumit.

Ada beberapa desain antarmuka pada aplikasi ini baik halaman untuk admin maupun halaman sistem informasi itu sendiri, yaitu desain halaman utama, entri data dan laporan.

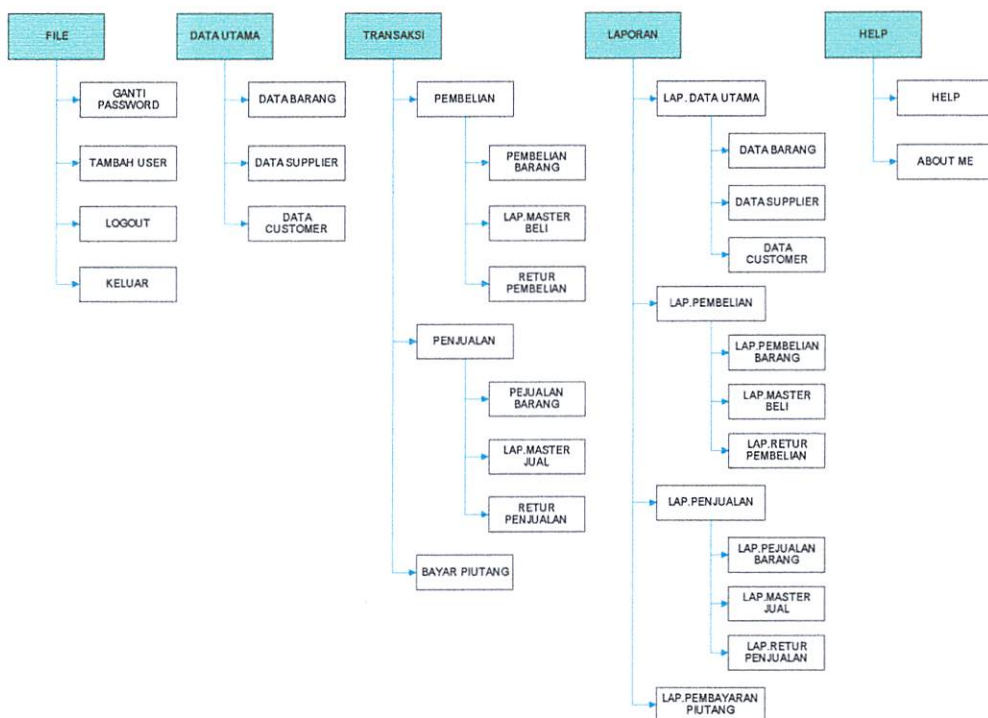
#### 3.3.1 Desain Form Utama



Gambar 3.7 Desain Form Utama

Pada halaman utama ini terdapat menu yang terdiri dari menu file, menu data utama, menu transaksi, dan menu laporan yang mempunyai fungsi untuk mengetahui laporan dan proses yang terjadi pada aplikasi ini.

### 3.3.2 Desain Menu Aplikasi Form Utama



Gambar 3.8 Desain Menu Aplikasi Form Utama

### 3.3.3 Desain Form Entri Data Barang

The screenshot shows a window titled 'PENGEMBANGAN SISTEM INFORMATIKA PERJUALAN DENGAN SISTEM KO...' with a sub-window 'FORM BARANG'. The form is for 'DAFTAR BARANG' and contains the following fields and controls:

- Code Barang: Text input field
- Nama Barang: Text input field
- Stok Minimal: Text input field with value '0'
- STOK: Text input field with value '0'
- HPP: Text input field with value '0'
- Nilai: Text input field with value '0'
- Harga Jual: Text input field with value '0'

At the bottom right, there are three buttons: 'Retry' (with a circular arrow icon), 'OK' (with a checkmark icon), and 'Close' (with a window icon).

Gambar 3.9 Desain Form Entri Data Barang

Form ini berfungsi untuk menginputkan data-data barang. Pada form ini terdapat beberapa tombol dengan fungsi sebagai berikut :

- RETRY : berfungsi untuk mengulang kembali perintah penginput data barang pada form barang.

- b. OK : berfungsi untuk menyimpan data barang ke daftar barang.
- c. CLOSE : berfungsi untuk keluar dari form barang.

### 3.3.4 Desain Form Entri Data Supplier

Form Gambar 3.10 Desain Form Entri Data lata-data supplier. Pada form ini terdapat beberapa tombol dengan fungsi sebagai berikut :

- a. RETRY : berfungsi untuk mengulang kembali perintah penginputan data supplier pada form Supplier.
- b. OK : berfungsi untuk menyimpan data supplier ke daftar Supplier.
- c. CLOSE : berfungsi untuk keluar dari form supplier.

### 3.3.5 Desain Form Entri Data Customer

Gambar 3.11 Desain Form Entri Data Customer

Form ini berfungsi untuk menginputkan data-data customer. Pada form ini terdapat beberapa tombol dengan fungsi sebagai berikut :

- a. RETRY : berfungsi untuk mengulang kembali perintah penginputan data customer pada form customer.
- b. OK : berfungsi untuk menyimpan data customer ke daftar customer.
- c. CLOSE : berfungsi untuk keluar dari form customer.

### 3.3.6 Desain Form Entri Data Pembelian Barang

Gambar 3.12 Form Entri Data Pembelian Barang

Form ini berfungsi untuk menginputkan data-data pembelian. Pada form ini terdapat beberapa tombol dengan fungsi sebagai berikut :

- a. CARI SUPPLIER : berfungsi untuk mencari daftar nama supplier.
- b. CARI BARANG : berfungsi untuk mencari daftar nama barang yang akan kita beli.
- c. OK : berfungsi untuk menyimpan data barang yang telah dibeli ke daftar barang.
- d. CLOSE : berfungsi untuk keluar dari form pembelian.

### 3.3.7 Desain Form Entri Data Retur Pembelian

Gambar 3.13 Desain Form Entri Data Retur Pembelian

Form ini berfungsi untuk menginputkan data-data barang yang diretur. Pada form ini terdapat beberapa tombol dengan fungsi sebagai berikut :

- d. **RETRY** : berfungsi untuk mengulang kembali perintah penginputan data retur pembelian pada form retur pembelian.
- e. **OK** : berfungsi untuk menyimpan data barang yang diretur ke daftar retur.
- f. **CLOSE** : berfungsi untuk keluar dari form retur pembelian.

### 3.3.8 Desain Form Entri Data Penjualan Barang

Gambar 3.14 Desain Form Entri Data penjualan

Form ini berfungsi untuk menginputkan data-data penjualan. Pada form ini terdapat beberapa tombol dengan fungsi sebagai berikut :

- a. **CARI CUSTOMER** : berfungsi untuk mencari daftar nama customer.



- b. CARI BARANG : berfungsi untuk mencari daftar nama barang yang akan kita jual.
- c. OK : berfungsi untuk menyimpan data barang yang telah dijual ke daftar barang.
- d. CLOSE : berfungsi untuk keluar dari form penjualan.

### 3.3.9 Desain Form Entri Data Retur Penjualan

Gambar 3.15 Desain Form Entri Data Retur

Form ini berfungsi untuk menginputkan data-data barang yang diretur. Pada form ini terdapat beberapa tombol dengan fungsi sebagai berikut :

- a. RETRY : berfungsi untuk mengulang kembali perintah penginputan data retur penjualan pada form retur penjualan.
- b. OK : berfungsi untuk menyimpan data barang yang diretur ke daftar retur.
- c. CLOSE : berfungsi untuk keluar dari form retur penjualan.

### 3.3.10 Desain Form Pembayaran Piutang

Gambar 3.16 Desain Form Pembayaran Piutang

Form ini berfungsi untuk melihat data-data pembayaran piutang customer dan melihat jatuh tempo pembayaran piutang.

### 3.3.11 Desain Laporan

Sistem laporan pada sistem informasi ini terdiri atas beberapa laporan, yaitu ;

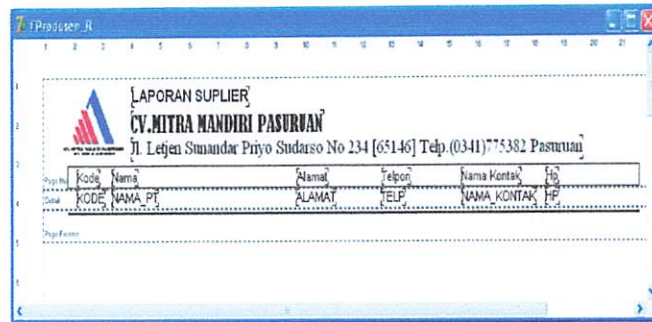
#### 1. Laporan Data Barang

Laporan ini dibuat untuk mengetahui data-data barang yang ada pada sistem informasi. Berikut ini desain tampilan laporan data barang :

Gambar 3.17 Desain Laporan Data Barang

#### 2. Laporan Data Supplier

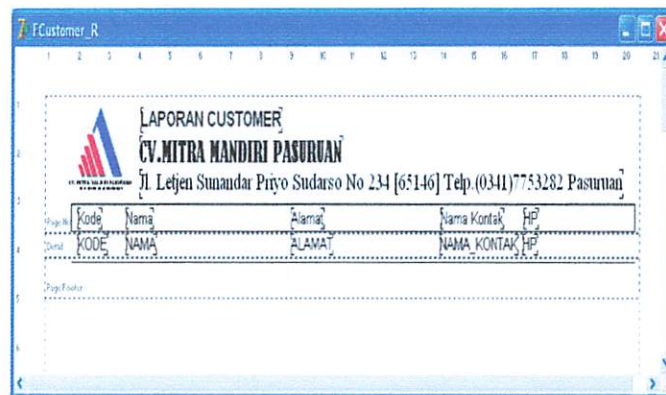
Laporan ini dibuat untuk mengetahui data-data supplier yang ada pada sistem informasi. Berikut ini desain tampilan laporan data supplier :



Gambar 3.18 Desain Laporan Data Supplier

### 3. Laporan Data Customer

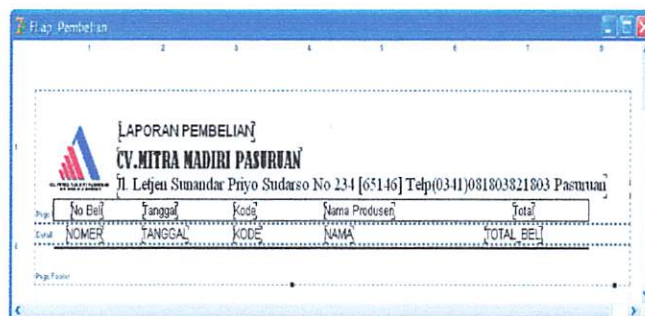
Laporan ini dibuat untuk mengetahui data-data customer yang ada pada sistem informasi. Berikut ini desain tampilan laporan data customer :



Gambar 3.19 Desain Laporan Data Customer

### 4. Laporan Data Pembelian

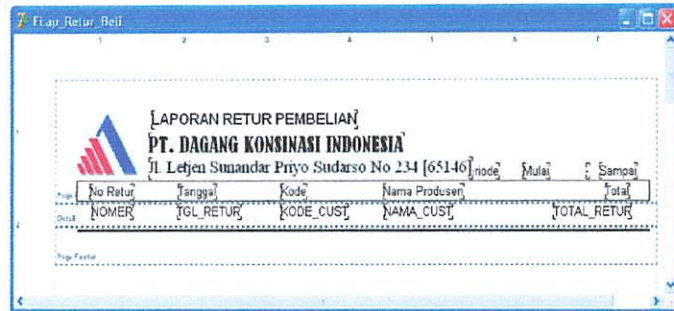
Laporan ini dibuat untuk mengetahui data-data pembelian yang ada pada sistem informasi. Berikut ini desain tampilan laporan data pembelian :



Gambar 3.20 Desain Laporan Data pembelian

### 5. Laporan Data Pembelian

Laporan ini dibuat untuk mengetahui data-data retur pembelian yang ada pada sistem informasi. Berikut ini desain tampilan laporan data retur pembelian :

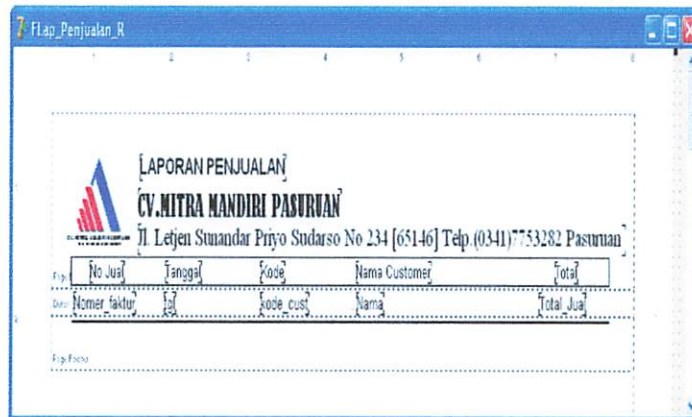


The screenshot shows a report design window for 'LAPORAN RETUR PEMBELIAN' from PT. DAGANG KONSINASI INDONESIA. The report header includes the company name and address: 'PT. DAGANG KONSINASI INDONESIA', 'Jl. Letjen Sunandar Priyo Sudarso No 234 [65146]'. Below the header is a table with columns: 'No Retur', 'Tanggal', 'Kode', 'Nama Produsen', and 'Total'. Below the table, there are fields for 'Nomer', 'TGL\_RETUR', 'KODE\_CUST', 'NAMA\_CUST', and 'TOTAL\_RETUR'.

Gambar 3.21 Desain Laporan Data Retur pembelian

### 6. Laporan Data Penjualan

Laporan ini dibuat untuk mengetahui data-data penjualan yang ada pada sistem informasi. Berikut ini desain tampilan laporan data penjualan :



The screenshot shows a report design window for 'LAPORAN PENJUALAN' from CV.NITRA MANDIRI PASURUAN. The report header includes the company name and address: 'CV.NITRA MANDIRI PASURUAN', 'Jl. Letjen Sunandar Priyo Sudarso No 234 [65146] Telp. (0341)7753282 Pasuruan'. Below the header is a table with columns: 'No Jual', 'Tanggal', 'Kode', 'Nama Customer', and 'Total'. Below the table, there are fields for 'Nomer faktur', 'Tgl', 'Kode cust', 'Nama', and 'Total Jual'.

Gambar 3.22 Desain Laporan Data Penjualan

### 7. Laporan Data Retur Penjualan

Laporan ini dibuat untuk mengetahui data-data retur penjualan yang ada pada sistem informasi. Berikut ini desain tampilan laporan data retur penjualan :

**LAPORAN RETURN PENJUALAN**  
**CV. MITRA MANDIRI PASURUAN**  
 Jl. Letjen Sumanar Priyo Sudarso No 234 [65146] Telp. (0341)7753282 Pasuruan

Page: 1 of 1

No Faktur: NO\_FAKTUR      No Retur: NOMER  
 Tanggal: TGL\_FAKTUR      Tanggal: TGL\_RETUR  
 Kode Custome: KODE\_CUST  
 Nama Custome: NAMA\_CUST

Kode	Nama Barang	Jml Jual	Harga	Sub Total	Keterangan
KODE	NAMA	JML_RETUR	HRG	SUB_TOTAL	KETERANGAN

Gambar 3.23 Desain Laporan Data Retur Penjualan

#### 8. Laporan Data Pembayaran Piutang

Laporan ini dibuat untuk mengetahui data-data pembayaran piutang yang ada pada sistem informasi. Berikut ini desain tampilan laporan data pembayaran piutang :

**LAPORAN PEMBAYARAN PIUTANG**  
**CV. MITRA MANDIRI PASURUAN**  
 Jl. Letjen Sumanar Priyo Sudarso No 234 [65146] Telp. (0341)7753282 Pasuruan

Kode Custome: Date  
 Nama Custome: Date  
 Nomor Faktur: Date  
 Tanggal: Date  
 Total Penjualan: Date

Jml Jatuh Tempo: Date  
 Sisa Piutang: Date  
 Total Bayar: Date

No Bayar	Tanggal	Jumlah Pembayaran
NOMER	TGL_BAYAR	JML_BAYAR

Gambar 3.24 Desain Laporan Data Pembayaran Piutang

## **BAB IV**

### **IMPLEMENTASI DAN PENGUJIAN SISTEM**

#### **4.1 IMPLEMENTASI SISTEM**

Tahap implementasi pengembangan perangkat lunak merupakan proses 43 perubahan spesifikasi sistem menjadi sistem yang dapat dijalankan. Tahap ini merupakan lanjutan dari proses perancangan yaitu proses pemrograman perangkat lunak sesuai dengan spesifikasi dan desain sistem.

Aplikasi sistem informasi ini menggunakan basis data MySQL yang berfungsi sebagai media penyimpanan data atau informasi yang terkumpul, yang terdiri dari beberapa tabel yang saling berhubungan. Implementasi dari desain aplikasi ini berupa implementasi struktur data dari masing-masing proses.

Implementasi dilakukan dengan menerapkan hasil desain yang telah dibuat kedalam bahasa pemrograman Borland Delphi 7, sehingga prosedur-prosedur yang telah dibuat dapat dimengerti oleh mesin sehingga menghasilkan keluaran seperti yang diharapkan. Berikut ini adalah perlengkapan yang digunakan dalam implementasi sistem:

No.	Perlengkapan	Spesifikasi	Keterangan
1	Software PC	Sistem Operasi	Windows XP Service Pack 2
		Bahasa Pemrograman	Borland Delphi 7.0
2	Personal Computer	Processor	Pentium 4 2,4 GHz
		Memori	512 DDR2

Tabel 4.1 Spesifikasi perlengkapan implementasi

Ada beberapa program utama yang sangat penting agar aplikasi ini dapat berjalan. Diantaranya adalah koneksi basisdata, pembacaan data, penambahan data, perubahan data, dan penghapusan data.

#### 4.1.1 Koneksi Basisdata

Teknik pemrograman dengan menggunakan aplikasi yang berbasis windows adalah teknik pemrograman yang berorientasi pada objek. Untuk itu, dibuat suatu *Module* untuk membuat program utama koneksi database, dalam hal ini yaitu *ZEOS*.

#### 4.1.2 Pengoperasian Basisdata

Setiap proses koneksi di atas, pengoperasian basisdata juga menggunakan *Module*, dalam hal ini dibuat *Module* sama dengan *Module* koneksi database. Proses utama dari pengoperasian basisdata meliputi pembacaan data, penyimpanan data, pengubahan data dan penghapusan data.

### 4.2 PENGUJIAN HASIL

#### 4.2.1 Form Login

Pada waktu akan masuk ke dalam tampilan menu utama sistem informasi ini, harus melakukan login terlebih dahulu. Berikut tampilan form login :

Gambar 4.1 Form Login Sebelum Entri Data

Pada saat kita akan menguji form login maka kita langsung masukan user name dan password setelah itu klik OK, jika berhasil akan muncul menu utama dan jika tidak berhasil maka program akan tertutup secara otomatis. Berikut ini tampilan form login setelah entri data:

PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM KON...

**CV. MITRA MANDIRI PASURUAN**  
DISTRIBUTOR MAKANAN RINGAN  
Jl. Letjen Sunandar Priyo Sudarso No 234 [65146] Telp. (0341)7753282 Pasuruan

User Names:

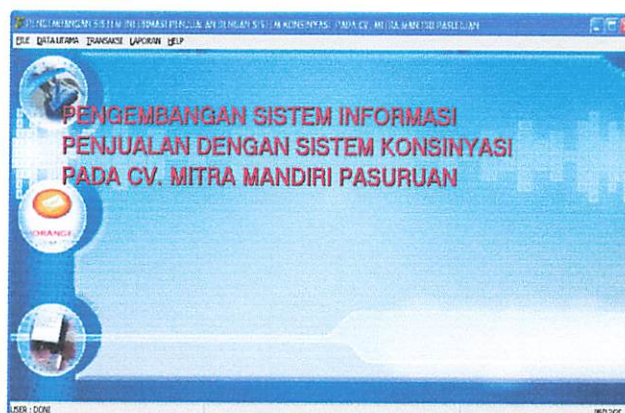
Password:

OK Close

Gambar 4.2 Form Login Setelah Entri Data

#### 4.2.2 Form Menu Utama

Pada menu utama terdapat berbagai fasilitas entri data-data yang digunakan sebagai data utama di setiap bidang data. Fasilitas data utama antara lain entri data customer, entri data supplier, entri data barang, entri data pembelian, entri data penjualan, entri retur data pembelian, entri retur penjualan. Berikut adalah tampilan menu utama pada aplikasi ini. Pada form ini terdapat beberapa menu yang bisa dipilih seperti pada gambar berikut:



Gambar 4.3 Menu Utama

#### 4.2.3 Form Ganti Password

Saat memilih menu file yang terdapat pada menu utama dan memilih ganti password maka akan tampil form ganti password yang berfungsi untuk mengganti password. Berikut ini tampilan form ganti password :



A screenshot of a Windows-style dialog box titled "Change Password". The dialog has a blue header with a close button (X) in the top right corner. On the left side, there are three circular icons: a globe, a person, and a person with a checkmark. The main area contains three text input fields. The first field is labeled "User Name" and contains the text "DONI". The second field is labeled "Old Password" and is empty. The third field is labeled "New Password" and is empty. At the bottom right, there is an "OK" button.

Gambar 4.4 Form Ganti Password Sebelum Entri Data

Pada saat kita akan menguji form ganti password maka kita langsung masukan password yang lama dan kemudian masukan password yang baru selanjutnya klik OK. Berikut ini tampilan form ganti password setelah entri data:

A screenshot of the same "Change Password" dialog box, but now the "Old Password" and "New Password" fields are filled with asterisks (\*), indicating that the user has entered their current and new passwords. The "User Name" field still contains "DONI" and the "OK" button remains at the bottom right.

Gambar 4.5 Form Ganti Password Sesudah Entri Data

#### 4.2.4 Form Tambah User

Saat memilih menu file yang terdapat pada menu utama dan memilih tambah user maka akan tampil form tambah user yang berfungsi untuk menambah user. Berikut ini tampilan form tambah user sebelum entri data :

A screenshot of a software window titled "Password Level User" with a subtitle "FORM PASSWORD ADMIN". The window has a blue header and a light blue background. On the left side, there are three circular icons. The main area contains two text input fields: the first is labeled "Administrator" and the second is labeled "Password". At the bottom right, there is a button labeled "OK".

Gambar 4.6 Form Tambah User Sebelum Entri Data

Pada saat kita akan menguji form tambah user maka kita langsung masukan nama administrator dan kemudian masukan password selanjutnya klik OK. Apabila sudah masuk ke form level user maka kita tinggal tentukan nama user name, password dan status. Berikut ini tampilan form tambah user setelah entri data:

 A screenshot of a software window titled "Form Level User" with a subtitle "FORM LEVEL USER". The window has a blue header and a light blue background. On the left side, there are three circular icons. The main area contains a text input field with the text "ISI". Below it are three fields: "User Name" with a dropdown menu showing "ANI", "Password" with the text "2", and "Status" with a dropdown menu showing "OPERATOR". At the bottom right, there is a button labeled "Simpan".

Gambar 4.7 Form Tambah User Sesudah Entri Data

#### 4.2.5 Form Data Customer

Saat memilih menu data utama yang terdapat pada menu utama dan memilih data customer maka akan tampil form data customer yang berfungsi untuk menambah, mengubah, menghapus data customer. Hasil dari proses entri data customer ditampilkan pada gambar berikut:

The screenshot shows a web-based form titled 'FORM CUSTOMER'. The form is set against a blue background with a globe icon on the left. The fields are as follows:

Field Label	Value
KODE CUSTOMER	
NAMA CUSTOMER	
ALAMAT	
TELEPON	
NAMA KONTAK	
TELEPON / HP	

At the bottom right, there are three buttons: 'Retry' (with a refresh icon), 'OK' (with a green checkmark icon), and 'Close' (with a window icon).

Gambar 4.8 Tampilan Form Data Customer Sebelum Entry Data

Pada saat kita akan menguji atau memasukkan data customer maka kita langsung masukan kode customer, nama customer, alamat, telepon, nama kontak dan no hp. Maka akan tampak seperti gambar di bawah ini:

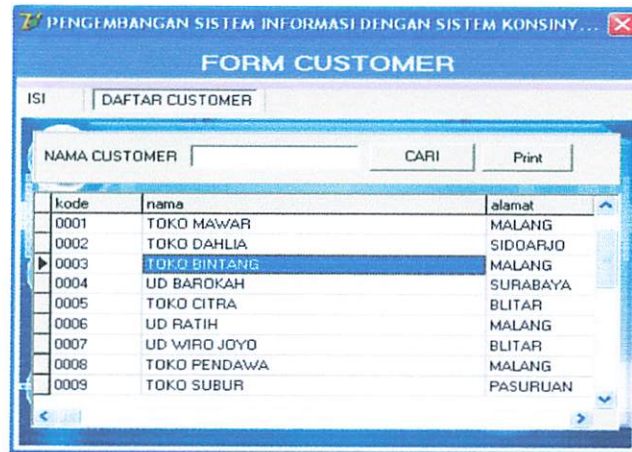
The screenshot shows the same 'FORM CUSTOMER' window, but now the input fields contain the following data:

Field Label	Value
KODE CUSTOMER	0003
NAMA CUSTOMER	TOKO BINTANG
ALAMAT	MALANG
TELEPON	0341-462457
NAMA KONTAK	SURYA
TELEPON / HP	03413525567

The 'OK' button is highlighted with a green checkmark, indicating that the data has been successfully entered and is ready to be saved.

Gambar 4.9 Tampilan Form Data Customer Setelah Entry Data

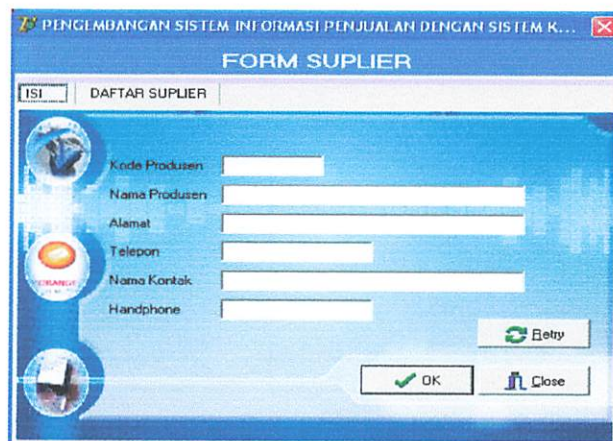
Setelah kita pilih tombol OK, maka data tersebut akan tersimpan pada tabel customer. Seperti pada tampilangambar berikut :



Gambar 4.10 Tampilan Form Tabel Data Customer

#### 4.2.6 Form Data Supplier

Saat memilih menu data utama yang terdapat pada menu utama dan memilih data supplier maka akan tampil form data supplier yang berfungsi untuk menambah, mengubah, dan menghapus data supplier. Hasil dari proses entri data supplier ditampilkan pada gambar berikut:



Gambar 4.11 Tampilan Form Data Supplier Sebelum Entry Data

Pada saat kita akan menguji atau memasukkan data Supplier maka kita langsung ketikkan kode produsen, nama produsen, alamat, telepon, nama kontak dan handphone. Maka akan tampak seperti gambar di bawah ini:

PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM K...

### FORM SUPLIER

ISI | DAFTAR SUPLIER

Kode Produsen: 104  
 Nama Produsen: PT. INDOFOOD SUKSES MAKMUR  
 Alamat: MALANG  
 Telepon: 0341-496956  
 Name Kontak: SUKARJO  
 Handphone: 0856778891

Buttons: Retry, OK, Close

Gambar 4.12 Tampilan Form Data Supplier Setelah di Entry Data

Setelah kita pilih tombol OK, maka data tersebut akan tersimpan pada tabel supplier. Maka akan tampak gambar dibawah ini :

PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM K...

### FORM SUPLIER

ISI | DAFTAR SUPLIER

Nama:  Cari Print

kode	nama_pt	alamat
100	PT. INAFODD	SURABAYA
101	PT. ANGKASA PURA	SURABAYA
102	PT. ARIMBI	MALANG
103	PT. ARGARINDO PERDANA	MALANG
104	PT. INDOFOOD SUKSES MAKMUR	MALANG
105	PT. INDOFLORA CIPTA MANDIRI	MALANG
106	PT. GIZNIDO PRIMA NUSANTARA	SURABAYA
107	PT. INDOMARCO ADI PRIMA	MALANG
108	PT. ITSUN FOOD INDONESIA	PASURUAN
109	PT. NUTRICIA INDONESIA SEJAHTERA	MALANG
110	PT. SINAR NIAGA SEJAHTERA	SIDDARJO
111	PT. SINDU AMRITA	PASURUAN

Gambar 4.13 Tampilan Form Tabel Data Supplier

#### 4.2.7 Form Data Barang

Saat memilih menu data utama yang terdapat pada menu utama dan memilih data barang maka akan tampil form data barang yang berfungsi untuk menambah, mengubah, dan menghapus data barang. Hasil dari proses entri data barang ditampilkan pada gambar berikut:

Gambar 4.14 Tampilan Form Data Barang  
Sebelum Entry Data

Pada saat kita akan menguji atau memasukkan data barang maka kita langsung ketikkan kode barang, nama barang, stok minimal dan untuk pengisian pada stok, hpp, nilai, harga jual kita bisa isikan pada form pembelian. Maka akan tampil seperti gambar di bawah ini:

Gambar 4.15 Tampilan Form Data Barang  
Setelah di Entry Data

Setelah kita pilih tombol OK, maka data tersebut akan tersimpan pada tabel barang. Maka akan tampil seperti dibawah ini:



PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM KONSINYAS...

### FORM BARANG

ISI DAFTAR BARANG

Nama:  Cari

Cek Stok Min.

kode	nama	stok_min	stok	hpp	nilai
028	POLONGKU	1000	0	0	
029	STAR SNACK	1000	2000	300	6
030	MOMOGI	1000	1000	300	3
031	TOYA - TOYA	1000	1458	400	5
033	FRENCH SNACK	1000	0	0	
032	LEO KRIPIK KENTANG	1000	0	0	
034	TARO STIK	1000	100	15000	15
035	SOBA MIE	1000	999	800	7
036	SUKRO	1000	1000	800	8
037	WAFELATOS	1000	1459	1800	26
039	GERY SALUT	1000	0	0	
039	O - DONAT	1000	999	1800	17
040	BIBIT SNACK	100	0	0	

Gambar 4.16 Tampilan Tabel Data Barang

#### 4.2.8 Form Transaksi Pembelian

Saat memilih menu transaksi yang terdapat pada menu utama dan memilih transaksi pembelian maka akan tampil form transaksi pembelian yang berfungsi untuk menambah, menjumlah total pembelian barang. Hasil dari proses transaksi pembelian ditampilkan pada gambar berikut:



PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM KONSINYAS PADA...

### FORM PEMBELIAN

ISI DAFTAR BARANG

No. Pembelian:

Tanggal:

Kode Suplier:

Nama Produsen:

No.	Kode Barang	Nama Barang	Jumlah	Harga	Sub Total
1					0
2					0
3					0
4					0
5					0
6					0

Total Pembelian:

Gambar 4.17 Tampilan Form Transaksi Pembelian Sebelum Entry Data

Pada saat kita akan menguji transaksi pembelian barang maka kita langsung ketikkan kode supplier maka akan muncul nama produsen sesuai kode produsen yang diminta. Setelah itu ketik kode barang yang

dikehendaki dan tentukan jumlah pembelian. Maka akan tampil seperti gambar di bawah ini:

**FORM PEMBELIAN**

ISI DAFTAR BARANG

No. Pembelian: 16  
 Tanggal: 05/12/2010  
 Kode Suplier: 102  
 Nama Produsen: PT. ARIMBI

No.	Kode Barang	Nama Barang	Jumlah	Harga	Sub Total
1	013	CHOKO STEAK SNACK	50	30.000	1.500.000
2					0
3					0
4					0
5					0
6					0

Total Pembelian: 1.500.000

OK Close

Gambar 4.18 Tampilan Form Transaksi Pembelian Setelah di Entry Data

Setelah kita pilih tombol OK, maka data tersebut akan tersimpan pada tabel pembelian. Maka akan tampil seperti gambar dibawah ini :

**FORM PEMBELIAN**

ISI DAFTAR BARANG

no	kode	nama	prl_beli	reg_beli	subs_total	prl_return
9	029	STARI SNACK	2000	300	600000	
9	031	TOYA - TOYA	1500	400	600000	
10	039	O - DONAT	1000	1800	1800000	2
11	052	TIC - TIK WAFER	1000	800	800000	
12	054	STIKO	1500	1000	1500000	1
13	058	NABATI SNACK	2000	300	600000	
14	013	CHOKO STEAK SNACK	1000	1000	1800000	
15	015	CHIRO SNACK	1000	300	300000	
16	024	KORO-KORO SNACK	1500	800	1200000	1
17	026	TARO SNACK	1000	800	800000	3
18	023	KACANG MAYASI	1000	300	300000	
19	025	SOBA MIE	1000	800	800000	1
20	027	SEACUPNH	1000	800	800000	
21	012	WAFFLET	1000	600	600000	
22	017	APPOLO	1500	1800	2700000	
23	030	MOMOGI	1000	300	300000	1
24	035	SURRO	1000	800	800000	
25	027	WABELATOS	1500	1800	2700000	1
26	024	TARO STIK	100	15000	1500000	
27	013	CHOKO STEAK SNACK	50	30000	1500000	

Gambar 4.19 Tampilan Tabel Transaksi Pembelian

#### 4.2.9 Form Transaksi Penjualan

Saat memilih menu transaksi yang terdapat pada menu utama dan memilih Transaksi Penjualan maka akan tampil form transaksi penjualan yang berfungsi untuk melakukan transaksi kepada customer. Hasil dari proses transaksi penjualan ditampilkan pada gambar berikut.

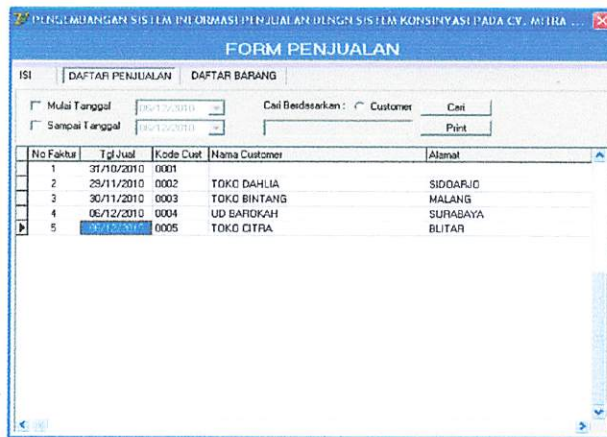


Gambar 4.20 Tampilan Form Transaksi penjualan Sebelum Entri Data

Pertama kita ketikkan kode customer yang akan kita pilih. Setelah itu kita ketikkan kode barang, jumlah barang dan harga barang yang dibeli oleh customer. Hasil dari proses ini akan tampil seperti gambar berikut :

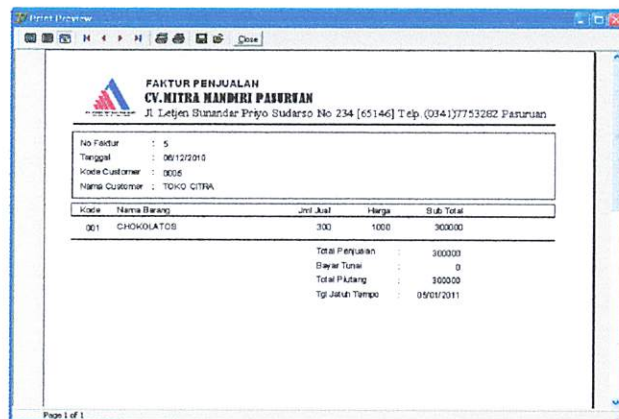
Gambar 4.21 Tampilan Form Transaksi Penjualan Setelah Entri Penjualan Data

Setelah kita pilih tombol OK, maka data tersebut akan tersimpan pada daftar penjualan. Maka akan tampil seperti gambar dibawah ini :



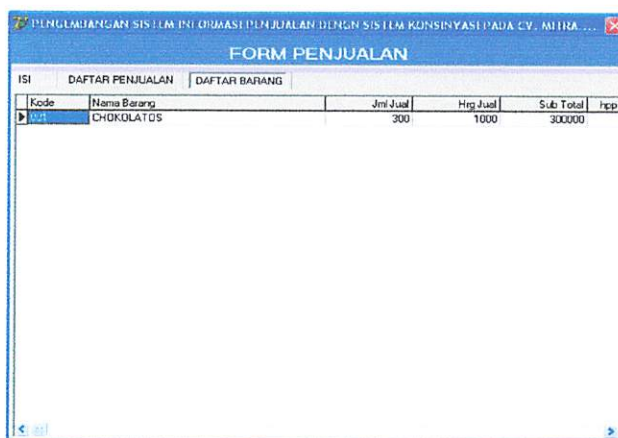
Gambar 4.22 Tampilan Daftar Customer Yang Melakukan Transaksi Penjualan

Setelah itu jika kita pilih tombol print maka akan tampil laporan faktur penjualan sebagai bukti penjualan. Berikut ini tampilan dari laporan faktur penjualan :



Gambar 4.23 Tampilan Laporan Faktur Penjualan Percustomer

Setelah proses tersebut selesai dilakukan untuk melihat daftar barang yang dijual, maka pilih daftar barang. Maka akan tampil seperti gambar berikut ini.



The screenshot shows a window titled 'FORM PENJUALAN' with a sub-tab 'DAFTAR BARANG'. It contains a table with the following data:

Kode	Nama Barang	Jml Jual	Hrg Jual	Sub Total	Jpp
▶	CHOKOLATIS	300	1000	300000	

Gambar 4.24 Tampilan Daftar Barang yang Telah Dijual

#### 4.2.10 Form Laporan Pembelian

Saat memilih menu transaksi yang terdapat pada menu utama dan memilih pembelian setelah itu memilih laporan master beli maka akan tampil form laporan master beli yang berfungsi untuk melihat semua data master pembelian. Tampilan dari laporan master beli seperti dibawah ini :




The screenshot shows a window titled 'Form Laporan Pembelian'. It contains two date selection fields, each with a checkbox and a dropdown menu:

- Mulai Tanggal: 07/12/2010
- Sampai Tanggal: 07/12/2010

Below these fields is a 'Print' button with a printer icon.

Gambar 4.25 Tampilan Form Laporan Pembelian

Setelah itu jika kita pilih tombol print maka akan tampil daftar laporan pembelian sebagai bukti laporan pembelian. Berikut ini adalah tampilan dari laporan pembelian :

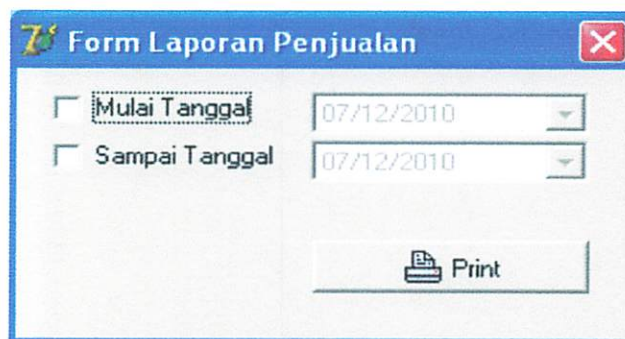


No Bill	Tanggal	Kode	Nama Produsen	Total
1	04/11/2010	100	PT. NAFOD	640000
2	04/11/2010	101	PT. ANGKASA PURA	1560000
3	04/11/2010	102	PT. ARIMBI	960000
4	04/11/2010	103	PT. ARGARINDO PERDANA	2400000
5	04/11/2010	104	PT. NDOFOOD SUKSES MAKMUR	2750000
6	04/11/2010	100	PT. NAFOD	600000
7	04/11/2010	106	PT. GENDIO PRIMA HUSANTARA	2100000
8	04/11/2010	107	PT. INDOMARCO ADI PRIMA	2000000
9	04/11/2010	108	PT. ITSUN FOOD INDONESIA	1100000
10	04/11/2010	109	PT. NUTRICA INDONESIA SEJAHTERA	800000
11	04/11/2010	111	PT. SINDU AMRITA	800000
12	04/11/2010	112	PT. SUMBER RASA	2700000
13	04/11/2010	113	PT. URIRAMA DUTA NAGA	300000
14	04/11/2010	114	PT. NESTLE INDONESIA	3500000
15	29/11/2010	111	PT. SINDU AMRITA	1500000

Gambar 4.26 Tampilan Laporan Pembelian

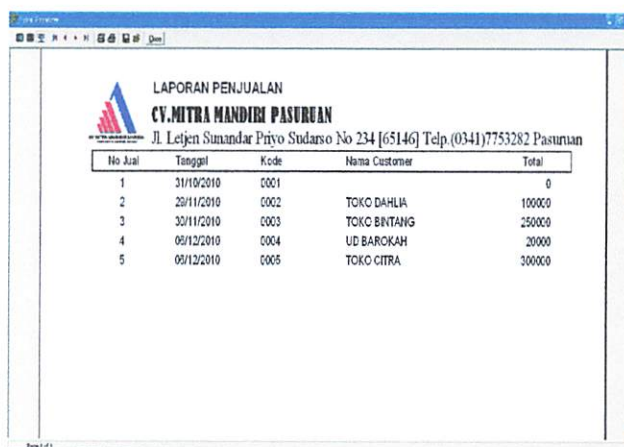
#### 4.2.11 Form Laporan Penjualan

Saat memilih menu transaksi yang terdapat pada menu utama dan memilih penjualan setelah itu memilih laporan master jual maka akan tampil form laporan master jual yang berfungsi untuk melihat semua data master penjualan. Tampilan dari laporan master jual seperti dibawah ini :



Gambar 4.27 Tampilan Form Laporan Penjualan

Setelah itu jika kita pilih tombol print maka akan tampil daftar laporan master penjualan seperti gambar berikut ini:



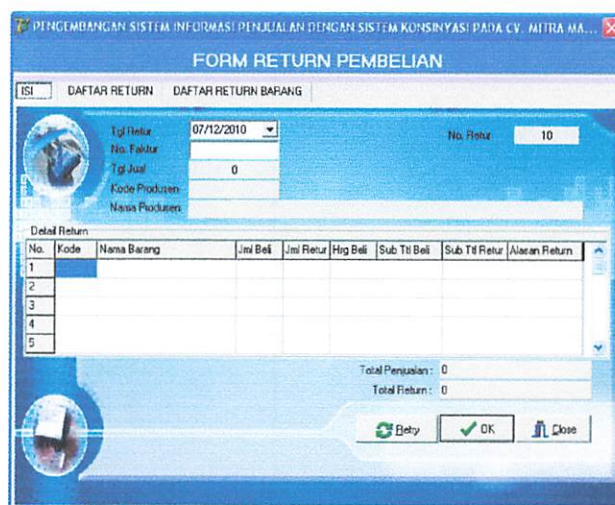
**LAPORAN PENJUALAN**  
**CV.MITRA MANDIRI PASURUAN**  
Jl. Letjen Sunandar Priyo Sudarmo No 234 (65146) Telp.(0341)7753282 Pasuruan

No Jual	Tanggal	Kode	Nama Customer	Total
1	31/10/2010	0001		0
2	20/11/2010	0002	TOKO DAHLIA	100000
3	30/11/2010	0003	TOKO BINTANG	250000
4	09/12/2010	0004	UD BAROKAH	20000
5	09/12/2010	0005	TOKO CITRA	300000

Gambar 4.28 Tampilan Laporan penjualan

#### 4.2.12 Form Retur Pembelian

Saat memilih menu transaksi yang terdapat pada menu utama dan memilih pembelian setelah itu memilih retur pembelian maka akan tampil form retur pembelian yang berfungsi untuk meretur barang yang rusak. Tampilan dari form retur pembelian seperti dibawah ini :



**FORM RETURN PEMBELIAN**

ISI DAFTAR RETURN DAFTAR RETURN BARANG

Tgl Retur: 07/12/2010 No. Retur: 10  
 No. Faktur:   
 Tgl. Jual: 0  
 Kode Produsen:   
 Nama Produsen:

Detail Retur:

No.	Kode	Nama Barang	Jml Beli	Jml Retur	Hrg Beli	Sub Ttl Beli	Sub Ttl Retur	Alasan Retur
1								
2								
3								
4								
5								

Total Penjualan: 0  
 Total Retur: 0

Batal OK Close

Gambar 4.29 Tampilan Form Retur Pembelian Sebelum Entri Data

Pada saat kita akan menguji form retur pembelian maka kita langsung ketikkan nomer faktur maka akan muncul nama produsen dan data barang yang akan diretur. Berikut ini adalah gambar form retur pembelian setelah entri data :

Gambar 4.30 Tampilan Form Retur Pembelian Setelah Entri Data

Setelah kita pilih tombol OK, maka data tersebut akan tersimpan pada daftar retur pembelian. Berikut ini adalah tampilan form daftar retur pembelian:

No Retur	Tgl Retur	No Beli	Tgl Beli	Kode	Nama Produsen	Total Beli / Retur
1	11/12/2010	1	11/12/2010	1	PT. ARIES CENTAURUS	13500000 27
2	12/12/2010	2	11/12/2010	2	PT. ENAK JAYA MAKMUR	17500000 35
3	12/12/2010	3	11/12/2010	3	PT. GEMINDO PRIMA NUSANTARA	14000000 42
4	12/12/2010	4	11/12/2010	4	PT. INDOFOOD SUKSES MAKMUR	14500000 125
5	12/12/2010	5	11/12/2010	5	PT. INDOMARCO ADI PRIMA	7000000 56
6	12/12/2010	7	11/12/2010	7	PT. NESTLE INDONESIA	12000000 66
7	12/12/2010	8	11/12/2010	8	PT. PRIMAFENDO PANGAN MAJ	14500000 71
8	12/12/2010	9	11/12/2010	9	PT. SARI PALMA SEMBILAN JAY	14000000 71
9	12/12/2010	11	11/12/2010	11	PT. SINDU AMRITHA	11000000 31
10	12/12/2010	12	11/12/2010	12	PT. GARUDA FOOD INDONESIA	18000000 86
11	13/12/2010	13	12/12/2010	6	PT. ITSUN FOOD INDONESIA	8500000 51

Gambar 4.31 Tampilan Form Daftar Retur Pembelian

Untuk melihat laporan daftar produsen untuk barang yang diretur, maka kita bisa memilih tombol print master. Berikut ini tampilan laporan daftar retur pembelian barang :

**LAPORAN RETUR PEMBELIAN**  
**CV. MITRA MANDIRI PASURUAN**  
 Jl. Letjen Sunandar Priyo Sudarmo No 234 [65146] Telp.(0341)7753282 Pasuruan

No Retur	Tanggal	Kode	Nama Produsen	Total
1	11/12/2010	1	PT. ARIES CENTAURIUS	270000
2	12/12/2010	2	PT. ENAK JAYA MAKMUR	350000
3	12/12/2010	3	PT. GIZNCO PRIMA NUSANTARA	450000
4	12/12/2010	4	PT. INDOFOOD SUKSES MAKMUR	120000
5	12/12/2010	5	PT. INDOMARCO ADI PRIMA	550000
6	12/12/2010	7	PT. NESTLE INDONESIA	500000
7	12/12/2010	8	PT. PRIMAFENDO PANGAN MAKMUR	720000
8	12/12/2010	9	PT. SARI PALMA SEMBLAN JAYA	750000
9	12/12/2010	11	PT. SINEU AMRITHA	350000
10	12/12/2010	12	PT. GARUDA FOOD INDONESIA	820000
11	13/12/2010	6	PT. ITSUN FOOD INDONESIA	510000

Gambar 4.32 Tampilan Laporan Daftar Produsen Untuk Barang Yang Di Retur

Untuk melihat laporan daftar barang yang diretur, maka kita bisa memilih tombol print master. Berikut ini tampilan laporan daftar barang yang diretur :

**LAPORAN RETUR PEMBELIAN**  
**CV. MITRA MANDIRI PASURUAN**  
 Jl. Letjen Sunandar Priyo Sudarmo No 234 [65146] Telp.(0341)7753282 Pasuruan

No Faktur : 13	No Retur : 11
Tanggal : 12/12/2010	Tanggal : 13/12/2010
Kode Produsen : 6	
Nama Produsen : PT. ITSUN FOOD INDONESIA	

Kode	Nama Barang	Jml Jual	Harga	Sub Total	Keterangan
1900	CHITATOZ	3	170000	510000	Rusak

Gambar 4.33 Tampilan Laporan Daftar Barang Yang Diretur

Untuk melihat daftar barang yang diretur percustomer, kita bisa pilih tombol daftar retur pembelian barang. Berikut ini tampilan form daftar retur pembelian barang percustomer :

Kode	Nama Barang	Jml Retur	Harga Jual	Sub Total Retur	Keterangan
0001	CHITATQZ	3	170000	510000	Rusak

Gambar 4.34 Tampilan Form Daftar Retur Pembelian Barang Percustomer

#### 4.2.13 Form Retur Penjualan

Saat memilih menu transaksi yang terdapat pada menu utama dan memilih penjualan setelah itu memilih retur penjualan maka akan tampil form retur penjualan yang berfungsi untuk meretur barang yang rusak. Tampilan dari form retur penjualan seperti dibawah ini :

Gambar 4.35 Tampilan Form Retur Pejualan Sebelum Entri Data

Pada saat kita akan menguji form retur penjualan maka kita langsung ketikkan nomer faktur maka akan muncul nama customer dan data barang yang akan diretur: Berikut ini adalah gambar form retur penjualan setelah entri data :



**FORM RETURN PENJUALAN**

ISI | DAFTAR RETURN | BARANG

Master Return  
 No. Faktur: 1  
 Tgl. Jual: 11/12/2010  
 Kode Cus: 0001  
 Nama Cus: TOKO CITRA  
 Alamat: MALANG  
 Tgl. Retur: 12/12/2010  
 No. Retur: 1

No	Kode	Nama Barang	Jml. Jual	Jml. Retur	Hrg. Jual	Sub. Tgl. Jual	Sub. Tgl. Retur	Absensi Retur
1	1000	CHOKOLATOS	5	1	150.000	750.000	150.000	Salah Kemasan
2	1100	TIC-TIC WAFER	5	1	170.000	850.000	170.000	Rusak

Total Penjualan: 1.800.000  
 Total Bayar: 600.000  
 Sisa Piutang: 1.500.000  
 Total Return: 320.000  
 Pengembalian Tunai: 0  
 Penolangan Piutang: 320.000

Batal | Simpan

Gambar 4.36 Tampilan Form Retur Pejualan Setelah Entri Data

Setelah kita pilih tombol SIMPAN, maka data tersebut akan tersimpan pada daftar retur penjualan. Berikut ini adalah tampilan form daftar customer yang melakukan retur penjualan:

**FORM DAFTAR CUSTOMER YANG MELAKUKAN RETURN PENJUALAN**

ISI | DAFTAR RETURN | BARANG

Mulai Tanggal: 12/12/2010  
 Sampai Tanggal: 12/12/2010  
 Cari | Print

No Retur	Tgl Retur	No Faktur	Tgl Faktur	Kode	Nama Customer	Total Jual	Total Retur
1	12/12/2010	1	11/12/2010	0001	TOKO CITRA	1600000	320000
2	13/12/2010	2	12/12/2010	0002	TOKO 29	1600000	220000
3	13/12/2010	3	12/12/2010	0003	UD DEWI	1500000	160000
4	13/12/2010	4	12/12/2010	0004	UD INTAN	2040000	240000
5	13/12/2010	5	12/12/2010	0005	TOKO JP	1660000	100000
6	13/12/2010	6	12/12/2010	0006	UD JAYA	2120000	200000

Gambar 4.37 Tampilan Form Daftar Customer Yang Melakukan Retur Pejualan

Untuk melihat laporan retur penjualan, maka kita bisa memilih tombol print. Berikut ini tampilan laporan retur penjualan barang :

**LAPORAN RETURN PENJUALAN**  
**CV. MITRA MANDIRI PASURUAN**  
 Jl. Letjen Sunandar Priyo Sudarso No 234 [65146] Telp. (0341)7753282 Pasuruan

No Faktur	: 1	No Retur	: 1
Tanggal	: 11/12/2010	Tanggal	: 12/12/2010
Kode Customer	: 0001		
Nama Customer	: TOKO CITRA		

Kode	Nama Barang	Jml Awal	Harga	Sub Total	Keterangan
1000	CHOKLATOS	1	150000	150000	Salah Kirim
1100	TIC-TIC WAFER	1	170000	170000	Rusak

Gambar 4.38 Tampilan laporan retur penjualan barang

Untuk melihat daftar barang yang diretur percustomer, kita bisa pilih tombol barang. Berikut ini tampilan form daftar retur penjualan barang percustomer :

**FORM RETURN PENJUALAN**

ISI    DAFTAR RETURN    **BARANG**

Kode	Nama Barang	Jml Retur	Harga Jual	Sub Total Retur	Keterangan
1000	CHOKLATOS	1	150000	150000	Salah Kirim
1100	TIC-TIC WAFER	1	170000	170000	Rusak

Gambar 4.39 Tampilan Form Daftar Retur Penjualan Barang Percustomer

#### 4.2.14 Form Pembayaran Piutang

Saat memilih menu transaksi yang terdapat pada menu utama dan memilih pembayaran piutang maka akan tampil form pembayaran piutang yang berfungsi untuk mengetahui semua piutang customer dan jatuh tempo pembayaran. Tampilan dari form pembayaran piutang semua customer seperti dibawah ini :

Kode Cust	Nama Customer
0001	TOKO MAWAR
0002	TOKO DAHLIA
0003	TOKO BINTANG
0004	UD BAROKAH
0005	TOKO CITRA

Gambar 4.40 Tampilan Form Daftar Piutang Semua Customer

Untuk mengecek daftar piutang percustomer kita bisa pilih piutang percustomer maka akan tampil daftar piutang percustomer. Berikut ini adalah tampilan form daftar piutang percustomer :

No Faktur	Tgl Jual	jatuh_tempo	Jumlah Hutang	Dibayar	Saldo Hutang
6	06/01/2011	06/01/2011	20000		20000

Gambar 4.41 Tampilan Form Daftar Piutang Percustomer

Untuk mengecek data pembayaran piutang percustomer kita bisa pilih pembayaran piutang maka akan tampil data pembayaran piutang percustomer. Berikut ini adalah tampilan form data pembayaran piutang percustomer :

**FORM BAYAR PIUTANG**

Daftar Piutang Semua Customer | Piutang/Customer | **Pembayaran Piutang** | Piutang Jatuh Tempo

DAFTAR PIUTANG PER CUSTOMER

Data Penjualan:  
 Nomor Faktur: 6  
 Tanggal Jual: 07/12/2010  
 Kode Cust: 0001  
 Nama Customer: TOKO MAWAR  
 Total Hutang: 20.000  
 Total Pembayaran: 0  
 Sisa Hutang: 20.000  
 Tgl Jatuh Tempo: 05/01/2011

Data Pembayaran Piutang:

No Bayar	Tanggal Bayar	Jumlah Pembayaran

Tanggal Bayar: 10/12/2010  
 Jumlah Bayar:   
 Simpan

Gambar 4.42 Tampilan Form Data Pembayaran Piutang Percustomer

Untuk mengecek jatuh tempo pembayaran piutang customer kita bisa pilih piutang jatuh tempo maka akan tampil data customer dan tanggal jatuh tempo pembayaran piutang customer. Berikut ini adalah tampilan form jatuh tempo pembayaran piutang customer :

**FORM BAYAR PIUTANG**

Daftar Piutang Semua Customer | Piutang/Customer | Pembayaran Piutang | **Piutang Jatuh Tempo**

Mulai Tanggal: 01/11/2010  
 Sampai Tanggal: 14/01/2011  
 Cari | Print

No Faktur	Tgl Jual	Tgl Jatuh Tempo	Kode Cust	Nama Customer	Total Piutang
2	29/11/2010	29/12/2010	0002	TOKO DAHLIA	10000
3	30/11/2010	30/12/2010	0003	TOKO BINTANG	150000
4	06/12/2010	05/01/2011	0004	UD BAROKAH	20000
5	06/12/2010	05/01/2011	0005	TOKO CITRA	300000
6	07/12/2010	05/01/2011	0001	TOKO MAWAR	20000

Gambar 4.43 Tampilan Form Jatuh Tempo Pembayaran Piutang Customer

Apabila kita pilih tombol print maka akan tampil laporan jatuh tempo pembayaran piutang untuk bukti laporan jatuh tempo pembayaran piutang. Berikut ini adalah tampilan laporan jatuh tempo pembayaran piutang :

**LAPORAN PIUTANG JATUH TEMPO**  
**CV. HETRA HANINDA PASTORAN**  
 Jl. Lajang Srengsari Permai Blok 3 No. 214 (01414) Telp. (0341) 775128 Pangkajene

No Faktur	No Faktur	Kode	Kategori	Jumlah	Debit	Kredit
1	001-221K	001	RO-014-001	10000	10000	
2	001-221K	001	RO-014-001	10000		10000
3	001-221K	001	RO-014-001	10000		10000
4	001-221K	001	RO-014-001	10000		10000
5	001-221K	001	RO-014-001	10000		10000
6	001-221K	001	RO-014-001	10000		10000

Gambar 4.44 Tampilan Laporan Jatuh Tempo  
 Pembayaran Piutang Customer

## **BAB V**

### **PENUTUP**

#### **5.1 KESIMPULAN**

Berdasarkan data yang didapat oleh penulis sehubungan dengan sistem dan aplikasi yang baru diterapkan pada CV.MITRA MANDIRI PASURUAN, maka penulis dapat mengambil beberapa kesimpulan yaitu :

1. Penyimpanan data transaksi dapat dilakukan lebih cepat dibandingkan dengan cara penyimpanan sebelumnya yaitu dengan dicatat pada buku transaksi sehingga kinerja karyawan lebih efisien.
2. Penyusunan laporan dapat dilakukan lebih cepat karena laporan dapat langsung dari form laporan pada aplikasi, sehingga pada tutup buku tiap bulan, karyawan dapat segera mencetak laporan yang dibutuhkan.
3. Stok barang yang kosong dapat diketahui dari form data barang, tanpa harus mengecek satu persatu barang, sehingga karyawan dapat segera melakukan transaksi pembelian.

#### **5.2 SARAN**

Dalam perencanaan dan pembuatan aplikasi ini terdapat beberapa hal yang perlu diperhatikan dalam pengembangan lebih lanjut yaitu :

1. Aplikasi ini masih dapat dikembangkan lebih jauh lagi ke dalam bentuk web yang terkoneksi dengan internet untuk memudahkan pelayanan pada pelanggan dan bisa memberikan informasi yang lebih luas kepada masyarakat.
2. Aplikasi ini bisa dikembangkan lagi menjadi berbasis mobile melalui telephone seluler atau sms gateway.

# LAMPIRAN



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAM TEKNOLOGI NASIONAL MALANG  
**INSTITUT TEKNOLOGI NASIONAL MALANG**

**FAKULTAS TEKNOLOGI INDUSTRI**  
**FAKULTAS TEKNIK SIPIL DAN PERENCANAAN**  
**PROGRAM PASCASARJANA MAGISTER TEKNIK**

T. BNI (PERSERO) MALANG  
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145  
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

---

**BERITA ACARA UJIAN SKRIPSI**  
**FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Donny Prasetyo  
NIM : 06.12.524  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika  
Judul Skripsi : PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN  
SISTEM KONSINYASI PADA CV.MITRA MANDIRI PASURUAN

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu ( S-1 ) pada :

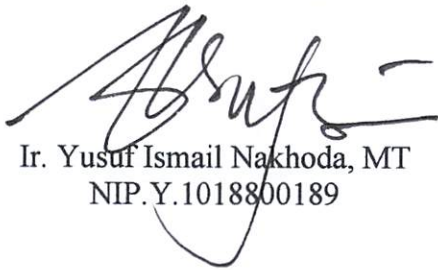
Hari : Sabtu

Tanggal : 12 Februari 2011

Dengan Nilai : 85,6 (A) *8*


Panitia Ujian Skripsi,

Ketua Majelis Penguji



Ir. Yusuf Ismail Nakhoda, MT  
NIP.Y.1018800189

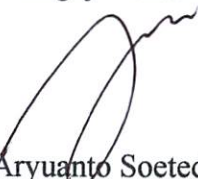
Sekretaris Majelis Penguji



Dr. Eng. Aryuanto Soetedjo, ST. MT  
NIP.Y.1030800417


Anggota Penguji,

Penguji Pertama



Dr. Eng. Aryuanto Soetedjo, ST. MT  
NIP.Y.1030800417

Penguji Kedua



Sandy Nataly Mantja, S.Kom  
NIP.P.1030800418



**FORMULIR PERBAIKAN SKRIPSI**

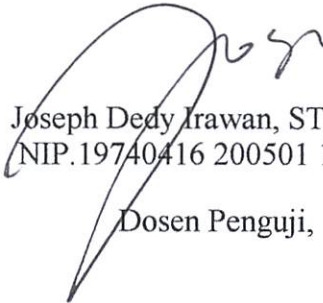
Dalam pelaksanaan Ujian Skripsi Jenjang Program Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer & Informatika, maka perlu adanya perbaikan Skripsi untuk mahasiswa :

Nama : Donny Prasetyo  
NIM : 06.12.524  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika  
Masa Bimbingan : 26 Oktober 2010 s/d 26 April 2011  
Judul Skripsi : PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM KONSINYASI PADA CV.MITRA MANDIRI PASURUAN

Tanggal	Uraian	Paraf
Penguji Pertama 12 Februari 2011	<ul style="list-style-type: none"><li>Penjelasan tentang konsinyasi, Perbaikan menu untuk konsinyasi, Perbaikan menu tidak perlu enter berkali-kali.</li></ul>	
Penguji Kedua 12 Februari 2011	<ul style="list-style-type: none"><li>Manajer memiliki hak akses penentuan harga pada program, Kalo ada teori yang tidak menunjang skripsi secara langsung tidak usah dicantumkan, Perbaikan dfd semua level dan cdm, Tampilan desain bukan sudah jadi (untuk tampilan), Tambahkan laporan rugi laba penjualan.</li></ul>	

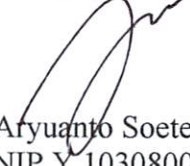
Mengetahui,

Dosen Pembimbing

  
Joseph Dedy Irawan, ST, MT  
NIP.19740416 200501 1 002

Dosen Penguji,

Penguji Pertama

  
Dr. Eng. Aryuanto Soetedjo, ST, MT  
NIP.Y.1030800417

Penguji Kedua

  
Sandy Nataly Mantja, S.Kom  
NIP.P.1030800418



# CV. MITRA MANDIRI PASURUAN

## DISTRIBUTOR MAKANAN RINGAN

Jl. Letjen Sunandar Priyo Sudarmo No 234 [65146] Telp (0341) 7753282 Pasuruan

Pasuruan, 18 Agustus 2010

Sifat : Biasa  
Lampiran : -  
Perihal : **Survey**

Saya yang bertanda tangan dibawah ini :

Nama : Ahmad Fahrudi  
Jabatan : Manager / Pemilik

Dengan ini menerangkan bahwa :

Nama : Donny Prasetyo  
Nim : 06.12.524  
Universitas : Institut Teknologi Nasional Malang  
Fakultas : Teknologi Industri  
Jurusan : Teknik Komputer dan Informatika S-1

Telah melaksanakan survey ke CV.Mitra Mandiri Pasuruan yang saya pimpin pada tanggal 15 Agustus 2010 sampai 18 agustus 2010. Untuk mencari informasi dan data - data yang dibutuhkan untuk penyusunan Skripsi dengan ketentuan sebagai berikut :

1. Data – data yang didapat dari CV. Mitra Mandiri Pasuruan hanya untuk keperluan skripsi.
2. Selama melaksanakan Survey agar mahasiswa mematuhi segala ketentuan yang berlaku di CV. Mitra Mandiri Pasuruan.

Demikian surat keterangan survey kami buat dengan sebenarnya.

  
CV. Mitra Mandiri Pasuruan  
Manager/Pemilik  
  
CV. MITRA MANDIRI PASURUAN  
DISTRIBUTOR MAKANAN RINGAN  
Ahmad Fahrudi, SE



INSTITUT TEKNOLOGI NASIONAL

Jl. Raya Karanglo Km 2

MALANG

### FORMULIR BIMBINGAN SKRIPSI

Nama : Donny Prasetyo  
Nim : 0612524  
Masa Bimbingan : 26 Oktober 2010 s/d 26 April 2011  
Judul Skripsi : PENGEMBANGAN SISTEM INFORMASI PENJUALAN  
DENGAN SISTEM KONSINYASI PADA CV.MITRA MANDIRI  
PASURUAN

No	Tanggal	Uraian	Paraf Pembimbing
1	10-11-2010	REVISI BATASAN MASALAH PADA BAB I	
2	15-11-2010	ACC BAB I	
3	15-11-2010	ACC BAB II	
4	05-01-2011	REVISI TAMPILAN DESAIN MENU UTAMA PADA BAB III	
5	05-01-2011	REVISI TAMPILAN LAPORAN PENJUALAN PADA BAB IV	
6	12-01-2011	ACC BAB III	
7	18-01-2011	ACC BAB IV DAN BAB V	
8			
9			
10			

Malang, 16 Maret 2011

Dosen Pembimbing

(Joseph Dedy Irawan, ST. MT)

NIP.19740416 200501 1 002

Form S-4b



DAFTAR PRESTASI AKADEMIK PRAKTIKUM  
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA

Nama Mahasiswa	:	Donny Prasetyo
NIM	:	06.12.524
Tempat, Tanggal Lahir	:	Malang, 1 Agustus 1986
Jenjang	:	Strata 1 (S1)
Fakultas	:	Teknologi Industri
Jurusan / Program Studi	:	Teknik Elektro
Konsentrasi	:	Teknik Komputer dan Informatika

Praktikum Laboratorium	Kode	Nama Praktikum	SKS	Nilai
I	EL-2215 wb	Fisika	1	B+
		Rangkaian Listrik		A
		Rangkaian Logika dan Digital		B+
		Dasar Komputer dan Pemrograman		A
II	EL-4216 wb	Dasar Elektronika	1	B
		Dasar Sistem Telekomunikasi		B5
		Mikrokontroler		B+
		Sistem Pengukuran		B+
		Dasar Sistem Kendali		B+
III	EL-5316 wb	Basis Data	1	B
		Administrasi Jaringan		B+
		Sistem Operasi		B
IV	EL-6317 wb	Pemrograman Internet	1	B
		Pemrograman Objek		B
		Rekayasa Perangkat Lunak Sistem Informasi		B
V PKM	EL-7318 wb	Peripheral dan Antar Muka	1	A
		Pemrosesan Sinyal Digital		B
		Multimedia		B
		Pemrograman Jaringan		A

Malang, \_\_\_\_\_

Recording  
Jurusan Teknik Elektro S1

  
Puji Handayani



## PERMOHONAN PERSETUJUAN SKRIPSI

Yang betanda tangan dibawah ini :

Nama : Donny Prasetyo  
 NIM : 06.18.529  
 Semester : VIII  
 Fakultas : Teknologi Industri  
 Jurusan : Teknik Elektro S-1  
 Konsentrasi : **TEKNIK ELEKTRONIKA**  
**TEKNIK ENERGI LISTRIK**  
**TEKNIK KOMPUTER DAN INFORMATIKA**  
 Alamat : .....

Dengan ini kami mengajukan permohonan untuk mendapatkan persetujuan untuk membuat **SKRIPSI Tingkat Sarjana**. Untuk melengkapi permohonan tersebut, bersama kami lampirkan persyaratan-persyaratan yang harus dipenuhi.

Adapun persyaratan-persyaratan pengambilan **SKRIPSI** adalah sebagai berikut :

1. Telah melaksanakan semua praktikum sesuai dengan konsentrasinya (.....)
2. Telah lulus dan menyerahkan Laporan Praktek Kerja (.....)
3. Telah lulus seluruh mata kuliah keahlian (MKB) sesuai konsentrasinya (.....)
4. Telah menempuh mata kuliah  $\geq 134$  sks dengan IPK  $\geq 2$  dan tidak ada nilai E (.....)
5. Telah mengikuti secara aktif kegiatan seminar skripsi yang diadakan Jurusan (.....)
6. Memenuhi persyaratan administrasi (.....)

Demikian permohonan ini untuk mendapatkan penyelesaian lebih lanjut dan atas perhatiannya kami ucapkan terima kasih.

Telah diteliti kebenaran data tersebut diatas  
 Recording Teknik Elektro

Handayani  
 (.....)

Malang, .....200  
 Pemohon

Donny Prasetyo  
 (.....)

Disetujui  
 Ketua Jurusan Teknik Elektro

Ir. F. Yudi Limpraptono, MT  
 NIP. P. 1039500274

Mengetahui  
 Dosen Wali

[Signature]  
 (.....)

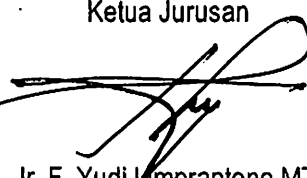
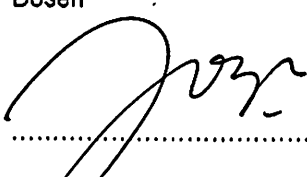
Catatan :  
 Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI agar membuat proposal dan mendapat persetujuan dari Ketua Jurusan/Sekretaris Jurusan T. Elektro S-1

1. IPK 117.5 / 138 = 2.03 .....
2. .....
3. - praktikum praktikum? lengkap. .....



## LEMBAR PENGAJUAN JUDUL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika/Teknik Komputer & Informatika\*)

1.	Nama Mahasiswa: <u>Donny Prasetyo</u>	Nim: <u>06.12.521</u>
2.	Waktu Pengajuan	Tanggal: _____ Bulan: _____ Tahun: _____
3.	Spesifikasi Judul (berilah tanda silang)**)	
	a. Sistem Tenaga Elektrik	e. Elektronika & Komponen
	b. Energi & Konversi Energi	f. Elektronika Digital & Komputer
	c. Tegangan Tinggi & Pengukuran	g. Elektronika Komunikasi
	d. Sistem Kendali Industri	h. lainnya .....
4.	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen*) <u>Yoseph Deddy ... ST MS</u>	Ketua Jurusan  Ir. F. Yudi Limpraptono, MT NIP. P. 1039500274
5.	Judul yang diajukan mahasiswa:	<u>Pengembangan Sistem Informasi penjualan dengan sistem bondingasi pada perusahaan dagang</u>
6.	Perubahan judul yang disetujui Dosen sesuai materi bidang ilmu	..... ..... .....
7.	Catatan: ..... ..... .....	Disetujui Dosen <span style="float: right;">200</span> 
	Persetujuan Judul skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu	

**Perhatian:**

1. Formulir pengajuan ini harap dikembalikan kepada jurusan paling lambat satu minggu setelah disetujui kelompok dosen keahlian dengan dilampirkan proposal skripsi beserta persyaratan skripsi sesuai form S-1
2. Keterangan: \*) Coret yang tidak perlu  
\*\*) dilingkari a, b, c, ..... atau g sesuai bidang keahlian

**INSTITUT TEKNOLOGI NASIONAL**  
**Jl. Raya Karanglo, Km. 2**  
**MALANG**

---

Lampiran : 1 (satu) berkas  
**Pembimbing Skripsi**

Kepada : Yth. Bapak Joseph Dedy Irawan, ST.,MT.  
Dosen Institut Teknologi Nasional  
MALANG

Yang bertanda tangan di bawah ini :

Nama : Donny Prasetyo  
Nim : 06.12.524  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika


Dengan ini mengajukan permohonan, kiranya Bapak bersedia menjadi Dosen Pembimbing Utama / ~~Pendamping~~ \*), untuk penyusunan Skripsi dengan judul (proposal terlampir) :

**PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM KONSINYASI PADA PERUSAHAAN DAGANG**


Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Tugas Akhir Sarjana Teknik.  
Demikian permohonan kami dan atas kesediaan Bapak kami ucapkan terima kasih.

Malang, 10 April 2010

**Mengetahui**  
**Ketua Jurusan Teknik Elektro**

  
**Ir. F. Yudi Limpraptono, MT**  
**NIP. Y. 1039500274**

**Hormat Kami,**

  
**Donny Prasetyo**  
**06.12.524**

\*) coret yang tidak perlu

Form S-3a

**PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI**

Sesuai permohonan dari Mahasiswa :

Nama : Donny Prasetyo

Nim : 06.12.524

Semester : 8

Jurusan : Teknik Elektro S-1

Konsentrasi : Teknik Komputer & Informatika

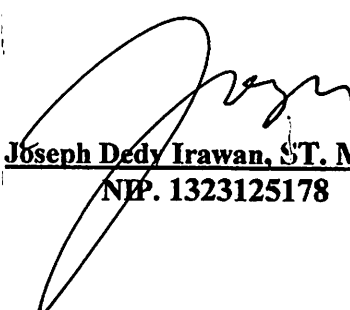
Dengan ini menyatakan bersedia / ~~tidak bersedia~~ \*) Membimbing skripsi dari Mahasiswa tersebut, dengan judul :

**PENGEMBANGAN SISTEM INFORMASI PENJUALAN DENGAN SISTEM  
KONSINYASI PADA PERUSAHAAN DAGANG**

Demikian Surat Pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang, 10 April 2010

**Kami yang Membuat Pernyataan,**

  
**Joseph Dedy Irawan, ST. MT**  
**NIP. 1323125178**

**Catatan :**

Setelah disetujui agar Formulir ini  
Diserahkan mahasiswa/I yang bersangkutan  
Kepada Jurusan untuk diproses lebih lanjut  
\*)Coret yang tidak perlu

Form S-3b



**BERITA ACARA SEMINAR PROPOSAL SKRIPSI  
 PROGRAM STUDI TEKNIK ELEKTRO S1**

ONSENTRASI      teknik komputer dan Informatika S-1

Nama Mahasiswa	Donny Prasetyo	NIM	06.12.829
Keterangan	Tanggal	Waktu	Tempat / Ruang
Pelaksanaan	21 April 2010		

Spesifikasi Judul (berilah tanda silang \*)

a. Sistem Tenaga Elektrik	e. Embedded System	<input checked="" type="checkbox"/>	Sistem Informasi
b. Konversi Energi	f. Antar Muka	<input type="checkbox"/>	Jaringan Komputer
c. Sistem Kendali	g. Elektronika Telekomunikasi	<input type="checkbox"/>	Web
d. Tegangan Tinggi	h. Elektronika Instrumentasi	<input type="checkbox"/>	Algoritma Cerdas



Judul Proposal yang diseminarkan Mahasiswa  
 Pengembangan Sistem Informasi penjualan dengan sistem konsinyasi pada perusahaan dagang

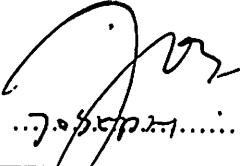
Perubahan Judul yang diusulkan oleh Kelompok Dosen Keahlian

Catatan :

Catatan :

Persetujuan Judul Skripsi

Disetujui, Dosen Keahlian I 	Disetujui, Dosen Keahlian II 
---	--

Mengetahui, Ketua Jurusan.  Ir. F. Yudi Limpraptono, MT NIP. Y. 1039500274	Disetujui, Calon Dosen Pembimbing	
	Pembimbing I  .....	Pembimbing II .....

angan :  
 ngkari a, b, c, ..... sesuai dengan bidang keahlian

## LAMPIRAN

### Sourcecode Form Barang :

```
unit UBarang;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
ComCtrls, Tabnotbk, StdCtrls, Db, Grids, DBGrids, DBTables, ExtCtrls,
Buttons, ZAbstractRODataset, ZAbstractDataset, ZDataset, jpeg, Menus;

type

TFBarang = class(TForm)

DataSource1: TDataSource;

PageControl1: TPageControl;

TabSheet1: TTabSheet;

TabSheet2: TTabSheet;

Panel1: TPanel;

Label7: TLabel;

Edit7: TEdit;

Button2: TButton;

DBGrid1: TDBGrid;

Label1: TLabel;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

Label5: TLabel;

Label6: TLabel;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;
```

**Fdit4: TFdit;**

**Edit5: TEdit;**

**Edit6: TEdit;**

**SpeedButton1: TSpeedButton;**

**Label8: TLabel;**

**Edit8: TEdit;**

**CheckBox1: TCheckBox;**

**Q1: TZQuery;**

**Panel2: TPanel;**

**Panel3: TPanel;**

**Panel4: TPanel;**

**Image1: TImage;**

**Button1: TBitBtn;**

**BitBtn1: TBitBtn;**

**BitBtn3: TBitBtn;**

**Image2: TImage;**

**PopupMenu1: TPopupMenu;**

**Delete1: TMenuItem;**

**ZQuery1: TZQuery;**

**procedure Edit1Exit(Sender: TObject);**

**procedure Edit1KeyPress(Sender: TObject; var Key: Char);**

**procedure Edit6KeyPress(Sender: TObject; var Key: Char);**

**procedure Edit6Exit(Sender: TObject);**

**procedure FormShow(Sender: TObject);**

**procedure Button2Click(Sender: TObject);**

**procedure Edit2KeyPress(Sender: TObject; var Key: Char);**

**procedure SpeedButton1Click(Sender: TObject);**

**procedure PageControl1Change(Sender: TObject);**

**procedure Edit8KeyPress(Sender: TObject; var Key: Char);**

**procedure Edit8Exit(Sender: TObject);**

**procedure Button1Click(Sender: TObject);**

```

procedure BitBtn3Click(Sender: TObject);

procedure Delete1Click(Sender: TObject);

private

  { Private declarations }

public

  { Public declarations }

end;

var

  FBarang: TFBarang;

implementation

uses UDM, UMain, UBarang_R;

  {$R *.DFM}

procedure TFBarang.Edit1Exit(Sender: TObject);

begin

  edit2.text := " ;

  edit3.text := '0' ;

  edit4.text := '0' ;

  edit5.text := '0' ;

  edit6.text := '0' ;

  if edit1.Text <> " then

    begin

      if DM.Barang.Locate('kode',Edit1.text,[loPartialKey]) then

        begin

          edit1.text := DM.BArang.fieldbyname('Kode').AsString ;

          edit2.text := DM.BArang.fieldbyname('Nama').AsString ;

          edit3.text := FormatFloat('#,0',DM.BArang.fieldbyname('Stok').AsFloat) ;

          edit4.text := FormatFloat('#,0',DM.BArang.fieldbyname('HPP').AsFloat) ;

          edit5.text := FormatFloat('#,0',DM.BArang.fieldbyname('Nilai').AsFloat) ;

          edit6.text := FormatFloat('#,0',DM.BArang.fieldbyname('Hrg_Jual').AsFloat) ;

          edit8.text := FormatFloat('#,0',DM.BArang.fieldbyname('Stok_Min').AsFloat) ;

```

```

    Button1.enabled := True ;

end

else if messagedlg('Kode barang tidak ditemukan, isi baru?', mtConfirmation, [mbYes, mbNo], 0) = mrYes then

begin

    edit2.SetFocus ;

    Button1.enabled := True ;

end

else

begin

    edit1.Text := '' ;

    Button1.enabled := False ;

end ;

end ;

end;

```

```

procedure TFBarang.Edit1KeyPress(Sender: TObject; var Key: Char);

```

```

begin

    if not(key in['0'..'9',#13,#8]) then key := #0 ;

    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);

end;

```

```

procedure TFBarang.Edit6KeyPress(Sender: TObject; var Key: Char);

```

```

begin

    if not(key in['0'..'9',#13,#8]) then key := #0 ;

    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);

end;

```

```

procedure TFBarang.Edit6Exit(Sender: TObject);

```

```

begin

    edit6.text := formatfloat('#,0',FMain.NR(edit6.text)) ;

end;

```

```

procedure TFBarang.FormShow(Sender: TObject);

```

```

begin

```

```

edit7.text := " ;
Button2click(sender) ;
edit1.text := " ;
edit2.text := " ;
edit3.text := '0' ;
edit4.text := '0' ;
edit5.text := '0' ;
edit6.text := '0' ;

end;

procedure TFBarang.Button2Click(Sender: TObject);
begin
    Q1.Close ;
    Q1.SQL.Clear ;
    Q1.SQL.Add('select * from barang Where Kode<>""') ;
    if CheckBox1.Checked then
        begin
            Q1.SQL.Add('And Stok_Min >=Stok') ;
        end;
    if edit7.text <> " then
        begin
            Q1.SQL.Add('And Nama like:xx ') ;
            Q1.parambyname('xx').AsString := '%' + edit7.text + '%' ;
        end;
    Q1.Open ;
end;

procedure TFBarang.Edit2KeyPress(Sender: TObject; var Key: Char);
begin
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFBarang.SpeedButton1Click(Sender: TObject);
begin

```

```

FBarang_R :=TFBarang_R.Create(Self);
FBarang_R.Q1.Close;
FBarang_R.Q1.SQL.Clear;
FBarang_R.Q1.SQL.Add('select * from barang' );
if edit7.text <> '' then
begin
FBarang_R.Q1.SQL.Add('where Nama like:xx ');
FBarang_R.Q1.parambyname('xx').AsString := '%' + edit7.text + '%';
end;
FBarang_R.Q1.Open;
FBarang_R.Preview;
FBarang_R.Q1.Close;
FBarang_R.Free;
end;
procedure TFBarang.PageControl1Change(Sender: TObject);
begin
Button2click(sender) ;
end;
procedure TFBarang.Edit8KeyPress(Sender: TObject; var Key: Char);
begin
if not(key in['0'..'9',#13,#8]) then key := #0 ;
if (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;
procedure TFBarang.Edit8Exit(Sender: TObject);
begin
edit8.text := formatfloat('#,0',FMain.NR(edit8.text)) ;
end;
procedure TFBarang.Button1Click(Sender: TObject);
begin //jhsghfjhsd\//sdhgjsdfg
if ((Edit1.text <> '') and (edit2.text <> '')) then
begin

```

```

if DM.Barang.Locate('kode',Edit1.text,[loPartialKey]) then
begin
    DM.Barang.edit ;
end
else
begin
    DM.Barang.Insert ;

    DM.Barang.FieldByName('Kode').AsString := Edit1.text ;
    DM.Barang.FieldByName('Stok').AsFloat := 0 ;
    DM.Barang.FieldByName('HPP').AsFloat := 0 ;
    DM.Barang.FieldByName('Nilai').AsFloat := 0 ;
end ;

DM.Barang.FieldByName('Nama').AsString := Edit2.text ;
DM.Barang.FieldByName('Hrg_Jual').AsFloat := FMain.NR(Edit6.text) ;
DM.Barang.FieldByName('Stok_min').AsFloat := FMain.NR(Edit8.text) ;
DM.Barang.Post ;
DM.Barang.ApplyUpdates ;
DM.Barang.CommitUpdates ;

edit1.text := " ;
edit2.text := " ;
edit3.text := '0' ;
edit4.text := '0' ;
edit5.text := '0' ;
edit6.text := '0' ;
edit1.setfocus ;
end
else
Begin
    MessageDlg('Kode barang dan nama barang harus diisi !', mtInformation, [mbOK], 0) ;
    Edit1.SetFocus;
end ;

```



```
/endksdjaksdklas  
end;  
procedure TFBarang.BitBtn3Click(Sender: TObject);  
begin  
    edit1.text := '' ;  
    edit2.text := '' ;  
    edit3.text := '0' ;  
    edit4.text := '0' ;  
    edit5.text := '0' ;  
    edit6.text := '0' ;  
    edit1.setfocus ;  
    Button1.Enabled := False ;  
end;  
procedure TFBarang.Delete1Click(Sender: TObject);  
begin  
    if DM.Barang.Locate('kode',DBGrid1.Fields[0].AsString,[loPartialKey]) then  
        begin  
            DM.Barang.Delete ;  
            DM.Barang.ApplyUpdates ;  
            DM.Barang.CommitUpdates ;  
            End;  
            Button2Click(Sender);  
        end;  
end;  
end.
```

## Sourcecode Form Customer :

```
unit UCustomer;

interface

uses

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ComCtrls, StdCtrls, Grids, DBGrids, ExtCtrls, Db, DBTables, Buttons,
  ZAbstractRODataset, ZAbstractDataset, ZDataset, jpeg, Menus;

type

  TFCustomer = class(TForm)

    PageControl1: TPageControl;

    TabSheet1: TTabSheet;

    TabSheet2: TTabSheet;

    Label1: TLabel;

    Label2: TLabel;

    Label3: TLabel;

    Label4: TLabel;

    Label6: TLabel;

    Label7: TLabel;

    Edit1: TEdit;

    Edit2: TEdit;

    Edit3: TEdit;

    Edit4: TEdit;

    Edit6: TEdit;

    Edit7: TEdit;

    Panel1: TPanel;

    Label10: TLabel;

    Edit9: TEdit;

    Button2: TButton;

    DBGrid1: TDBGrid;

    DataSource1: TDataSource;
```

```
SpeedButton1: TSpeedButton;

Q1: TZQuery;

Panel2: TPanel;

Panel3: TPanel;

Panel4: TPanel;

Image1: TImage;

Button1: TBitBtn;

BitBtn2: TBitBtn;

BitBtn3: TBitBtn;

Image2: TImage;

PopupMenu1: TPopupMenu;

Delete1: TMenuItem;

procedure FormShow(Sender: TObject);

procedure ComboBox1KeyPress(Sender: TObject; var Key: Char);

procedure Edit2KeyPress(Sender: TObject; var Key: Char);

procedure Edit1KeyPress(Sender: TObject; var Key: Char);

procedure PageControl1Change(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure SpeedButton1Click(Sender: TObject);

procedure Edit1Exit(Sender: TObject);

procedure BitBtn3Click(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure Delete1Click(Sender: TObject);

private
  { Private declarations }

public
  { Public declarations }

end;

var
  FCustomer: TFCustomer;

implementation
```

```

uses UDM, UMain, UCustomer_R;

{$R *.DFM}

procedure TFCustomer.FormShow(Sender: TObject);

begin
    Button2click(sender) ;

end;

procedure TFCustomer.ComboBox1KeyPress(Sender: TObject; var Key: Char);

begin
If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0) Else Key:=#0;

end;

procedure TFCustomer.Edit2KeyPress(Sender: TObject; var Key: Char);

begin
If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);

end;

procedure TFCustomer.Edit1KeyPress(Sender: TObject; var Key: Char);

begin
    if not(key in['0'..'9',#13,#8]) then key := #0 ;

    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);

end;

procedure TFCustomer.PageControl1Change(Sender: TObject);

begin
    Button2click(sender) ;

end;

procedure TFCustomer.Button2Click(Sender: TObject);

begin
    Q1.Close ;

    Q1.SQL.Clear ;

    Q1.SQL.Add('select * from Customer') ;

    if edit9.text <> " then

begin
    Q1.SQL.Add('where Nama like:xx ');

```

```

    Q1.parambyname('xx').AsString := '%' + edit9.text + '%';
end;

Q1.Open ;

end;

procedure TFCustomer.SpeedButton1Click(Sender: TObject);
begin
    FCustomer_R :=TFCustomer_R.Create(Self);

    FCustomer_R.Q1.Close;

    FCustomer_R.Q1.Sql.Clear;

    FCustomer_R.Q1.SQL.Add('select * from CUsomer');

    if edit9.text <> " then
        begin
            FCustomer_R.Q1.SQL.Add('where Nama like:xx ');

            FCustomer_R.Q1.parambyname('xx').AsString := '%' + edit9.text + '%';

            end;

        FCustomer_R.Q1.Open ;

        FCustomer_R.Preview;

        FCustomer_R.Q1.Close;

        FCustomer_R.Free;

end;

procedure TFCustomer.Edit1Exit(Sender: TObject);
begin
    edit2.Text := " ;

    edit3.Text := " ;

    edit4.Text := " ;

    edit6.Text := " ;

    edit7.Text := " ;

    if edit1.Text <> " then
        begin

            If Dm.Customer.Locate('kode',Edit1.text,[loPartialKey]) Then

        begin

```

```

edit1.Text := DM.Customer.FieldByName('Kode').AsString ;
edit2.Text := DM.Customer.FieldByName('Nama').AsString ;
edit3.Text := DM.Customer.FieldByName('Alamat').AsString ;
edit4.Text := DM.Customer.FieldByName('Telp').AsString ;
edit6.Text := DM.Customer.FieldByName('Nama_Kontak').AsString ;
edit7.Text := DM.Customer.FieldByName('HP').AsString ;

Button1.enabled := True ;

end

else if messagedlg('Kode customer tidak ditemukan, isi baru?', mtConfirmation, [mbYes, mbNo], 0) = mrYes then
begin
    edit2.SetFocus ;

    Button1.enabled := True ;

end

else
begin
    edit1.Text := '' ;

    Button1.enabled := False ;

end ;

end ;

end;

procedure TFCustomer.BitBtn3Click(Sender: TObject);
begin
    edit2.Text := '' ;
    edit3.Text := '' ;
    edit4.Text := '' ;
    edit6.Text := '' ;
    edit7.Text := '' ;

    button1.Enabled := false ;

end;

procedure TFCustomer.Button1Click(Sender: TObject);
begin

```

```

if ((Edit1.text <> "") and (edit2.text <> "")) then
begin
    If Dm.Customer.Locate('kode',Edit1.text,[loPartialKey]) Then
DM.Customer.edit else DM.Customer.Insert ;
DM.Customer.FieldByName('Kode').AsString := Edit1.text ;
DM.Customer.FieldByName('Nama').AsString := Edit2.text ;
DM.Customer.FieldByName('Alamat').AsString := Edit3.text ;
DM.Customer.FieldByName('Telp').AsString := Edit4.text ;
DM.Customer.FieldByName('Nama_Kontak').AsString := Edit6.text ;
DM.Customer.FieldByName('HP').AsString := Edit7.text ;
DM.Customer.Post ;
DM.Customer.ApplyUpdates ;
DM.Customer.CommitUpdates ;
edit1.text := " ;
edit2.text := " ;
edit3.text := " ;
edit4.text := " ;
edit6.Text := " ;
edit7.Text := " ;
edit1.setfocus ;
end
else
Begin
    Messagedlg('Kode dan nama customer harus diisi !', mtInformation, [mbOK], 0) ;
    edit1.setfocus ;
End;
end;

procedure TFCustomer.Delete1Click(Sender: TObject);
begin
    if DM.Customer.Locate('kode',DBGrid1.Fields[0].AsString,[loPartialKey]) then
        begin

```

**DM.Customer.Delete ;**

**DM.Customer.ApplyUpdates ;**

**DM.Customer.CommitUpdates ;**

**End;**

**Button2Click(Sender);**

**end;**

**end.**



## Sourcecode Form Produsen :

```
unit UProdusen;

interface

uses

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ComCtrls, StdCtrls, Grids, DBGrids, ExtCtrls, Db, DBTables, Buttons, jpeg,
  ZAbstractRODataset, ZAbstractDataset, ZDataset, Menus;

type

  TFFProdusen = class(TForm)

    PageControl1: TPageControl;

    TabSheet1: TTabSheet;

    Label1: TLabel;

    TabSheet2: TTabSheet;

    Label2: TLabel;

    Label3: TLabel;

    Label4: TLabel;

    Label5: TLabel;

    Label6: TLabel;

    Edit1: TEdit;

    Edit2: TEdit;

    Edit3: TEdit;

    Edit4: TEdit;

    Edit5: TEdit;

    Edit6: TEdit;

    Panel1: TPanel;

    Label7: TLabel;

    Edit7: TEdit;

    Button2: TButton;

    DBGrid1: TDBGrid;

    DataSource1: TDataSource;
```

```
SpeedButton1: TSpeedButton;

Panel2: TPanel;

Panel3: TPanel;

Panel4: TPanel;

Image2: TImage;

Image1: TImage;

Button1: TBitBtn;

BitBtn2: TBitBtn;

BitBtn3: TBitBtn;

Q1: TZQuery;

PopupMenu1: TPopupMenu;

Delete1: TMenuItem;

procedure Edit1KeyPress(Sender: TObject; var Key: Char);

procedure Edit1Exit(Sender: TObject);

procedure FormShow(Sender: TObject);

procedure Button2Click(Sender: TObject);

procedure PageControl1Change(Sender: TObject);

procedure Edit7KeyPress(Sender: TObject; var Key: Char);

procedure Edit2KeyPress(Sender: TObject; var Key: Char);

procedure SpeedButton1Click(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure BitBtn3Click(Sender: TObject);

procedure Delete1Click(Sender: TObject);

procedure Edit6KeyPress(Sender: TObject; var Key: Char);

private
{ Private declarations }

public
{ Public declarations }

end;

var
FProdusen: TFProdusen;
```

implementation

uses UDM, UProdusen\_R;

{\$R \*.DFM}

procedure TFProdusen.Edit1KeyPress(Sender: TObject; var Key: Char);

begin

if not(key in['0'..'9','.',',','#13,#8]) then key := #0 ;

If (Key=Chr(VK\_return)) Then SendMessage(Handle,WM\_NEXTDLGCTL,0,0);

end;

procedure TFProdusen.Edit1Exit(Sender: TObject);

begin

edit2.text := " ;

edit3.text := " ;

edit4.text := " ;

edit5.text := " ;

edit6.text := " ;

if edit1.Text <> " then

begin

If Dm.Produsen.Locate('kode',Edit1.text,[loPartialKey]) Then

begin

edit1.text := DM.Produsen.fieldbyname('Kode').AsString ;

edit2.text := DM.Produsen.fieldbyname('Nama\_PT').AsString ;

edit3.text := DM.Produsen.fieldbyname('Alamat').AsString ;

edit4.text := DM.Produsen.fieldbyname('Telp').AsString ;

edit5.text := DM.Produsen.fieldbyname('Nama\_Kontak').AsString ;

edit6.text := DM.Produsen.fieldbyname('HP').AsString ;

Button1.enabled := True ;

end

else if messagedlg('Kode produsen tidak ditemukan, isi baru?', mtConfirmation, [mbYes, mbNo], 0) = mrYes then

begin

edit2.SetFocus ;

Button1.enabled := True ;

```

    end
else
begin
    edit1.Text := " ;
    Button1.enabled := False ;
end ;
end ;
end;

procedure TFProdusen.FormShow(Sender: TObject);
begin
    edit7.text := " ;
    Button2click(sender) ;
end;

procedure TFProdusen.Button2Click(Sender: TObject);
begin
    Q1.Close ;
    Q1.SQL.Clear ;
    Q1.SQL.Add('select * from produsen') ;
    if edit7.text <> " then
begin
    Q1.SQL.Add('where Nama_PT like:xx ') ;
    Q1.parambyname('xx').AsString := '%' + edit7.text + '%' ;
end;
    Q1.Open ;
end;

procedure TFProdusen.PageControl1Change(Sender: TObject);
begin
    Button2click(sender) ;
end;

procedure TFProdusen.Edit7KeyPress(Sender: TObject; var Key: Char);
begin

```

```

    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFProdusen.Edit2KeyPress(Sender: TObject; var Key: Char);
begin
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
End;

procedure TFProdusen.SpeedButton1Click(Sender: TObject);
begin
    FProdusen_R :=TFProdusen_R.Create(Self);
    FProdusen_R.Q1.Close;
    FProdusen_R.Q1.Sql.Clear;
    FProdusen_R.Q1.SQL.Add('select * from Produsen');
    if edit7.text <> '' then
        begin
            FProdusen_R.Q1.SQL.Add('where Nama_PT like:xx ');
            FProdusen_R.Q1.parambyname('xx').AsString := '%' + edit7.text + '%';
        end;
    FProdusen_R.Q1.Open ;
    FProdusen_R.Preview;
    FProdusen_R.Q1.Close;
    FProdusen_R.Free;
end;

procedure TFProdusen.Button1Click(Sender: TObject);
begin
    if ((Edit1.text <> '') and (edit2.text <> '')) then
        begin
            If Dm.Produsen.Locate('kode',Edit1.text,[loPartialKey]) Then
                Dm.Produsen.edit else Dm.Produsen.Insert ;
            Dm.Produsen.FieldByName('Kode').AsString := Edit1.text ;
            Dm.Produsen.FieldByName('Nama_PT').AsString := Edit2.text ;
            Dm.Produsen.FieldByName('Alamat').AsString := Edit3.text ;
        end;
    end;
end;

```

```

DM.Produsen.FieldByName('Telp').AsString := Edit4.text ;
DM.Produsen.FieldByName('Nama_Kontak').AsString := Edit5.text ;
DM.Produsen.FieldByName('HP').AsString := Edit6.text ;
DM.Produsen.Post ;
DM.Produsen.ApplyUpdates ;
DM.Produsen.CommitUpdates ;

edit1.text := " ;
edit2.text := " ;
edit3.text := " ;
edit4.text := " ;
edit5.text := " ;
edit6.text := " ;
edit1.setfocus ;

end
else
Begin
MessageDlg('Kode Suplier dan nama Suplier harus diisi !', mtInformation, [mbOK], 0) ;
Edit1.SetFocus;
End;

end;

procedure TFProduken.BitBtn3Click(Sender: TObject);
begin
edit1.text := " ;
edit2.text := " ;
edit3.text := " ;
edit4.text := " ;
edit5.text := " ;
edit6.text := " ;
edit1.setfocus ;

Button1.Enabled := False ;

end;

```

```
procedure TFProdusen.Delete1Click(Sender: TObject);
begin
  if DM.Produsen.Locate('kode',DBGrid1.Fields[0].AsString,[loPartialKey]) then
    begin
      DM.Produsen.Delete ;
      DM.Produsen.ApplyUpdates ;
      DM.Produsen.CommitUpdates ;
      End;
      Button2Click(Sender);
    end;

  procedure TFProdusen.Edit6KeyPress(Sender: TObject; var Key: Char);
  begin
    if not(key in['0'..'9',#13,#8]) then key := #0 ;

    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
  end;
end.
```

Sourcecode Form Pembelian :

```
unit UBeli;

interface

uses

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ComCtrls, StdCtrls, Grids, ExtCtrls, Db, DBTables, DBGrids, Buttons,
  Menus, jpeg, ZAbstractRODataset, ZAbstractDataset, ZDataset;

type

TFBeli = class(TForm)
  PageControl1: TPageControl;
  TabSheet1: TTabSheet;
  TabSheet2: TTabSheet;
  TabSheet3: TTabSheet;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Panel1: TPanel;
  DateTimePicker1: TDateTimePicker;
  Edit1: TEdit;
  Panel2: TPanel;
  S: TStringGrid;
  Panel3: TPanel;
  Label5: TLabel;
  DBGrid1: TDBGrid;
  Panel4: TPanel;
  Button2: TButton;
  DataSource1: TDataSource;
  DateTimePicker2: TDateTimePicker;
  DateTimePicker3: TDateTimePicker;
```



```
CheckBox1: TCheckBox;
CheckBox2: TCheckBox;
DBGrid2: TDBGrid;
DataSource2: TDataSource;
DataSource3: TDataSource;
SpeedButton1: TSpeedButton;
SpeedButton2: TSpeedButton;
SpeedButton3: TSpeedButton;
PopupMenu1: TPopupMenu;
HapusItem1: TMenuItem;
Label16: TLabel;
RadioButton1: TRadioButton;
Edit3: TEdit;
Panel5: TPanel;
Panel6: TPanel;
Panel7: TPanel;
Image2: TImage;
button1: TBitBtn;
BitBtn2: TBitBtn;
Q1: TZQuery;
Q2: TZQuery;
procedure FormShow(Sender: TObject);
procedure Edit1Exit(Sender: TObject);
procedure SKeyPress(Sender: TObject; var Key: Char);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure CheckBox1Click(Sender: TObject);
procedure CheckBox2Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure SpeedButton1Click(Sender: TObject);
procedure SpeedButton2Click(Sender: TObject);
procedure SExit(Sender: TObject);
```

```

procedure PageControl1Change(Sender: TObject);
procedure DateTimePicker2Change(Sender: TObject);
procedure DateTimePicker3Change(Sender: TObject);
procedure SpeedButton3Click(Sender: TObject);
procedure DateTimePicker1KeyPress(Sender: TObject; var Key: Char);
procedure Button3Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure HapusItemDel1Click(Sender: TObject);
procedure HapusItem1Click(Sender: TObject);
procedure Panel4Click(Sender: TObject);
procedure RadioButton1Click(Sender: TObject);
procedure button1Click(Sender: TObject);

private

  { Private declarations }

  procedure SubTotal ;

public

  { Public declarations }

end;

var

  FBeli: TFBel;

implementation

uses UDM, UMain, UProdusen, UBarang, Ubeli_R;

  {$R *.DFM}

procedure TFBel.FormShow(Sender: TObject);

  var i : Integer ;

begin

  Panel1.Caption := Inttostr(DM.Mbeli.RecordCount+1) ;

  DateTimePicker1.DateTime := Date;

  DateTimePicker2.DateTime := Date;

  DateTimePicker3.DateTime := Date;

  S.Cells[0,0] := 'No.' ;

```

S.Cells[1,0] := 'Kode Barang' ;

S.Cells[2,0] := 'Nama Barang' ;

S.Cells[3,0] := 'Jumlah' ;

S.Cells[4,0] := 'Harga' ;

S.Cells[5,0] := 'Sub Total' ;

S.ColWidths[0] := 30 ;

S.ColWidths[1] := 60 ;

S.ColWidths[2] := 200 ;

S.ColWidths[3] := 75 ;

S.ColWidths[4] := 75 ;

S.ColWidths[5] := 75 ;

for i := 1 to S.RowCount -1 do

begin

S.Cells[0,i]:= IntToStr(i) ;

end ;

Edit1.Text := " ;

Panel2.Caption := " ;

Panel3.Caption := '0' ;

Panel1.Caption := Inttostr(DM.Mbeli.RecordCount+1) ;

for i := 1 to S.RowCount -1 do

begin

S.Cells[0,i]:= IntToStr(i) ;

S.Cells[1,i]:= " ;

S.Cells[2,i]:= " ;

S.Cells[3,i]:= " ;

S.Cells[4,i]:= " ;

S.Cells[5,i]:= '0' ;

end ;

DateTimePicker1.DateTime := Date ;

DateTimePicker2.DateTime := Date ;

```

    DateTimePicker3.DateTime := Date ;
end;
procedure TFBeli.Edit1Exit(Sender: TObject);
begin
    panel2.caption := " ;
    if edit1.Text <> " then
        begin
            If Dm.Produsen.Locate('kode',Edit1.text,[loPartialKey]) Then
                begin
                    panel2.Caption := dm.Produsen.fieldbyname('Nama_PT').AsString ;
                    subtotal ;
                end
            else
                begin
                    subtotal ;
                    messagedlg('Kode Suplier tidak ditemukan', mtInformation, [mbOK],0) ;
                    edit1.setfocus ;
                    exit ;
                end ;
            end ;
        end;
procedure TFBeli.SKeyPress(Sender: TObject; var Key: Char);
var j, baris : Integer ;
begin
    if S.Col in [1,3,4] then
        begin
            if not (key in ['0'..'9', #8, #13]) then key := #0 ;
            end ;
            if ((S.col = 1) and (key = #13)) then
                begin
                    baris := S.Row ;

```

```

for j := 1 to S.RowCount-1 do
begin
  if j <> baris then
  begin
    if S.Cells[1,j] = S.Cells[1,baris] then
      begin
        S.Cells[1,baris] := " ;
      end ;
    end ;
  end ;
end ;
if Dm.Barang.Locate('kode',S.Cells[1,S.Row],[loPartialKey]) Then
begin
  S.Cells[2,S.Row] := dm.barang.fieldbyname('Nama').AsString ;
end
else
begin
  messageDlg('Kode Barang tidak ditemukan atau sudah ada', mtInformation, [mbOK], 0);
  S.Col := 1 ;
  exit ;
end ;
end ;
if S.Col in [2,5] then key := #0 ;
if ((S.Col in [3,4]) and (key = #13)) then
begin
  S.Cells[3,S.Row] := formatFloat('#,0', FMain.NR(S.Cells[3,S.Row])) ;
  S.Cells[4,S.Row] := formatFloat('#,0', FMain.NR(S.Cells[4,S.Row])) ;
end ;
if key = #13 then
  Begin
  subtotal ;
  If S.Col=1 Then S.Col :=3 Else

```

```

Begin
  If S.Col=3 Then S.Col :=4 Else
    Begin
      If S.Col=4 Then
        Begin
          S.Col :=1;
          S.ROw:=S.ROw+1;
        End;
      End;
    End;
  End;
End;
end;
procedure TFBeli.SubTotal ;
var i : Integer ;
    total : Double ;
    cek : boolean ;
begin
  total := 0 ;
  cek := true ;

  for i := 1 to S.RowCount-1 do
    begin
      S.Cells[5,i] := formatfloat('#,0', FMain.NR(S.Cells[3,i])*FMain.NR(S.Cells[4,i])) ;
      total := total + FMain.NR(S.Cells[5,i]) ;
      If ((fmain.NR(s.cells[5,i]) <=0) And (s.cells[2,i]<>"")) Then
        cek := false ;
      end ;
      panel3.caption := formatfloat('#,0', total) ;
      if ((panel2.caption <> "") and (cek = true) and (total > 0)) then
        button1.enabled := true
      else button1.enabled := false ;
    end ;
  end ;
end ;

```

```
end ;

procedure TFBeli.Edit1KeyPress(Sender: TObject; var Key: Char);

begin
    if not(key in ['0'..'9',#13,#8]) then key := #0 ;
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFBeli.CheckBox1Click(Sender: TObject);

begin
    if checkbox1.Checked then
        begin
            DateTimePicker2.Enabled := True ;
            DateTimePicker2.Color := clWhite ;
        end
    else
        begin
            DateTimePicker2.Enabled := False ;
            DateTimePicker2.Color := clMenu ;
        end ;
    end;

procedure TFBeli.CheckBox2Click(Sender: TObject);

begin
    if checkbox2.Checked then
        begin
            DateTimePicker3.Enabled := True ;
            DateTimePicker3.Color := clWhite ;
        end
    else
        begin
            DateTimePicker3.Enabled := False ;
            DateTimePicker3.Color := clMenu ;
        end ;
    end ;
```

```

end;

procedure TFBeli.Button2Click(Sender: TObject);
begin
    Q1.Close ;
    Q1.SQL.Clear ;
    Q1.SQL.Add('select * from MBeli where Nomer > 0') ;
    if CheckBox1.Checked then
        begin
            Q1.SQL.Add('and Tanggal >=:xx ');
            Q1.parambyname('xx').AsDateTime := DateTimePicker2.Date ;
        end;
    if CheckBox2.Checked then
        begin
            Q1.SQL.Add('and Tanggal <=:yy ');
            Q1.parambyname('yy').AsDateTime := DateTimePicker3.Date ;
        end;
    if ((radiobutton1.Checked) and (edit3.Text <> '')) then
        begin
            Q1.SQL.Add(' and Nama like:xx ');
            Q1.parambyname('xx').AsString := '%' + edit3.text + '%';
        end ;
    Q1.Open ;
    Q2.Close;
    Q2.Open ;
end;

```

```

procedure TFBeli.SpeedButton1Click(Sender: TObject);
begin
    FProdusen.ShowModal ;
    Edit1.text:=FProdusen.DBGrid1.Fields[0].AsString;
    Edit1Exit(Sender);

```



```

end;

procedure TFBeli.SpeedButton2Click(Sender: TObject);
begin
    FBarang.ShowModal ;
    S.Cells[1, S.Row]:=FBarang.DBGrid1.Fields[0].AsString;
end;

procedure TFBeli.SExit(Sender: TObject);
begin
    subtotal ;
end;

procedure TFBeli.PageControl1Change(Sender: TObject);
begin
    Q2.Open;
    If PageControl1.ActivePageIndex=1 Then
        Button2Click(Sender);
end;

procedure TFBeli.DateTimePicker2Change(Sender: TObject);
begin
    Button2Click(Sender);
end;

procedure TFBeli.DateTimePicker3Change(Sender: TObject);
begin
    Button2Click(Sender);
end;

procedure TFBeli.SpeedButton3Click(Sender: TObject);
begin
    FBeli_R :=TFBeli_R.Create(Self);
    FBeli_R.Q1.Close;
    FBeli_R.Q1.Sql.Clear;
    FBeli_R.q1.sql.add('select * from mBeli where Nomer=:XX');
    FBeli_R.q1.ParamByName('xx').asString:=DbGrid1.Fields[0].AsString;

```

```

    FBeli_R.Q1.Open;
    FBeli_R.Q2.Open;
    FBeli_R.Preview;
    FBeli_R.Q2.Close;
    FBeli_R.Q1.Close;
    FBeli_R.Free;
end;

procedure TFBeli.DateTimePicker1KeyPress(Sender: TObject; var Key: Char);
begin
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFBeli.Button3Click(Sender: TObject);
var i : integer ;
begin
    Edit1.Text := " ;
    Panel2.Caption := " ;
    Panel3.Caption := '0' ;
    Panel1.Caption := Inttostr(DM.Mbeli.RecordCount+1) ;
    for i := 1 to S.RowCount -1 do
    begin
        S.Cells[0,i]:= IntToStr(i) ;
        S.Cells[1,i]:= " ;
        S.Cells[2,i]:= " ;
        S.Cells[3,i]:= " ;
        S.Cells[4,i]:= " ;
        S.Cells[5,i]:= '0' ;
    end ;
    button1.Enabled := false ;
end;

procedure TFBeli.FormClose(Sender: TObject; var Action: TCloseAction);
var i : integer ;

```

```

begin
    Edit1.Text := "";
    Panel2.Caption := "";
    Panel3.Caption := '0' ;
    Panel1.Caption := Inttostr(DM.Mbeli.RecordCount+1) ;
    for i := 1 to S.RowCount -1 do
        begin
            S.Cells[0,i]:= IntToStr(i) ;
            S.Cells[1,i]:= " ;
            S.Cells[2,i]:= " ;
            S.Cells[3,i]:= " ;
            S.Cells[4,i]:= " ;
            S.Cells[5,i]:= '0' ;
        end ;
        button1.Enabled := false ;
    end;
    procedure TFBeli.HapusItemDel1Click(Sender: TObject);
    Var
        I: Integer;
    begin
        With S Do
            Begin
                Rows[Row].Clear;
                Cells[0,Row]:= IntToStr(Row)+'.';
                For i:= Row To RowCount - 1 Do
                    Begin
                        IF (i <> RowCount - 1) Then
                            Begin
                                Cells[1,i]:= Cells[1,i+1];
                                Cells[2,i]:= Cells[2,i+1];
                                Cells[3,i]:= Cells[3,i+1];
                            end
                        end
                    end
                end
            end
        end
    end;

```

```

        Cells[4,i]:= Cells[4,i+1];
        Cells[5,i]:= Cells[5,i+1];
    End;
End;
SubTotal;
End;
end;
procedure TFBeli.HapusItem1Click(Sender: TObject);
Var
    I: Integer;
begin
    With S Do
        Begin
            Rows[Row].Clear;
            Cells[0,Row]:= IntToStr(Row)+'.';
            For i:= Row To RowCount - 1 Do
                Begin
                    IF (i <> RowCount - 1) Then
                        Begin
                            Cells[1,i]:= Cells[1,i+1];
                            Cells[2,i]:= Cells[2,i+1];
                            Cells[3,i]:= Cells[3,i+1];
                            Cells[4,i]:= Cells[4,i+1];
                            Cells[5,i]:= Cells[5,i+1];
                        End;
                    End;
                SubTotal;
            End;
        end;
    procedure TFBeli.Panel4Click(Sender: TObject);
    begin

```

```
radiobutton1.Checked := false ;  
edit3.Enabled := false ;  
edit3.Text := " ;  
edit3.Color := clBtnFace ;  
end;
```

```
procedure TFBeli.RadioButton1Click(Sender: TObject);
```

```
begin
```

```
  if radiobutton1.Checked then
```

```
  begin
```

```
    edit3.Enabled := true ;
```

```
    edit3.Color := clWhite ;
```

```
  end
```

```
  else
```

```
  begin
```

```
    edit3.Enabled := false ;
```

```
    edit3.Color := clBtnFace ;
```

```
  end ;
```

```
end;
```

```
procedure TFBeli.button1Click(Sender: TObject);
```

```
var i, Nomer : Integer ;
```

```
begin
```

```
  DM.MBeli.Insert ;
```

```
  DM.Mbeli.FieldByName('Nomer').AsInteger := DM.MBeli.RecordCount + 1 ;
```

```
  Nomer := DM.Mbeli.FieldByName('Nomer').AsInteger ;
```

```
  DM.Mbeli.FieldByName('Tanggal').AsDateTime := DateTimePicker1.Date ;
```

```
  DM.Mbeli.FieldByName('Kode').AsString := Edit1.Text ;
```

```
  DM.Mbeli.FieldByName('Nama').AsString := Panel2.Caption ;
```

```
  DM.Mbeli.FieldByName('Total_Beli').AsFloat := FMain.NR(Panel3.Caption) ;
```

```
  DM.Mbeli.Post ;
```

```
  DM.Mbeli.ApplyUpdates ;
```

```

DM.Mbeli.CommitUpdates ;

DM.Mbeli.Refresh ;

for i := 1 to S.RowCount - 1 do

begin

if ((S.Cells[1,i] <> "") and (S.Cells[2,i] <> "") and (FMain.NR(S.Cells[5,i]) > 0)) then

begin

    DM.Dbeli.Insert ;

    DM.Dbeli.FieldByName('Nomer').AsInteger := DM.Dbeli.RecordCount + 1 ;

    DM.Dbeli.FieldByName('No_Mbeli').AsInteger := Nomer ;

    DM.Dbeli.FieldByName('Kode').AsString := S.Cells[1,i] ;

    DM.Dbeli.FieldByName('Nama').AsString := S.Cells[2,i] ;

    DM.Dbeli.FieldByName('Jml_Beli').AsFloat := FMain.NR(S.Cells[3,i]) ;

    DM.Dbeli.FieldByName('Hrg_Beli').AsFloat := FMain.NR(S.Cells[4,i]) ;

    DM.Dbeli.FieldByName('Sub_Total').AsFloat := FMain.NR(S.Cells[5,i]) ;

    DM.Dbeli.Post ;

    DM.Dbeli.ApplyUpdates ;

    DM.Dbeli.CommitUpdates ;

    DM.Dbeli.Refresh ;

    If Dm.Barang.Locate('kode',S.Cells[1,i],[loPartialKey]) Then

begin

DM.Barang.edit ;

DM.Barang.FieldByName('Stok').AsFloat := DM.Barang.FieldByName('Stok').AsFloat + FMain.NR(S.Cells[3,i]) ;

DM.Barang.FieldByName('Nilai').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat + FMain.NR(S.Cells[5,i]) ;

if ((DM.Barang.FieldByName('Stok').AsFloat > 0) and (DM.Barang.FieldByName('Nilai').AsFloat > 0)) then

begin

    DM.Barang.FieldByName('HPP').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat /
DM.Barang.FieldByName('Stok').AsFloat ;

end

else

begin

    DM.Barang.FieldByName('Stok').AsFloat := 0 ;

```

```
DM.Barang.FieldByName('HPP').AsFloat := 0 ;
DM.Barang.FieldByName('Nilai').AsFloat := 0 ;

end ;

DM.Barang.Post ;

DM.Barang.ApplyUpdates ;

DM.Barang.CommitUpdates ;

DM.Barang.Refresh ;

end ;

end ;

end ;

Edit1.Text := '' ;

Panel2.Caption := '' ;

Panel3.Caption := '0' ;

Panel1.Caption := Inttostr(DM.Mbeli.RecordCount+1) ;

for i := 1 to S.RowCount -1 do
begin
S.Cells[0,i]:= IntToStr(i) ;
S.Cells[1,i]:= '' ;
S.Cells[2,i]:= '' ;
S.Cells[3,i]:= '' ;
S.Cells[4,i]:= '' ;
S.Cells[5,i]:= '0' ;
end ;

Button1.Enabled:=False;

end;

end.
```

## Sourcecode Form Penjualan :

```
unit UJual;

interface

uses

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ComCtrls, StdCtrls, Grids, ExtCtrls, Db, DBTables, DBGrids, Menus,
  Buttons, jpeg, ZAbstractRODataset, ZAbstractDataset, ZDataset;

type

TFJual = class(TForm)

  PageControl1: TPageControl;

  TabSheet1: TTabSheet;

  TabSheet2: TTabSheet;

  TabSheet3: TTabSheet;

  Label1: TLabel;

  Label2: TLabel;

  Label3: TLabel;

  Label4: TLabel;

  Panel1: TPanel;

  Edit1: TEdit;

  Panel2: TPanel;

  S: TStringGrid;

  Panel3: TPanel;

  Label5: TLabel;

  DBGrid1: TDBGrid;

  Panel4: TPanel;

  Button2: TButton;

  DataSource1: TDataSource;

  DateTimePicker2: TDateTimePicker;

  DateTimePicker3: TDateTimePicker;

  CheckBox1: TCheckBox;
```



**CheckBox2: TCheckBox;**  
**DBGrid2: TDBGrid;**  
**DataSource2: TDataSource;**  
**DataSource3: TDataSource;**  
**PopupMenu1: TPopupMenu;**  
**Hapus1: TMenuItem;**  
**Label12: TLabel;**  
**Panel5: TPanel;**  
**Edit2: TEdit;**  
**Panel6: TPanel;**  
**DateTimePicker4: TDateTimePicker;**  
**Label13: TLabel;**  
**Label14: TLabel;**  
**Label15: TLabel;**  
**DateTimePicker1: TDateTimePicker;**  
**SpeedButton1: TSpeedButton;**  
**SpeedButton2: TSpeedButton;**  
**SpeedButton3: TSpeedButton;**  
**RadioButton1: TRadioButton;**  
**Label16: TLabel;**  
**Edit3: TEdit;**  
**Panel7: TPanel;**  
**Panel8: TPanel;**  
**Panel9: TPanel;**  
**Image1: TImage;**  
**button1: TBitBtn;**  
**BitBtn2: TBitBtn;**  
**BitBtn3: TBitBtn;**  
**Q1: TZQuery;**  
**Q2: TZQuery;**  
**procedure FormShow(Sender: TObject);**

```

procedure Edit1Exit(Sender: TObject);
procedure SKeyPress(Sender: TObject; var Key: Char);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure CheckBox1Click(Sender: TObject);
procedure CheckBox2Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Hapus1Click(Sender: TObject);
procedure Edit2Exit(Sender: TObject);
procedure DateTimePicker1KeyPress(Sender: TObject; var Key: Char);
procedure PageControl1Change(Sender: TObject);
procedure Edit2KeyPress(Sender: TObject; var Key: Char);
procedure SpeedButton2Click(Sender: TObject);
procedure SExit(Sender: TObject);
procedure SpeedButton3Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure RadioButton1Click(Sender: TObject);
procedure Panel4Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure button1Click(Sender: TObject);
procedure SpeedButton1Click(Sender: TObject);

private
  { Private declarations }
  procedure SubTotal ;

public
  { Public declarations }

end;

var
  FJual: TFJual;

implementation

uses UDM, UMain, UBarang, UPenjualan_R, UCustomer;
{$R *.DFM}

```

```

procedure TFJual.FormShow(Sender: TObject);

var i : Integer ;

begin

    Panel1.Caption := InttoStr(DM.MJual.RecordCount+1) ;

    DateTimePicker1.DateTime := Date ;

    DateTimePicker4.DateTime := Date + 30 ;

    edit3.Text := '' ;

    S.Cells[0,0] := 'No.' ;

    S.Cells[1,0] := 'Kode Barang' ;

    S.Cells[2,0] := 'Nama Barang' ;

    S.Cells[3,0] := 'Stok' ;

    S.Cells[4,0] := 'Jumlah' ;

    S.Cells[5,0] := 'Harga' ;

    S.Cells[6,0] := 'Sub Total' ;

    S.ColWidths[0] := 30 ;

    S.ColWidths[1] := 60 ;

    S.ColWidths[2] := 200 ;

    S.ColWidths[3] := 75 ;

    S.ColWidths[4] := 75 ;

    S.ColWidths[5] := 75 ;

    S.ColWidths[6] := 75 ;

    for i := 1 to S.RowCount -1 do

        begin

            S.Cells[0,i]:= IntToStr(i) ;

        end ;

    DateTimePicker1.Date := Now ;

    DateTimePicker1.DateTime := Date;

    DateTimePicker2.Date := Now ;

    DateTimePicker2.DateTime := Date;

    DateTimePicker3.Date := Now ;

    DateTimePicker3.DateTime := Date;

```

```

end;

procedure TFJual.Edit1Exit(Sender: TObject);
begin
    panel2.caption := " ";
    if edit1.Text <> " " then
        begin
            If Dm.Customer.Locate('kode',Edit1.text,[loPartialKey]) Then
                begin
                    panel2.Caption := dm.Customer.fieldbyname('Nama').AsString ;
                    panel5.Caption := dm.Customer.fieldbyname('Alamat').AsString ;
                    subtotal ;
                end
            else
                begin
                    subtotal ;
                    messagedlg('Kode customer tidak ditemukan', mtInformation, [mbOK],0) ;
                    edit1.setfocus ;
                    exit ;
                end ;
            end ;
        end;

procedure TFJual.SKeyPress(Sender: TObject; var Key: Char);
var j, baris : Integer ;
begin
    if S.Col in [1,4] then
        begin
            if not (key in ['0'..'9', #8, #13]) then key := #0 ;
        end ;
        if ((S.col = 1) and (key = #13)) then
            begin
                baris := S.Row ;
            end ;
        end ;
    end ;
end ;

```

```

for j := 1 to S.RowCount-1 do
begin
  if j <> baris then
  begin
    if S.Cells[1,j] = S.Cells[1,baris] then
      begin
        S.Cells[1,baris] := " ;
      end ;
    end ;
  end ;
dm.barang.Refresh;
If Dm.Barang.Locate('kode',S.Cells[1,S.Row],[loPartialKey]) Then
begin
S.Cells[2,S.Row] := dm.barang.fieldbyname('Nama').AsString ;
S.Cells[3,S.Row] := FormatFloat('#,0',dm.barang.fieldbyname('Stok').AsFloat) ;
S.Cells[5,S.Row] := FormatFloat('#,0',dm.barang.fieldbyname('Hrg_Jual').AsFloat) ;
end
else
begin
  messageDlg('Kode Barang tidak ditemukan atau sudah ada', mtInformation, [mbOK], 0);
  S.Col := 1 ;
  exit ;
end ;
end ;
if S.Col in [2,3,5,6] then key := #0 ;
if ((S.Col in [4]) and (key = #13)) then
begin
  if FMain.NR(S.Cells[4,S.Row]) > FMain.NR(S.Cells[3,S.Row]) then
    S.Cells[4,S.Row] := S.Cells[3,S.Row] ;
  S.Cells[4,S.Row] := formatFloat('#,0', FMain.NR(S.Cells[4,S.Row])) ;
end ;

```

```

if key = #13 then
  Begin
    subtotal ;
    If S.Col=1 Then S.Col :=4 Else
      Begin
        If S.Col=4 Then
          Begin
            S.Col :=1;
            S.ROw:=S.ROw+1;
          End;
        End;
      End;
    End;
  End;
end;
procedure TFJual.SubTotal ;
var i : Integer ;
    total : Double ;
    cek : boolean ;
begin
  total := 0 ;
  cek := true ;
  for i := 1 to S.RowCount-1 do
    begin
      S.Cells[6,i] := formatfloat('#,0', FMain.NR(S.Cells[4,i])*FMain.NR(S.Cells[5,i]));
      total := total + FMain.NR(S.Cells[6,i]) ;
      If ((fmain.NR(s.cells[6,i]) <=0) And (s.cells[2,i]<>')) Then
        cek := false ;
      end ;
      panel3.caption := formatfloat('#,0', total) ;
      panel6.caption := formatfloat('#,0', total) ;
      if ((panel2.caption <> '') and (cek = true) and (total > 0)) then
        button1.enabled := true
    end
  end
end

```

```

    else button1.enabled := false ;
end ;
procedure TFJual.Edit1.KeyPress(Sender: TObject; var Key: Char);
begin
    if not(key in ['0'..'9',#13,#8]) then key := #0 ;
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;
procedure TFJual.CheckBox1Click(Sender: TObject);
begin
    if checkbox1.Checked then
        begin
            DateTimePicker2.Enabled := True ;
            DateTimePicker2.Color := clWhite ;
        end
    else
        begin
            DateTimePicker2.Enabled := False ;
            DateTimePicker2.Color := clMenu ;
        end ;
end;
procedure TFJual.CheckBox2Click(Sender: TObject);
begin
    if checkbox2.Checked then
        begin
            DateTimePicker3.Enabled := True ;
            DateTimePicker3.Color := clWhite ;
        end
    else
        begin
            DateTimePicker3.Enabled := False ;
            DateTimePicker3.Color := clMenu ;
        end ;
end;

```

```

    end ;
end;
procedure TFJual.Button2Click(Sender: TObject);
begin
    Q1.Close ;
    Q1.SQL.Clear ;
    Q1.SQL.Add('select * from MJual where Nomer_Faktur > 0') ;
    if CheckBox1.Checked then
        begin
            Q1.SQL.Add('and Tgl >=:xx ') ;
            Q1.parambyname('xx').AsDateTime := DateTimePicker2.Date ;
        end;
    if CheckBox2.Checked then
        begin
            Q1.SQL.Add('and Tgl <=:yy ') ;
            Q1.parambyname('yy').AsDateTime := DateTimePicker3.Date ;
        end;
    if ((radiobutton1.Checked) and (edit3.Text <> '')) then
        begin
            Q1.SQL.Add(' and Nama like:xx ') ;
            Q1.parambyname('xx').AsString := '%' + edit3.text + '%' ;
        end ;
    Q1.Open ;
    Q2.Close ;
    Q2.Open ;
end;
procedure TFJual.Hapus1Click(Sender: TObject);
Var
    I: Integer;
begin
    With S Do

```



```

Begin
Rows[Row].Clear;
For i:= Row To RowCount - 1 Do
Begin
  IF (i <> RowCount - 1) Then
  Begin
    Cells[1,i]:= Cells[1,i+1];
    Cells[2,i]:= Cells[2,i+1];
    Cells[3,i]:= Cells[3,i+1];
    Cells[4,i]:= Cells[4,i+1];
    Cells[5,i]:= Cells[5,i+1];
    Cells[6,i]:= Cells[6,i+1];
  End;
End;
SubTotal;
End;
end;
procedure TFJual.Edit2Exit(Sender: TObject);
begin
If (Fmain.NR(Edit2.Text)>=Fmain.NR(Panel3.Caption)) Then
  Begin
    Edit2.Text:=Panel3.Caption;
    DateTimePicker4.Enabled:=False;
    DateTimePicker4.Color:=ClMenu;
    Panel6.Caption:='0';
  End Else
  Begin
    Panel6.Caption:=FormatFloat('#,0',Fmain.NR(Panel3.Caption)-Fmain.NR(Edit2.Text));
    DateTimePicker4.Enabled:=True;
    DateTimePicker4.Color:=ClWhite;
  End;
End;

```

```

end;

procedure TFJual.DateTimePicker1KeyPress(Sender: TObject; var Key: Char);
begin
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFJual.PageControl1Change(Sender: TObject);
begin
    If PageControl1.ActivePageIndex=1 Then
        Begin
            edit3.Text := " ";
            Panel4Click(Sender) ;
            Button2Click(Sender) ;
        End;
end;

procedure TFJual.Edit2KeyPress(Sender: TObject; var Key: Char);
begin
    if not(key in ['0'..'9',#13,#8]) then key := #0 ;
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFJual.SpeedButton2Click(Sender: TObject);
begin
    FBarang.ShowModal ;
    s.cells[1,s.row]:=FBarang.DBGrid1.Fields[0].AsString;
end;

procedure TFJual.SExit(Sender: TObject);
begin
    SubTotal;
end;

procedure TFJual.SpeedButton3Click(Sender: TObject);
begin
    FPenjualan_R :=TFPenjualan_R.Create(Self);

```

```

FPenjualan_R.Q1.Close;
FPenjualan_R.Q1.Sql.Clear;
FPenjualan_R.q1.sql.add('select * from mJual where Nomer_faktur=:XX');
FPenjualan_R.q1.ParamByName('xx').asstring:=DbGrid1.Fields[0].AsString;
FPenjualan_R.Q1.Open;
FPenjualan_R.Q2.Open;
FPenjualan_R.Preview;
FPenjualan_R.Q2.Close;
FPenjualan_R.Q1.Close;
FPenjualan_R.Free;

end;

procedure TFJual.FormClose(Sender: TObject; var Action: TCloseAction);
var i : integer ;
begin
    Edit1.Text := " ;
    Panel2.Caption := " ;
    Panel5.Caption := " ;
    Panel3.Caption := '0' ;
    Panel6.Caption := '0' ;
    Edit2.Text := '0';
    for i := 1 to S.RowCount -1 do
        begin
            S.Cells[0,i]:= IntToStr(i) ;
            S.Cells[1,i]:= " ;
            S.Cells[2,i]:= " ;
            S.Cells[3,i]:= " ;
            S.Cells[4,i]:= " ;
            S.Cells[5,i]:= " ;
            S.Cells[6,i]:= '0' ;
        end ;
    end;

```

```
procedure TFJual.RadioButton1Click(Sender: TObject);
```

```
begin
```

```
  if radiobutton1.Checked then
```

```
    begin
```

```
      edit3.Enabled := true ;
```

```
      edit3.Color := clWhite ;
```

```
    end
```

```
  else
```

```
    begin
```

```
      edit3.Enabled := false ;
```

```
      edit3.Color := clBtnFace ;
```

```
    end ;
```

```
end;
```

```
procedure TFJual.Panel4Click(Sender: TObject);
```

```
begin
```

```
  radiobutton1.Checked := false ;
```

```
  edit3.Enabled := false ;
```

```
  edit3.Text := " ;
```

```
  edit3.Color := clBtnFace ;
```

```
end;
```

```
procedure TFJual.BitBtn3Click(Sender: TObject);
```

```
var i : integer ;
```

```
begin
```

```
  Edit1.Text := " ;
```

```
  Panel2.Caption := " ;
```

```
  Panel5.Caption := " ;
```

```
  Panel3.Caption := '0' ;
```

```
  Panel6.Caption := '0' ;
```

```
  Edit2.Text := '0';
```

```
  for i := 1 to S.RowCount -1 do
```

```

begin
    S.Cells[0,i]:= IntToStr(i) ;
    S.Cells[1,i]:= " ;
    S.Cells[2,i]:= " ;
    S.Cells[3,i]:= " ;
    S.Cells[4,i]:= " ;
    S.Cells[5,i]:= " ;
    S.Cells[6,i]:= '0' ;
end ;
end;
procedure TFJual.button1Click(Sender: TObject);
var i, Nomer : Integer ;
Ada_user_Lain_mengurangi: Boolean;
begin

//Mengecek Client Server
Ada_user_Lain_mengurangi:=False;
for i:=1 to s.rowcount-1 do
begin
    dm.barang.rEFRESH;
    If Dm.Barang.Locate('kode',S.Cells[1,i],[loPartialKey]) Then
    begin
        S.Cells[2,i] := dm.barang.fieldbyname('Nama').AsString ;
        S.Cells[3,i] := FormatFloat('#,0',dm.barang.fieldbyname('Stok').AsFloat) ;
        S.Cells[5,i] := FormatFloat('#,0',dm.barang.fieldbyname('Hrg_Jual').AsFloat) ;
        If fmain.NR(s.cells[4,i]) > fmain.NR(s.cells[3,i]) Then
            Ada_user_Lain_mengurangi:=True;
        End;
    end;
end;
If Ada_user_Lain_mengurangi=True Then
    Begin

```

```

MessageDlg('Stok tidak cukup !'+#13+
           'Karena ada user lain yang mengurangi'+#13+
           'Cek lagi stoknya', mtInformation, [mbOK],0);

//SpeedButton1.Enabled:=False;

s.SetFocus;

Exit;

End;

DM.MJual.Insert ;

DM.MJual.FieldByName('Nomer_Faktur').AsInteger := DM.MJual.RecordCount + 1 ;

Nomer := DM.MJual.FieldByName('Nomer_Faktur').AsInteger ;

DM.MJual.FieldByName('Tgl').AsDateTime:= DateTimePicker1.Date ;

DM.MJual.FieldByName('Kode_Cust').AsString := Edit1.Text ;

DM.MJual.FieldByName('Nama').AsString := Panel2.Caption ;

DM.MJual.FieldByName('Alamat').AsString := Panel5.Caption ;

DM.MJual.FieldByName('Total_Jual').AsFloat := FMain.NR(Panel3.Caption) ;

DM.MJual.FieldByName('Tunai').AsFloat := FMain.NR(Edit2.Text) ;

DM.MJual.FieldByName('Hutang').AsFloat := FMain.NR(Panel6.Caption) ;

DM.MJual.FieldByName('Saldo').AsFloat := FMain.NR(Panel6.Caption) ;

If DateTimePicker4.Enabled=True Then

DM.MJual.FieldByName('Jatuh_Tempo').AsDateTime := DateTimePicker4.Date ;

DM.MJual.Post ;

DM.MJual.ApplyUpdates ;

DM.MJual.CommitUpdates ;

DM.MJual.Refresh ;

for i := 1 to S.RowCount - 1 do
begin
if ((S.Cells[1,i] <> "") and (S.Cells[2,i] <> "") and (FMain.NR(S.Cells[6,i]) > 0)) then
begin
DM.DJual.Insert ;

DM.DJual.FieldByName('Nomer').AsInteger := DM.DJual.RecordCount + 1 ;

DM.DJual.FieldByName('No_Faktur').AsInteger := Nomer ;

```

```

DM.DJual.FieldByName('Tgl').AsDateTime := StrToDate(FormatDateTime('dd/mm/yyyy',DateTimePicker1.Date));
DM.DJual.FieldByName('Kode').AsString := S.Cells[1,i];
DM.DJual.FieldByName('Nama').AsString := (S.Cells[2,i]);
DM.DJual.FieldByName('Jml_Jual').AsFloat := FMain.NR(S.Cells[4,i]);
DM.DJual.FieldByName('Harga_Jual').AsFloat := FMain.NR(S.Cells[5,i]);
DM.DJual.FieldByName('Sub_Total').AsFloat := FMain.NR(S.Cells[6,i]);
If Dm.Barang.Locate('kode',S.Cells[1,i],[loPartialKey]) Then
DM.DJual.FieldByName('HPP').AsFloat := DM.BArang.fieldbyname('HPP').AsFloat;
DM.DJual.FieldByName('Nilai').AsFloat := FMain.NR(S.Cells[4,i]) * DM.DJual.FieldByName('HPP').AsFloat;
DM.DJual.FieldByName('Laba').AsFloat := DM.DJual.FieldByName('Sub_Total').AsFloat -
DM.DJual.FieldByName('Nilai').AsFloat;
DM.DJual.FieldByName('Jml_retur').AsFloat := 0;
DM.DJual.FieldByName('Nilai_Retur').AsFloat := 0;
DM.DJual.FieldByName('Nilai_HPP_Retur').AsFloat := 0;
DM.DJual.FieldByName('Rugi').AsFloat := 0;
DM.DJual.FieldByName('Laba_Bersih').AsFloat := DM.DJual.FieldByName('Laba').AsFloat -
DM.DJual.FieldByName('Rugi').AsFloat;
DM.DJual.Post;
DM.DJual.ApplyUpdates;
DM.DJual.CommitUpdates;
DM.DJual.Refresh;
If Dm.Barang.Locate('kode',S.Cells[1,i],[loPartialKey]) Then
begin
DM.Barang.edit;
DM.Barang.FieldByName('Stok').AsFloat := DM.Barang.FieldByName('Stok').AsFloat - FMain.NR(S.Cells[4,i]);
DM.Barang.FieldByName('Nilai').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat - (FMain.NR(S.Cells[4,i]) *
DM.Barang.FieldByName('hpp').AsFloat);
if ((DM.Barang.FieldByName('Stok').AsFloat > 0) and (DM.Barang.FieldByName('Nilai').AsFloat > 0)) then
begin
DM.Barang.FieldByName('HPP').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat /
DM.Barang.FieldByName('Stok').AsFloat;
end
else

```

```

begin
    DM.Barang.FieldByName('Stok').AsFloat := 0 ;
    DM.Barang.FieldByName('HPP').AsFloat := 0 ;
    DM.Barang.FieldByName('Nilai').AsFloat := 0 ;
end ;
DM.Barang.Post ;
DM.Barang.ApplyUpdates ;
DM.Barang.CommitUpdates ;
DM.Barang.Refresh ;
end ;
end ;
Edit1.Text := '' ;
Panel2.Caption := '' ;
Panel5.Caption := '' ;
Panel3.Caption := '0' ;
Panel6.Caption := '0' ;
Edit2.Text := '0';

for i := 1 to S.RowCount -1 do
begin
    S.Cells[0,i]:= IntToStr(i) ;
    S.Cells[1,i]:= '' ;
    S.Cells[2,i]:= '' ;
    S.Cells[3,i]:= '' ;
    S.Cells[4,i]:= '' ;
    S.Cells[5,i]:= '' ;
    S.Cells[6,i]:= '0' ;
end ;
Button1.Enabled:=False;
DateTimePicker1.DateTime := Date ;

```



```
DateTimePicker4.DateTime := Date + 30 ;  
Panel1.Caption := InttoStr(DM.MJual.RecordCount+1) ;  
end;  
procedure TFJual.SpeedButton1Click(Sender: TObject);  
begin  
Fcustomer.ShowModal;  
Edit1.text:=Fcustomer.DBGrid1.Fields[0].AsString;  
Edit1Exit(Sender);  
end;  
  
end.
```

## Sourcecode Form Retur Pembelian :

```
unit URetur_Beli;

interface

uses

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Grids, ComCtrls, ExtCtrls, StdCtrls, Db, DBTables, DBGrids, Buttons,
  Menus, jpeg, ZAbstractRODataset, ZAbstractDataset, ZDataset;

type

TFretur_beli = class(TForm)

  PageControl1: TPageControl;

  TabSheet1: TTabSheet;

  TabSheet2: TTabSheet;

  TabSheet3: TTabSheet;

  GroupBox1: TGroupBox;

  S: TStringGrid;

  Label9: TLabel;

  N1: TPanel;

  ComboBox5: TComboBox;

  Label15: TLabel;

  N4: TPanel;

  Panel6: TPanel;

  Button2: TButton;

  DateTimePicker2: TDateTimePicker;

  DateTimePicker3: TDateTimePicker;

  CheckBox1: TCheckBox;

  CheckBox2: TCheckBox;

  DBGrid1: TDBGrid;

  DataSource1: TDataSource;

  DBGrid2: TDBGrid;
```

**DataSource2: TDataSource;**  
**DataSource3: TDataSource;**  
**SpeedButton1: TSpeedButton;**  
**PopupMenu1: TPopupMenu;**  
**HapusItem1: TMenuItem;**  
**SpeedButton2: TSpeedButton;**  
  
**Panel5: TPanel;**  
**Panel7: TPanel;**  
**Panel8: TPanel;**  
  
**Image2: TImage;**  
  
**Label1: TLabel;**  
**Label2: TLabel;**  
**Label3: TLabel;**  
**Label4: TLabel;**  
**Label5: TLabel;**  
**Label7: TLabel;**  
  
**Edit1: TEdit;**  
  
**Panel2: TPanel;**  
**Panel3: TPanel;**  
**Panel4: TPanel;**  
**Panel1: TPanel;**  
  
**D1: TDateTimePicker;**  
  
**Button1: TBitBtn;**  
  
**BitBtn2: TBitBtn;**  
**BitBtn3: TBitBtn;**  
  
**Image1: TImage;**  
  
**Q1: TZQuery;**  
**Q11: TZQuery;**  
**Q2: TZQuery;**  
  
**procedure FormShow(Sender: TObject);**

```

procedure ComboBox5Exit(Sender: TObject);

procedure SClick(Sender: TObject);

procedure Edit1Exit(Sender: TObject);

procedure SKeyPress(Sender: TObject; var Key: Char);

procedure Edit1KeyPress(Sender: TObject; var Key: Char);

procedure D1KeyPress(Sender: TObject; var Key: Char);

procedure Button2Click(Sender: TObject);

procedure CheckBox1Click(Sender: TObject);

procedure CheckBox2Click(Sender: TObject);

procedure PageControl1Change(Sender: TObject);

procedure ComboBox5KeyPress(Sender: TObject; var Key: Char);

procedure SpeedButton1Click(Sender: TObject);

procedure FormClose(Sender: TObject; var Action: TCloseAction);

procedure SpeedButton2Click(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure BitBtn3Click(Sender: TObject);

private

  { Private declarations }

  procedure KaliJes;

public

  { Public declarations }

end;

var

  Fretur_beli: TFretur_beli;

implementation

uses UDM, UMain, URetur_R, URetur_BeliR, ULap_Pembelian, ULap_Retur_Beli;

  {$R *.DFM}

procedure TFretur_beli.FormShow(Sender: TObject);

  var i : Integer ;

begin

```

```

Panel1.Caption := Inttostr(DM.Mretur_beli.RecordCount+1);
D1.DateTime := Date;
//DateTimePicker4.DateTime := Date + 30;
S.Cells[0,0] := 'No.';
S.Cells[1,0] := 'Kode';
S.Cells[2,0] := 'Nama Barang';
S.Cells[3,0] := 'Jml Beli';
S.Cells[4,0] := 'Jml Retur';
S.Cells[5,0] := 'Hrg Beli';
S.Cells[6,0] := 'Sub Ttl Beli';
S.Cells[7,0] := 'Sub Ttl Retur';
S.Cells[8,0] := 'Alasan Return' ;//Rusak, Salah kirim
S.Cells[9,0] := 'HPP Jual';
S.Cells[10,0] := 'No DBeli';
S.ColWidths[0] := 30;
S.ColWidths[1] := 45;
S.ColWidths[2] := 150;
S.ColWidths[3] := 50;
S.ColWidths[4] := 50;
S.ColWidths[5] := 50;
S.ColWidths[6] := 70;
S.ColWidths[7] := 70;
S.ColWidths[8] := 80;
S.ColWidths[9] := 0;
S.ColWidths[10] := 0;
for i := 1 to S.RowCount -1 do
begin
  S.Cells[0,i]:= IntToStr(i);
end;
D1.Date := Date;

```

```

D1.Date := Now ;

DateTimePicker2.Date := Now ;

DateTimePicker2.DateTime := Date;

DateTimePicker3.Date := Now ;

DateTimePicker3.DateTime := Date;

end;

procedure TFretur_beli.ComboBox5Exit(Sender: TObject);

begin

S.Cells[8,S.Row]:=Combobox5.Text;

Combobox5.Visible:=False;

end;

procedure TFretur_beli.SClick(Sender: TObject);

var

r:TRect;

indexRow,indexCol:Integer;

begin

indexCol:=S.Col;

indexRow:=S.Row;

r:=S.CellRect(S.Col,S.Row);

case S.Col of

8: begin

If S.Row <> S.RowCount Then

Begin

Combobox5.Left:=S.Left+r.Left+1;

Combobox5.Top:=S.Top+r.Top;

Combobox5.Visible:=True;

if S.Cells[indexCol,indexRow]<>" then

Combobox5.Text:=S.Cells[indexCol,indexRow];

Combobox5.ItemIndex:=0;

Combobox5.SetFocus;

```

```

        Exit;
    End;
end;
end;
end;
end;
end;
end;
end;
end;
end;

procedure TFretur_beli.Edit1Exit(Sender: TObject);
Var I : Integer;
    Sudah_Retur:Boolean;
begin
    Sudah_Retur:=False;
    if edit1.Text <> " " then
    begin
        If Dm.Mbeli.Locate('nomer',Fmain.NR(Edit1.Text),[loPartialKey]) Then
            Begin
                Panel3.Caption := DM.Mbeli.FieldByName('Nomer').AsString;
                Panel3.Caption := DM.Mbeli.FieldByName('Tanggal').AsString;
                Panel4.Caption:=DM.Mbeli.FieldByName('Kode').AsString;
                Panel2.Caption:=DM.Mbeli.FieldByName('Nama').AsString;
                N1.Caption:=FormatFloat('#,0',DM.Mbeli.FieldByName('Total_Beli').AsFloat);
                Q1.Close;
                Q1.Sql.Clear;
                Q1.Sql.Add('Select * from DBeli Where No_MBeli=:XX');
                Q1.ParamByName('XX').AsInteger:=StrToInt(Edit1.Text);
                Q1.Sql.Add('Order By Kode');
                Q1.Open;
                If Q1.RecordCount > 0 Then
                    Begin
                        S.RowCount:=Q1.RecordCount + 1 ;
                        Q1.First;
                        I:=1;
                    End;
                End;
            End;
        End;
    End;
end;

```

```

repeat
    S.Cells[1,I]:=Q1.FieldBYname('Kode').AsString;
    S.Cells[2,I]:=Q1.FieldBYname('Nama').AsString;
    S.Cells[3,I]:=FormatFloat('#,0',Q1.FieldBYname('Jml_beli').AsFloat);
    S.Cells[5,I]:=FormatFloat('#,0',Q1.FieldBYname('Hrg_Beli').AsFloat);
    S.Cells[6,I]:=FormatFloat('#,0',Q1.FieldBYname('Sub_Total').AsFloat);
    S.Cells[9,I]:=FormatFloat('#,0',Q1.FieldBYname('HPP').AsFloat);
    S.Cells[10,I]:=FormatFloat('#,0',Q1.FieldBYname('Nomer').AsInteger);
    S.Cells[8,I]:='Rusak';

    If Q1.FieldBYname('Jml_Return').AsFloat > 0 Then Sudah_Retur:=True;
    I:=I+1;
    Q1.Next;
Until Q1.Eof;
End;
//Cek Retur
If ((StrToDate(Panel3.Caption)+7 < D1.Date) Or (Sudah_Retur=True))Then
Begin
    Panel3.Caption:='0';
    Panel4.Caption:="";
    Panel2.Caption:="";
    Panel1.Caption:="";
    N1.Caption:='0';
    N4.Caption:='0';
    Edit1.text:="";
    For I:=1 To S.RowCount-1 Do
        Begin
            S.Cells[1,I]:="";
            S.Cells[2,I]:="";
            S.Cells[3,I]:="";
            S.Cells[4,I]:="";

```



S.Cells[5,I]:="";

S.Cells[6,I]:="";

S.Cells[7,I]:="";

S.Cells[8,I]:="";

S.Cells[9,I]:="";

S.Cells[10,I]:="";

End;

MessageDlg('Faktur Lebih dari 7 Hari atau'+#13+

'Sudah pernah diretur, tidak boleh diretur lagi',MTInformation,[MBOK],0);

Button1.Enabled:=False;

Edit1.SetFocus;

End Else Button1.Enabled:=True;

End

else

begin

if messagedlg('No. faktur tidak ditemukan, periksa lagi', mtInformation, [mbOK],0) = mrOK then

begin

edit1.SetFocus ;

end ;

end ;

end ;

end;

procedure TFretur\_beli.KaliJes;

Var

I: Integer;

Total:Double;

Begin

Total:=0;

For I:=1 To S.RowCount-1 DO

Begin

```

If Trim(S.Cells[3,i])="" Then S.Cells[3,i]:='0';

S.Cells[7,i]:=FormatFloat("#,0",Fmain.NR(S.Cells[4,i]) * Fmain.NR(S.Cells[5,i]));

Total:=Total+ Fmain.NR(S.Cells[7,i]);

End;

N4.Caption:=FormatFloat("#,0",Total);

If ((Fmain.NR(N4.Caption)>0) AND (Panel4.Caption<>"")) Then

Button1.Enabled:=True Else Button1.Enabled:=False;

End;

procedure TFretur_beli.SKeyPress(Sender: TObject; var Key: Char);

begin

if S.Col=4 then

begin

if not (key in ['0'..'9', #8, #13]) then key := #0 ;

If Fmain.NR(S.Cells[4,S.ROw])>=Fmain.NR(S.Cells[3,S.ROw]) Then

S.Cells[4,S.ROw]:=S.Cells[3,S.ROw];

end Else Key:=#0;

KaliJes;

end;

procedure TFretur_beli.Edit1KeyPress(Sender: TObject; var Key: Char);

begin

if not(key in['0'..'9',#13,#8]) then key := #0 ;

If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);

end;

procedure TFretur_beli.D1KeyPress(Sender: TObject; var Key: Char);

begin

If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);

end;

procedure TFretur_beli.Button2Click(Sender: TObject);

begin

Q11.Close ;

```

```

Q11.SQL.Clear ;
Q11.SQL.Add('select * from MRETUR_beli where Nomer > 0') ;
if CheckBox1.Checked then
begin
Q11.SQL.Add('and Tgl_RETUR >=:xx ') ;
Q11.parambyname('xx').AsDateTime := DateTimePicker2.Date ;
end;
if CheckBox2.Checked then
begin
Q11.SQL.Add('and Tgl_RETUR <=:yy ') ;
Q11.parambyname('yy').AsDateTime := DateTimePicker3.Date ;
end;
Q11.Open ;
Q2.Close ;
Q2.Open ;
end;
procedure TFretur_beli.CheckBox1Click(Sender: TObject);
begin
if checkbox1.Checked then
begin
DateTimePicker2.Enabled := True ;
DateTimePicker2.Color := clWhite ;
end
else
begin
DateTimePicker2.Enabled := False ;
DateTimePicker2.Color := clMenu ;
end ;
end;
procedure TFretur_beli.CheckBox2Click(Sender: TObject);

```

```

begin
    if checkbox2.Checked then
        begin
            DateTimePicker3.Enabled := True ;
            DateTimePicker3.Color := clWhite ;
        end
    else
        begin
            DateTimePicker3.Enabled := False ;
            DateTimePicker3.Color := clMenu ;
        end ;
    end;
procedure TFretur_beli.PageControl1Change(Sender: TObject);
begin
    If PageControl1.ActivePageIndex=1 Then
        Begin
            Button2Click(Sender);
        End;
    end;
procedure TFretur_beli.ComboBox5KeyPress(Sender: TObject; var Key: Char);
begin
    Key:=#0;
end;
procedure TFretur_beli.SpeedButton1Click(Sender: TObject);
begin
    FRetur_beliR :=TFRetur_beliR.Create(Self);
    FRetur_beliR.Q1.Close;
    FRetur_beliR.Q1.Sql.Clear;
    FRetur_beliR.q1.sql.add('select * from mretur_beli where Nomer=:XX');
    FRetur_beliR.q1.ParamByName('xx').asstring:=DbGrid1.Fields[0].AsString;

```

```
FRetur_beliR.Q1.Open;
FRetur_beliR.Q2.Open;
FRetur_beliR.Preview;
FRetur_beliR.Q2.Close;
FRetur_beliR.Q1.Close;
FRetur_beliR.Free;
end;
procedure TFretur_beli.FormClose(Sender: TObject; var Action: TCloseAction);
var i : integer ;
begin
    Panel3.Caption:='0';
    Panel4.Caption:='';
    Panel2.Caption:='';
    Panel1.Caption:='';
    N1.Caption:='0';
    N4.Caption:='0';
    Edit1.text:='';
    For i:=1 To S.RowCount-1 Do
        Begin
            S.Cells[1,i]:='';
            S.Cells[2,i]:='';
            S.Cells[3,i]:='';
            S.Cells[4,i]:='';
            S.Cells[5,i]:='';
            S.Cells[6,i]:='';
            S.Cells[7,i]:='';
            S.Cells[8,i]:='';
            S.Cells[9,i]:='';
            S.Cells[10,i]:='';
        End;
    End;
```

```

end;

procedure TFretur_beli.SpeedButton2Click(Sender: TObject);

begin
    Flap_Retur_beli := TFlap_Retur_beli.Create(Self);

    Flap_Retur_beli.Q11.Close;

    Flap_Retur_beli.Q11.Sql.Clear;

    Flap_Retur_beli.Q11.SQL.Add('Select * From MRetur_Beli Where Nomer > 0');

    if CheckBox1.Checked then

        begin

            Flap_Retur_beli.Q11.SQL.Add('and Tgl_retur >=:xx ');

            Flap_Retur_beli.Q11.parambyname('xx').AsDateTime := DateTimePicker2.Date ;

        end;

        if CheckBox2.Checked then

            begin

                Flap_Retur_beli.Q11.SQL.Add('and Tgl_retur <=:yy ');

                Flap_Retur_beli.Q11.parambyname('yy').AsDateTime := DateTimePicker3.Date ;

            end;

            Flap_Retur_beli.Q11.Open;

            if ((checkbox1.Checked = false) and (checkbox2.Checked = false)) then

                begin

                    Flap_Retur_beli.TM.Caption:= 'Seluruhnya' ;

                    Flap_Retur_beli.TS.Enabled := false ;

                    Flap_Retur_beli.QRLabel21.Enabled := false ;

                end

            else

                begin

                    Flap_Retur_beli.TM.Caption:= DateToStr(DateTimePicker2.date) ;

                    Flap_Retur_beli.TS.Caption:= DateToStr(DateTimePicker3.date) ;

                end ;

            Flap_Retur_beli.Preview ;

```

```

    Flap_Retur_beli.Q11.Close;

    Flap_Retur_beli.Free;

end;

procedure TFretur_beli.Button1Click(Sender: TObject);

var i, Nomer, Cari : Integer ;

begin

    Q1.Close ;

    Q1.SQL.Clear ;

    Q1.SQL.Add('select * from Mretur_beli where Nomer=:XX') ;

    Q1.ParamByName('XX').AsString:=Edit1.text;

    Q1.Open ;

    Q1.First;

If Q1.RecordCount = 0 Then

    Cari:=0 Else

    Cari:=StrToInt(Q1.FieldByName('Nomer').AsString);

Begin

If Dm.Mbeli.Locate('nomer',Cari,[loPartialKey]) Then

    Begin

        MessageDlg('Sudah diretur User lain'+#13+ 'Tidak Boleh Diretur Lagi',MTInformation,[MBOK],0);

        Exit;

    End Else

    Begin

        DM.Mretur_beli.Insert ;

        DM.Mretur_beli.FieldByName('Nomer').AsInteger := DM.Mretur_beli.RecordCount + 1 ;

        Nomer := DM.Mretur_beli.FieldByName('Nomer').AsInteger ;

        DM.Mretur_beli.FieldByName('No_faktur').AsInteger :=StrToInt(Edit1.Text);

        DM.Mretur_beli.FieldByName('Tgl_Faktur').AsDateTime := Strtodate(Panel3.caption);

        DM.Mretur_beli.FieldByName('Tgl_retur').AsDateTime := D1.Date ;

        DM.Mretur_beli.FieldByName('Kode_Cust').AsString := Panel4.Caption ;

        DM.Mretur_beli.FieldByName('Nama_Cust').AsString := Panel2.Caption ;

```

```

DM.Mretur_beli.FieldByName('Total_Jual').AsFloat := FMain.NR(N1.Caption) ;
DM.Mretur_beli.FieldByName('Total_Retur').AsFloat := FMain.NR(N4.Caption) ;
DM.Mretur_beli.Post ;
DM.Mretur_beli.ApplyUpdates ;
DM.Mretur_beli.CommitUpdates ;
DM.Mretur_beli.Refresh ;
for i := 1 to S.RowCount - 1 do
begin
if FMain.NR(S.Cells[4,i]) > 0 then
begin
DM.Dretur_beli.Insert ;
DM.Dretur_beli.FieldByName('Nomer').AsInteger := DM.Dretur_beli.RecordCount + 1 ;
DM.Dretur_beli.FieldByName('No_retur').AsInteger := Nomer ;
DM.Dretur_beli.FieldByName('Tgl').AsDateTime := D1.Date ;
DM.Dretur_beli.FieldByName('Kode').AsString := S.Cells[1,i] ;
DM.Dretur_beli.FieldByName('Nama').AsString := (S.Cells[2,i]) ;
DM.Dretur_beli.FieldByName('Jml_retur').AsFloat := FMain.NR(S.Cells[4,i]) ;
DM.Dretur_beli.FieldByName('Hrg').AsFloat := FMain.NR(S.Cells[5,i]) ;
DM.Dretur_beli.FieldByName('Sub_Total').AsFloat := FMain.NR(S.Cells[7,i]) ;
DM.Dretur_beli.FieldByName('Keterangan').AsString := S.Cells[8,i] ;
DM.Dretur_beli.Post ;
DM.Dretur_beli.ApplyUpdates ;
DM.Dretur_beli.CommitUpdates ;
DM.Dretur_beli.Refresh ;

If Dm.DBeli.Locate('nomer',S.Cells[10,i],[loPartialKey]) Then
begin
DM.DBeli.edit;
DM.DBeli.FieldByName('Jml_Return').AsFloat :=FMain.NR(S.Cells[4,i]) ;
{DM.DBeli.FieldByName('Nilai_retur').AsFloat := FMain.NR(S.Cells[7,i]) ;

```



```

DM.DBeli.FieldByName('Nilai_Hpp_Retur').AsFloat := DM.DBeli.FieldByName('HPP').AsFloat *
FMain.NR(S.Cells[4,i]);

DM.DBeli.FieldByName('Rugi').AsFloat := DM.DBeli.FieldByName('Nilai_retur').AsFloat-
DM.DBeli.FieldByName('Nilai_Hpp_Retur').AsFloat;

DM.DBeli.FieldByName('Laba_Bersih').AsFloat := DM.DBeli.FieldByName('Laba').AsFloat -
DM.DBeli.FieldByName('Rugi').AsFloat;

}DM.DBeli.Post ;

DM.DBeli.ApplyUpdates ;

DM.DBeli.CommitUpdates ;

DM.DBeli.Refresh ;

End;

//Mencari Di Kode Barang, Mengemablikan stok dan nilai
If Dm.Barang.Locate('kode',S.Cells[1,i],[loPartialKey]) Then

begin

DM.Barang.edit ;

DM.Barang.FieldByName('Stok').AsFloat := DM.Barang.FieldByName('Stok').AsFloat - FMain.NR(S.Cells[4,i]) ;

DM.Barang.FieldByName('Nilai').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat - ( FMain.NR(S.Cells[4,i]) *
DM.Barang.FieldByName('HPP').AsFloat );

if ((DM.Barang.FieldByName('Stok').AsFloat > 0) and (DM.Barang.FieldByName('Nilai').AsFloat > 0)) then

begin

DM.Barang.FieldByName('HPP').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat /
DM.Barang.FieldByName('Stok').AsFloat ;

end

else

begin

DM.Barang.FieldByName('Stok').AsFloat := 0 ;

DM.Barang.FieldByName('HPP').AsFloat := 0 ;

DM.Barang.FieldByName('Nilai').AsFloat := 0 ;

end ;

DM.Barang.Post ;

DM.Barang.ApplyUpdates ;

DM.Barang.CommitUpdates ;

DM.Barang.Refresh ;

```

```

        end ;
    end ;
end ;
Panel3.Caption:='0';
Panel4.Caption:="";
Panel2.Caption:="";
Panel1.Caption:="";
N1.Caption:='0';
N4.Caption:='0';
Edit1.text:="";
Button1.Enabled:=False;
For I:=1 To S.RowCount-1 Do
    Begin
        S.Cells[1,I]:="";
        S.Cells[2,I]:="";
        S.Cells[3,I]:="";
        S.Cells[4,I]:="";
        S.Cells[5,I]:="";
        S.Cells[6,I]:="";
        S.Cells[7,I]:="";
        S.Cells[8,I]:="";
        S.Cells[9,I]:="";
        S.Cells[10,I]:="";
    End;
Edit1.SetFocus;
End;
End;
Panel1.Caption := IntToStr(DM.Mretur_beli.RecordCount+1) ;
end;
procedure TFretur_beli.BitBtn3Click(Sender: TObject);

```

```
var i : integer ;
begin
  Panel3.Caption:='0';
  Panel4.Caption:="";
  Panel2.Caption:="";
  Panel1.Caption:="";
  N1.Caption:='0';
  N4.Caption:='0';
  Edit1.text:="";
  For i:=1 To S.RowCount-1 Do
    Begin
      S.Cells[1,i]:="";
      S.Cells[2,i]:="";
      S.Cells[3,i]:="";
      S.Cells[4,i]:="";
      S.Cells[5,i]:="";
      S.Cells[6,i]:="";
      S.Cells[7,i]:="";
      S.Cells[8,i]:="";
      S.Cells[9,i]:="";
      S.Cells[10,i]:="";
    End;
  button1.Enabled := false ;
end;
end.
```

## Sourcecode Form Retur Penjualan :

```
unit URetur;

interface

uses

  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Grids, ComCtrls, ExtCtrls, StdCtrls, Db, DBTables, DBGrids, Buttons,
  Menus, ZAbstractRODataset, ZAbstractDataset, ZDataset;

type

TFretur = class(TForm)

  PageControl1: TPageControl;

  TabSheet1: TTabSheet;

  TabSheet2: TTabSheet;

  TabSheet3: TTabSheet;

  GroupBox1: TGroupBox;

  GroupBox2: TGroupBox;

  Label1: TLabel;

  Label2: TLabel;

  Label3: TLabel;

  Label4: TLabel;

  Label12: TLabel;

  Edit1: TEdit;

  Panel2: TPanel;

  Panel5: TPanel;

  Panel3: TPanel;

  Panel4: TPanel;

  Label5: TLabel;

  Label7: TLabel;

  Panel1: TPanel;

  D1: TDateTimePicker;

  S: TStringGrid;
```

Label9: TLabel;  
N1: TPanel;  
Label13: TLabel;  
N2: TPanel;  
ComboBox5: TComboBox;  
N3: TPanel;  
Label15: TLabel;  
N4: TPanel;  
Button1: TButton;  
Panel6: TPanel;  
Button2: TButton;  
DateTimePicker2: TDateTimePicker;  
DateTimePicker3: TDateTimePicker;  
CheckBox1: TCheckBox;  
CheckBox2: TCheckBox;  
DBGrid1: TDBGrid;  
DataSource1: TDataSource;  
DBGrid2: TDBGrid;  
DataSource2: TDataSource;  
DataSource3: TDataSource;  
PN1: TPanel;  
Label18: TLabel;  
Label14: TLabel;  
Label17: TLabel;  
PN2: TPanel;  
SpeedButton1: TSpeedButton;  
Button3: TButton;  
PopupMenu1: TPopupMenu;  
HapusItem1: TMenuItem;  
Panel7: TPanel;  
Panel8: TPanel;

```
Panel9: TPanel;
Q1: TZQuery;
Q11: TZQuery;
Q2: TZQuery;
procedure FormShow(Sender: TObject);
procedure ComboBox5Exit(Sender: TObject);
procedure SClick(Sender: TObject);
procedure Edit1Exit(Sender: TObject);
procedure SKeyPress(Sender: TObject; var Key: Char);
procedure Button1Click(Sender: TObject);
procedure Edit1KeyPress(Sender: TObject; var Key: Char);
procedure D1KeyPress(Sender: TObject; var Key: Char);
procedure Button2Click(Sender: TObject);
procedure CheckBox1Click(Sender: TObject);
procedure CheckBox2Click(Sender: TObject);
procedure PageControl1Change(Sender: TObject);
procedure ComboBox5KeyPress(Sender: TObject; var Key: Char);
procedure SpeedButton1Click(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure Button3Click(Sender: TObject);
private
  { Private declarations }
  procedure KaliJes;
public
  { Public declarations }
end;
var
  Fretur: TFretur;
implementation
uses UDM, UMain, URetur_R;
```

```

{$R *.DFM}

procedure TFretur.FormShow(Sender: TObject);

var i : Integer ;

begin
    Panel1.Caption := Inttostr(DM.Mretur.RecordCount+1) ;
    D1.DateTime := Date ;
    //DateTimePicker4.DateTime := Date + 30 ;
    S.Cells[0,0] := 'No.' ;
    S.Cells[1,0] := 'Kode' ;
    S.Cells[2,0] := 'Nama Barang' ;
    S.Cells[3,0] := 'Jml Jual' ;
    S.Cells[4,0] := 'Jml Retur' ;
    S.Cells[5,0] := 'Hrg Jual' ;
    S.Cells[6,0] := 'Sub Ttl Jual' ;
    S.Cells[7,0] := 'Sub Ttl Retur' ;
    S.Cells[8,0] := 'Alasan Return' ; //Rusak, Salah kirim
    S.Cells[9,0] := 'HPP Jual';
    S.Cells[10,0] := 'No Djual';
    S.ColWidths[0] := 30 ;
    S.ColWidths[1] := 45 ;
    S.ColWidths[2] := 150 ;
    S.ColWidths[3] := 50 ;
    S.ColWidths[4] := 50 ;
    S.ColWidths[5] := 50 ;
    S.ColWidths[6] := 70 ;
    S.ColWidths[7] := 70 ;
    S.ColWidths[8] := 80 ;
    S.ColWidths[9] := 0 ;
    S.ColWidths[10] := 0 ;

    for i := 1 to S.RowCount -1 do
begin

```

```

    S.Cells[0,i]:= IntToStr(i) ;
end ;
D1.Date := Date;
D1.Date := Now ;
DateTimePicker2.Date := Now ;
DateTimePicker2.DateTime := Date;
DateTimePicker3.Date := Now ;
DateTimePicker3.DateTime := Date;
// DateTimePicker2.DateTime := Date;
// DateTimePicker3.DateTime := Date;
end;
procedure TFretur.ComboBox5Exit(Sender: TObject);
begin
    S.Cells[8,S.Row]:=Combobox5.Text;
    Combobox5.Visible:=False;
end;
procedure TFretur.SClick(Sender: TObject);
var
r:TRect;
indexRow,indexCol:Integer;
begin
    indexCol:=S.Col;
    indexRow:=S.Row;
    r:=S.CellRect(S.Col,S.Row);
    case S.Col of
    8: begin
        If S.Row <> S.RowCount Then
            Begin
                Combobox5.Left:=S.Left+r.Left+1;
                Combobox5.Top:=S.Top+r.Top;
                Combobox5.Visible:=True;
            end;
        end;
    end;
end;

```



```

    if S.Cells[indexCol,indexRow]<>" then
        Combobox5.Text:=S.Cells[indexCol,indexRow];
        Combobox5.ItemIndex:=0;
        Combobox5.SetFocus;
        Exit;
        End;
    end;
end;
end;
procedure TFretur.Edit1Exit(Sender: TObject);
Var I : Integer;
    Sudah_Retur:Boolean;
begin
    For i:=1 To S.RowCount-1 Do
        Begin
            S.Cells[1,i]:= "";
            S.Cells[2,i]:= "";
            S.Cells[3,i]:= "";
            S.Cells[4,i]:= "";
            S.Cells[5,i]:= "";
            S.Cells[6,i]:= "";
            S.Cells[7,i]:= "";
            S.Cells[8,i]:= "";
            S.Cells[9,i]:= "";
            S.Cells[10,i]:= "";
        End;
    Sudah_Retur:=False;
    if edit1.Text <> " then
        begin
            If Dm.Mjual.Locate('nomer_faktur',Fmain.NR(Edit1.Text),[loPartialKey]) Then
                Begin

```

```

Panel3.Caption := DM.MJual.FieldByName('Nomer_Faktur').AsString;
Panel3.Caption := DM.MJual.FieldByName('Tgl').AsString;
Panel4.Caption:=DM.MJual.FieldByName('Kode_Cust').AsString;
Panel2.Caption:=DM.MJual.FieldByName('Nama').AsString;
Panel5.Caption:=DM.MJual.FieldByName('Alamat').AsString;
N1.Caption:=FormatFloat('#,0',DM.MJual.FieldByName('Total_Jual').AsFloat);
N2.Caption:=FormatFloat('#,0',DM.MJual.FieldByName('Tunai').AsFloat+DM.MJual.FieldByName('Dibayar').AsFloat);
N3.Caption:=FormatFloat('#,0',DM.MJual.FieldByName('Saldo').AsFloat);
Q1.Close;
Q1.Sql.Clear;
Q1.Sql.Add('Select * from Djual Where No_Faktur=:XX');
Q1.ParamByName('XX').AsInteger:=StrToInt(Edit1.Text);
Q1.Sql.Add('Order By Kode');
Q1.Open;
If Q1.RecordCount > 0 Then
  Begin
    S.RowCount:=Q1.RecordCount +1 ;
    Q1.First;
    I:=1;
    repeat
      S.Cells[1,I]:=Q1.FieldByName('Kode').AsString;
      S.Cells[2,I]:=Q1.FieldByName('Nama').AsString;
      S.Cells[3,I]:=FormatFloat('#,0',Q1.FieldByName('Jml_Jual').AsFloat);
      S.Cells[5,I]:=FormatFloat('#,0',Q1.FieldByName('Harga_jual').AsFloat);
      S.Cells[6,I]:=FormatFloat('#,0',Q1.FieldByName('Sub_Total').AsFloat);
      S.Cells[9,I]:=FormatFloat('#,0',Q1.FieldByName('HPP').AsFloat);
      S.Cells[10,I]:=FormatFloat('#,0',Q1.FieldByName('Nomer').AsInteger);
      S.Cells[8,I]:'Rusak';
      If Q1.FieldByName('Jml_Retur').AsFloat > 0 Then Sudah_Retur:=True;
      I:=I+1;
    Q1.Next;
  End

```

Until Q1.Eof;

End;

//Cek Retur

If ((StrToDate(Panel3.Caption)+7 < D1.Date) Or (Sudah\_Retur=True))Then

Begin

Panel3.Caption:='0';

Panel4.Caption:=";

Panel2.Caption:=";

Panel5.Caption:=";

Panel1.Caption:=";

N1.Caption:='0';

N2.Caption:='0';

N3.Caption:='0';

N4.Caption:='0';

Edit1.text:=";

For I:=1 To S.RowCount-1 Do

Begin

S.Cells[1,I]:=";

S.Cells[2,I]:=";

S.Cells[3,I]:=";

S.Cells[4,I]:=";

S.Cells[5,I]:=";

S.Cells[6,I]:=";

S.Cells[7,I]:=";

S.Cells[8,I]:=";

S.Cells[9,I]:=";

S.Cells[10,I]:=";

End;

MessageDlg('Faktur Lebih dari 7 Hari atau'+#13+

'Sudah pernah diretur, tidak boleh diretur lagi',MTInformation,[MBOK],0);

Button1.Enabled:=False;

```

    Edit1.SetFocus;

    End Else Button1.Enabled:=True;

End

else

begin

    if messagedlg('No. faktur tidak ditemukan, periksa lagi', mtInformation, [mbOK],0) = mrOK then

    begin

        edit1.SetFocus ;

        end ;

    end ;

    end ;

end;

procedure TFretur.KaliJes;

Var

l: Integer;

Total:Double;

Begin

Total:=0;

For l:=1 To S.RowCount-1 DO

Begin

    If Trim(S.Cells[3,l])='' Then S.Cells[3,l]:= '0';

        S.Cells[7,l]:=FormatFloat('#,0',Fmain.NR(S.Cells[4,l]) * Fmain.NR(S.Cells[5,l]));

        Total:=Total+ Fmain.NR(S.Cells[7,l]);

End;

N4.Caption:=FormatFloat('#,0',Total);

If Fmain.NR(N4.Caption) > Fmain.NR(N3.Caption) Then

Begin

    PN1.Caption:=FormatFloat('#,0',Fmain.NR(N4.Caption)-Fmain.NR(N3.Caption) );

    PN2.Caption:=FormatFloat('#,0',Fmain.NR(N3.Caption) );

End Else

Begin

```

```

PN1.Caption:='0';
PN2.Caption:=FormatFloat('#,0',Fmain.NR(N4.Caption) );
End;
If ((Fmain.NR(N4.Caption)>0) ANd (Panel4.Caption<>'')) Then
  Button1.Enabled:=True Else Button1.Enabled:=False;
End;
procedure TFretur.SKeyPress(Sender: TObject; var Key: Char);
begin
  if S.Col=4 then
    begin
      if not (key in ['0'..'9', #8, #13]) then key := #0 ;
      If Fmain.NR(S.Cells[4,S.ROw])>=Fmain.NR(S.Cells[3,S.ROw]) Then
        S.Cells[4,S.ROw]:=S.Cells[3,S.ROw];
      end Else Key:=#0;
    end
  KaiJes;
end;
procedure TFretur.Button1Click(Sender: TObject);
var i, Nomer, Cari : Integer ;
begin
  Q1.Close ;
  Q1.SQL.Clear ;
  Q1.SQL.Add('select * from Mretur where No_Faktur =:XX' );
  Q1.ParamByName('XX').AsString:=Edit1.text;
  Q1.Open ;
  Q1.First;
  If Q1.RecordCount = 0 Then
    Cari:=0 Else
    Cari:=StrToInt(Q1.FieldByName('Nomer').AsString);
  Begin
  If Dm.MRretur.Locate('nomer',Cari,[loPartialKey]) Then
    Begin

```

```

MessageDlg('Sudah diretur User lain'+#13+ 'Tidak Boleh Diretur Lagi',MTInformation,[MBOK],0);

Exit;

End Else

Begin

DM.MRetur.Insert ;

DM.MRetur.FieldByName('Nomer').AsInteger := DM.MRetur.RecordCount + 1 ;

Nomer := DM.MRetur.FieldByName('Nomer').AsInteger ;

DM.MRetur.FieldByName('No_faktur').AsInteger := StrToInt(Edit1.Text);

DM.MRetur.FieldByName('Tgl_Faktur').AsDateTime := StrToDate(Panel3.caption);

DM.MRetur.FieldByName('Tgl_retur').AsDateTime := D1.Date ;

DM.MRetur.FieldByName('Kode_Cust').AsString := Panel4.Caption ;

DM.MRetur.FieldByName('Nama_Cust').AsString := Panel2.Caption ;

DM.MRetur.FieldByName('Total_Jual').AsFloat := FMain.NR(N1.Caption) ;

DM.MRetur.FieldByName('Total_Retur').AsFloat := FMain.NR(N4.Caption) ;

DM.MRetur.Post ;

DM.MRetur.ApplyUpdates ;

DM.MRetur.CommitUpdates ;

DM.MRetur.Refresh ;

for i := 1 to S.RowCount - 1 do

begin

if FMain.NR(S.Cells[4,i]) > 0 then

begin

DM.DRetur.Insert ;

DM.DRetur.FieldByName('Nomer').AsInteger := DM.DRetur.RecordCount + 1 ;

DM.DRetur.FieldByName('no_retur').AsInteger := Nomer ;

DM.DRetur.FieldByName('Tgl').AsDateTime := D1.Date ;

DM.DRetur.FieldByName('Kode').AsString := S.Cells[1,i] ;

DM.DRetur.FieldByName('Nama').AsString := (S.Cells[2,i]) ;

DM.DRetur.FieldByName('Jml_retur').AsFloat := FMain.NR(S.Cells[4,i]) ;

DM.DRetur.FieldByName('Hrg').AsFloat := FMain.NR(S.Cells[5,i]) ;

DM.DRetur.FieldByName('Sub_Total').AsFloat := FMain.NR(S.Cells[7,i]) ;

```

```

DM.DRetur.FieldByName('Keterangan').AsString := S.Cells[8,i] ;

DM.DRetur.Post ;

DM.DRetur.ApplyUpdates ;

DM.DRetur.CommitUpdates ;

DM.DRetur.Refresh ;

If Dm.Djual.Locate('kode',S.Cells[10,i],[loPartialKey]) Then

begin

DM.Djual.edit;

DM.Djual.FieldByName('Jml_Retur').AsFloat :=FMain.NR(S.Cells[4,i]) ;

DM.Djual.FieldByName('Nilai_retur').AsFloat := FMain.NR(S.Cells[7,i]) ;

DM.Djual.FieldByName('Nilai_Hpp_Retur').AsFloat := DM.Djual.FieldByName('HPP').AsFloat *
FMain.NR(S.Cells[4,i]);

DM.Djual.FieldByName('Rugi').AsFloat := DM.Djual.FieldByName('Nilai_retur').AsFloat-
DM.Djual.FieldByName('Nilai_Hpp_Retur').AsFloat;

DM.Djual.FieldByName('Laba_Bersih').AsFloat := DM.Djual.FieldByName('Laba').AsFloat -
DM.Djual.FieldByName('Rugi').AsFloat;

DM.Djual.Post ;

DM.Djual.ApplyUpdates ;

DM.Djual.CommitUpdates ;

DM.Djual.Refresh ;

End;

// If S.Cells[8,i]≠'Salah Kirim' Then

// Begin

//Mencari Di Kode Barang, Mengemablkan stok dan nilai

If Dm.Barang.Locate('kode',S.Cells[1,i],[loPartialKey]) Then

begin

DM.Barang.edit ;

DM.Barang.FieldByName('Stok').AsFloat := DM.Barang.FieldByName('Stok').AsFloat + FMain.NR(S.Cells[4,i]) ;

DM.Barang.FieldByName('Nilai').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat + ( FMain.NR(S.Cells[4,i]) *
DM.Barang.FieldByName('HPP').AsFloat ) ;

if ((DM.Barang.FieldByName('Stok').AsFloat > 0) and (DM.Barang.FieldByName('Nilai').AsFloat > 0)) then

begin

```

```

        DM.Barang.FieldByName('HPP').AsFloat := DM.Barang.FieldByName('Nilai').AsFloat /
DM.Barang.FieldByName('Stok').AsFloat ;

        end

        else

        begin

            DM.Barang.FieldByName('Stok').AsFloat := 0 ;

            DM.Barang.FieldByName('HPP').AsFloat := 0 ;

            DM.Barang.FieldByName('Nilai').AsFloat := 0 ;

        end ;

        DM.Barang.Post ;

        DM.Barang.ApplyUpdates ;

        DM.Barang.CommitUpdates ;

        DM.Barang.Refresh ;

    end ;

// End;

end ;

end ;

If (Fmain.NR(Pn2.Caption) > 0 ) Then

    Begin

        dm.Dbyr_piutang.Insert;

        dm.Dbyr_piutang.Fieldbyname('Nomer').asinteger:=dm.Dbyr_piutang.recordcount+1;

        dm.Dbyr_piutang.Fieldbyname('No_Faktur').asinteger:=StrToInt(Edit1.text);

dm.Dbyr_piutang.Fieldbyname('Tgl_bayar').asdatetime:=StrToDate(FormatDateTime('dd/mm/yyyy',D1.date));

        dm.Dbyr_piutang.Fieldbyname('Jml_Bayar').asfloat:=fmain.nr(PN2.Caption);

        dm.Dbyr_piutang.post;

        dm.Dbyr_piutang.ApplyUpdates;

        dm.Dbyr_piutang.CommitUpdates;

        dm.Dbyr_piutang.refresh;

    End;

    If Dm.Mjual.Locate('nomer_faktur',StrToInt(Edit1.Text),[loPartialKey]) Then

        Begin

```



```

dm.Mjual.Edit;
dm.Mjual.Fieldbyname('Dibayar').asfloat:=
dm.Mjual.Fieldbyname('Dibayar').asfloat+ fmain.nr(PN2.Caption);
dm.Mjual.Fieldbyname('Saldo').asfloat:=
dm.Mjual.Fieldbyname('hutang').asfloat-
dm.Mjual.Fieldbyname('Dibayar').asfloat;
dm.Mjual.post;
dm.Mjual.ApplyUpdates;
dm.Mjual.CommitUpdates;
dm.Mjual.refresh;
End;
Panel3.Caption:='0';
Panel4.Caption:="";
Panel2.Caption:="";
Panel5.Caption:="";
Panel1.Caption:="";
N1.Caption:='0';
N2.Caption:='0';
N3.Caption:='0';
N4.Caption:='0';

PN1.Caption:='0';
PN2.Caption:='0';
Edit1.text:="";
Button1.Enabled:=False;
For I:=1 To S.RowCount-1 Do
Begin
S.Cells[1,I]:="";
S.Cells[2,I]:="";
S.Cells[3,I]:="";
S.Cells[4,I]:="";

```

```

        S.Cells[5,I]:="";
        S.Cells[6,I]:="";
        S.Cells[7,I]:="";
        S.Cells[8,I]:="";
        S.Cells[9,I]:="";
        S.Cells[10,I]:="";

    End;

    Edit1.SetFocus;

End;

End;

end;

procedure TFretur.Edit1KeyPress(Sender: TObject; var Key: Char);
begin
    if not(key in ['0'..'9',#13,#8]) then key := #0 ;
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFretur.D1KeyPress(Sender: TObject; var Key: Char);
begin
    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure TFretur.Button2Click(Sender: TObject);
begin
    Q11.Close ;
    Q11.SQL.Clear ;
    Q11.SQL.Add('select * from MRETUR where Nomer > 0');
    if CheckBox1.Checked then
    begin
        Q11.SQL.Add('and Tg_RETUR >=:xx ');
        Q11.parambyname('xx').AsDateTime := DateTimePicker2.Date ;
    end;
    if CheckBox2.Checked then

```

```

begin
    Q11.SQL.Add('and Tgl_RETUR <=:yy ');
    Q11.parambyname('yy').AsDateTime := DateTimePicker3.Date ;
end;
Q11.Open ;
Q2.Close ;
Q2.Open ;
end;
procedure TFretur.CheckBox1Click(Sender: TObject);
begin
    if checkbox1.Checked then
        begin
            DateTimePicker2.Enabled := True ;
            DateTimePicker2.Color := clWhite ;
        end
    else
        begin
            DateTimePicker2.Enabled := False ;
            DateTimePicker2.Color := clMenu ;
        end ;
end;
procedure TFretur.CheckBox2Click(Sender: TObject);
begin
    if checkbox2.Checked then
        begin
            DateTimePicker3.Enabled := True ;
            DateTimePicker3.Color := clWhite ;
        end
    else
        begin
            DateTimePicker3.Enabled := False ;

```

```

    DateTimePicker3.Color := clMenu ;
end ;
end;
procedure TFretur.PageControl1Change(Sender: TObject);
begin
    If PageControl1.ActivePageIndex=1 Then
        Begin
            Button2Click(Sender);
        End;
    end;
procedure TFretur.ComboBox5KeyPress(Sender: TObject; var Key: Char);
begin
    Key:=#0;
end;
procedure TFretur.SpeedButton1Click(Sender: TObject);
begin
    FRetur_R :=TFRetur_R.Create(Self);
    FRetur_R.Q1.Close;
    FRetur_R.Q1.Sql.Clear;
    FRetur_R.q1.sql.add('select * from mretur where Nomer=:XX');
    FRetur_R.q1.ParamByName('xx').asstring:=DbGrid1.Fields[0].AsString;
    FRetur_R.Q1.Open;
    FRetur_R.Q2.Open;
    FRetur_R.Preview;
    FRetur_R.Q2.Close;
    FRetur_R.Q1.Close;
    FRetur_R.Free;
end;
procedure TFretur.FormClose(Sender: TObject; var Action: TCloseAction);
var i : integer ;
begin

```

```
Panel3.Caption:='0';
Panel4.Caption:="";
Panel2.Caption:="";
Panel5.Caption:="";
Panel1.Caption:="";
N1.Caption:='0';
N2.Caption:='0';
N3.Caption:='0';
N4.Caption:='0';
Edit1.text:="";
For i:=1 To S.RowCount-1 Do
Begin
S.Cells[1,i]:="";
S.Cells[2,i]:="";
S.Cells[3,i]:="";
S.Cells[4,i]:="";
S.Cells[5,i]:="";
S.Cells[6,i]:="";
S.Cells[7,i]:="";
S.Cells[8,i]:="";
S.Cells[9,i]:="";
S.Cells[10,i]:="";
End;
end;
procedure TFretur.Button3Click(Sender: TObject);
var i : integer ;
begin
Panel3.Caption:='0';
Panel4.Caption:="";
Panel2.Caption:="";
Panel5.Caption:="";
```

```
Panel1.Caption:="";
N1.Caption:='0';
N2.Caption:='0';
N3.Caption:='0';
N4.Caption:='0';
Edit1.text:="";
For i:=1 To S.RowCount-1 Do
Begin
S.Cells[1,i]:="";
S.Cells[2,i]:="";
S.Cells[3,i]:="";
S.Cells[4,i]:="";
S.Cells[5,i]:="";
S.Cells[6,i]:="";
S.Cells[7,i]:="";
S.Cells[8,i]:="";
S.Cells[9,i]:="";
S.Cells[10,i]:="";
End;
button1.Enabled := false ;
end;
end.
```

## Sourcecode Form Pembayaran Piutang :

```
unit Ubayar_piutang;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,

ComCtrls, Grids, DBGrids, ExtCtrls, Db, DBTables, StdCtrls, jpeg, Buttons,

ZAbstractRODataset, ZAbstractDataset, ZDataset;

type

TFbayar_piutang = class(TForm)

    PageControl1: TPageControl;

    TabSheet1: TTabSheet;

    TabSheet2: TTabSheet;

    TabSheet3: TTabSheet;

    DataSource1: TDataSource;

    Panel1: TPanel;

    DBGrid1: TDBGrid;

    DBGrid2: TDBGrid;

    Panel2: TPanel;

    DataSource2: TDataSource;

    DataSource3: TDataSource;

    Panel3: TPanel;

    Label8: TLabel;

    Label9: TLabel;

    Label10: TLabel;

    E: TEdit;

    DataSource4: TDataSource;

    SpeedButton1: TSpeedButton;

    Panel4: TPanel;
```

**GroupBox3: TGroupBox;**  
**D: TDBGrid;**  
**GroupBox2: TGroupBox;**  
**Label1: TLabel;**  
**Label2: TLabel;**  
**date1: TLabel;**  
**Label4: TLabel;**  
**Label5: TLabel;**  
**Label3: TLabel;**  
**Label6: TLabel;**  
**Label11: TLabel;**  
**P2: TPanel;**  
**P1: TPanel;**  
**P4: TPanel;**  
**P5: TPanel;**  
**P6: TPanel;**  
**P7: TPanel;**  
**P8: TPanel;**  
**P3: TPanel;**  
**SpeedButton2: TSpeedButton;**  
**TabSheet4: TTabSheet;**  
**Panel5: TPanel;**  
**DBGrid3: TDBGrid;**  
**Y1: TDateTimePicker;**  
**Y2: TDateTimePicker;**  
**Label7: TLabel;**  
**Label12: TLabel;**  
**Button1: TButton;**  
**DataSource5: TDataSource;**



```
Button2: TButton;

DT: TDateTimePicker;

Panel7: TPanel;

Panel8: TPanel;

Panel9: TPanel;

Query1: TZQuery;

Query2: TZQuery;

Q3: TZQuery;

Q4: TZQuery;

procedure PageControl1Change(Sender: TObject);

procedure EExit(Sender: TObject);

procedure EKeyPress(Sender: TObject; var Key: Char);

procedure FormShow(Sender: TObject);

procedure SpeedButton1Click(Sender: TObject);

procedure DtKeyPress(Sender: TObject; var Key: Char);

procedure SpeedButton2Click(Sender: TObject);

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

Fbayar_piutang: TFbayar_piutang;

implementation

uses UDM, UMain, UBayar_Piutang_R, UJatuh_Tempo_R;

{$R *.DFM}
```

```

procedure TFbayar_piutang.PageControl1Change(Sender: TObject);

begin
// Panel2.Caption:='Daftar Putang dari : '+DbGrid1.Fields[0].AsString+' : '+DbGrid1.Fields[1].AsString;

// Panel3.Caption:='Daftar Putang dari : '+DbGrid1.Fields[0].AsString+' : '+DbGrid1.Fields[1].AsString +' dg No Faktur :
'+DbGrid2.Fields[0].AsString+' : '+DbGrid1.Fields[2].AsString;

If Dm.MJual.Locate('nomer_faktur',DbGrid2.Fields[0].AsString,[loPartialKey]) Then

Begin

P3.Caption:=DM.MJual.FieldByName('NOMER_FAKTUR').AsString;

P4.Caption:=DM.MJual.FieldByName('Tgl').AsString;

P1.Caption:=DM.MJual.FieldByName('Kode_Cust').AsString;

P2.Caption:=DM.MJual.FieldByName('Nama').AsString;

P5.Caption:=FormatFloat('#,0',Dm.MJual.FieldByName('Hutang').AsFloat);

P8.Caption:=FormatFloat('#,0',Dm.MJual.FieldByName('Dibayar').AsFloat);

P7.Caption:=FormatFloat('#,0',Dm.MJual.FieldByName('Saldo').AsFloat);

P6.Caption:=Dm.MJual.FieldByName('Jatuh_tempo').AsString;

Q3.Close;

Q3.Sql.Clear;

Q3.Sql.Add('Select * from DByr_Piutang Where NO_FAKTUR=:XX');

Q3.ParamByName('Xx').AsInteger:=StrToInt(P3.Caption);

Q3.Open;

If Fmain.NR(P7.Caption) >0 Then

Begin

E.Enabled:=TRUE;

E.COlor:=CLWhite;

DT.Enabled:=TRUE;

Dt.COlor:=CLWhite;

SpeedButton1.Enabled:=True;

End Else

```

```

begin

E.Enabled:=False;

E.COlor:=CLMenu;

DT.Enabled:=False;

Dt.COlor:=CLMenu;

SpeedButton1.Enabled:=False;

End;

ENd;

If PageControl1.ActivePageIndex=3 Then

    Button1Click(Sender);

end;

procedure TFbayar_piutang.EExit(Sender: TObject);

begin

    If Fmain.NR(E.Text) > Fmain.NR(P7.Caption) Then

        Begin

            E.Text:=P7.Caption;

            End;

            E.Text:=FormatFloat('#,0',Fmain.NR(E.Text));

end;

procedure TFbayar_piutang.EKeyPress(Sender: TObject; var Key: Char);

begin

    If Not(Key In ['0'..'9', #8, #13]) Then Key:=#0;

    If (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);

end;

procedure TFbayar_piutang.FormShow(Sender: TObject);

begin

    DT.DateTime := Date;

    Y1.date := Date ;

    Y2.date := Date ;

```

```

Query1.Close;

Query2.Close;

Query1.Open;

Query2.Open;

end;

procedure TFbayar_piutang.SpeedButton1Click(Sender: TObject);

begin

    If Fmain.NR(E.Text) <=0 Then

        Begin

            Exit;

        End;

        //Mengecek Client Server

        DM.Mjual.Refresh;

        If Dm.Mjual.Locate('nomer_faktur',StrToInt(P3.Caption),[loPartialKey]) Then

            Begin

                If (DM.Mjual.FieldByName('Saldo').AsFloat<>Fmain.NR(P7.Caption)) Then

                    Begin

                        MessageDlg('Ada User lain membayar piutang !'+#13+

                            'Jadi.....'+#13+

                            'Cek Lagi Total Pembayaran', mtInformation, [mbOK],0);

                        PageControl1.ActivePageIndex:=0;

                        Query1.Close;

                        Query2.Close;

                        Query1.Open;

                        Query2.Open;

                        Exit;

                    End;

                End;

            End;

            dm.Dbyr_piutang.Insert;

```

```
dm.Dbyr_piutang.Fieldbyname('Nomer').asinteger:=dm.Dbyr_piutang.recordcount+1;
dm.Dbyr_piutang.Fieldbyname('No_Faktur').asinteger:=StrToInt(P3.Caption);
dm.Dbyr_piutang.Fieldbyname('Tgl_bayar').asdatetime:=StrToDate(FormatDateTime('dd/mm/yyyy',DT.date));
dm.Dbyr_piutang.Fieldbyname('Jml_Bayar').asfloat:=fmain.nr(E.Text);
dm.Dbyr_piutang.post;
dm.Dbyr_piutang.ApplyUpdates;
dm.Dbyr_piutang.CommitUpdates;
dm.Dbyr_piutang.refresh;
If Dm.Mjual.Locate('nomer_faktur',StrToInt(P3.Caption),[[oPartialKey]]) Then
  Begin
    dm.Mjual.Edit;
    dm.Mjual.Fieldbyname('Dibayar').asfloat:=
    dm.Mjual.Fieldbyname('Dibayar').asfloat+ Fmain.NR(E.Text);
    dm.Mjual.Fieldbyname('Saldo').asfloat:=
    dm.Mjual.Fieldbyname('hutang').asfloat-
    dm.Mjual.Fieldbyname('Dibayar').asfloat;
    dm.Mjual.post;
    dm.Mjual.ApplyUpdates;
    dm.Mjual.CommitUpdates;
    dm.Mjual.refresh;
  End;
PageControl1.ActivePageIndex:=0;
Query1.Close;
Query2.Close;
Query1.Open;
Query2.Open;
E.Text:='0';
Exit;
end;
```

```

procedure Tfbayar_piutang.DtKeyPress(Sender: TObject; var Key: Char);
begin
    if (Key=Chr(VK_return)) Then SendMessage(Handle,WM_NEXTDLGCTL,0,0);
end;

procedure Tfbayar_piutang.SpeedButton2Click(Sender: TObject);
begin
    FBayar_Piutang_R :=TFBayar_Piutang_R.Create(Self);
    FBayar_Piutang_R.A1.CAption:=P1.Caption;
    FBayar_Piutang_R.A2.CAption:=P2.Caption;
    FBayar_Piutang_R.A3.CAption:=P3.Caption;
    FBayar_Piutang_R.A4.CAption:=P4.Caption;
    FBayar_Piutang_R.A5.CAption:=P5.Caption;
    FBayar_Piutang_R.A6.CAption:=P6.Caption;
    FBayar_Piutang_R.A7.CAption:=P7.Caption;
    FBayar_Piutang_R.A8.CAption:=P8.Caption;
    FBayar_Piutang_R.Q3.Close;
    FBayar_Piutang_R.Q3.Sql.Clear;
    FBayar_Piutang_R.Q3.Sql.Add('Select * from DByr_Piutang Where No_Faktur=:XX');
    FBayar_Piutang_R.Q3.ParamByName('Xx').AsInteger:=StrToInt(P3.Caption);
    FBayar_Piutang_R.Q3.Open;
    FBayar_Piutang_R.Preview;
    FBayar_Piutang_R.Q3.Close;
    FBayar_Piutang_R.Free;
end;

procedure Tfbayar_piutang.Button1Click(Sender: TObject);
begin
    Q4.Close;
    Q4.Sql.Clear;
    Q4.Sql.Add('Select * from Mjual Where Saldo >0');

```

**If Y1.Checked Then**

**Begin**

Q4.Sql.Add('And Jatuh\_Tempo>=:XX');

Q4.ParambyName('Xx').AsdateTime:=Y1.DateTime;

**End;**

**If Y2.Checked Then**

**Begin**

Q4.Sql.Add('And Jatuh\_Tempo<=:YY');

Q4.ParambyName('YY').AsdateTime:=Y2.DateTime;

**End;**

**Q4.Open;**

**end;**

**procedure Tfbayar\_piutang.Button2Click(Sender: TObject);**

**begin**

FJatuh\_Tempo\_R :=TFJatuh\_Tempo\_R.Create(Self);

FJatuh\_Tempo\_R.Q1.Close;

FJatuh\_Tempo\_R.Q1.Sql.Clear;

FJatuh\_Tempo\_R.Q1.Sql.Add('Select \* from Mjual Where Saldo >0');

**If Y1.Checked Then**

**Begin**

FJatuh\_Tempo\_R.Q1.Sql.Add('And Jatuh\_Tempo>=:XX');

FJatuh\_Tempo\_R.Q1.ParambyName('Xx').AsdateTime:=Y1.DateTime;

**End;**

**If Y2.Checked Then**

**Begin**

FJatuh\_Tempo\_R.Q1.Sql.Add('And Jatuh\_Tempo<=:YY');

FJatuh\_Tempo\_R.Q1.ParambyName('YY').AsdateTime:=Y2.DateTime;

**End;**

**FJatuh\_Tempo\_R.Q1.Open;**

FJatuh\_Tempo\_R.Preview;

FJatuh\_Tempo\_R.Q1.Close;

FJatuh\_Tempo\_R.Free;

end;

end.