

# **SKRIPSI**

## **PENGEMBANGAN SISTEM INFORMASI PRODUKSI ROTI DI UD SUTRA SIDOARJO**



**Disusun Oleh :**

**HANI LESTYOADI**

**NIM : 05.12.608**

**JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER & INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2010**

---

LEMBAR PERSETUJUAN

PENGEMBANGAN SISTEM INFORMASI PRODUKSI ROTI DI UD  
SUTRA SIDOARJO

SKRIPSI

Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh  
Gelar Sarjana Teknik Komputer dan Informatika Strata Satu (S-1)

Disusun Oleh :

**HANI LESTYOADI**

NIM : 05.12.608

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II

Irmalia Suryani Faradisa, ST, MT  
NIP. P. 1030000365

Ahmad Faisal, ST

Mengetahui

Ketua Jurusan Teknik Elektro S-1

Yusuf Ismail Nakhoda, MT  
NIP. Y. 1018800189

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

2010

## PENGEMBANGAN SISTEM INFORMASI PRODUKSI ROTI DI UD SUTRA SIDOARJO

**Hani Lestyoadi**

Jurusan Teknik Elektro S-1, Konsentrasi Teknik Komputer dan Informatika  
Fakultas Teknologi Industri, Institut Teknologi Nasional Malang  
Jl. Raya Karanglo Km 2 Malang  
Email : [sathagolf@ymail.com](mailto:sathagolf@ymail.com)

### **Abstrak**

*Pada era komputerisasi saat ini peran sistem informasi salah satu faktor keberhasilan dalam persaingan bisnis, yang mana dapat membantu dalam mekanisme komunikasi yang lebih efektif dengan pelanggan, pemasok dan karyawan. Diharapkan dengan penerapan system informasi dapat menyederhanakan sejumlah proses internal sehingga dapat dihitung secara nyata.*

*Dengan persaingan bisnis dalam usaha retail yang begitu ketat maka salah satu langkah peretail dengan mempertahankan keunggulan adalah mengefisienkan proses didalam usaha retail itu sendiri agar lebih efisien dan efektif, sehingga dengan metodologi perencanaan yang baik akan didapatkan hasil yang lebih baik dan bermanfaat untuk periode yang panjang. Perencanaan strategis sistem informasi akan memberikan gambaran menyeluruh tentang kebutuhan sistem informasi masa kini dan masa yang akan datang, diselaraskan dengan visi, misi, dan strategi bisnis jangka panjang telah ditetapkan.*

*Pengembangan sistem informasi ini diharapkan dapat menunjang kinerja di UD.Sutra sidoarjo, dimana perusahaan tersebut masih menggunakan sistem manual, sehingga dengan adanya sistem informasi ini segala kegiatan seperti pengolahan data dan pembuatan laporan dapat lebih efisien, efektif serta system ini mudah digunakan oleh para karyawan UD.Sutra*

**Kata Kunci :** *Sistem Informasi Manajemen pada Usaha Dagang Roti Sutra Sidoarjo*

# BREAD PRODUCTION INFORMATION SYSTEM DEVELOPMENT IN UD SIDOARJO SUTRA

**Hani Lestyoadi**

Majors Elektronics Engineering S-1, Concentration Computer and Informatika  
industrial faculty of technology, Institute Technology Nasional Malang  
Raya Karanglo Street Km 2 Malang  
Email : [sathagolf@yahoo.com](mailto:sathagolf@yahoo.com)

## *Abstract*

*in the current era of computerized information systems roles one success factor in business competition. which can assist in more effective communication mechanisms with customers, suppliers and employees. Expected with the implementation of information systems can simplify a number of internal processes that can be computed in real.*

*With business competition in the retail business is so tight, so one step peretail to maintain excellence is streamline the processes involved in the retail business itself to be more efficient and effective, so that with good planning methodology will get better results and is useful for long periods. Strategic planning system will give more comprehensive information about the needs of today's information systems and years to come, aligned with the vision, mission and long-term business strategy has been determined.*

*Information system development is expected to bolster the performance in UD.Sutra sidoarjo, where the company is still using a manual system, so using this information system all activities such as data processing and reporting can be more efficient, effective and the system is easy to use by people at UD . Sutra*  
**Keywords:** *Management Information System on Bread Trade Enterprises Sutra Sidoarjo*

## KATA PENGANTAR

### **Bismillahirrahmaanirrahim**

Syukur alhamdulillah penulis haturkan kehadiran Allah SWT atas rahmat dan karuniaNya, selanjutnya shalawat dan salam penulis sampaikan kepada junjungan kita Nabi besar Muhammad SAW beserta sahabat dan keluarganya yang telah membawa kita dari alam kegelapan kealam yang berilmu pengetahuan sehingga penulis dapat menyelesaikan skripsi ini dengan judul **“PENGEMBANGAN SISTEM INFORMASI PRODUKSI ROTI DI UD SUTRA SIDOARJO”**. Guna memenuhi syarat-syarat dalam menyelesaikan pendidikan program S1 Teknik Informatika pada Institut Teknologi Nasional Malang.

Dengan ketulusan hati yang sedalam-dalamnya penulis menyampaikan penghargaan dan penghormatan yang setinggi-tingginya kepada Ayahanda dan Ibunda yang telah memberikan cinta dan kasih sayangnya kepada penulis baik itu melalui doa, dukungan moril maupun materil selama ini, serta ucapan terima kasih penulis untuk kakak sehingga terselesaikannya penulisan ini selain itu penulis ingin menyampaikan terima kasih yang amat sangat kepada teman-teman di Mojokerto, HKRT, B-Speed atas motivasinya dalam penulisan skripsi ini.

Selama pengerjaan skripsi ini, banyak mendapat bantuan dan bimbingan dari berbagai pihak, untuk itu pada kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi
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. Ibu Irmalia Suryani Faradisa, ST, MT., selaku Dosen Pembimbing I
5. Bapak Ahmad Faisol, ST., selaku Dosen Pembimbing II
6. Seluruh Dosen dan staf pengajaran Institiut Teknologi Nasional Malang.
7. seluruh karyawan/i Fakultas Teknik Indutri khususnya yang telah banyak memberikan informasi kepada penulis.
8. Segenap civitas mahasiswa Institut Nasional Malang yang turut memberikan semangat kepada penulis.

Akhirnya penulis menyadari sepenuhnya bahwa skripsi ini jauh dari sempurna, baik isi, pemilihan bahasa maupun sistematika penulisannya, namun penulis mengharapkan bantuan berupa saran dan kritik yang membangun dari semua pihak demi kesempurnaan dan mutu penulisan skripsi ini.

Semoga tulisan ini dapat bermanfaat bagi ilmu pengetahuan dan pembaca khususnya serta mendapat keridhaan Allah SWT.

Amin... Yaarabbal 'Alamin

Malang, Agustus 2010

Penulis

## DAFTAR ISI

	Halaman
LEMBARAN PENGESAHAN.....	i
ABSTRAK .....	ii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR TABEL .....	viii
DAFTAR GAMBAR.....	ix
DAFTAR LAMPIRAN.....	xi
<b>BAB I PENDAHULUAN</b>	
1.1 Latar Belakang.....	1
1.2 Tujuan Penulisan.....	3
1.3 Perumusan Masalah.....	3
1.4 Batasan Masalah.....	3
1.5 Metodologi Penelitian.....	4
1.6 Sistematika Penulisan.....	6
<b>BAB II DASAR TEORI</b>	
2.1 Pengertian Dasar Sistem.....	7
2.2 Data Flow Diagram .....	8
2.3 Data Management Sistem.....	13
2.4 Pengertian Peramalan.....	14
2.5 Perencanaan Produksi.....	16
2.6 Penjadwalan Produksi.....	16
<b>BAB III ANALISA DAN DESAIN SISTEM</b>	
3.1 Analisa Sistem.....	18
3.1.1 Analisa Sistem yang Lama.....	18
3.1.2 Analisa Permasalahan.....	19
3.1.3 Analisa Kebutuhan.....	19
3.2 Desain Sistem.....	21
3.2.1 Data Flow Diagram.....	22
3.2.2 Program Flowchart Peramalan.....	32
3.2.3 Program Flowchart Penjadwalan.....	33
3.2.4 Conceptual Data Model (CDM).....	35
3.2.5 Physical Data Model (PDM).....	38

3.2.6 Desain Tabel Database.....	39
<b>BAB IV HASIL DAN ANALISA</b>	
4.1 Instalasi Program.....	50
4.1.1 Kebutuhan Perangkat Lunak.....	50
4.1.2 Setup Program.....	51
4.2 Tampilan Program.....	51
4.2.2 Tampilan Menu Utama.....	52
4.2.3 Tampilan Input Data Bahan Baku.....	54
4.2.4 Tampilan Input Data Barang Jadi.....	55
4.2.5 Tampilan Input Data Supplier.....	55
4.2.6 Tampilan Input Order.....	56
4.2.7 Tampilan Input Resep.....	57
4.2.8 Tampilan Kapasitas Produksi.....	57
4.3.9 Tampilan Input Data Produksi.....	58
4.3.10 Tampilan Input Produksi Pesan.....	58
4.3.11 Tampilan Input Data Customer.....	59
4.3.12 Tampilan Input Data Penjualan.....	59
4.3.13 Tampilan Input Data Pemesanan.....	60
4.3.14 Tampilan Bayar Pesan.....	60
4.3.15 Tampilan Input Data Karyawan.....	61
4.3.16 Tampilan Input Useracc.....	61
4.3.17 Tampilan Update Password.....	62
4.3.18 Tampilan Exit.....	62
4.3.19 Laporan Bahan Baku.....	63
4.3.20 Laporan Barang Jadi.....	63
4.3.21 Laporan Supplier.....	64
4.3.22 Laporan Order.....	64
4.3.23 Laporan Produksi.....	65
4.3.24 Laporan Produksi Pesan.....	66
4.3.25 Laporan Customer.....	67
4.3.26 Laporan Jual.....	68
4.3.27 Laporan Pesan.....	69
4.3.28 Laporan Peramalan.....	69
<b>BAB V PENUTUP</b>	
5.1. Kesimpulan.....	70
5.2. Saran.....	70
<b>DAFTAR PUSTAKA.....</b>	<b>72</b>



## DAFTAR TABEL

Tabel 3.1	Tabel T_Bahan_Baku.....	39
Tabel 3.2	Tabel T_Barang_Jadi.....	39
Tabel 3.3	Tabel T_Customer.....	40
Tabel 3.4	Tabel T_Item_Order.....	40
Tabel 3.5	Tabel T_Item_Pesan.....	41
Tabel 3.6	Tabel T_Item_Jual.....	41
Tabel 3.7	Tabel T_Item_Produksi.....	42
Tabel 3.8	Tabel T_Item_Produksi_Pesan.....	42
Tabel 3.9	Tabel T_Order.....	43
Tabel 3.10	Tabel T_Pemesanan.....	43
Tabel 3.11	Tabel T_Penjadwalan.....	44
Tabel 3.12	Tabel T_Penjualan.....	44
Tabel 3.13	Tabel T_Peramalan.....	45
Tabel 3.14	Tabel T_Produksi.....	45
Tabel 3.15	Tabel T_Resep.....	46
Tabel 3.16	Tabel T_Supplier.....	46
Tabel 3.17	Tabel T_Kapasitas_Produksi.....	47
Tabel 3.18	Tabel T_Karyawan.....	47
Tabel 3.19	Tabel T_Bayar_Pesan.....	48
Tabel 3.20	Tabel T_Produksi_Pesan.....	48
Tabel 3.21	Tabel T_Useracc.....	49

## DAFTAR GAMBAR

Gambar 2.1 Entity .....	10
Gambar 2.2 Proses.....	10
Gambar 2.3 Data Flow.....	11
Gambar 2.4 Data Strorage.....	11
Gambar 2.5 One to one Relationship .....	12
Gambar 2.6 One to many Relationship.....	12
Gambar 2.7 Many to many .....	13
Gambar 3.1 Konteks Diagram.....	22
Gambar 3.2 DFD Level 1 .....	24
Gambar 3.3 DFD Level 2 Pemesanan dan Pengiriman Produk .....	27
Gambar 3.4 DFD Level 2 Proses Penjadwalan Produksi .....	29
Gambar 3.5 DFD Level 2 Proses Peramalan .....	31
Gambar 3.6 Flowchart Peramalan.....	32
Gambar 3.7 Flowchart Penjadwalan.....	33
Gambar 3.8 Conceptual Data Model (CDM).....	35
Gambar 3.9 Physical Data Model (PDM).....	38
Gambar 4.1 Form Login .....	51
Gambar 4.2 Form Menu Utama .....	54
Gambar 4.3 Form Data Bahan Baku .....	54
Gambar 4.4 Form Data Barang Jadi .....	55
Gambar 4.5 Form Data Supplier .....	55

Gambar 4.6 Form Order.....	56
Gambar 4.7 Form Resep.....	57
Gambar 4.8 Form Kapasitas Produksi.....	57
Gambar 4.9 Form Data Produksi.....	58
Gambar 4.10 Form Produksi Pesan.....	58
Gambar 4.11 Form Data Customer.....	59
Gambar 4.12 Form Data Penjualan.....	59
Gambar 4.13 Form Data Pemesanan.....	60
Gambar 4.14 Form Bayar Pesan.....	60
Gambar 4.15 Form Data Karyawan.....	61
Gambar 4.16 Form UserACC.....	61
Gambar 4.17 Form Update Password.....	62
Gambar 4.18 Form Exit.....	62
Gambar 4.19 Laporan Bahan Baku.....	63
Gambar 4.20 Laporan Barang Jadi.....	63
Gambar 4.21 Laporan Supplier.....	64
Gambar 4.22 Laporan Order.....	65
Gambar 4.23 Laporan Produksi.....	66
Gambar 4.24 Laporan Produksi Pesan.....	67
Gambar 4.25 Laporan Customer.....	67
Gambar 4.26 Laporan Jual.....	68
Gambar 4.27 Laporan Pesan.....	69
Gambar 4.28 Laporan Peramalan.....	69

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

UD “Sutra” merupakan sebuah perusahaan yang bergerak di bidang industri makanan, yang khusus memproduksi roti perusahaan ini dipimpin oleh seorang pimpinan yang mengatur segala sesuatu yang ada pada perusahaan itu mulai dari pembelian material, perencanaan produksi, sampai pada pengendalian kualitas. Selama ini keputusan yang diambil oleh perusahaan seringkali hanya berdasarkan pada pengalaman yang diperoleh dari sang pemimpin. tidak ada perencanaan, produksi, dan tidak ada penjadwalan.

Dengan hanya bergantung pada pengalaman sang pemimpin yang juga sangat sibuk, sangatlah sulit dalam memperkirakan jumlah produk yang akan diproduksi sesuai permintaan dari pasar, sehingga seringkali terjadi kesulitan dalam memenuhi permintaan pasar. Pihak perusahaan memang memiliki data permintaan masa lalu, tetapi data tersebut tidak digunakan untuk mengatur tingkat produksinya.

Perusahaan ini memproduksi tiga jenis rasa yang berbeda ( coklat, strobery, keju,dll) sehingga secara total ada tiga variasi jenis produksi. Sering kali ada permintaan dari pasar, perusahaan ini tidak memiliki persediaan barang jadi yang cukup untuk memenuhi permintaan tersebut dan membutuhkan waktu yang cukup lama dalam memenuhi permintaan tersebut<sup>[3]</sup>. Perusahaan tidak mempunyai perencanaan produksi yang optimal, sehingga pada periode tertentu terjadi kekurangan. Data permintaan dari perusahaan belum tersusun dengan baik, sehingga tidak adanya peramalan dalam melakukan produksi (produksi

berlangsung tanpa memperhatikan jumlah permintaan)<sup>[3]</sup>. persediaan bahan baku yang ada kurang terkontrol ( kelebihan bahan baku, terkadang juga kekurangan bahan baku ). Sehingga berdampak dalam melakukan penjadwalan produksi. Serta dalam melakukan penjadwalan sering tidak efisien sehingga sering gagal untuk memenuhi permintaan yang ada, karena waktu pengiriman yang telah disejui tidak melalui hitungan yang akurat dari pihak perusahaan. Kondisi ini memberikan kesempatan para competitor untuk merebut pansa pasar.

Pada perusahaan roti ini merupakan usaha yang dipengaruhi oleh *trend* musiman. Serta kadang ada kemungkinan dilain hal tersebut, Oleh karena itu kita menggunakan metode *winter* , karena dalam metode ini mencakup pola data *trend*, musiman dan acak. Pada penjadwalan produksi yang digunakan adalah metode FIFO (First In First out)<sup>[3]</sup>, karena pelanggan terkadang meminta agar permintaannya dipenuhi dalam kurun waktu tertentu. Pada metode FIFO (First In First out) digunakan untuk menyelesaikan semua proses yang ada agar cepat selesai. Untuk itu pada skripsi ini akan dilakukan perbaikan pada system yang lama menjadi system yang baru.

## **1.2. Tujuan Penulisan**

Tujuan dari penulisan tugas akhir ini membuat suatu perangkat lunak yang dapat membantu dalam meramalkan perencanaan produksi, serta menentukan perencanaan penjadwalan produksi, pemesanan, untuk memenuhi permintaan konsumen.

## **1.3. Perumusan Masalah**

Berdasarkan latar belakang masalah yang telah di jelaskan, maka dapat dirumuskan permasalahan yang ada, antara lain:

- a. Bagaimana membuat system yang mampu melakukan peramalan untuk melakukan perencanaan produksi pada periode tertentu agar bahan baku tidak mengalami kekurangan ataupun kelebihan.
- b. Bagaimana membuat system yang mampu melakukan penjadwalan perencanaan produksi.

## **1.4. Batasan Masalah**

Agar permasalahan mengarah sesuai dengan tujuan, maka pembahasan skripsi ini dibatasi hal-hal berikut :

- a. Aplikasi sistem informasi Produksi ini hanya digunakan pada Perusahaan UD Roti Sutra Sidoarjo.
- b. Tidak membahas sistem jaringan yang digunakan.
- c. Proses Peramalan Produksi ini dengan menggunakan metode winter yang berfungsi untuk meramalkan jumlah produksi pada bulan kedepan,

- d. Perencanaan produksi sesuai dengan yang sudah direncanakan dan dapat dilaksanakan berdasarkan data-data laporan peramalan dan pemesanan yang telah diperoleh.
- e. Proses Penjadwalan produksi ini dengan menggunakan metode FIFO (First In First Out)<sup>(3)</sup>
- f. Perhitungan laba rugi perusahaan diluar pengerjaan skripsi ini.
- g. Perangkat lunak yang dipakai sebagai penunjang dalam skripsi ini adalah:
  - *MySQL* untuk membuat database.
  - *Borland Delphi 7.0* untuk desain aplikasi program.
  - *Power Designer 6.32* untuk membuat DFD dan ERD.
  - *Crystal Report 10.0* untuk membuat laporan program.

### **1.5 Metodologi Penelitian**

Pengerjaan skripsi ini dilaksanakan dengan langkah pengerjaan sebagai berikut:

#### **1. Melakukan Studi literature**

- Teori perencanaan produksi
- Teori peramalan
- Teori penjadwalan

#### **2. Melakukan survey pada perusahaan**

- Teknik pengumpulan data dengan cara terjun secara langsung dan mencatat secara sistematis terhadap proyek masalah
- Melakukan wawancara terhadap pimpinan perusahaan dan pengamatan secara langsung dilapangan.

### **3. Analisis dan perencanaan system**

- Melakukan analisis dan menentukan sistem yang akan digunakan untuk mendukung penjadwalan produksi pada perusahaan.
- Desain interface yang dibuat menarik dan user friendly.
- Perencanaan database yang akan digunakan.

### **4. Membuat program**

- Merencanakan perangkat lunak aplikasi untuk kemudian digunakan dalam melakukan penjadwalan produksi dari sumber-sumber yang didapat pada perusahaan.

### **5. Pengujian dan analisis**

- Setelah aplikasi selesai dibuat, dilakukan pengujian terhadap aplikasi tersebut, untuk mengetahui apakah dari aplikasi tersebut tujuan dari pengerjaan tugas akhir ini sudah sesuai.

### **6. Pembuatan laporan**

- Penyusunan laporan skripsi.



## **1.6 Sistematika Penyusunan Laporan**

Sistematika penulisan yang digunakan dalam penyusunan skripsi ini adalah sebagai berikut:

### **BAB I : PENDAHULUAN**

Bab ini berisi latar belakang penelitian, perumusan masalah, tujuan penelitian, batasan masalah dan sistematika penulisan skripsi.

### **BAB II : LANDASAN TEORI**

Bab ini berisi tentang teori-teori pendukung yang digunakan dalam penelitian ini dan yang menunjang dalam pembuatan skripsi.

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

Bab ini berisi tentang analisis terhadap sistem yang ada pada perusahaan, perencanaan dan desain sistem yang dijelaskan dengan menggunakan Data Flow Diagram, Entity, Relationship diagram, perancangan file – file database, menu, dan juga desain interface.

### **BAB IV : HASIL DAN ANALISA**

Bab ini berisi tentang hasil pengujian yang telah dibuat secara keseluruhan Dari awal hingga akhir proses.

### **BAB V : PENUTUP**

Bab ini berisi kesimpulan dan saran-saran dari hasil pembahasan skripsi ini.

## BAB II

### LANDASAN TEORI

#### 2.1 Sistem Informasi

Sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Informasi adalah data yang telah diproses atau dapat dikatakan data yang memiliki sebuah arti. Sedangkan data adalah angka dan fakta yang menggambarkan peristiwa yang terjadi dalam suatu organisasi atau lingkungan fisik yang belum diatur atau diproses. Jadi, sistem informasi dapat didefinisikan sebagai perangkat elemen yang bekerja sama dalam mengumpulkan, memproses, menyimpan dan menyebarkan informasi untuk mendukung pengambilan keputusan, koordinasi, pengawasan, analisa, dan visualisasi dalam organisasi<sup>[2]</sup>.

Pengertian secara umum, sistem informasi merupakan kumpulan komponen atau elemen yang bekerja sama untuk mencapai tujuan tertentu sistem merupakan sumber daya yang akan mengubah input menjadi output. Informasi merupakan data yang sudah diolah sehingga berguna dalam pengambilan keputusan. Sistem informasi merupakan susunan dari komponen – komponen yang berhubungan yang saling berhubungan yang saling berinteraksi untuk mendukung kegiatan<sup>[2]</sup>, manajemen informasi dan pengambilan informasi yang dibutuhkan oleh perusahaan atau organisasi

## 2.2 Data Flow Diagram DFD

Data Flow Diagram (DFD) adalah penggambaran sistem secara logika yang menggunakan bentuk - bentuk simbol untuk menggambarkan aliran data melalui suatu proses yang saling terkait tahapan pada pembuatan DFD adalah sebagai berikut:

### ❖ Context Diagram

Context Diagram menggambarkan keseluruhan ruang lingkup dari suatu system perusahaan yang menunjukkan batasan sistem external entities yang berinteraksi dengan perusahaan dan aliran informasi yang ada di antara entities dengan sistem. Pada context Diagram biasanya biasanya terdapat satu proses saja, karena merupakan gambaran keseluruhan system.

### ❖ Level 1

Level 0 dari DFD adalah penjabaran dari Context Diagram. Didalamnya terdapat semua proses yang terjadi dalam sistem. Semua proses disini adalah proses – proses utama dalam system.

### ❖ Level 2

Level 2 dari DFD adalah penjabaran dari setiap proses yang ada didalam level 1 DFD. Setiap proses memiliki penjabaran masing – masing sehingga terdapat beberapa DFD level 2. Hal ini yang membedakan dengan Context Diagram dan dari DFD yang masing – masing hanya ada satu. Level 2 dari DFD adalah sebuah penjabaran spesifik dan merupakan batas minimal penjabaran proses – proses yang ada didalam system.

❖ Dan seterusnya

Ada beberapa hal yang harus diperhatikan dalam pembuatan DFD hal – hal tersebut adalah :

- Nama proses dalam Context Diagram harus sama dengan nama system tersebut.
- Setiap bagian dari DFD harus terletak pada satu halaman, tidak boleh bersambung di halaman yang lain.
- Setiap symbol harus memiliki nama yang unik serta menjelaskan mengenai isi symbol dsn konsisten dalam setiap bagian DFD baik pada Context Diagram, level 1, level2 dan seterusnya
- Hindari garis berpotongan jika memungkinkan, jika tidak beri tanda khusus.
- Setiap proses dalam DFD selalu memiliki input dan output.
- Setiap data storage hanya memiliki hubungan input dan output dengan proses saja.
- Data flow tidak boleh menghubungkan antara external entities dengan data storage maupun dengan entities yang lain.

Berikut ini adalah beberapa simbol – symbol yang dipakai dalam pembuatan DFD :

a. External entities (entitas luar)



Gambar 2.1 entity

External entity berada diluar sistem. Elemen yang melambangkan kesatuan dilingkungan diluar sistem yang dapat berupa orang, organisasi, maupun sistem lain yang dapat memberikan output maupun menerima input dari system.

b. Proses



Gambar 2.2 Proses

Proses melambangkan kegiatan yang dilakukan orang, mesin, atau computer dari hasil suatu arus data yang masuk kedalam suatu proses untuk menghasilkan suatu arus data yang keluar dari proses . proses ini juga diberikan sebuah nomor indentifikasi yang unik untuk menunjukan level dari Diagram.

c. Data Flow



Gambar 2.3 Data Flow

Data flow menggambarkan aliran data yang dapat berupa input ataupun output dari proses

d. Data Storage (Penyimpanan Data)



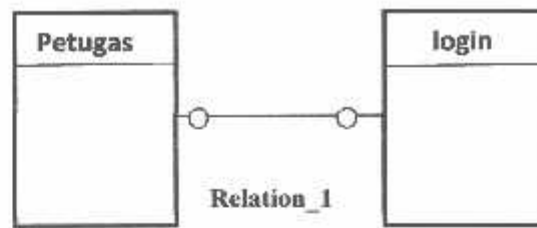
Gambar 2.4 Data Storage

Data Storage adalah tempat untuk menyimpan data yang akan digunakan sebagai referensi. Data storage tidak menjelaskan detail karakteristik seperti bagaimana pengaturannya maupun bagaimana pengaksesannya.

Ada beberapa macam hubungan antar entity, yaitu :

a) One - to - one

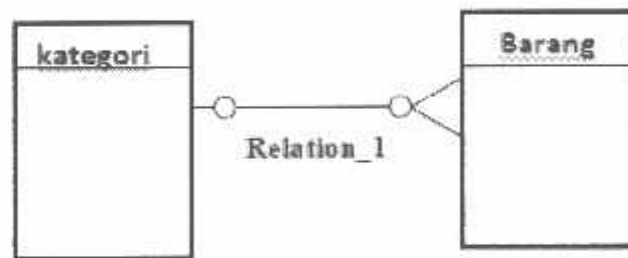
One - to - one relationship merupakan suatu hubungan dimana suatu anggota entity mempunyai hubungan dengan satu anggota entity yang lain. Contoh : satu petugas hanya mempunyai satu login, demikian pula satu login hanya dimiliki oleh satu petugas.



gambar 2.5. One - to - one relationship

b) One - to - many

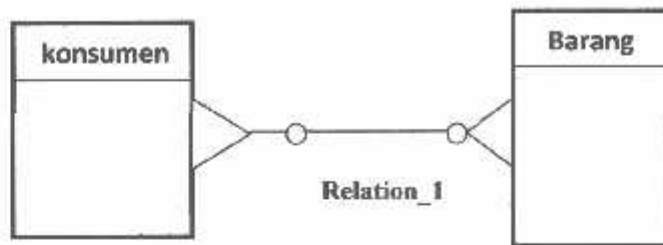
One - to - many relationship merupakan hubungan antara beberapa anggota entity yang satu dengan satu dari entiti yang lain. Contoh: setiap barang harus berada dalam satu katagori, tetapi satu kategori dapat terdiri dari beberapa barang.



gambar 2.6. One - to - many relationship

c) Many - to - many

Many - to - many relationship merupakan hubungan antara beberapa anggota entity yang satu dengan beberapa anggota entity yang lain, contoh : satu macam barang dapat dibeli oleh beberapa konsumen, dan satu konsumen dapat membeli beberapa macam barang.



Gambar 2.7 Many - to - many relationship

### 2.3 Data Management System

Database management system ( DBMS) adala suatu kumpulan basis data yang saling berelasi satu sama lain dan suatu bentuk program untuk mengakses basis data tersebut. 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 memperbarui data data yang ada. Pembuatan data base yang baik harus melalui tahap tahap dibawah ini antara lain:

- a) Planning, tujuan dari tahap ini adalah untuk menderteminasi apakah system yang dibuat layak atau tidak
- b) Analisis, pada tahap ini aktivitas yang dilakukan adalah mengidentifikasi kebutuhan user seperti akan kebutuhan perangkat keras serta perangkat lunak yang diperlukan dalam pembuatan basis data, pengindentifikasian seperti interview diperusahaan bersangkutan.



c) Desain , tahap ini dibagi menjadi tiga bagian, yaitu:

- 1) Konseptual desain, konsep yang dilakukan pada tahap ini adalah menerjemahkan data requirement menjadi konseptual skema dari basis data atau bias disebut DFD
- 2) Logica desain setelah konseptual skema selesai dibuat, dilakukan pemilihan database management system (DBMS) yang akan digunakan untuk menerjemahkan konseptual model tersebut, agar dapat diterapkan pada(DBMS) yang telah dipilih.
- 3) Fisikal desain, pada tahap ini dilakukan pembuatan fisik skema yang didasarkan fisik desain yang telah dibuat sebelumnya.
- d) Coding, pada tahap ini desain yang telah dibuat sebelumnya diubah kedalam bentuk program aplikasi basis data.
- e) Implementasi, melakukan uji coba pada program aplikasi yang baru dibuat
- f) Operational dan maintenance, kegiatan yang telah dilakukan memonitoring kinerja yang dibuat serta melakukan maintenance akan program yang telah dibuat.

#### **2.4. Pengertian Peramalan Metode Winter.**

Peramalan adalah proses untuk memperkirakan beberapa kebutuhan dimasa mendatang yang meliputi kebutuhan dalam ukuran kuantitas, kualitas, waktu dan lokasi yang dibutuhkan dalam rangka memenuhi permintaan barang atau jasa<sup>[3]</sup>. Dengan semakin kompleks dan dinamisnya kondisi pasar juga dipengaruhi oleh keadaan social, ekonomi, politik, competitor . peramalan yang akurat merupakan informasi yang sangat dibutuhkan dalam pengambilan keputusan manajemen.

Peramalan winter merupakan tingkat permintaan produk – produk yang diharapkan akan terrealisasi dalam batas waktu yang tertentu pada masa yang

mendatang. Peramalan winter ini menjadi masukan yang sangat penting dalam keputusan perencanaan dan pengendalian perusahaan<sup>[3]</sup>. Karena bagian operational produksi bertanggung jawab terhadap pembuatan produk yang dibutuhkan konsumen, maka keputusan operasi produk sangat dipengaruhi dari hasil peramal permintaan dari produk yang bersifat bebas. Metode winter mencoba untuk memberikan hasil peramalan tanpa memperdulikan pola data yang terbentuk apakah bersifat stasioner atau mengandung factor musiman karena metode ini dilakukan dengan memberikan pemulusan untuk kedua factor tersebut.

Pola data yang mungkin muncul adalah trend, siklus, musiman dan acak trend adalah suatu sifat jangka panjang dalam suatu data rangkaian waktu yang dapat digambarkan dengan suatu garis lurus atau kurva , dengan kecenderungan naik, kecenderungan turun atau naik juga mungkin konstan. Analisa trend mempunyai implikasi sebagai perencanaan Manajerial jangka panjang. Sirkulus adalah suatu pola rangkaian data yang berulang secara periodic. Biasanya lebih dari satu tahun. musiman (sesasional) adalah factor yang berulang tiap tahun misalnya perubahan cuaca dan musim. Analisa(sesasional) pada data rangkaian waktu mempunyai implikasi jangka pendek. Sedangkan acak adalah factor yang tidak bias dicari polanya, biasanya disebabkan oleh hal-hal yang tidak dapat dikendalikan.

Beberapa sifat hasil peramalan:

1. Peramalan pasti mengandung kesalahan, artinya peramalan hanya bias mengurangi ketidakpastian yang terjadi, tetapi tidak menghilangkan ketidakpastian tersebut.

2. Peramalan jangka pendek lebih akurat dibandingkan peramalan jangka panjang hal ini disebabkan karena dalam peramalan jangka pendek, faktor-faktor yang mempengaruhi permintaan relatif masih konstan.

## 2.5 Perencanaan Produksi

Pada dasarnya perencanaan produksi merupakan suatu proses penetapan output manufaktur secara keseluruhan guna memenuhi tingkat penjualan yang direncanakan dan inventori yang diinginkan<sup>[4]</sup>, perencanaan produksi menetapkan kerangka kerja untuk penjadwalan produksi dan pelaksanaan dari manufaktur, proses yang terjadi dalam perencanaan produksi dapat dijelaskan dalam empat langkah utama, yaitu :

- 1) Menumpulkan data yang relevan dengan perencanaan produksi
- 2) Menggunakan data yang relevan itu menjadi informasi yang teratur
- 3) Menggunakan kapasitas produksi, berkaitan dengan sumber daya yang ada
- 4) Partnership meeting, dimana para manajer atau direktur yang bersangkutan melakukan pertemuan untuk mengambil keputusan akan perencanaan produksi, rencana produksi mengacu pada permintaan total, sehingga formula umum perencanaan produksi, ialah

$$\text{Rencana produksi} = (\text{permintaan total} - \text{inventori awal}) + \text{inventori akhir}$$

## 2.6 Penjadwalan Produksi

Penjadwalan merupakan proses penagaturan pembuatan produk secara menyeluruh pada sejumlah proses tertentu

*First – come, first – served (FCFS)* tugas pertama datang ke pusat kerja diproses terlebih dahulu metode ini digunakan apabila waktu kerja yang tersisa untuk

*completing order* relative sama<sup>13)</sup>, FCFS akan cocok untuk flow proses karena memiliki remaining times yang serupa. Dapat dilakukan beberapa perhitungan sebagai informasi guna mengukur efektifitas dari metode FCFS *First – come, first – served (FCFS)* yang merupakan proses pengurutan pembuatan produk secara menyeluruh pada batas tenggang waktu yang tercepat, juga disertai dengan prioritas beban produksi minimum sehingga *job – job* yang ada akan dijadwalkan dengan baik, karena hal tersebut juga mempengaruhi penjadwalan proses produksi.

## **BAB III**

### **ANALISIS DAN DESAIN SISTEM**

#### **3.1. Analisis Sistem**

##### **3.1.1 Analisis Sistem yang Lama**

Pada saat ini system informasi yang dimiliki oleh UD Sutra yang bergerak dibidang makanan (Roti) masih secara manual, dalam arti masih belum menggunakan teknologi computer untuk melakukan penyimpanan dan pengolahan data yang ada dalam perusahaan. Proses pemesanan yang ada juga terjadi secara langsung yaitu melalui kontak langsung dengan pemilik perusahaan. Melakukan produksi pun masih dengan mengandalkan firasat pemilik yang diperoleh dari pengalamannya sendiri. Data - data yang tersimpan pada pemilik perusahaan tersebut juga tidak digunakan untuk membenahi kinerja produksi dari perusahaan, system yang ada pada bagian proses produksi hanya melaksanakan apa yang diminta oleh pemilik perusahaan, sehingga apabila ada terjadi pemesanan produk yang tidak dapat dipenuhi oleh stok sekarang maka pemilik sering menghadapi permasalahan pada perencanaan produksi yang sedang berlangsung saat itu. Hal ini dikarenakan perusahaan ini masih belum memiliki system peramalan dan penjadwalan produksi.

### **3.1.2 Analisis Permasalahan**

Dengan keadaan system lama yang ada dalam perusahaan pada saat ini, maka beberapa kendala yang dihadapi adalah sebagai berikut :

- a. Kesulitan dalam penyimpanan dan mencari data permintaan yang pernah terjadi pada perusahaan.
- b. Kesulitan untuk mengetahui data permintaan yang terjadi pada periode waktu tertentu karena tidak adanya laporan yang pasti.
- c. Kesulitan untuk menjalankan proses produksi yang efisien karena tidak adanya penjadwalan produksi, sehingga dalam memenuhi permintaan dari pasar sering mengalami kendala.
- d. Kesulitan dalam mengatur stok agar tidak berlebihan atau kekurangan pada bagian gudang.
- e. Kesulitan dalam pengolahan data maupun laporan – laporan yang ada pada perusahaan.

### **3.1.3 Analisis kebutuhan**

Dari analisis system lama dan permasalahan yang ada pada perusahaan. Maka analisis baru yang dibutuhkan oleh perusahaan yang dibutuhkan adalah sebagai berikut:

- a. Suatu system yang dapat menyimpan dan pemesanan yang terjadi pada perusahaan.

- b. Suatu system yang dapat membantu untuk meramalakan pemesanan permintaan suatu periode tertentu, sehingga stok perusahaan dapat disiapkan agar tidak terjadi selisih persediaan yang banyak pada perusahaan.
- c. Suatu system yang dapat memberikan perencanaan proses produksi sehingga didapat informasi kapan produksi di mulai dan kapan proyek tersebut selesai.
- d. Suatu system yang dapat membantu menjadwalkan produksi secara efisien yang mengacu pada pemesanan yang terjadi pada perusahaan.
- e. Suatu system yang dapat memberikan laporan-laporan yang diinginkan sesuai kebutuhan.

Dari keterangan diatas Perusahaan ini menunjukkan system baru yang akan digunakan oleh perusahaan dengan adanya dari pembuatan aplikasi progam ini, alur system yang baru dapat diterangkan sebagai berikut. Pelangan yang melakukan pemesanan akan dikumpulkan terlebih dahulu sebelum melakukan proses perproduksi, setelah pemesanan telah di kumpulkan akan dilakukan pengiriman barang yang dapat dipenuhi dari stok barang yang ada pada perusahaan. Apabila stok tidak dapat dipenuhi maka akan dilakukan perencanaan produksi untuk memenuhi pemesanan tersebut. Setelah didapat perencanaan produksi maka akan dilakukan perhitungan penjadwalan berdasarkan perencanaan yang telah dibuat, setelah itu penjadwalan akan dipilih oleh perusahaan berdasarkan metode yang ada yang diinginkan oleh perusahaan. Setelah itu akan dilakukan pelaksanaan realisasi produksi. Pada saat penutupan produksi akan dilakukan penyesuaian untuk mengupdate data stok barang yang sudah selesai produksi serta penggunaan akan bahan baku yang terpakai. Setelah produksi selesai

maka pemesanan yang ada akan dipenuhi dari hasil produksi yang telah dilakukan oleh perusahaan, berdasarkan dari data pemesanan produk ke pelanggan yang dimiliki perusahaan akan dapat dilakukan peramalan apabila data yang dimiliki cukup untuk dilakukan perhitungan peramalan dengan menggunakan metode winter,<sup>[3]</sup> dari hasil peramalan maka data ketersediaan kebutuhan bahan baku untuk perusahaan akan dapat dipenuhi. Penyimpanan data – data yang diperlukan oleh perusahaan akan tersimpan dengan baik pada *database* yang telah disediakan oleh aplikasi, dan data-data tersebut dapat digunakan untuk melakukan perencanaan yang lebih baik bagi perusahaan.

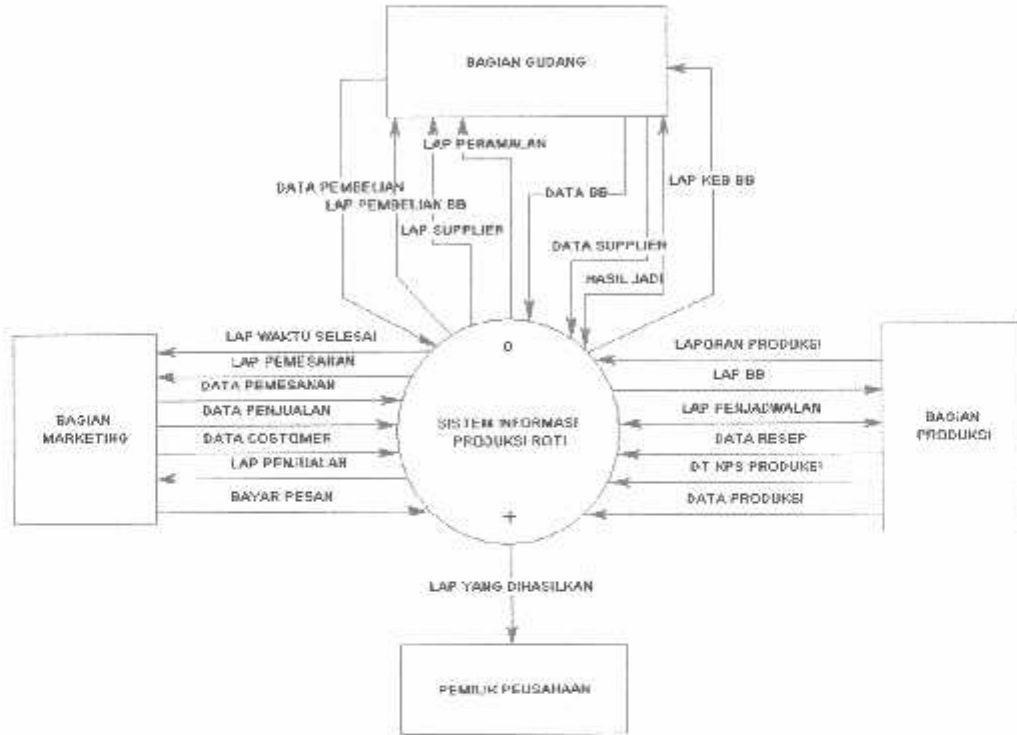
### **3.2 Desain Sistem**

Pembuatan system meliputi pembuatan *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD), desain table *Database* serta desain *Interface*



### 3.2.1 Data Flow Diagram (DFD)

Konteks diagram untuk untuk perusahaan makanan dapat digambarkan pada gambar 3.1 Konteks Diagram, ini merupakan gambaran secara umum aktifitas-aktifitas dalam UD Roti.



Gambar 3.1 Konteks Diagram



Konteks Diagram ini kemudian akan di jabarkan menjadi DFD level 1 Pada gambar 3.2. pada DFD level 1 ada beberapa proses utama yaitu :

➤ Proses 1: Proses pemesanan.

Proses ini berfungsi untuk memasukan data pemesanan dan pengiriman produk.

➤ Proses 2 : Proses Penerimaan Bahan Baku.

Proses ini berfungsi untuk melakukan Penyimpanan data bahan baku yang diperlukan untuk kebutuhan perusahaan dalam melaksanakan produksi

➤ Proses 3 : Proses Penjadwalan dan Produksi.

Proses ini berfungsi untuk melakukan penjadwalan dan produksi produk dari perencanaan yang didapat dari data pemesanan atau perencanaan yang dibuat sendiri dari pemilik perusahaan.

➤ Proses 4 : Proses Peramalan.

Proses ini berfungsi untuk meramalkan jumlah produk yang akan diproduksi pada periode tertentu, juga sebagai persiapan kebutuhan bahan baku yang akan digunakan pada waktu melakukan produksi pada perusahaan.

➤ Proses 5 : Proses Penjualan.

Proses ini berfungsi untuk melakukan penjualan produk dari bagian marketing yang di dapat dari data pemesanan yang dibuat oleh bagian marketing atas permintaan pelanggan yang pesan.

➤ Proses 6 : Inventori Bahan Baku

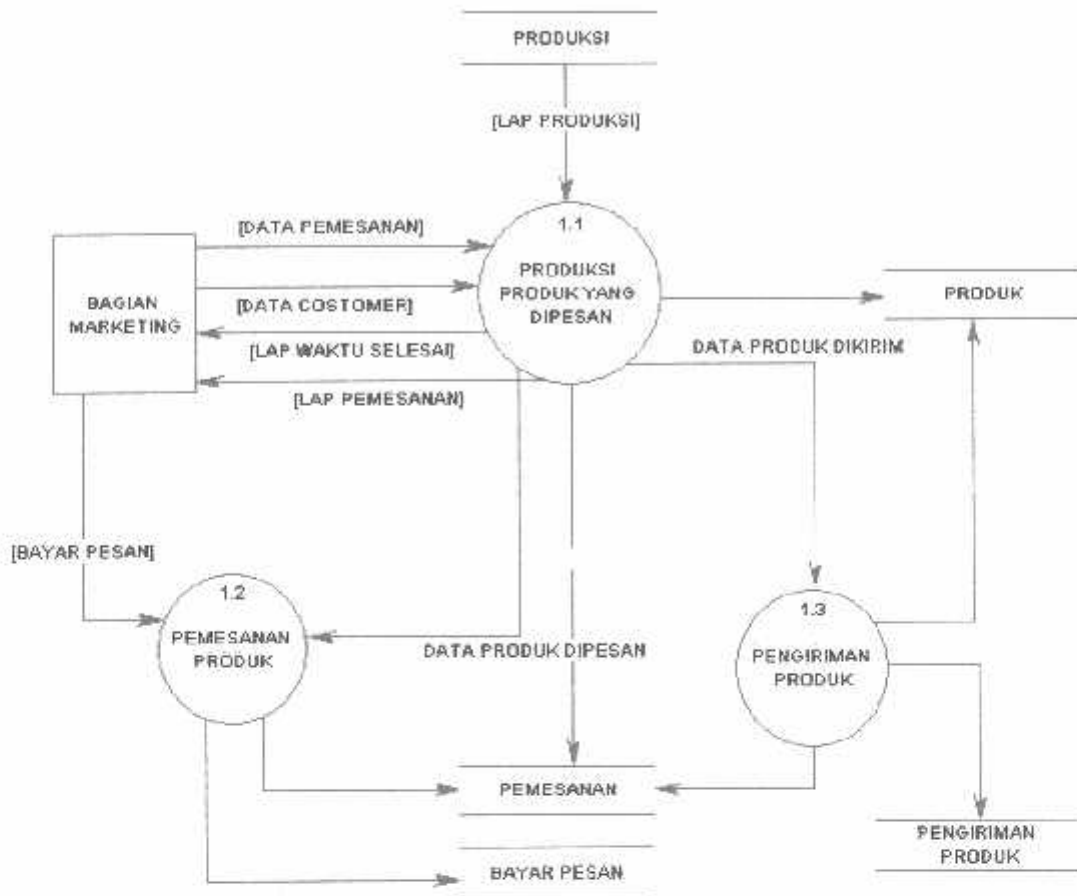
Proses ini berfungsi untuk melakukan Penyimpanan data bahan baku yang diperlukan untuk kebutuhan perusahaan dalam melaksanakan produksi yang diperoleh dari data supplier yang dibuat oleh proses pembelian.

➤ Proses 7 : Pembelian

Proses ini berfungsi untuk melakukan pembelian produk, yang data tersebut diperoleh dari data supplier.

➤ Proses 8 : Pembuatan laporan.

Proses ini akan membuat laporan berdasarkan pada data – data yang ada sesuai dengan kebutuhan perusahaan.



**Gambar 3.3 DFD Level 2**

### **Proses Pemesanan dan Pengiriman Produk**

Ada beberapa Proses pada DFD level 1 ini yang dapat di pecah lagi menjadi DFD level 2. Proses – proses tersebut adalah proses produksi produk yang dipesan, proses pemesanan produk, dan proses pengiriman produk. Proses pemesanan dan pengiriman produk pada DFD level 2 terdapat 3 proses yaitu:

➤ Proses 1.1 : Proses produksi produk yang dipesan

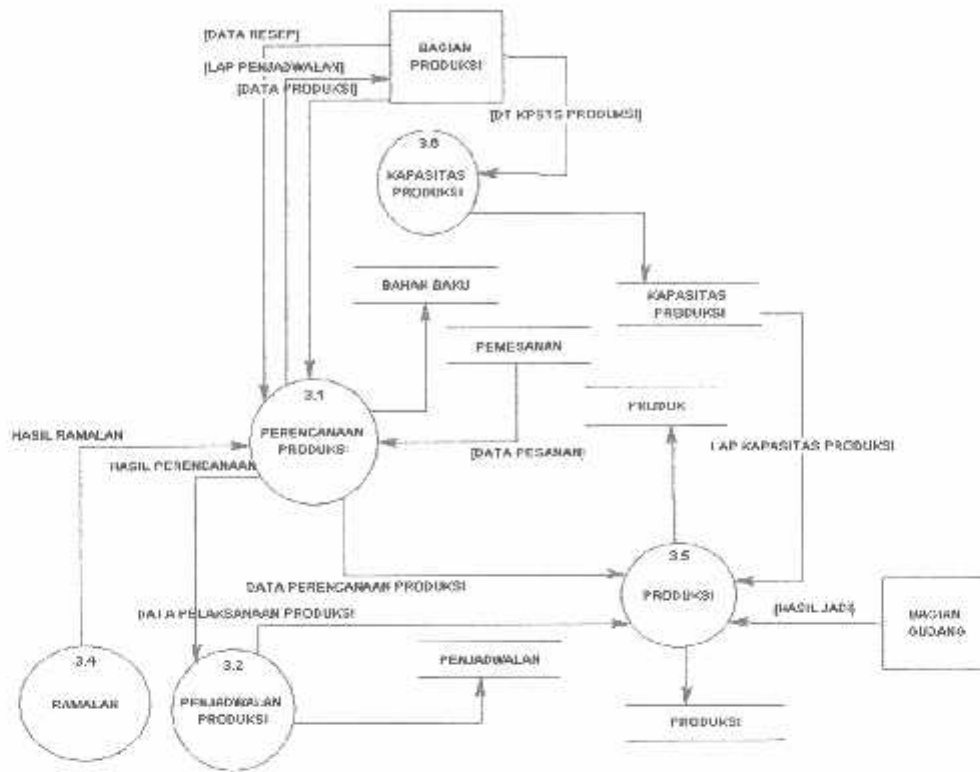
Proses ini akan melakukan pengecekan produk yang dipesan dapat dipenuhi dengan stok yang ada atau tidak. Jika stok tidak mencukupi akan dilakukan pengecekan pada produksi apakah produk yang di pesan sedang di produksi atau tidak, bila sedang diproduksi maka akan menunggu sampai produk selesai, apabila belum diproduksi maka akan dilakukan pemesanan produk.

➤ Proses 1.2: Proses Pemesanan Produk

Proses ini akan memberikan info ke bagian produksi tentang produk yang di pesan.

➤ Proses 1.3 : Proses Pengiriman Produk.

Proses ini akan mengurangi data produk yang dikirim serta data yang dikirim ke pembeli masuk data pengiriman produk yang akan menjadi data actual perusahaan.



**Gambar 3.4 Level 2 proses Penjadwalan produksi**

Pada DFD level 2 proses penjadwalan produksi terdapat 5 proses, yaitu :

➤ **Proses 3.1 Proses Rencana Produksi.**

Proses ini berfungsi untuk melakukan perencanaan produksi dari data yang dimasukkan oleh bagian produksi, pada proses ini data bahan baku akan dikurangi terlebih dahulu sebagai persiapan dalam produksi dan dapat dilakukan pengecekan minimum produksi.

➤ Proses 3.2 Proses Penjadwalan Produksi

Proses ini akan membuat penjadwalan produksi dari hasil perencanaan dengan metode FIFO, informasi Penjadwalan akan diberikan kebagian produksi atau ditentukan pelaksanaannya produksi.

➤ Proses 3.3 proses peramalan

Proses ini berfungsi untuk meramalkan banyaknya produk yang perlu diproduksi, info ini kemudian diberikan kebagian gudang untuk menentukan kebutuhan bahan baku pada periode yang dibutuhkan.

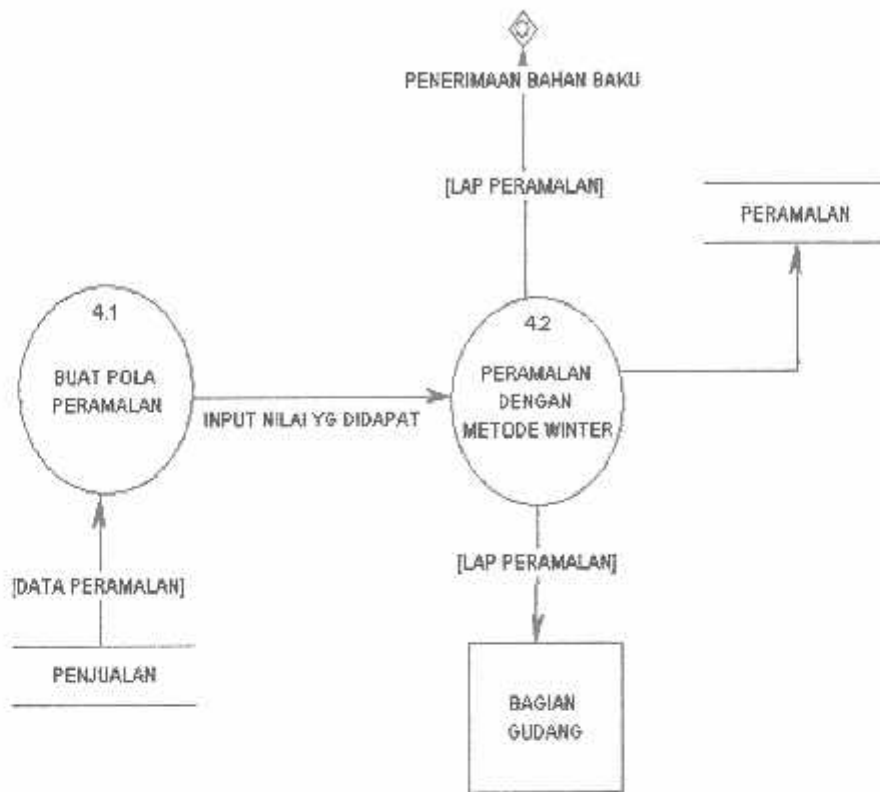
➤ Proses 3.4 Produksi

Proses ini akan melaksanakan produksi sesuai dengan data dari bagian produksi, pada pelaksanaan ini akan dicatat realisasi data waktu produk pada tiap proses, serta data bahan baku yang terpakai dan danproduk yang terjadi secara actual akan di update pada waktu penutupan produksi.

➤ Proses 3.5 Kapasitas Produksi

Proses ini berfungsi untuk menyimpan kapasitas produksi, dari banyaknya produk yang akan diproduksi info ini akan diberikan ke proses produksi untuk melakukan produksi.





**Gambar 3.5 DFD Level 2 Peramalan**

Pada DFD level 2 proses peramalan produksi terdapat 2 proses,yaitu:

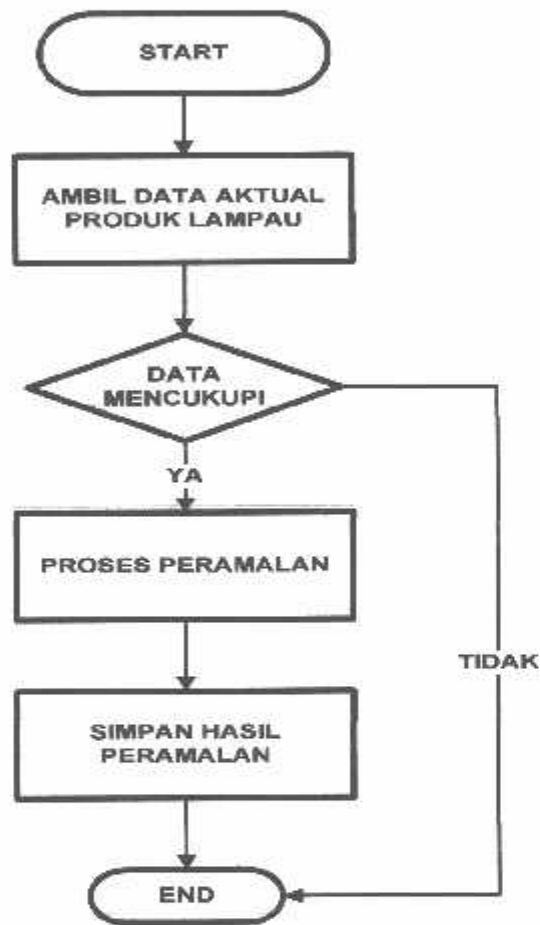
- Proses 1. Proses pembuatan Pola Peramalan.

Proses ini berfungsi untuk menentukan sendiri nilai Alpa, Beta, Gama sesuai dengan yang diinginkan, nilai ini akan digunakan untuk melakukan peramalan.

➤ Proses 2. Proses Peramalan

Proses ini berfungsi untuk meramalkan banyaknya produk yang perlu diproduksi, info ini kemudian diberikan ke bagian gudang untuk menentukan kebutuhan bahan baku pada periode yang dibutuhkan.

**3.2.2 Program Flowchart Peramalan**



**Gambar 3.6 Program Flowchart Peramalan**

Pada gambar 3.6 menunjukkan program flowchart peramalan. Pada sistem yang baru yang ada pada perusahaan. Untuk memulai peramalan dilakukan

pengambilan data actual masa lampau yang diperlukan agar dapat dilakukan peramalan dengan metode Winter untuk memperoleh nilai-nilai trend, musiman, dan acak. Kemudian nilai tersebut akan digunakan untuk meramalkan perencanaan produksi pada periode ke depan yang akan dilakukan oleh perusahaan. Data dari peramalan akan kemudian akan di simpan oleh perusahaan.

### 3.2.3 Program Flowchart Penjadwalan FIFO (First In First Out)

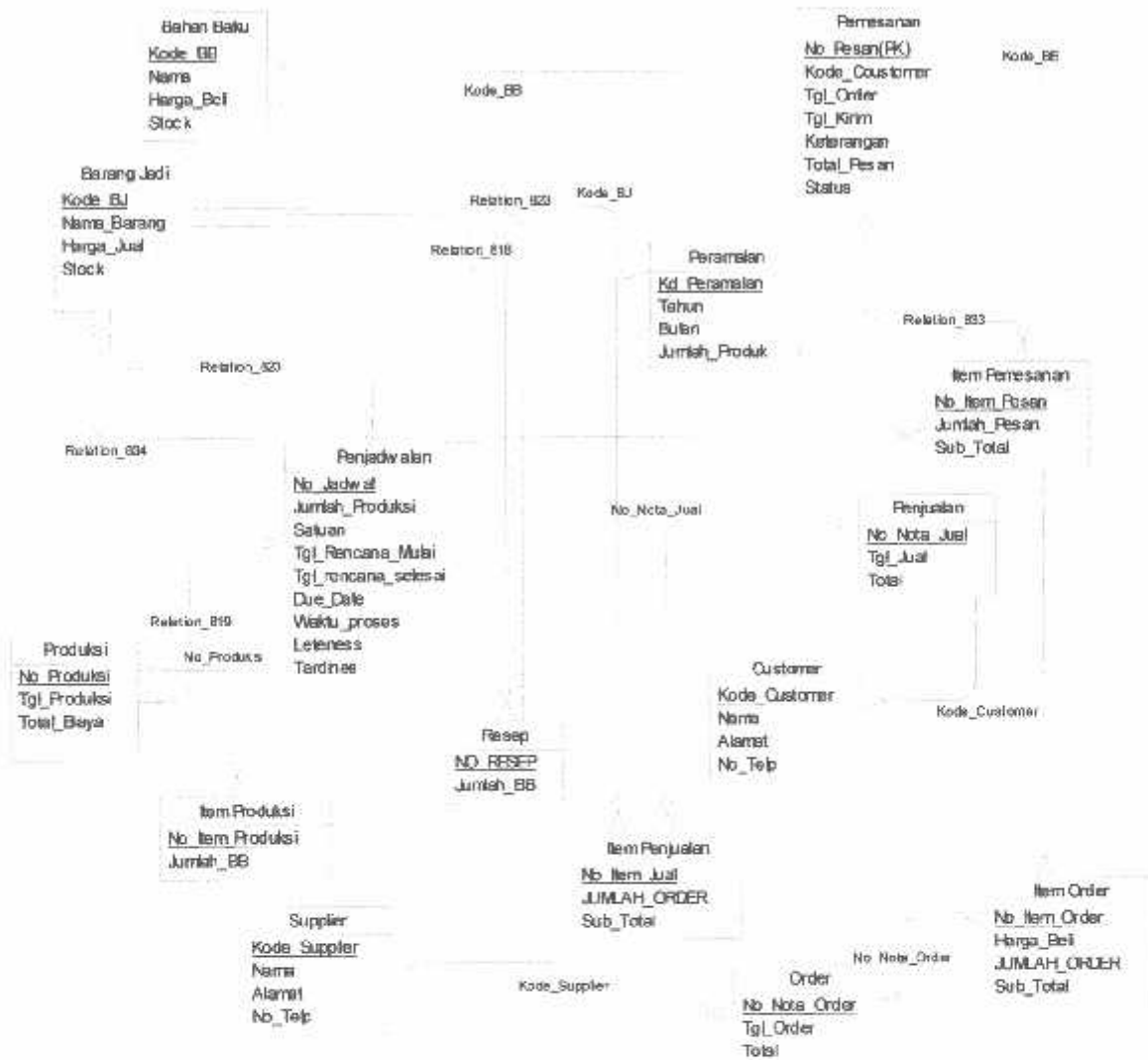


**Gambar 3.7 Program Flowchart Penjadwalan (FIFO)**

Gambar 3.7 menunjukkan program flowchart penjadwalan FIFO pada system baru yang ada dalam perusahaan. Penjadwalan ini dilakukan dengan mengumpulkan

data permintaan kemudian dibuat perencanaan produksi yang dapat disesuaikan dengan (n) hari untuk ditentukan penjadwalan produksinya. Perencanaan produksi tersebut diurutkan berdasarkan batas waktu yang terdekat. Tetapi bila ada data dengan batas waktu yang sama maka akan dipilih nilai terkecil dari selisi antara waktu proses dengan batas waktu permintaan tersebut.

### 3.2.4 Conceptual Data Model (CDM)



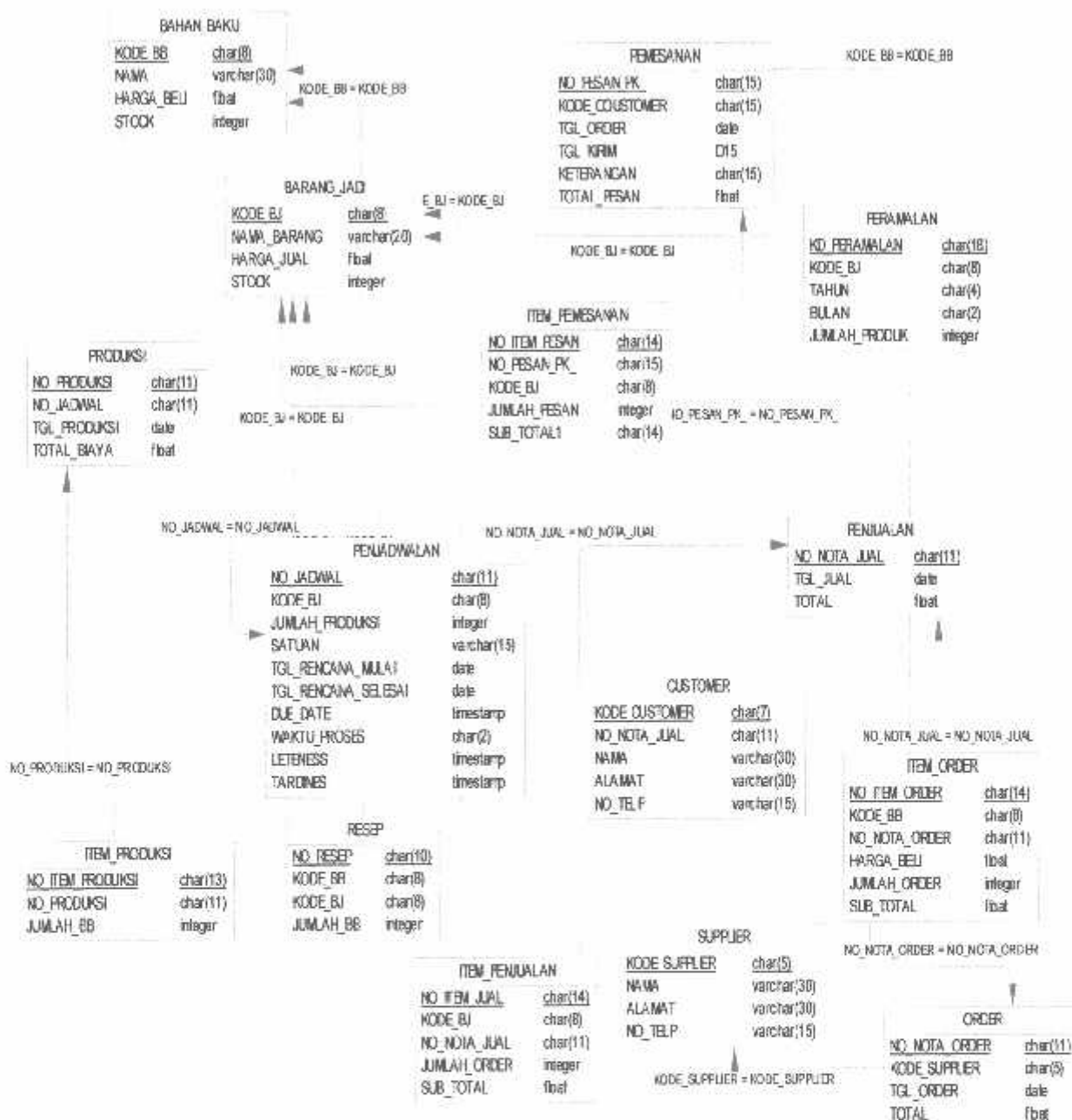
**Gambar 3.8 Conceptual Data Model (CDM)**

Gambar 3.8 menunjukan rancangan conceptual Entity Relationship Diagram dari system yang dibuat.berikut ini adalah penjelasannya relasi antara table :

1. Entitas T\_Bahan\_baku dengan T\_Item\_Order memiliki hubungan one to many, yang memiliki arti bahwa dari setiap bahan baku dapat melakukan beberapa Order,.
2. Entitas T\_Bahan\_baku dengan T\_Resep memiliki hubungan one to many, yang memiliki arti bahwa dari setiap bahan baku dapat melakukan beberapa Resep,.
3. Entitas T\_Barang\_Jadi dengan T\_Item\_Penjualan memiliki hubungan one to many, yang memiliki arti bahwa dari setiap barang jadi memiliki beberapa item penjualan, sedangkan item penjualan hanya dapat memiliki satu barang jadi.
4. Entitas T\_Barang\_Jadi dengan T\_Resep memiliki hubungan one to many, yang memiliki arti bahwa dari setiap Barang\_Jadi dapat memiliki beberapa Resep
5. Entitas T\_Barang\_Jadi dengan T\_Peramalan memiliki hubungan one to many, yang memiliki arti bahwa dari setiap Barang\_Jadi dapat memiliki beberapa Peramalan.
6. Entitas T\_Barang\_Jadi dengan T\_Penjadwalan memiliki hubungan one to many, yang memiliki arti bahwa dari setiap barang jadi dapat dapat memiliki beberapa penjadwalan produk, sedangkan penjadwalan produk hanya dapat satu barang jadi
7. Entitas T\_Barang\_Jadi dengan T\_Item\_Pemesanan memiliki hubungan one to many, yang memiliki arti bahwa dari setiap barang jadi dapat memiliki beberapa item pemesanan, sedangkan setiap item pemesanan hanya dapat memiliki satu barang jadi.
8. Entitas T\_Pemesanan dengan T\_Item\_Pemesanan memiliki hubungan one to many, yang memiliki arti bahwa dari setiap proses pemesanan dapat memiliki beberapa Item pemesanan,.

9. Entitas T\_Penjadwalan dengan T\_Produksi memiliki hubungan one to many, yang memiliki arti bahwa dari setiap penjadwalan dapat menjadwalkan beberapa produksi,
10. Entitas T\_Produksi dengan T\_Item\_Produksi memiliki hubungan one to many, yang memiliki arti bahwa dari setiap produksi dapat memiliki beberapa item produksi, sedangkan setiap item produksi dapat memiliki satu produksi.
11. Entitas T\_Penjualan dengan T\_Customer memiliki hubungan one to many, yang memiliki arti bahwa dari setiap penjualan dapat memiliki beberapa customer .
12. Entitas T\_Penjualan dengan T\_item\_Penjualan memiliki hubungan one to many, yang memiliki arti bahwa dari setiap penjualan dapat memiliki beberapa item penjualan.
13. Entitas T\_Supplier dengan T\_Order memiliki hubungan one to many, yang memiliki arti bahwa dari setiap supplier dapat memiliki beberapa order, sedangkan setiap order hanya dapat memiliki satu supplier.
14. Entitas T\_Order dengan T\_Item\_Order memiliki hubungan one to many, yang memiliki arti bahwa dari setiap order dapat memiliki beberapa item order.

### 3.2.5 Physical Data Model ( PDM)



Gambar 3,9 Physical Data Model ( PDM)



### 3.2.6 Desain Tabel Database

#### 1. Tabel Bahan Baku

Tabel ini digunakan untuk menyimpan bahan baku yang dimiliki oleh perusahaan

Name	Type	Length	Decimals	Allow Null	
▶ kode_bb	char	8	0	<input type="checkbox"/>	
nama	varchar	30	0	<input checked="" type="checkbox"/>	
harga_beli	float	0	0	<input checked="" type="checkbox"/>	
stock	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.1 T\_Bahan\_baku

#### 2. Tabel Barang Jadi

Tabel ini digunakan untuk menyimpan semua barang jadi yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ kode_bj	char	8	0	<input type="checkbox"/>	
nama	varchar	20	0	<input checked="" type="checkbox"/>	
harga_jual	float	0	0	<input checked="" type="checkbox"/>	
stock	int	11	0	<input checked="" type="checkbox"/>	
stock_pesanan	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.2 T\_Barang\_jadi

### 3. Tabel Customer

Tabel ini digunakan untuk menyimpan semua data dari customer yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ kode_customer	char	7	0	<input type="checkbox"/>	
nama	varchar	30	0	<input checked="" type="checkbox"/>	
alamat	varchar	30	0	<input checked="" type="checkbox"/>	
no_telp	varchar	15	0	<input checked="" type="checkbox"/>	

Tabel 3.3 T\_Customer

### 4. Tabel Item Order

Tabel digunakan untuk menyimpan data item order yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ no_item_order	char	14	0	<input type="checkbox"/>	
No_nota_order	char	11	0	<input checked="" type="checkbox"/>	
kode_bb	char	6	0	<input checked="" type="checkbox"/>	
harga_beli	float	0	0	<input checked="" type="checkbox"/>	
jumlah_order	int	11	0	<input checked="" type="checkbox"/>	
sub_total	float	0	0	<input checked="" type="checkbox"/>	

Tabel 3.4 T\_Item\_Order

## 5. Tabel Item Pesan

Tabel ini digunakan untuk menyimpan semua data item pesan yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
no_item_pesan	char	14	0	<input type="checkbox"/>	
no_pesan	char	11	0	<input checked="" type="checkbox"/>	
kode_bj	char	8	0	<input checked="" type="checkbox"/>	
jumlah_pesan	int	4	0	<input checked="" type="checkbox"/>	
sub_total	float	0	0	<input checked="" type="checkbox"/>	

Tabel 3.5 T\_Item\_Pesan

## 6. Tabel Item Jual

Tabel ini digunakan untuk menyimpan semua data item jual yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
no_item_jual	char	14	0	<input type="checkbox"/>	
no_nota_jual	char	11	0	<input checked="" type="checkbox"/>	
kode_bj	char	8	0	<input checked="" type="checkbox"/>	
jumlah_jual	float	11	0	<input checked="" type="checkbox"/>	
sub_total	float	0	0	<input checked="" type="checkbox"/>	

Tabel 3.6 T\_Item\_jual

## 7. Tabel Item Produksi

Tabel ini digunakan untuk menyimpan data item produksi yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
no_itemproduksi	char	13	0	<input type="checkbox"/>	
noproduksi	char	11	0	<input checked="" type="checkbox"/>	
no_resep	char	16	0	<input checked="" type="checkbox"/>	
jumlah_bb	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.7 T\_Item\_Produksi

## 8. Tabel Item Produksi pesan

Tabel ini digunakan untuk menyimpan data item produksi pesan yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
no_itemproduksi	char	13	0	<input type="checkbox"/>	
noproduksi	char	11	0	<input checked="" type="checkbox"/>	
no_resep	char	16	0	<input checked="" type="checkbox"/>	
jumlah_bb	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.8 T\_Item\_Produksi\_pesan

## 9. Tabel Order

Tabel ini digunakan untuk menyimpan semua data dari order yang dimiliki perusahaan.

Name	Type	Length	Decimals	Allow Null	
no_nota_order	char	11	0	<input type="checkbox"/>	
kode_supplier	char	15	0	<input checked="" type="checkbox"/>	
tgl_order	date	0	0	<input checked="" type="checkbox"/>	
total	float	0	0	<input checked="" type="checkbox"/>	

Tabel 3.9 T\_Order

## 10. Tabel Pemesanan

Tabel ini digunakan untuk menyimpan semua data pemesanan yang dimiliki perusahaan.

Name	Type	Length	Decimals	Allow Null	
no_pesan	char	11	0	<input type="checkbox"/>	
kode_customer	char	7	0	<input checked="" type="checkbox"/>	
tgl_pesan	date	0	0	<input checked="" type="checkbox"/>	
total_pesan	float	0	0	<input checked="" type="checkbox"/>	
tgl_selesai	date	0	0	<input checked="" type="checkbox"/>	
Uang_Muka	float	0	0	<input checked="" type="checkbox"/>	
status	tinyint	1	0	<input checked="" type="checkbox"/>	

Tabel 3.10 T\_Pemesanan

## 11. Tabel Penjadwalan

Tabel ini berfungsi untuk menyimpan detail proses penjadwalan produksi dengan metode FIFO.

Name	Type	Length	Decimals	Allow Null	
▶ No_jadwal	char	12	0	<input type="checkbox"/>	
Tgl_jadwal	date	0	0	<input checked="" type="checkbox"/>	
no_pesan	char	11	0	<input checked="" type="checkbox"/>	
kode_bj	char	8	0	<input checked="" type="checkbox"/>	
jumlah	float	11	0	<input checked="" type="checkbox"/>	
status	tinyint	4	0	<input checked="" type="checkbox"/>	

Tabel 3.11 T\_Penjadwalan

## 12. Tabel Penjualan

Tabel ini berfungsi untuk menyimpan data penjualan yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ no_nota_jual	char	11	0	<input type="checkbox"/>	
tgl_jual	date	0	0	<input checked="" type="checkbox"/>	
kode_customer	char	7	0	<input checked="" type="checkbox"/>	
total	float	0	0	<input checked="" type="checkbox"/>	

Tabel 3.12 T\_Penjualan

### 13. Tabel Peramalan

Tabel ini berfungsi untuk menyimpan hasil dari peramalan dengan data actual yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
kd_peramalan	char	18	0	<input type="checkbox"/>	
kode_bj	char	8	0	<input checked="" type="checkbox"/>	
tahun	char	4	0	<input checked="" type="checkbox"/>	
bulan	char	2	0	<input checked="" type="checkbox"/>	
jumlah_produk	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.13 T\_Peramalan

### 14. Tabel Produksi


Tabel ini berfungsi untuk menyampaikan hasil diproduksi yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
no_produk	char	11	0	<input type="checkbox"/>	
tgl_produk	date	0	0	<input checked="" type="checkbox"/>	
Kode_bj	char	8	0	<input checked="" type="checkbox"/>	
Jumlah_Produksi	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.14 T\_Produksi

## 15. Tabel Resep


Tabel ini berfungsi untuk menyimpan data resep yang akan digunakan untuk produksi yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ no_resep	char	10	0	<input type="checkbox"/>	
kode_bb	char	8	0	<input checked="" type="checkbox"/>	
kode_bj	char	8	0	<input checked="" type="checkbox"/>	
jumlah_bb	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.15 T\_Resep

## 16. Tabel Supplier

Tabel ini berfungsi untuk untuk menyimpan semua data dari supplier yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ kode_supplier	char	5	0	<input type="checkbox"/>	
nama	varchar	30	0	<input checked="" type="checkbox"/>	
alamat	varchar	30	0	<input checked="" type="checkbox"/>	
no_telp	varchar	15	0	<input checked="" type="checkbox"/>	

Tabel 3.16 T\_Supplier



### 17. Tabel Kapasitas Produksi

Tabel ini berfungsi untuk untuk menyimpan semua data kapasitas produksi yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ KODE_KAPASITAS	char	8	0	<input type="checkbox"/>	
TGL_SETTING	date	0	0	<input checked="" type="checkbox"/>	
MAX_PRODUKSI	int	11	0	<input checked="" type="checkbox"/>	
MAX_PRODUKSI_PESAN	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.17 T\_Kapaitas\_Produksi

### 18. Tabel Karyawan

Tabel ini berfungsi untuk untuk menyimpan semua data dari kapasitas produksi yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ NIP	char	10	0	<input type="checkbox"/>	
NAMA	varchar	30	0	<input checked="" type="checkbox"/>	
ALAMAT	varchar	30	0	<input checked="" type="checkbox"/>	
NO_TELP	varchar	15	0	<input checked="" type="checkbox"/>	
JENIS_KELAMIN	int	11	0	<input checked="" type="checkbox"/>	
JABATAN	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.18 T\_Karyawan

## 19. Tabel Bayar Pesan

Tabel ini berfungsi untuk menyimpan data pembayaran dari pemesanan barang yang dilakukan oleh customer.

Name	Type	Length	Decimals	Allow Null	
▶ No_Bayar	char	12	0	<input type="checkbox"/>	
Tgl_Bayar	date	0	0	<input checked="" type="checkbox"/>	
no_pesanan	char	11	0	<input checked="" type="checkbox"/>	
Total_Bayar	float	0	0	<input checked="" type="checkbox"/>	

Tabel 3.19 T\_Bayar\_Pesan

## 20. Tabel Produksi Pesan


Tabel ini berfungsi untuk menyimpan hasil diproduksi pesan yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ no_produksi	char	11	0	<input type="checkbox"/>	
tgl_produksi	date	0	0	<input checked="" type="checkbox"/>	
Kode_bj	char	8	0	<input checked="" type="checkbox"/>	
Jumlah_Produksi	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.20 T\_Produksi\_Pesan

## 21. Tabel User acc

Tabel ini berfungsi untuk menyimpan hasil data hak akses karyawan yang dimiliki oleh perusahaan.

Name	Type	Length	Decimals	Allow Null	
▶ NIP	char	10	0	<input type="checkbox"/>	
Passwd	char	6	0	<input checked="" type="checkbox"/>	
Hak_Akses	int	11	0	<input checked="" type="checkbox"/>	

Tabel 3.21 T\_Useracc

## BAB IV

### HASIL DAN ANALISA

Hasil program adalah implementasi jalannya sistem yang telah dibuat sehingga diharapkan dengan adanya implementasi ini dapat dipahami jalannya suatu sistem. Sebelum melakukan implementasi sistem kita harus mempersiapkan kebutuhan – kebutuhan dari program yang akan kita implementasikan seperti perangkat lunak (*software*) komputer. Program ini dibuat dengan menggunakan *MySQL* untuk membuat database, dan pembuatan *interface* dengan *Borland Delphi 7.0* untuk desain aplikasi program<sup>[1]</sup>.

#### 4.1 Instalasi Program

##### 4.1.1 Kebutuhan perangkat lunak

Perangkat lunak yang digunakan untuk menjalankan sistem ini adalah :

1. Microsoft Windows XP.
2. *MySQL* untuk membuat database.
3. *Borland Delphi 7.0* untuk desain aplikasi program.
4. *Power Designer 6.32* untuk membuat DFD dan ERD.
5. *Crystal Report 10.0* untuk membuat laporan program.

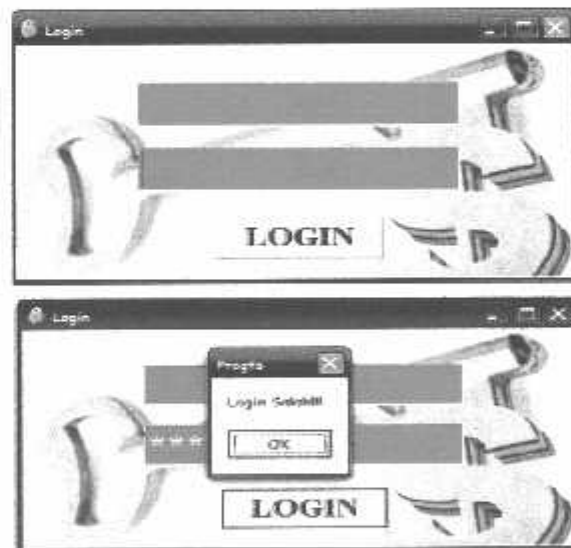
## 4.1.2 Setup program

1. Instal Microsoft Windows
2. Jalankan File EXE Aplikasi

## 4.2 Penjelasan Pemakaian Program

### 4.2.1 Tampilan Login

Form login digunakan untuk menginputkan data user yang telah disesuaikan pada database yang sudah dibuat untuk membuka form-form pada menubar yang tersedia pada menu utama untuk melakukan proses inputan data yang sudah tersedia pada form-form tertentu. Jika username dan password salah, maka user tidak dapat melakukan proses selanjutnya. Jika password benar, maka menubar akan aktif dan selanjutnya bisa melakukan proses pada form-form yang telah tersedia.



Gambar 4.1 Form Login

#### **4.2.2 Tampilan Menu Utama**

Menu Utama adalah Form menu yang berisi menu - menu yang terdapat pada sistem informasi Produksi ini. Antara lain Bagian Gudang, Bagian Produksi, Bagian Marketing, Laporan, User Akses serta Exit. Pada menu - menu tersebut terdapat sub menu seperti di bawah ini :

##### **1. Bagian Gudang**

- a. Data Bahan Baku
- b. Data Hasil Jadi
- c. Data Supliar
- d. Data Pembelian

##### **2. Bagian Produksi**

- a. Resep
- b. Kapasitas Produksi
- c. Data Produksi
- d. Produksi Pesan.

##### **3. Bagian Marketing**

- a. Data Customer
- b. Data Penjualan
- c. Data Pemesanan
- d. Bayar Pesan

##### **4. Laporan**

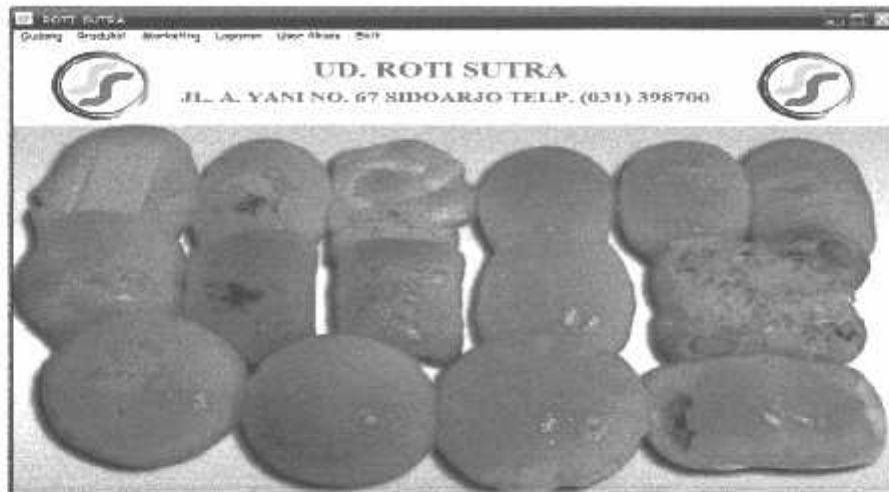
- a. Laporan Bahan Baku

- b. Laporan hasil jadi
- c. Laporan Supplier
- d. Laporan Pembelian
- e. Laporan produksi
- f. Laporan Produksi Pesan
- g. Laporan Customer
- h. Laporan Penjualan
- i. Laporan Pemesanan
- j. Laporan Peramalan

**5. User Akses**

- a. Data Karyawan
- b. Data User Akses
- c. Update Password
- d. Log Out

**6. Exit**



Gambar 4.2 Form Menu Utama

#### 4.2.3 Tampilan input data Bahan Baku

Form ini berfungsi untuk meng-*set* data Bahan Baku yang diinputkan oleh bagian gudang, jika data yang diblok warna coklat maka button save berubah menjadi button update dan kita dapat mengupdate data tersebut

Kode Bahan Baku	Nama	Harga Beli (Rp)	Satuan
BB-0001	TEPIUNG TERIGUAT		55000 GRAM
BB-0002	TELOR11		400 BUTIR
BB-0003	KFJ1115		300 BJI
BB-0004	COKLATE1		10000 GRAM
BB-0006	KACANG2		1000 FAKJIFA

Gambar 4.3 Form Data Bahan baku



#### 4.2.4 Tampilan input data Barang jadi

Form ini berfungsi untuk menginputkan atau memasukkan daftar barang yang sudah jadi .

The screenshot shows a window titled 'Barang Jadi' with a header 'BARANG JADI'. It contains three input fields: 'Kode Produk' (empty), 'Nama' (filled with 'RDTI KETIL'), and 'Harga Jual' (filled with '5000'). Below these is a table with the following data:

Kode Barang Jadi	Nama	Harga Jual (Rp)	Stock
BJ-0001	RDTI PISANG REJUT		5000
BJ-0002	RDTI PISANG COKLAT		5000
BJ-0003	RDTI COKLAT ISI		5000
BJ-0004	RDTI KACANG HIJAU		5000
BJ-0005	RDTI GORENGS		5000
BJ-0007	RDTI KEJU		5000

At the bottom of the window are four buttons: 'Update', 'Cancel', 'Exit', and 'Delete'.

Gambar 4.4 Form Data Barang Jadi

#### 4.2.5 Tampilan Input data supplier

Form ini berisi tentang keterangan data dari supplier, pengoperasian form sama seperti form-form diatas.

The screenshot shows a window titled 'Supplier' with a header 'DATA SUPPLIER'. It contains three input fields: 'Kode Supplier' (empty), 'Nama' (filled with 'CV MERDEKA'), and 'No. Telp' (filled with '452627'). Below these is a table with the following data:

Kode Supplier	Nama	Alamat	No. Telp
S-001	CV MERDEKA	JL MANGGA 45	4526
S-002	CV MAWANI	JLN BULU-PRU NY MALANG	8641
S-003	UD BOLON	JLN KARTIKA	031
S-004	RADI		1062
S-005	ANTONI I		7634

At the bottom of the window are four buttons: 'Update', 'Cancel', 'Exit', and 'Delete'.

Gambar 4.5 Form Data Supplier

#### 4.2.6 Tampilan Inputan Order

Form ini berfungsi sebagai penyuplai atau pembelian bahan baku yang diinputkan oleh bagian gudang form ini di lengkapi cari data supplier, apabila kita lupa kode supplier yaitu tinggal tekan F5 pada keyboard maka akan muncul form berikut ini.

The image shows two screenshots of a software application. The top window is titled 'Pembelian' and contains a form for entering purchase order details. It includes fields for 'No Faktur', 'Kode Supplier', and 'Nama Supplier'. A table lists items with columns for 'No', 'Kode Barang', 'Nama Barang', 'Harga Beli', 'Jumlah', 'Kupon', and 'Sub Total'. Below the table are fields for 'Kode Bahan Baku', 'Harga Beli', and 'Jumlah', along with a 'Total (Rp)' field and buttons for 'Cari', 'Klik', and 'Ok'. The bottom window is titled 'CARI DATA SUPPLIER' and displays a list of suppliers with columns for 'Kode Supplier', 'Nama', 'Alamat', and 'No'.

No	Kode Barang	Nama Barang	Harga Beli	Jumlah	Kupon	Sub Total
1	0000000	TEPUNG TERIGU 11	200	2	UMAM	400
2	0000000	TELOR 11	300	2	DLTIF	600

Kode Supplier	Nama	Alamat	No
S-001	CV MERDEKA	JL MANGGA 45	451
S-002	TYO	JLN KARTIKA	301
S-003	SINTA	MAJELANG UTARA	103
S-004	CV MADINAH	JLN MANKO	051

Gambar 4.6 Form Order

#### 4.2.7 Tampilan Resep

Form ini berisi keterangan resep yang akan digunakan untuk produksi pengoperasian form sama seperti form – form diatas,

Kode Bahan Baku	Nama Bahan Baku	Satuan	Jumlah
BB-00001	TELUNG	GEMAN	5000
BB-00002	TELUR	BUTIR	5000

Gambar 4.7 Form Resep

#### 4.2.8 Tampilan Kapasitas Produksi

Form ini berisi data kapasitas produksi yang dihasilkan oleh perusahaan. Pengoperasian form ini seperti form data bahan baku diatas jika data yang diblok warna coklat maka button save berubah menjadi button update dan kita dapat mengupdate data tersebut.

Kode Kapasitas	Tgl Ending	Total Produksi	Max. Produksi Perhari
R-1-00001	20/07/2010	1100	50
R-1-00002	04/09/2010	0	1
R-1-00003	04/09/2010	0	0
R-1-00004	04/09/2010	0	0

Gambar 4.8 Form Kapasitas Produksi

### 4.3.9 Tampilan Data Produksi

Form ini berfungsi untuk menampilkan data – data produksi yang dibutuhkan oleh bagian produksi untuk dilakukan proses produksi yang dihasilkan oleh perusahaan.

KODE BARANG	NAMA BARANG	JUMLAH PRODUKSI	STOK
BB-00001	TEPLUNG	5000	500000
BB-00002	TELUR	10000	1000000

Gambar 4.9 Form Data Produksi

### 4.3.10 Tampilan Produksi Pesan

Form produksi pesan berfungsi untuk menampilkan barang jadi hari berikutnya yang sesuai dengan inputan pesanan jika menginputkan kode barang jadi maka secara otomatis akan terhubung dengan tabel data produksi dan tabel kebutuhan bahan baku sesuai dengan resep yang kita inputkan pada form resep diatas.

KODE BARANG	NAMA BARANG	JUMLAH PRODUKSI	STOK
BB-00001	TEPLUNG	40000	1000000
BB-00002	TELUR	40000	500000

Gambar 4.10 Form produksi Pesan

### 4.3.11 Tampilan Data Customer

Form ini berisi tentang keterangan data-data customer Pengoprasian form ini seperti form data bahan baku diatas jika data yang diblok warna coklat maka button save berubah menjadi button update dan kita dapat mengupdate data tersebut.

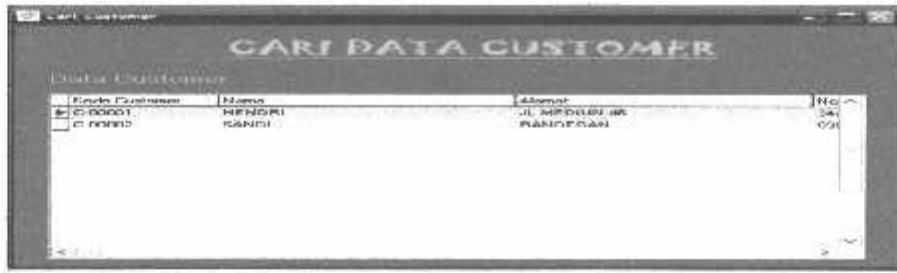
Kode Customer	Nama	Alamat	No Telp
C-00001	ARTINI	Jl. M. BUDI	81234567
C-00002	RIDHO 121	MALANG	87654321
C-00003	HSDHSD 112	JGDUSAN	90123456

Gambar 4.11 Form data customer

### 4.3.12 Tampilan Data Penjualan

Form ini berfungsi untuk mengetahui keterangan transaksi hasil penjualan pesanan atau pembelian dari customer. jika kode tersebut sudah ada dalam database maka otomatis nama customer akan terisi dan tabel akan aktif, jika kode belum ada dalam database maka nama customer tidak akan terisi.

Kode Barang	Nama Barang	Stok	Jumlah	Total
BJ-0001	RDTI KACANG	100	2	200



Gambar 4.12 Form data Penjualan

### 4.3.13 Tampilan Data Pemesanan

Form ini berfungsi untuk menampilkan data pemesanan roti yang di pesan oleh customer



Gambar 4.13 Form data Pemesanan

### 4.3.14 Tampilan bayar Pesan

Form ini berfungsi untuk menampilkan pembayaran atas pemesanan barang yang dilakukan oleh customer.



Gambar 4.14 Form data Pemesanan

### 4.3.15 Tampilan Data Karyawan

Form ini berfungsi untuk mengetahui data – data karyawan pada perusahaan tersebut.

The screenshot shows a window titled 'DATA KARYAWAN'. It features several input fields: 'Nama' (Name) with value 'GADSI', 'Alamat' (Address) with value 'Jl. Pahlawan 12', 'No. Telp' (Phone Number) with value '08123456789', 'Jenis Kelamin' (Gender) with a dropdown menu set to 'WANITA', and 'Jabatan' (Position) with a dropdown menu set to 'KARYAWAN'. Below these fields is a table with columns: 'NIP', 'Nama', 'Alamat', 'No. Telp', and 'Jenis Kelamin'. The table contains three rows of data. At the bottom, there are buttons for 'Update', 'Cancel', 'Edit', and 'Delete'.

NIP	Nama	Alamat	No. Telp	Jenis Kelamin
111111111	HAZRAH	Jl. Pahlawan 12	08123456789	PRIA
111111112	HAZRAH	Jl. Pahlawan 12	08123456789	WANITA
111111113	HAZRAH	Jl. Pahlawan 12	08123456789	PRIA

Gambar 4.14 Form data karyawan

### 4.3.16 Tampilan User Accounts

Form ini berfungsi untuk menginputkan hak akses yang dimiliki oleh karyawan.

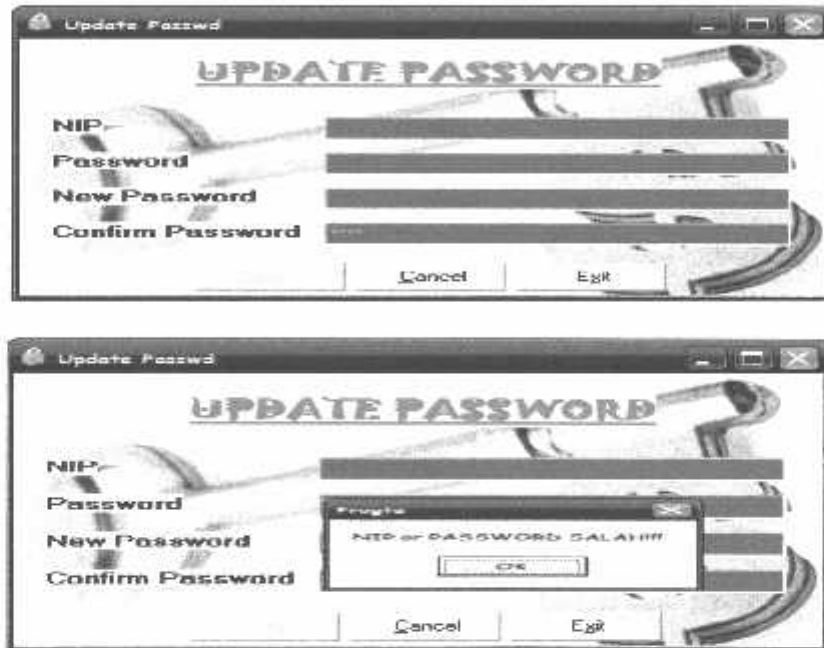
The screenshot shows a window titled 'USER ACCOUNTS'. It features input fields for 'NIP' (value: 111111111), 'Nama' (value: HAZRAH), and 'Hak Akses' (dropdown menu: Bagian Gudang). Below these fields are buttons for 'Save', 'Delete', 'Cancel', and 'Edit'. At the bottom, there is a table with columns: 'NIP' and 'Hak Akses'. The table contains three rows of data. On the right side of the table, there are up and down arrow buttons.

NIP	Hak Akses
111111111	Bagian Gudang
111111112	Bagian Marketing
111111113	Bagian Produksi

Gambar 4.15 Form user accounts

### 4.3.17 Tampilan Update Password

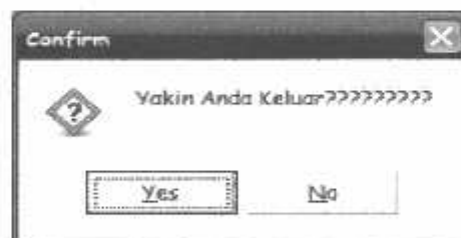
Form ini berfungsi untuk memperbarui password pada bagian yang telah ditentukan oleh perusahaan dan apabila dalam mengupdate pasword salah maka akan muncul warning seperti pada gambar di bawah ini.



Gambar 4.16 Form update password

### 4.3.18 Tampilan exit

Form ini digunakan apabila kita mau keluar secara langsung dari menu utama




Gambar 4.17 Form Exit



#### 4.3.19 Laporan Bahan Baku

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia




The screenshot shows a window titled 'Lap Bahan Baku'. It contains a search criteria form with two dropdown menus: 'KODE BAHAN BAKU' and 'LIKE'. Below the form is a 'Cetak' button. Below the window is a printed report titled 'LAPORAN BAHAN BAKU' dated '01/08/2010'. The report is a table with the following data:

Kode Bahan Baku	Nama	Harga Beli	Satuan	Stock
BB-00001	TEPUNG	Rp 20	GRAM	4.589.986
BB-00002	TELOR	Rp 50	BUTIR	4.507.970
BB-00003	GULA JAWA	Rp 2	GRAM	0
BB-00004	TAPIOKA	Rp 39	GRAM	0
BB-00005	COKLAT	Rp 21	GRAM	0
BB-00006	HKHKHK	Rp 5	HGJHG	0

Gambar 4.18 Laporan bahan baku

#### 4.3.20 Laporan Barang Jadi

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia



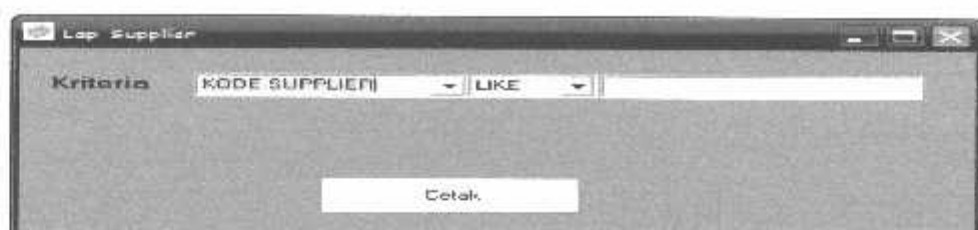
The screenshot shows a window titled 'Lap Barang Jadi'. It contains a search criteria form with two dropdown menus: 'KODE BARANG JADI' and 'LIKE'. Below the form is a 'Cetak' button. Below the window is a printed report titled 'LAPORAN BARANG JADI' dated '01/08/2010'. The report is a table with the following data:

Kode Barang Jadi	Nama	Harga Jual	Stock	Stock Pesan
BJ-00001	ROTI KACANG	Rp 100	8	160
BJ-00002	ROTIKEDELAI	Rp 130	15	0
BJ-00003	ROTI COKLAT	Rp 150	5	0
BJ-00004	ROTI GORENG	Rp 300	0	0
BJ-00005	ROTI ABON	Rp 100	0	0

Gambar 4.19 Laporan Barang Jadi

### 4.3.21 Laporan Supplier

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia



LAPORAN SUPPLIER			
01/09/2010			
Kode Suplier	Nama	Alamat	No telp
S-001	CV MERDEKA	JL MANGGA 45	452627
S-002	CV MAWAR1	JLN GOLF NO 67 MALANG	8381919
S-003	UD BOLIEM	JLN KARTIKA	031 9876545
S-004	DAS21		12321431

Gambar 4.20 Laporan supplier

### 4.3.22 Laporan Order

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia



## LAPORAN ORDER

01/08/2010

No Faktur: O-100700001  
 Kode Suplier: S-001  
 Nama Suplier: CV MERDEKA

Tgl Order: 13/03/2010

No	Kode BB	Nama Bahan Baku	Harga Beli	Satuan	Jumlah	Sub Total
1	BB-00001	TEPUNG	Rp 8.000	GRAM	20	Rp 160.000
<b>Total</b>						<b>Rp 160.000</b>

No Faktur: O-100700002  
 Kode Suplier: S-001  
 Nama Suplier: CV MERDEKA

Tgl Order: 27/03/2010

No	Kode BB	Nama Bahan Baku	Harga Beli	Satuan	Jumlah	Sub Total
1	BB-00002	TELOR	Rp 200	BUTIR	5	Rp 1.000
1	BB-00001	TEPUNG	Rp 1.000	GRAM	10	Rp 10.000
<b>Total</b>						<b>Rp 11.000</b>

No Faktur: O-100700001  
 Kode Suplier: S-002  
 Nama Suplier: TYO

Tgl Order: 30/07/2010

No	Kode BB	Nama Bahan Baku	Harga Beli	Satuan	Jumlah	Sub Total
1	BB-00001	TEPUNG	Rp 2.000	GRAM	100	Rp 200.000
<b>Total</b>						<b>Rp 200.000</b>

No Faktur: O-100700002  
 Kode Suplier: S-001  
 Nama Suplier: SINTA

Tgl Order: 30/07/2010

No	Kode BB	Nama Bahan Baku	Harga Beli	Satuan	Jumlah	Sub Total
1	BB-00002	TELOR	Rp 100	BUTIR	30	Rp 3.000
<b>Total</b>						<b>Rp 3.000</b>

No Faktur: O-100800001  
 Kode Suplier: S-001  
 Nama Suplier: CV MERDEKA

Tgl Order: 01/08/2010

No	Kode BB	Nama Bahan Baku	Harga Beli	Satuan	Jumlah	Sub Total
1	BB-00001	TEPUNG	Rp 1	GRAM	1	Rp 1
<b>Total</b>						<b>Rp 1</b>

Gambar 4.21 Laporan order

### 4.3.23 Laporan Produksi

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia

The screenshot shows a window titled "Lap Produksi" with the following fields:

- Kriteria 1:** A dropdown menu showing "NO PRODUKSI" and a "LIKE" dropdown.
- Kriteria 2:** A checkbox labeled "Tot Produk" which is checked, followed by a date range from "01/08/2010" to "01/08/2010".
- Buttons:** A "Cetak" button is located at the bottom center of the window.

## LAPORAN PRODUKSI

01/08/2010

No Produksi **P-100730006**  
Tgl Produksi 30/07/2010  
Kode BJ BJ-00001  
Nama BJ ROTI KACANG  
Jumlah Produksi 3

No Kode BB	Nama BB	Jumlah BB
2 BB-00002	TELOR	18.000

No Produksi **P-100730007**  
Tgl Produksi 30/07/2010  
Kode BJ BJ-00002  
Nama BJ ROTIKEDELAI  
Jumlah Produksi 2

No Kode BB	Nama BB	Jumlah BB
2 BB-00002	TELOR	20
1 BB-00001	TEPUNG	10

No Produksi **P-100730008**  
Tgl Produksi 30/07/2010  
Kode BJ BJ-00001  
Nama BJ ROTI KACANG  
Jumlah Produksi 1

No Kode BB	Nama BB	Jumlah BB
2 BB-00002	TELOR	6.000
1 BB-00001	TEPUNG	5.000

Gambar 4.22 Laporan Produksi

### 4.3.24 Laporan Produksi Pesan

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia

Lap Produksi Pesan

Kriteria 1 NO PRODUKSI LIKE

Kriteria 2  Tgl Produksi 01/08/2010 s/d 01/08/2010

Cetak

01.08.2010

No Produksi P-100801001  
Kode Produk BJ-00001  
Nama Produk ROTI KACANG  
No Jadwal JD-100801001  
Jumlah Produksi 80

Tgl Produksi 01.08/2010

No	Kode bahan	Nama Bahan	Jumlah
1	BB-00001	TEPUNG	400.000
2	BB-00002	TELOR	480.000

Gambar 4.23 Laporan Produksi Pesan

#### 4.3.25 Laporan Customer

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia



LAPORAN CUSTOMER			
01.08.2010			
Kode Customer	Nama	Alamat	No Telp
C-00001	HENDRI	Jl. MEDUN 46	3421313
C-00002	SANDI	RANDEGAN	0909898

Gambar 4.24 Laporan Customer

### 4.3.26 Laporan jual

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia

The screenshot shows a window titled "Lap Jual" with the following fields:

- Kriteria 1: NO NOTA (dropdown), LIKE (dropdown)
- Kriteria 2:  Tgl Jual
- Date Range: 01/09/2010 to 31/09/2010
- Button: Cetak

LAPORAN PENJUALAN					
01/08/2010					
No Nota	J-100300001			Tgl Jual	19/03/2010
Kode Customer	C-00001				
Nama Customer	HENDRI				
No	Kode Produk	Nama Produk	Harga Jual	Jumlah	Sub Total
1	BF-00001	ROTI KACANG	Rp 6.300	2	Rp 12.600
				Total	Rp 12.600
No Nota	J-100300002			Tgl Jual	19/03/2010
Kode Customer	C-00001				
Nama Customer	HENDRI				
No	Kode Produk	Nama Produk	Harga Jual	Jumlah	Sub Total
1	BF-00002	ROTI KEDELAI	Rp 6.300	3	Rp 19.500
				Total	Rp 19.500
No Nota	J-100700001			Tgl Jual	30/07/2010
Kode Customer	C-00001				
Nama Customer	HENDRI				
No	Kode Produk	Nama Produk	Harga Jual	Jumlah	Sub Total
1	BF-00001	ROTI KACANG	Rp 300	1	Rp 300
				Total	Rp 300
No Nota	J-100800001			Tgl Jual	01/08/2010
Kode Customer	C-00002				
Nama Customer	SANDI				
No	Kode Produk	Nama Produk	Harga Jual	Jumlah	Sub Total
1	BF-00001	ROTI KACANG	Rp 100	2	Rp 200
				Total	Rp 200

Gambar 4.25 Laporan Penjualan

### 4.3.27 Laporan Pesan

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia

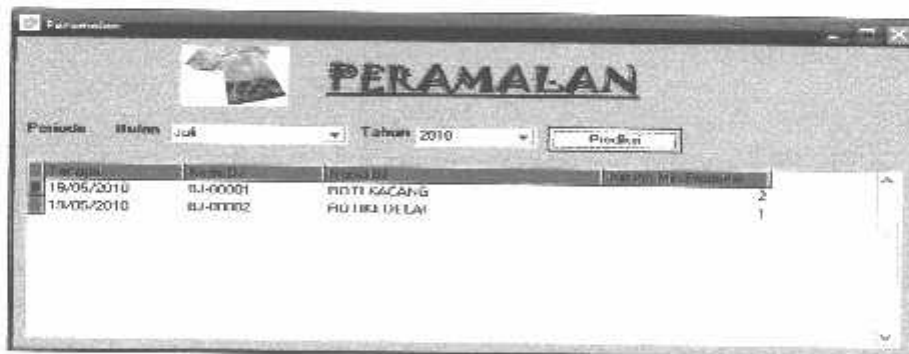


LAPORAN PESAN					
01-08-2010					
No Nota	P-100600001			Tgl Pesan	24-06-2010
Kode Customer	C-00001			Tgl Selesai	24-06-2010
Nama Customer	HENDRI				
No	Kode Produk	Nama Produk	Harga Jual	Jumlah	Sub Total
1	BJ-00001	ROTI KACANG	Rp 6.300	100	Rp 630.000
				Total	Rp 630.000
				Uang Muka	Rp -
				Status	PROSES

Gambar 4.26 Laporan pesan

### 4.3.28 Laporan Peramalan

Kita bisa melihat data laporan sesuai dengan kriteria – kriteria yang ter sedia



Periode	Bulan	Juli	Tahun 2010	Produk
18/05/2010		BJ-00001		ROTI KACANG
18/05/2010		BJ-00002		ROTI LAYU LAYU

Gambar 4.27 Laporan Peramalan

## **BAB V**

### **PENUTUP**

#### **51 Kesimpulan**

Berdasarkan hasil pengujian sistem yang dilakukan, dapat diambil kesimpulan beberapa hal antara lain :

1. Sistem informasi yang dikembangkan ini dapat membantu pihak pegawai Perusahaan UD sutra sidoarjo dalam melakukan pengolahan data- data perusahaan, laporan bahan baku, laporan barang jadi, laporan supplier, laporan pembelian, laporan produksi, laporan produksi pesan, laporan customer, laporan penjualan, dan laporan pemesanan, serta laporan peramalan.
2. Dengan adanya sistem ini kita dapat memudahkan perusahaan dalam menangani pengolahan data produksi dan efisiensi waktu.
3. Bagaimana membuat suatu perangkat lunak yang dapat membantu dalam meramalkan perencanaan produksi, serta menentukan perencanaan penjadwalan produksi, pemesanan, untuk memenuhi permintaan konsumen.

#### **52 Saran**

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

1. Mengingat bahwa salah satu factor penting dalam system ini adalah pengolahan data, maka dalam pengolahan data secara berkala sangat penting untuk



dilakukan sehingga keamanan data dalam system dapat terjamin. Salah satunya adalah mengencrypsi data- data di dalamnya.

2. Sistem informasi ini masih perlu penyempurnaan baik dari teknologi maupun fasilitasnya. Seperti menambahkan beberapa menu laporan keuangan seperti laporan buku besar, jurnal, dan neraca.
3. Untuk mengurangi permasalahan jika terjadi kerusakan data atau sistem maka disarankan untuk membuat fasilitas backup data, sehingga data dapat dikembalikan seperti sebelumnya.
4. System informasi ini akan menjadi lebih baik jika dilengkapi dengan web untuk sarana promosi dan juga untuk transaksi online bagi para costomer.

## Daftar Pustaka

- [1]. Alam, M.Agus.J, *Mengolah Database dengan Borland Delphi 7*. PT Elexmedia Komputindo, Jakarta, 2003.
- [2]. Kadir, Abdul. 2003. *Pengenalan Sistem Informasi*. Yogyakarta : Andi.
- [3]. H.S, Suryadi D., & Bunawan. 1995. *Pengantar Metodologi Pengembangan Sistem Informasi*. Jakarta: Gunadarma.
- [4]. Modul Praktikum Basisdata Institut Teknologi Nasional Malang.
- [5]. Modul Praktikum RPL (*Rekayasa Perangkat Lunak*) Institut Teknologi Nasional Malang.
- [6]. [www.ilmukomputer.com](http://www.ilmukomputer.com)
- [7]. Bernaridho I Hutabarat. 2007. *My sql*. Jakarta: pc media.
- [8]. <http://id.wikipedia.org/wiki/DFD>
- [9]. [http://id.wikipedia.org/wiki/Sistem\\_Informasi](http://id.wikipedia.org/wiki/Sistem_Informasi)
- [10] Martine,inge. 2006. *36 Jam Belajar Komputer: Microsoft My SQL*. Jakarta: PT Elexmedia Komputindo.
- [11] Teddy Marcus, Agus Prijono, Josef Widiadhi.2007. *Delphi developer dan sql*.jakarta: informatika.



PT. BNI (PERSERO) MALANG  
BANK NIAGA MALANG

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG  
INSTITUT TEKNOLOGI NASIONAL MALANG

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

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

BERITA ACARA UJIAN SKRIPSI

FAKULTAS TEKNOLOGI INDUSTRI

Nama : HANI LESTYOADI  
NIM : 0512608  
Jurusan : T.Elektro S-1  
Konsentrasi : T.Komputer dan Informatika  
Judul Skripsi : PENGEMBANGAN SISTEM INFORMASI PRODUKSI ROTI DI  
UD SUTRA SIDOARJO

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

Hari : Rabu  
Tanggal : 18 Agustus 2010  
Dengan Nilai : 77,5 (B+) *B4*

Panitia Ujian Skripsi

Ketua Majelis Penguji

Ir. Yusuf Ismail Nahkoda, MT

NIP. P 1018600189

Penguji I

(Dr. Eng. Aryanto Soetedjo, ST.MT)

NIP.P. 1030800417

Penguji II

(Sandy Nataly M, S.Kom)



**FORMULIR PERBAIKAN SKRIPSI**

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

Nama : Hani Lestyoadi  
NIM : 05 12 608  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika  
Masa Bimbingan : 30 Mei 2010 s/d 30 November 2010.  
Judul Skripsi : Pengembangan Sistem Informasi Produksi Roti Di UD Sutra Sidoarjo.

Penguji / Tanggal	Uraian	Paraf
Penguji I 18 Agustus 2010	Tampilkan informasi kode bahan baku dan harga, Koneksi database belum ada, jelaskan peramalan secara lengkap, daftar pustaka.	
Penguji II 18 Agustus 2010	Masukkan nama barang bukan kode barang, tidak bisa menunjukkan cara kerja peramalan dan letak di scripy, tambahkan laporan penjualan keseluruhan, sistem informasi yang penting, dan daftar pustaka, form peramalan eror, ERD tidak perlu ditampilkan, perbaiki saran.	

**Mengetahui,**

**Dosen Pembimbing I**

**(Irmalia Suryani Faradisa, ST. MT)**  
**NIP.P.1030000365**

**Dosen Pembimbing II**

**(Ahmad Faisol, ST)**

**Dosen Penguji,**

**Penguji I**

**(Dr. Eng. Aryuanto Soetedjo, ST.MT)**  
**NIP.P.1030800417**

**Penguji II**

**(Sandy Nataly M, S.Kom)**



## FORMULIR BIMBINGAN SKRIPSI

Nama : Hani Lestyoadi  
Nim : 05.12.608  
Masa Bimbingan : 30 Mei 2010 - 30 November 2010  
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI PRODUKSI  
ROTI DI UD SUTRA SIDOARJO**

NO	Tanggal	Uraian	Paraf Pembimbing
1.	12/07 2010	Revisi : - Format Penulisan Laporan, Spasi, Penulisan BAB.	
2.	19/07 2010	Ace BAB I II	
3.	27/07 2010	Perbaiki Program yang error	
4.	2/08	program diperbaiki	
5.	3/08	Ace mklh smmr	
6.	4/08	Ace Bab III	
7.	5/08	Ace Bab IV	
8.	5/08	Ace Bab V	
9.	6/08	Ace Komare	
10.	12/08	Ace Jilid	

Malang,  
Dosen Pembimbing I

**(Irmalia Suryani Faradisa, ST.MT)**  
NIP.P 1030000365

Form S-4b



## FORMULIR BIMBINGAN SKRIPSI

Nama : Hani Lestyoadi  
Nim : 05.12.608  
Masa Bimbingan : 30 Mei 2010 - 30 November 2010  
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI PRODUKSI  
ROTI DI UD SUTRA SIDOARJO**

NO	Tanggal	Uraian	Paraf Pembimbing
1.	28/07/2010	Demo program (Perbaiki program yg lama)	Is
2.	28/07/2010	Acc Bab I, II	Is
3.	28/07/2010	Revisi Bab II, Perbaiki DFD	Is
4.	2/08	Acc Bab III	Is
5.	3/08	Acc Melakahi Seminar hari	Is
6.			
7.			
8.			
9.			
10.			

Malang,  
Dosen Pembimbing II

**Ahmad Faisol, ST**



### Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : Hans Lestyoadi  
NIM : 1612608  
Perbaikan meliputi :

- Nama variabel dan kode bahan baku & barang - ditanya dan kerendahan harga -> Musti
- Reseptur -> Informasi kode -> tampil saat input.
- Konsentrasi ditambahkan bahan awal.
- Setiap menu masih perlu mengentry hal yg sama
- Parameter perlu dijelaskan secara lengkap karena merupakan bagian skripsi.
- Daftar pustaka ditambal

Malang, 10/8/ 2000

()  
Aryananto



### Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : HANI LESTYADI  
 NIM : 0512602  
 Perbaikan melalui :

1. PADA CODED PEMBELIAN JGV MEMAJUKKAN KODE BARANG TP APALAH ~~NAMA~~ BERDASARKAN NAMA BARANG DEWAS PULSA ID BAE LAIN MA TIDAK GAL MANDUKAN NAMA PRTI OOL
2. PERUBAHAN CASIANY
3. TAMBAHNYAN DAFTAR PUSTAKA
4. APA COMBO TABA KODELE NAMA BARANG. TAMBAHNYAN
5. TDK BISA MEMURJIKAN CARA KERJA PERALATAN DAN LETAK DI SCRIPT
6. LAPORAN PERFORMAN <sup>KETERANGAN</sup> PRTI JECAPA GAMIS PERALATAN NILAI PERUBULAN TANDA MENYANTUNNYAN PANDA CUSTOMER
7. FORM PERANTAUAN ~~DIRUBAH~~
8. - ISAB II ETAP TDK PERLU DITAMPILKAN PRN TDK DIBAHAS  
 - PERALATAN BUKAN UMM TP TTD WINDOR (PAMING)  
 - TAMBAHNYAN LSI- INF PRODUSEN YG PERLU

Malang, 18 - 2 - 2000

( SANDJ WATALY )



## Lampiran Script

```
unit UDatabase;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, DBXpress, FMTBcd, DB, DBClient, Provider, SqlExpr, strutils;

type
  TFDatabase = class(TForm)
    AdoConn: TSQLConnection;
    Search: TSQLQuery;
    Query: TSQLQuery;
    DataSetProvider: TDataSetProvider;
    ClientDataSet: TClientDataSet;
    DataSource: TDataSource;
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
  TBahan_Baku=Class
    Kode_BB:string;
    Nama:String;
    Harga_Beli:string;
    Satuan: String;
    Stock:integer;
  Private
    Sqlcmd:String;
  Public
    Function Get_Kode_BB:string;
    Function Find_Kode_BB(Kode_BB1:string):Boolean;
    Function Find_BB_InResep(Kode_BB1:string):Boolean;
    Function Find_BB_InItemOrder(Kode_BB1:string):Boolean;
    Procedure Save_Data(Kode_BB1,NAMA1,Harga_Beli1,Satuan1,Stock1:string);
    Procedure Update_Data(Kode_BB1,NAMA1,Harga_Beli1,Satuan1:string);
    Procedure Update_StockBB(Kode_BB1,Stock1:string);
    Procedure Delete_Data(Kode_BB1:string);
    Constructor Create;
  end;
  TSupplier=Class
    Kode_Supplier:string;
```

---

```

Nama:String;
Alamat:string;
No_Telp: String;
Private
    Sqlcmd:String;
Public
    Function Get_Kode_Supp:string;
    Function Find_Kode_Supp(Kode_Supp1:string):Boolean;
    Function Find_Supp_InOrder(Kode_Supp1:string):Boolean;
    Procedure Save_Data(Kode_Supp1,NAMA1,Alamat1,No_Telp1:string);
    Procedure Update_Data(Kode_Supp1,NAMA1,Alamat1,No_Telp1:string);
    Procedure Delete_Data(Kode_Supp1:string);
    Constructor Create;
end;
TBarang_Jadi=Class
    Kode_BJ:string;
    Nama:String;
    Harga_Jual:string;
    Stock:integer;
    Stock_Pesan:integer;
Private
    Sqlcmd:String;
Public
    Function Get_Kode_BJ:string;
    Function Find_Kode_BJ(Kode_BJ1:string):Boolean;
    Function Find_BJ_InItemPenjualan(Kode_BJ1:string):Boolean;
    Function Find_BJ_InPeramalan(Kode_BJ1:string):Boolean;
    Function Find_BJ_InResep(Kode_BJ1:string):Boolean;
    Function Find_BJ_InPenjadwalan(Kode_BJ1:string):Boolean;
    Function Find_BJ_InItemPemesanan(Kode_BJ1:string):Boolean;

    Procedure Save_Data(Kode_BJ1, NAMA1, Harga_Jual1,Stock1,Stock_Pesan1:
string);
    Procedure Update_Data(Kode_BJ1,NAMA1,Harga_Jual1:string);
    Procedure Delete_Data(Kode_BJ1:string);
    Procedure Update_StockBJ(Kode_BJ1,Stock1:string);
    Procedure Update_StockPesan(Kode_BJ1,Stock_Pesan1:string);
    Constructor Create;
end;
TCustomer=Class
    Kode_Customer:string;
    Nama:String;
    Alamat:string;
    No_Telp: String;
Private

```

---

```

    SqlCommand:String;
Public
    Function Get_Kode_Customer:string;
    Function Find_Kode_Customer(Kode_Customer1:string):Boolean;
    Function Find_Customer_InPenjualan(Kode_Customer1:string):Boolean;
    Procedure Save_Data(Kode_Customer1,NAMA1,Alamat1,No_Telp1:string);
    Procedure Update_Data(Kode_Customer1,NAMA1,Alamat1,No_Telp1:string);
    Procedure Delete_Data(Kode_Customer1:string);
    Constructor Create;
end;
TOrder = Class
Private
    SqlCommand:String;
Public
    Function Get_NoNotaOrder:string;
    Procedure
Save_Data_Order(No_Nota_Order1,Kode_Supplier1,Tgl_Order1,Total1:string);
    Procedure
Save_Data_Order_Detail(no_item_order1,No_Nota_Order1,Kode_bb1,harga_bel
i1,Jumlah1,Sub_Total1:string);
    end;
TPenjualan = Class
Private
    SqlCommand:String;
Public
    Function Get_NoNotaJual:string;
    Procedure
Save_Data_Jual(No_Nota_Jual1,Tgl_Jual1,Kode_Customer1,Total1:string);
    Procedure
Save_Data_Jual_Detail(no_item_Jual1,No_Nota_Jual1,Kode_bj1,Jumlah1,Sub_To
tal1:string);
    end;
TResep = Class
    No_Resep:string;
    Kode_BJ:String;
    Kode_BB:string;
    Jumlah_BB:integer;
private
    SqlCommand:String;
Public
    Function Get_No_Resep(Kode_BJ1: string; Kode_BB1: string):string;
    Function Find_No_Resep(No_Resep:String):Boolean;
    Function Find_Resep(Kriteria:string):Boolean;
    Procedure
Save_Data_Resep(No_Resep1,Kode_BB1,Kode_BJ1,Jumlah_BB1:string);

```

---

```

    Procedure Delete_Data_Resep(Kode_BJ1:string);
    Constructor Create;
end;
TPesan = Class
    no_pesan      : string;
    kode_customer : string;
    tgl_pesan     : string;
    total_pesan   : string;
    tgl_selesai  : string;
    Uang_muka     : string;
    status        : integer;
Private
    SqlCommand:String;
Public
    Constructor Create;
    Function Get_NoNotaPesan:string;
    Procedure
Save_Data_Pesan(No_Nota_Pesan1,Kode_Customer1,Tgl_Pesan1,total_pesan1,T
gl_selesai1,uang_muka1,status1:string);
    Procedure
Save_Data_Pesan_Detail(no_item_pesan1,No_pesan1,Kode_bj1,Jumlah_pesan1,
Sub_Total1:string);
    Procedure Update_Tgl_Selesai(No_Nota_Pesan1,Tgl_Selesai1:string);
    Function Find_Pesanan_BelumSelesai(No_Pesan1: String):Boolean;
    Procedure Update_StatusPesan(No_Pesan1,Status1:String);
    Function Find_Pesanan(No_Pesan1:string):Boolean;
end;

TKapasitas_Produksi = Class
    KODE_KAPASITAS : string;
    TGL_SETTING    : string;
    MAX_PRODUKSI   : string;
    MAX_PRODUKSI_PESAN : string;
Private
    SqlCommand:String;
Public
    Function Get_Kode_Kapasitas:string;
    Function Find_Kode_Kapasitas(Kode_Kapasitas1:string):Boolean;
    Procedure
Save_Data(Kode_Kapasitas1,Tgl_Setting1,Max_Produksi1,Max_Produksi_Pesan1:
string);
    Procedure
Update_Data(Kode_Kapasitas1,Tgl_Setting1,Max_Produksi1,Max_Produksi_Pesa
n1:string);
    Procedure Delete_Data(Kode_Kapasitas1: string);

```

---

```

    Constructor Create;
end;
TPenjadwalan = Class
    No_Jadwal : string;
    Tgl_jadwal : string;
    no_pesan : string;
    kode_bj : string;
    jumlah : string;
    status : byte;
Private
    Sqlcmd : string;
Public
    //Function Get_No_Jadwal(Tgl_Jadwal:string):string;
    Constructor Create;
    function Get_No_Jadwal(Tgl_Jadwal1: string): string;
    Function Get_Sisa_Kapasitas_Produksi(Tgl_Jadwal:string):integer;
    Function Get_Kapasitas_Produksi(Tgl_Jadwal:string):integer;
    Function Get_Juml_Pesanan(Tgl_Jadwal:string):integer;
    Procedure
Save_Data(no_jadwal1,Tgl_jadwal1,no_pesan1,kode_bj1,jumlah1,status1:string)
;
    Procedure Update_Status(no_jadwal1,status1:string);
    Function Find_No_Jadwal(No_Jadwal1:string):Boolean;
end;
TProduksi_Pesan = Class
Private
    Sqlcmd :string;
Public
    Function Get_NoProduksiPesan:string;
    Procedure
Save_Produksi_Pesan(no_produksi1,no_jadwal1,tgl_produksi1,Juml_Produksi1:s
tring);
    Procedure
Save_Produksi_Pesan_Detail(no_item_produksi1,no_produksi1,no_resep1,jumla
h_BB1:string);
end;
TProduksi = Class
Private
    Sqlcmd :string;
Public
    Function Get_NoProduksi:string;
    Procedure
Save_Produksi(no_produksi1,tgl_produksi1,Kode_bj1,Jumlah_Produksi1:string);

```

---

```

    Procedure
Save_Produksi_Detail(no_item_produksi1,no_produksi1,no_resep1,jumlah_BB1:
string);
end;
TKaryawan = Class
    NIP      :string;
    NAMA      :string;
    ALAMAT    :string;
    NO_TELP   :string;
    JENIS_KELAMIN:byte;
    JABATAN   :byte;
    STATUS    :byte;
Public
    constructor Create; overload; virtual;
    Function Find_NIP(NIP1:string):boolean;
    Procedure
Save_Data(NIP1,NAMA1,ALAMAT1,NO_TELP1,JENIS_KELAMIN1,JABATAN1,STAT
US1:string);
    Procedure
Update_Data(NIP1,NAMA1,ALAMAT1,NO_TELP1,JENIS_KELAMIN1,JABATAN1,ST
ATUS1:string);
    Procedure Delete_Data(NIP1:string);
Private
    Sqlcmd:String;
end;
T_UserAcc = Class
    NIP      : string;
    Passwd   : String;
    Hak_Akses : STRING;
    Hak_Akses1 : integer;
Public
    Function Find_UserID(NIP: string): Boolean;
    Procedure Save_UserAcc(NIP1,Passwd1,Hak_Akses1: string);
    Procedure Update_UserAcc(NIP1,Passwd1,Hak_Akses1: string);
    Procedure Delete_UserAcc(NIP1:string);
    Function Find_UserID1(NIP1, Passwd1: String): Boolean;
    Procedure Update_Password(NIP1,Passwd1: string);
    constructor Create; overload; virtual;
Private
    Sqlcmd:String;
end;
TBayar_Pesanan =class
Public
    Function Get_NoBayarPesan:string;

```

---

```

    Procedure
Save_BayarPesanan(no_Bayar1,tgl_Bayar1,No_Pesan1,Total_Bayar1:string);
    Private
        Sqlcmd:String;
    end;

var
    FDatabase: TFDatabase;

implementation

{$R *.dfm}

{ TBahan_Baku }

constructor TBahan_Baku.Create;
begin
    Kode_BB:="";
    Nama:="";
    Harga_Beli:="";
    Satuan:="";
    Stock:=0;
end;

procedure TBahan_Baku.Delete_Data(Kode_BB1: string);
begin
    Sqlcmd:='Delete From T_Bahan_Baku Where Kode_BB=' +
    Quotedstr(Kode_BB1) +";
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TBahan_Baku.Find_BB_InItemOrder(Kode_BB1: string): Boolean;
begin
    Result:=false;
    sqlcmd:='Select * From T_Item_Order Where Kode_BB=' +
    Quotedstr(Kode_BB1) +";
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:=True;
        end;
    end;
end;

```

---

```

end;

function TBahan_Baku.Find_BB_InResep(Kode_BB1: string): Boolean;
begin
    Result:=false;
    sqlcmd:='Select * From T_Resep Where Kode_BB='+ Quotedstr(Kode_BB1) +';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:=True;
        end;
    end;
end;

function TBahan_Baku.Find_Kode_BB(Kode_BB1: string): Boolean;
begin
    Result:=false;
    sqlcmd:='Select * From T_Bahan_Baku Where Kode_BB='+
    Quotedstr(Kode_BB1) +";
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            begin
                Result:=True;
                Nama:=Fieldbyname('Nama').AsString;
                Harga_Beli:=Fieldbyname('Harga_Beli').AsString;
                Satuan:=Fieldbyname('Satuan').AsString;
                Stock:=Fieldbyname('Stock').AsInteger;
            end;
        end;
    end;
end;

function TBahan_Baku.Get_Kode_BB: string;
begin
    sqlcmd:='Select Kode_BB From T_Bahan_Baku order by Kode_BB desc Limit 1';
    With FDatabase.Search do
    begin
        close;

```

---



```

        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:='BB-' + copy('00000',1,5-
length(inttostr(strtoint(RightStr(Fieldbyname('Kode_BB').AsString,5)) +1))) +
inttostr(strtoint(RightStr(Fieldbyname('Kode_BB').AsString,5))+1)
        else
            Result:='BB-00001';
        end;
    end;
end;

procedure TFDatabase.FormCreate(Sender: TObject);
begin
    adoconn.Open;
end;

procedure TBahan_Baku.Save_Data(Kode_BB1, NAMA1, Harga_Beli1, Satuan1,
    Stock1: string);
begin
    Sqlcmd:= 'INSERT INTO T_BAHAN_BAKU VALUES('+ Quotedstr(KODE_BB1)
+','+ Quotedstr(NAMA1) +',' +
    "+ HARGA_BELI1 +','+ Quotedstr(SATUAN1) +','+ STOCK1 +')';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TBahan_Baku.Update_Data(Kode_BB1, NAMA1, Harga_Beli1,
    Satuan1: string);
begin
    Sqlcmd:= 'UPDATE T_BAHAN_BAKU SET NAMA= '+ Quotedstr(NAMA1) +',' +
    'HARGA_BELI= '+ HARGA_BELI1 + ' WHERE KODE_BB = '+
    Quotedstr(KODE_BB1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TBahan_Baku.Update_StockBB(Kode_BB1, Stock1: string);
begin
    Sqlcmd:= 'UPDATE T_BAHAN_BAKU SET STOCK= STOCK + '+ STOCK1 + ' WHERE
KODE_BB = '+ Quotedstr(KODE_BB1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ TSupplier }

```

---

```

constructor TSupplier.Create;
begin
  Kode_Supplier:="";
  Nama:="";
  Alamat:="";
  No_Telp:="";
end;

procedure TSupplier.Delete_Data(Kode_Supp1: string);
begin
  Sqlcmd:='Delete From T_Supplier Where Kode_Supplier= '+
Quotedstr(Kode_Supp1) +";
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TSupplier.Find_Kode_Supp(Kode_Supp1: string): Boolean;
begin
  Result:=false;
  sqlcmd:='Select * From T_Supplier Where Kode_Supplier='+
Quotedstr(Kode_Supp1) +";
  With FDatabase.Search do
  begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (isempty) then
    begin
      Result:=True;
      Nama:=Fieldbyname('Nama').AsString;
      Alamat:=Fieldbyname('Alamat').AsString;
      No_Telp:=Fieldbyname('No_Telp').AsString;
    end;
  end;
end;

function TSupplier.Find_Supp_InOrder(Kode_Supp1: string): Boolean;
begin
  Result:=false;
  sqlcmd:='Select Kode_Supplier From T_Order Where Kode_Supplier='+
Quotedstr(Kode_Supp1) +";
  With FDatabase.Search do
  begin
    close;
    sql.Clear;

```

---

```

        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:=True;
        end;
    end;

function TSupplier.Get_Kode_Supp: string;
begin
    sqlcmd:='Select Kode_Supplier From T_Supplier order by Kode_Supplier desc
Limit 1';
    With FDatabase.Search do
        begin
            close;
            sql.Clear;
            sql.Add(sqlcmd);
            open;
            if not (isempty) then
                Result:='S-' + copy('000',1,3-
length(inttostr(strtoint(RightStr(Fieldbyname('Kode_Supplier').AsString,3)) +1)))
+
inttostr(strtoint(Rightstr(Fieldbyname('Kode_Supplier').AsString,3))+1)
            else
                Result:='S-001';
            end;
        end;
    end;

procedure TSupplier.Save_Data(Kode_Supp1, NAMA1, Alamat1,
    No_Telp1: string);
begin
    Sqlcmd:= 'INSERT INTO T_SUPPLIER VALUES('+ Quotedstr(KODE_SUPP1) +','+
Quotedstr(NAMA1) +',' +
    '+ Quotedstr(ALAMAT1) +','+ Quotedstr(NO_TELP1) +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TSupplier.Update_Data(Kode_Supp1, NAMA1, Alamat1,
    No_Telp1: string);
begin
    Sqlcmd:= 'UPDATE T_SUPPLIER SET NAMA= '+ QUOTEDSTR(NAMA1)
+' ,ALAMAT= '+ QUOTEDSTR(ALAMAT) +',' +
    'NO_TELP= '+ NO_TELP1 +' WHERE KODE_SUPPLIER = '+
Quotedstr(KODE_SUPP1) +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

```

---

```

end;

{ TCustomer }

constructor TCustomer.Create;
begin
    Kode_Customer:='';
    Nama:='';
    Alamat:='';
    No_Telp:='';
end;

procedure TCustomer.Delete_Data(Kode_Customer1: string);
begin
    Sqlcmd:='Delete From T_Customer Where Kode_Customer=' +
    Quotedstr(Kode_Customer1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TCustomer.Find_Customer_InPenjualan(
    Kode_Customer1: string): Boolean;
begin
    Result:=false;
    sqlcmd:='Select * From T_Penjualan Where Kode_Customer=' +
    Quotedstr(Kode_Customer1) +'';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:=True;
        end;
    end;
end;

function TCustomer.Find_Kode_Customer(Kode_Customer1: string): Boolean;
begin
    Result:=false;
    sqlcmd:='Select * From T_Customer Where Kode_Customer=' +
    Quotedstr(Kode_Customer1) +'';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;

```

---

```

sql.Add(sqlcmd);
open;
if not (isempty) then
begin
    Result:=True;
    Nama:=Fieldbyname('Nama').AsString;
    Alamat:=Fieldbyname('Alamat').AsString;
    No_Telp:=Fieldbyname('No_Telp').AsString;
end;
end;
end;

function TCustomer.Get_Kode_Customer: string;
begin
    sqlcmd:='Select Kode_Customer From T_Customer order by Kode_Customer
desc Limit 1';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:='C-' + copy('00000',1,5-
length(inttostr(strtoint(RightStr(Fieldbyname('Kode_Customer').AsString,5)) +1)))
+
inttostr(strtoint(Rightstr(Fieldbyname('Kode_Customer').AsString,5))+1)
        else
            Result:='C-00001';
        end;
    end;
end;

procedure TCustomer.Save_Data(Kode_Customer1, NAMA1, Alamat1,
    No_Telp1: string);
begin
    Sqlcmd:= 'INSERT INTO T_CUSTOMER VALUES('+
Quotedstr(KODE_CUSTOMER1) +',' + Quotedstr(NAMA1) +',' + ' +
"+ Quotedstr(ALAMAT1) +',' + Quotedstr(NO_TELP1) +')';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TCustomer.Update_Data(Kode_Customer1, NAMA1, Alamat1,
    No_Telp1: string);
begin

```

---

```

    Sqlcmd:= 'UPDATE T_CUSTOMER SET NAMA= '+ QUOTEDSTR(NAMA1)
+ ',ALAMAT= '+ QUOTEDSTR(ALAMAT) +', ' +
    'NO_TELP= '+ NO_TELP1 +' WHERE KODE_CUSTOMER = '+
Quotedstr(KODE_CUSTOMER1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ TBarang_Jadi }

constructor TBarang_Jadi.Create;
begin
    Kode_BJ:='';
    Nama:='';
    Harga_Jual:='';
    Stock:=0;
    Stock_Pesan:=0;
end;

procedure TBarang_Jadi.Delete_Data(Kode_BJ1: string);
begin
    Sqlcmd:='Delete From T_Barang_jadi Where Kode_BJ='+ Quotedstr(Kode_BJ1)
+'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TBarang_Jadi.Find_BJ_InItemPemesanan(Kode_BJ1: string): Boolean;
begin
    Result:=false;
    sqlcmd:='Select Kode_BJ From T_Item_Pemesanan Where Kode_BJ='+
Quotedstr(Kode_BJ1) +'';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:=True;
        end;
    end;
end;

function TBarang_Jadi.Find_BJ_InItemPenjualan(Kode_BJ1: string): Boolean;
begin
    Result:=false;

```

---

```
    sqlcmd:='Select Kode_BJ From T_Item_Penjualan Where Kode_BJ='+  
Quotedstr(Kode_BJ1) +";  
    With FDatabase.Search do  
    begin  
        close;  
        sql.Clear;  
        sql.Add(sqlcmd);  
        open;  
        if not (isempty) then  
            Result:=True;  
    end;  
end;
```

```
function TBarang_Jadi.Find_BJ_InPenjadwalan(Kode_BJ1: string): Boolean;  
begin  
    Result:=false;  
    sqlcmd:='Select Kode_BJ From T_Penjadwalan Where Kode_BJ='+  
Quotedstr(Kode_BJ1) +";  
    With FDatabase.Search do  
    begin  
        close;  
        sql.Clear;  
        sql.Add(sqlcmd);  
        open;  
        if not (isempty) then  
            Result:=True;  
    end;  
end;
```

```
end;
```

```
function TBarang_Jadi.Find_BJ_InPeramalan(Kode_BJ1: string): Boolean;  
begin  
    Result:=false;  
    sqlcmd:='Select Kode_BJ From T_Peramalan Where Kode_BJ='+  
Quotedstr(Kode_BJ1) +";  
    With FDatabase.Search do  
    begin  
        close;  
        sql.Clear;  
        sql.Add(sqlcmd);  
        open;  
        if not (isempty) then  
            Result:=True;  
    end;  
end;
```

---

end;

function TBarang\_Jadi.Find\_BJ\_InResep(Kode\_BJ1: string): Boolean;

begin

Result:=false;

sqlcmd:='Select Kode\_BJ From T\_Resep Where Kode\_BJ='+  
Quotedstr(Kode\_BJ1) +";

With FDatabase.Search do

begin

close;

sql.Clear;

sql.Add(sqlcmd);

open;

if not (isempty) then

Result:=True;

end;

end;

function TBarang\_Jadi.Find\_Kode\_BJ(Kode\_BJ1: string): Boolean;

begin

Result:=false;

sqlcmd:='Select \* From T\_Barang\_jadi Where Kode\_BJ='+  
Quotedstr(Kode\_BJ1) +";

With FDatabase.Search do

begin

close;

sql.Clear;

sql.Add(sqlcmd);

open;

if not (isempty) then

begin

Result:=True;

Nama:=Fieldbyname('Nama').AsString;

Harga\_Jual:=Fieldbyname('Harga\_Jual').AsString;

Stock:=Fieldbyname('Stock').AsInteger;

Stock\_Pesan:= Fieldbyname('Stock\_Pesan').AsInteger;

end;

end;

end;

function TBarang\_Jadi.Get\_Kode\_BJ: string;

begin

sqlcmd:='Select Kode\_BJ From T\_Barang\_Jadi order by Kode\_BJ desc Limit 1';

With FDatabase.Search do

begin



```

close;
sql.Clear;
sql.Add(sqlcmd);
open;
if not (isempty) then
    Result:='BJ-' + copy('00000',1,5-
length(inttostr(strtoint(RightStr(Fieldbyname('Kode_BJ').AsString,5)) +1))) +
inttostr(strtoint(RightStr(Fieldbyname('Kode_BJ').AsString,5))+1)
    else
        Result:='BJ-00001';
    end;
end;

procedure TBarang_Jadi.Save_Data(Kode_BJ1, NAMA1, Harga_Jual1,
    Stock1,Stock_Pesan1: string);
begin
    Sqlcmd:= 'INSERT INTO T_BARANG_JADI VALUES('+ Quotedstr(KODE_BJ1) +','+
    Quotedstr(NAMA1) +',' +
    '+ HARGA_JUAL1 +','+ STOCK1 +','+ STOCK_PESAN1 +')';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TBarang_Jadi.Update_Data(Kode_BJ1, NAMA1, Harga_Jual1: string);
begin
    Sqlcmd:= 'UPDATE T_BARANG_JADI SET NAMA= '+ Quotedstr(NAMA1) +',' +
    'HARGA_JUAL= '+ HARGA_JUAL1 + ' WHERE KODE_BJ= '+
    Quotedstr(KODE_BJ1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TBarang_Jadi.Update_StockBJ(Kode_BJ1, Stock1: string);
begin
    Sqlcmd:= 'UPDATE T_BARANG_JADI SET STOCK= STOCK + '+ STOCK1 + ' WHERE
    KODE_BJ= '+ Quotedstr(KODE_BJ1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TBarang_Jadi.Update_StockPesan(Kode_BJ1, Stock_Pesan1: string);
begin
    Sqlcmd:= 'UPDATE T_BARANG_JADI SET STOCK_PESAN= STOCK_PESAN + '+
    STOCK_PESAN1 + ' WHERE KODE_BJ= '+ Quotedstr(KODE_BJ1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

```

---

```

{ TOrder }

function TOrder.Get_NoNotaOrder: string;
Var
  Temp:string;
begin
  Temp:='O-' + Formatdatetime('yyMM',date)+'%';
  sqlcmd:='Select No_Nota_order From T_Order Where No_Nota_Order Like '+
Quotedstr(Temp) +' order by No_Nota_Order desc Limit 1';
  With FDatabase.Search do
  begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (Isempy) then
      result:='O-' + Formatdatetime('yyMM',date)+ copy('00000',1,5-
length(inttostr(strtoint(RightStr(Fieldbyname('No_Nota_Order').AsString,5)) +1)))
+
inttostr(strtoint(Rightstr(Fieldbyname('No_Nota_Order').AsString,5))+1)
    else
      result:='O-' + Formatdatetime('yyMM',date)+ '00001';
    end;
  end;

procedure TOrder.Save_Data_Order(No_Nota_Order1, Kode_Supplier1,
Tgl_Order1,
  Total1: string);
begin
  Sqlcmd:= 'INSERT INTO T_ORDER VALUES('+ Quotedstr(No_Nota_Order1) +','+
Quotedstr(Kode_Supplier1) +',' +
"+ Quotedstr(Tgl_Order1) +','+ TOTAL1 +')';
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TOrder.Save_Data_Order_Detail(no_item_order1, No_Nota_Order1,
  Kode_bb1, harga_beli1, Jumlah1, Sub_Total1: string);
begin
  Sqlcmd:= 'INSERT INTO T_ITEM_ORDER VALUES(+
Quotedstr(No_item_Order1) +','+ Quotedstr(No_Nota_Order1) +',' +
"+ Quotedstr(Kode_bb1) +','+ harga_beli1 +','+ Jumlah1 +','+ Sub_Total1 +')';
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

```

---

```

{ TPenjualan }

function TPenjualan.Get_NoNotaJual: string;
Var
  Temp:string;
begin
  Temp:='J-' + Formatdatetime('yyMM',date)+'%';
  sqlcmd:='Select No_Nota_Jual From T_Penjualan Where No_Nota_Jual Like '+
Quotedstr(Temp) +' order by No_Nota_Jual desc Limit 1';
  With FDatabase.Search do
  begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (isempty) then
      result:='J-' + Formatdatetime('yyMM',date)+ copy('00000',1,5-
length(inttostr(strtoint(RightStr(Fieldbyname('No_Nota_Jual').AsString,5)) +1))) +
inttostr(strtoint(Rightstr(Fieldbyname('No_Nota_Jual').AsString,5))+1)
    else
      result:='J-' + Formatdatetime('yyMM',date)+ '00001';
    end;
  end;
end;

procedure TPenjualan.Save_Data_Jual(No_Nota_Jual1, Tgl_Jual1,
  Kode_Customer1, Total1: string);
begin
  Sqlcmd:= 'INSERT INTO T_PENJUALAN VALUES('+ Quotedstr(No_Nota_Jual1)
+' '+ Quotedstr(Tgl_Jual1) +', ' +
  "+ Quotedstr(Kode_Customer1) +','+ TOTAL1 +)';
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TPenjualan.Save_Data_Jual_Detail(no_item_Jual1, No_Nota_Jual1,
  Kode_bj1, Jumlah1, Sub_Total1: string);
begin
  Sqlcmd:= 'INSERT INTO T_ITEM_PENJUALAN VALUES('+
Quotedstr(No_item_jual1) +','+ Quotedstr(No_Nota_jual1) +', ' +
  "+ Quotedstr(Kode_bj1) +','+ Jumlah1 +','+ Sub_Total1 +)';
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ TResep }

```

---

```

constructor TResep.Create;
begin
  No_Resep:="";
  Kode_BJ:="";
  Kode_BB:="";
  Jumlah_BB:=0;
end;

procedure TResep.Delete_Data_Resep(Kode_BJ1:string);
begin
  Sqlcmd:='Delete From T_Resep Where Kode_BJ= '+ Quotedstr(Kode_BJ1) +";
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TResep.Find_No_Resep(No_Resep: String): Boolean;
begin
  result:=false;
  Sqlcmd:='Select * From T_Resep Where No_Resep= '+ Quotedstr(No_Resep) +
';
  With FDatabase.Search do
  begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (isempty) then
    begin
      result:=true;
      No_Resep:=Fieldbyname('No_Resep').AsString;
      Kode_BJ:=Fieldbyname('Kode_BJ').AsString;
      Kode_BB:=Fieldbyname('Kode_BB').AsString;
      Jumlah_BB:=Fieldbyname('Jumlah_BB').AsInteger;
    end;
  end;
end;

function TResep.Find_Resep(Kriteria: string): Boolean;
begin
  result:=false;
  Sqlcmd:='Select * From T_Resep ' + Kriteria + 'order by No_Resep Limit 1 ';
  With FDatabase.Search do
  begin
    close;
    sql.Clear;

```

---

```

sql.Add(sqlcmd);
open;
if not (isempty) then
begin
    result:=true;
    No_Resep:=Fieldbyname('No_Resep').AsString;
    Kode_BJ:=Fieldbyname('Kode_BJ').AsString;
    Kode_BB:=Fieldbyname('Kode_BB').AsString;
    Jumlah_BB:=Fieldbyname('Jumlah_BB').AsInteger;
end;
end;
end;

function TResep.Get_No_Resep(Kode_BJ1: string; Kode_BB1: string): string;
begin
    Result:=Kode_BJ1 + Kode_BB1;
end;

procedure TResep.Save_Data_Resep(No_Resep1, Kode_BB1, Kode_BJ1,
    Jumlah_BB1: string);
begin
    Sqlcmd:= 'INSERT INTO T_RESEP VALUES(' + Quotedstr(No_Resep1) + ',' +
    Quotedstr(Kode_BB1) + ',' +
    " + Quotedstr(Kode_BJ1) + ',' + Jumlah_BB1 + ')';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ TPesan }

constructor TPesan.Create;
begin
    no_pesan := "";
    kode_customer := "";
    tgl_pesan := "";
    total_pesan := "";
    tgl_selesai := "";
    Uang_muka := "";
    status := 0;
end;

function TPesan.Find_Pesanan(No_Pesan1: string): Boolean;
begin
    Result:=False;
    Sqlcmd:='Select * From T_Pemesanan Where No_Pesan= ' +
    Quotedstr(No_Pesan1) + ' ';

```

---

```

With FDatabase.Search do
begin
  close;
  sql.Clear;
  sql.Add(sqlcmd);
  open;
  if not (isempty) then
  begin
    result:=True;
    no_pesanan      := fieldbyname('no_pesanan').AsString;
    kode_customer  := fieldbyname('kode_customer').AsString;
    tgl_pesanan    := fieldbyname('tgl_pesanan').AsString;
    total_pesanan  := fieldbyname('total_pesanan').AsString;
    tgl_selesai    := fieldbyname('tgl_pesanan').AsString;
    Uang_muka      := fieldbyname('uang_muka').AsString;
    status         := fieldbyname('status').AsInteger;
  end;
end;

end;

function TPesanan.Find_Pesanan_BelumSelesai(No_Pesanan1: String): Boolean;
begin
  Result:=False;
  Sqlcmd:='Select * From T_Penjadwalan Where Status=0 And No_Pesanan= '+
Quotedstr(No_Pesanan1) +'';
  With FDatabase.Search do
  begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (isempty) then
      result:=True;
    end;
  end;
end;

function TPesanan.Get_NoNotaPesanan: string;
Var
  Temp:string;
begin
  Temp:='P-' + Formatdatetime('yyMM',date)+'%';
  sqlcmd:='Select No_Pesanan From T_Pemesanan Where No_Pesanan Like '+
Quotedstr(Temp) +'' order by No_Pesanan desc Limit 1';
  With FDatabase.Search do

```

---

```

begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (isempty) then
        result:='P-' + Formatdatetime('yyMM',date)+ copy('00000',1,5-
length(inttostr(strtoint(RightStr(Fieldbyname('No_Pesan').AsString,5)) +1))) +
inttostr(strtoint(Rightstr(Fieldbyname('No_Pesan').AsString,5))+1)
    else
        result:='P-' + Formatdatetime('yyMM',date)+ '00001';
    end;
end;

procedure
TPesan.Save_Data_Pesan(No_Nota_Pesan1,Kode_Customer1,Tgl_Pesan1,total_p
esan1,
    Tgl_selesai1,uang_muka1,status1:string);
begin
    Sqlcmd:= 'INSERT INTO T_PEMESANAN VALUES('+
Quotedstr(No_Nota_Pesan1) +','+ Quotedstr(Kode_Customer1) +',' +
    "+ Quotedstr(Tgl_Pesan1) +','+ total_pesan1 +','+ Quotedstr(Tgl_selesai1)
+','+ uang_muka1 +','+ status1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TPesan.Save_Data_Pesan_Detail(no_item_pesan1, No_pesan1,
    Kode_bj1, Jumlah_pesan1, Sub_Total1: string);
begin
    Sqlcmd:= 'INSERT INTO T_ITEM_PEMESANAN VALUES('+
Quotedstr(No_item_pesan1) +','+ Quotedstr(No_pesan1) +',' +
    "+ Quotedstr(Kode_bj1) +','+ Jumlah_pesan1 +','+ Sub_Total1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TPesan.Update_StatusPesan(No_Pesan1, Status1: String);
begin
    Sqlcmd:='Update T_PEMESANAN Set Status= '+ Status1 + ' Where No_Pesan =
'+ Quotedstr(No_Pesan1) +";
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TPesan.Update_Tgl_Selesai(No_Nota_Pesan1, Tgl_Selesai1: string);

```

---

```

begin
    Sqlcmd:='Update T_PEMESANAN Set Tgl_Selesai= '+ Quotedstr(Tgl_Selesai) +'
Where No_Pesan = '+ Quotedstr(No_Nota_Pesan1) +";
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ TKapasitas_Produksi }

constructor TKapasitas_Produksi.Create;
begin
    KODE_KAPASITAS := "";
    TGL_SETTING := "";
    MAX_PRODUKSI := "";
    MAX_PRODUKSI_PESAN := "";
end;

procedure TKapasitas_Produksi.Delete_Data(Kode_Kapasitas1: string);
begin
    Sqlcmd:= 'Delete From T_Kapasitas_Produksi Where Kode_Kapasitas= '+
Quotedstr(Kode_Kapasitas) +";
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TKapasitas_Produksi.Find_Kode_Kapasitas(
    Kode_Kapasitas1: string): Boolean;
begin
    Result:=false;
    sqlcmd:='Select * From T_Kapasitas_Produksi Where Kode_Kapasitas='+
Quotedstr(Kode_Kapasitas1) +";
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            begin
                Result:=True;
                KODE_KAPASITAS := Fieldbyname('Kode_Kapasitas').AsString;
                TGL_SETTING := Fieldbyname('Tgl_Setting').AsString;
                MAX_PRODUKSI := Fieldbyname('Max_Produksi').AsString;
                MAX_PRODUKSI_PESAN := Fieldbyname('Max_Produksi_Pesan').AsString;
            end;
        end;
    end;
end;

```

---



```

function TKapasitas_Produksi.Get_Kode_Kapasitas: string;
Var
  Temp:string;
begin
  Temp:='K-' + Formatdatetime('yyMM',date)+'%';
  sqlcmd:='Select KODE_KAPASITAS From T_Kapasitas_Produksi Where
Kode_Kapasitas Like '+ Quotedstr(Temp) +' order by Kode_Kapasitas desc Limit
1';
  With FDatabase.Search do
  begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (isempty) then
      result:='K-' + Formatdatetime('yyMM',date)+ copy('00',1,2-
length(inttostr(strtoint(RightStr(Fieldbyname('Kode_Kapasitas').AsString,2)) +1)))
+
inttostr(strtoint(Rightstr(Fieldbyname('Kode_Kapasitas').AsString,2))+1)
      else
        result:='K-' + Formatdatetime('yyMM',date)+ '01';
    end;
  end;
end;

procedure TKapasitas_Produksi.Save_Data(Kode_Kapasitas1, Tgl_Setting1,
  Max_Produksi1, Max_Produksi_Pesan1: string);
begin
  Sqlcmd:= 'INSERT INTO T_KAPASITAS_PRODUKSI VALUES('+
Quotedstr(Kode_Kapasitas1) +','+ Quotedstr(Tgl_Setting1) +',' +
  "+ Max_Produksi1 +','+ Max_Produksi_Pesan1 +')';
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TKapasitas_Produksi.Update_Data(Kode_Kapasitas1, Tgl_Setting1,
  Max_Produksi1, Max_Produksi_Pesan1: string);
begin
  Sqlcmd:= 'UPDATE T_KAPASITAS_PRODUKSI SET TGL_SETTING= '+
QUOTEDSTR(TGL_SETTING1) +',MAX_PRODUKSI= '+ MAX_PRODUKSI +',' +
  'MAX_PRODUKSI_PESAN= '+ MAX_PRODUKSI_PESAN1 +' WHERE
KODE_KAPASITAS = '+ Quotedstr(KODE_KAPASITAS1) +'';
  FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

```

---

```

{ TPenjadwalan }
function TPenjadwalan.Get_Juml_Pesanan(Tgl_Jadwal:string): integer;
begin
    result:=0;
    Sqlcmd:= 'Select sum(Jumlah) From T_Penjadwalan Where Tgl_Jadwal= '+
Quotedstr(Tgl_Jadwal) + ' ' +
    'order by Tgl_Jadwal desc Limit 1 ';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            Result:= Fields[0].AsInteger;
        end;
    end;
end;

function TPenjadwalan.Get_No_Jadwal(Tgl_Jadwal1: string): string;
Var
    Temp:string;
begin
    Temp:='JD-' + Formatdatetime('yyMMdd',strtodate(Tgl_Jadwal1))+'%';
    sqlcmd:='Select No_Jadwal From T_Penjadwalan Where No_Jadwal Like '+
Quotedstr(Temp) + ' order by No_Jadwal desc Limit 1';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
            result:='JD-' + Formatdatetime('yyMMdd',strtodate(Tgl_Jadwal1))+
copy('000',1,3-
length(inttostr(strtoint(RightStr(Fieldbyname('No_Jadwal').AsString,3)) +1))) +
inttostr(strtoint(Rightstr(Fieldbyname('No_Jadwal').AsString,3))+1)
        else
            result:='JD-' + Formatdatetime('yyMMdd',strtodate(Tgl_Jadwal1))+ '001';
        end;
    end;
end;

function TPenjadwalan.Get_Kapasitas_Produksi(
    Tgl_Jadwal: string): integer;

```

---

```

Var
    Hasil:integer;
    Max_Pesan:real;
begin
    Hasil:=0;
    Sqlcmd:='select MAX_PRODUKSI,MAX_PRODUKSI_PESAN from
t_kapasitas_produksi ' +
    'where (TGL_SETTING < '+ Quotedstr(Tgl_Jadwal) +)';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
        if not (isempty) then
        begin
            Max_Pesan:=Fieldbyname('Max_Produksi_Pesan').AsInteger/100;
            Hasil:= Round(Fieldbyname('Max_Produksi').AsInteger * Max_Pesan);
        end;
    end;
    Result:=Hasil;
end;

procedure TPenjadwalan.Save_Data(no_jadwal1, Tgl_jadwal1, no_pesanan1,
kode_bj1, jumlah1, status1: string);
begin
    Sqlcmd:='INSERT INTO T_PENJADWALAN VALUES('+ Quotedstr(no_jadwal1)
+', '+ Quotedstr(Tgl_jadwal1) +', ' +
    '+ Quotedstr(no_pesanan1) +', '+ Quotedstr(kode_bj1) +', '+ jumlah1 +', '+
status1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TPenjadwalan.Update_Status(no_jadwal1, status1: string);
begin
    sqlcmd:='Update T_Penjadwalan set status= '+ status1 + ' Where No_Jadwal=
'+ Quotedstr(no_jadwal1) +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TPenjadwalan.Get_Sisa_Kapasitas_Produksi(
    Tgl_Jadwal: string): integer;
begin
    result:=Get_Kapasitas_Produksi(Tgl_Jadwal)-Get_Juml_Pesanan(Tgl_Jadwal);
end;

```

```

end;

constructor TPenjadwalan.Create;
begin
    No_Jadwal := "";
    Tgl_jadwal := "";
    no_pesanan := "";
    kode_bj := "";
    jumlah := "";
    status := 0;
end;

function TPenjadwalan.Find_No_Jadwal(No_Jadwal1: string): Boolean;
begin
    result:=false;
    Sqlcmd:='Select * From T_Penjadwalan Where No_Jadwal = '+
    Quotedstr(No_Jadwal1) +';
    with FDatabase do
    begin
        search.Close;
        search.SQL.Clear;
        search.SQL.Add(sqlcmd);
        search.Open;
        if not (search.IsEmpty) then
        begin
            result:=True;
            No_Jadwal :=Search.FieldByName('No_Jadwal').AsString;
            Tgl_jadwal :=Search.FieldByName('Tgl_jadwal').AsString;
            no_pesanan :=Search.FieldByName('No_pesanan').AsString;
            kode_bj :=Search.FieldByName('kode_bj').AsString;
            jumlah :=Search.FieldByName('jumlah').AsString;
            status :=Search.FieldByName('Status').AsInteger;
        end;
    end;
end;

{ TProduksi_Pesan }

function TProduksi_Pesan.Get_NoProduksiPesan: string;
Var
    Temp:string;
begin
    Temp:='P-' + Formatdatetime('yyMMdd',date)+'%';
    sqlcmd:='Select No_Produksi From T_Produksi_Pesan Where No_Produksi Like
'+ Quotedstr(Temp) +' order by No_Produksi desc Limit 1';

```

---

```

With FDatabase.Search do
begin
    close;
    sql.Clear;
    sql.Add(sqlcmd);
    open;
    if not (isempty) then
        result:='P-' + Formatdatetime('yyMMdd',date)+ copy('000',1,3-
length(inttostr(strtoint(RightStr(Fieldbyname('No_Produksi').AsString,3)) +1))) +
inttostr(strtoint(Rightstr(Fieldbyname('No_Produksi').AsString,3))+1)
    else
        result:='P-' + Formatdatetime('yyMMdd',date)+ '001';
    end;
end;

procedure TProduksi_Pesan.Save_Produksi_Pesan(no_produksi1, no_jadwal1,
    tgl_produksi1,Juml_Produksi1: string);
begin
    Sqlcmd:= 'INSERT INTO T_PRODUKSI_PESAN VALUES('+
    Quotedstr(no_produksi1) +','+ Quotedstr(no_jadwal1) +',' +
    "+ Quotedstr(tgl_produksi1) +','+ Juml_Produksi1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TProduksi_Pesan.Save_Produksi_Pesan_Detail(no_item_produksi1,
    no_produksi1,no_resep1,jumlah_BB1: string);
begin
    Sqlcmd:= 'INSERT INTO T_ITEM_PRODUKSI_PESAN VALUES('+
    Quotedstr(no_item_produksi1) +','+ Quotedstr(no_produksi1) +',' +
    "+ Quotedstr(no_resep1) +','+ jumlah_BB1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ TProduksi }

function TProduksi.Get_NoProduksi: string;
Var
    Temp:string;
begin
    Temp:='P-' + Formatdatetime('yyMMdd',date)+'%';
    sqlcmd:='Select No_Produksi From T_Produksi Where No_Produksi Like '+
    Quotedstr(Temp) +' order by No_Produksi desc Limit 1';
    With FDatabase.Search do
        begin

```

---

```

close;
sql.Clear;
sql.Add(sqlcmd);
open;
if not (isempty) then
    result:='P-' + Formatdatetime('yyMMdd',date)+ copy('000',1,3-
length(inttostr(strtoint(RightStr(Fieldbyname('No_Produksi').AsString,3)) +1))) +
inttostr(strtoint(Rightstr(Fieldbyname('No_Produksi').AsString,3))+1)
    else
        result:='P-' + Formatdatetime('yyMMdd',date)+ '001';
    end;
end;

procedure TProduksi.Save_Produksi(no_produksi1, tgl_produksi1,
    Kode_bj1,Jumlah_Produksi1: string);
begin
    Sqlcmd:= 'INSERT INTO T_PRODUKSI VALUES('+ Quotedstr(no_produksi1) +','+
    Quotedstr(tgl_produksi1) +','+ Quotedstr(kode_bj1) +','+ Jumlah_Produksi1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TProduksi.Save_Produksi_Detail(no_item_produksi1, no_produksi1,
    no_resep1,jumlah_BB1: string);
begin
    Sqlcmd:= 'INSERT INTO T_ITEM_PRODUKSI VALUES('+
    Quotedstr(no_item_produksi1) +','+ Quotedstr(no_produksi1) +','+
    '+ Quotedstr(no_resep1) +','+ jumlah_BB1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ TKaryawan }

constructor TKaryawan.Create;
begin
    NIP:="";
    NAMA:="";
    ALAMAT:="";
    NO_TELP:="";
    JENIS_KELAMIN:=0;
    JABATAN:=0;
    STATUS:=0;
end;

procedure TKaryawan.Delete_Data(NIP1: string);

```

---

```

begin
    Sqlcmd:='Delete From T_Karyawan Where NIP= '+ Quotedstr(NIP1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

function TKaryawan.Find_NIP(NIP1: string): boolean;
begin
    result:=false;
    Sqlcmd:='Select * From T_Karyawan Where NIP = '+ Quotedstr(NIP1) +'';
    with FDatabase do
    begin
        search.Close;
        search.SQL.Clear;
        search.SQL.Add(sqlcmd);
        search.Open;
        if not (search.IsEmpty) then
        begin
            result:=True;
            NAMA:=Search.FieldByName('Nama').AsString;
            ALAMAT:=Search.FieldByName('Alamat').AsString;
            NO_TELP:=Search.FieldByName('No_Telp').AsString;
            JENIS_KELAMIN:=Search.FieldByName('Jenis_Kelamin').AsInteger;
            JABATAN:=search.Fieldbyname('Jabatan').AsInteger;
            STATUS:=Search.FieldByName('Status').AsInteger;
        end;
    end;
end;

procedure TKaryawan.Save_Data(NIP1, NAMA1, ALAMAT1, NO_TELP1,
    JENIS_KELAMIN1, JABATAN1, STATUS1: string);
begin
    Sqlcmd:= 'INSERT INTO T_KARYAWAN VALUES('+ Quotedstr(NIP1) +','+
    Quotedstr(NAMA1) +',' +
    "+ Quotedstr(ALAMAT1) +','+ Quotedstr(NO_TELP1) +','+ JENIS_KELAMIN1
    +',' +
    "+ JABATAN1 +','+ STATUS1 +')';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure TKaryawan.Update_Data(NIP1, NAMA1, ALAMAT1, NO_TELP1,
    JENIS_KELAMIN1, JABATAN1, STATUS1: string);
begin
    Sqlcmd:= 'UPDATE T_KARYAWAN SET NAMA='+ Quotedstr(NAMA1) +',' +
    'ALAMAT='+ Quotedstr(ALAMAT1) +','+ NO_TELP='+ Quotedstr(NO_TELP1)
    +','+ JENIS_KELAMIN='+ JENIS_KELAMIN1 +',' +

```

---

```
        'JABATAN='+ JABATAN1 +',STATUS='+ STATUS1 +' WHERE NIP= '+  
Quotedstr(NIP1) +";  
        FDatabase.ADOConn.ExecuteDirect(Sqlcmd);  
end;
```

```
{ T_UserAcc }
```

```
constructor T_UserAcc.Create;  
begin  
    NIP      :=";  
    Passwd   :=";  
    Hak_Akses :=";  
end;
```

```
procedure T_UserAcc.Delete_UserAcc(NIP1: string);  
begin  
    Sqlcmd:= 'Delete From T_UserAcc Where NIP = '+ Quotedstr(NIP1) +";  
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);  
end;
```

```
function T_UserAcc.Find_UserID(NIP: string): Boolean;  
begin  
    result:= False;  
    Sqlcmd:= 'Select * From T_UserAcc Where NIP = '+ Quotedstr(NIP) + ";  
    FDatabase.search.Close;  
    FDatabase.search.SQL.Clear;  
    FDatabase.search.SQL.Add(sqlcmd);  
    FDatabase.search.Open;  
    if not (FDatabase.search.IsEmpty) then  
    begin  
        result:= True;  
        Hak_Akses:= FDatabase.Search.Fieldbyname('Hak_Akses').AsString;  
    end;  
end;
```

```
function T_UserAcc.Find_UserID1(NIP1, Passwd1: String): Boolean;  
begin  
    Result:= False;  
    Sqlcmd:= 'Select NIP ,PASSWORD,HAK_AKSES From T_USERACC Where NIP = '+  
Quotedstr(NIP1) +' and Passwd = '+ Quotedstr(Passwd1)+";  
    FDatabase.search.Close;  
    FDatabase.search.SQL.Clear;  
    FDatabase.search.SQL.Add(sqlcmd);  
    FDatabase.search.Open;
```

---



```

if not (FDatabase.search.IsEmpty) then
begin
    Result:= True;
    HAK_AKSES1:= FDatabase.Search.Fieldbyname('Hak_Akses').Asinteger;
End;
end;

procedure T_UserAcc.Save_UserAcc(NIP1, Passwd1, Hak_Akses1: string);
begin
    Sqlcmd:= 'INSERT INTO T_USERACC VALUES('+ Quotedstr(NIP1) +','+
    Quotedstr(PASSWD1) +',' +
    '+ HAK_AKSES1 +)';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure T_UserAcc.Update_Password(NIP1, Passwd1: string);
begin
    Sqlcmd:= 'UPDATE T_USERACC SET PASSWD= '+ QUOTEDSTR(Passwd1) +' ' +
    'WHERE NIP= '+ Quotedstr(NIP1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

procedure T_UserAcc.Update_UserAcc(NIP1, Passwd1, Hak_Akses1: string);
begin
    Sqlcmd:= 'UPDATE T_USERACC SET PASSWD= '+ QUOTEDSTR(PASSWD) +' ' +
    'HAK_AKSES= '+ HAK_AKSES1 +' ' +
    'WHERE NIP= '+ Quotedstr(NIP1) +'';
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
end;

{ T_Bayar_Pesanan }

function TBayar_Pesanan.Get_NoBayarPesan: string;
Var
    Temp:string;
begin
    Temp:='BP-' + Formatdatetime('yyMM',date)+'%';
    sqlcmd:='Select No_Bayar From T_Bayar_Pesan Where No_Bayar Like '+
    Quotedstr(Temp) +' order by No_Bayar desc Limit 1';
    With FDatabase.Search do
    begin
        close;
        sql.Clear;
        sql.Add(sqlcmd);
        open;
    end;
end;

```

---

```
    if not (isempty) then
        result:='BP-' + Formatdatetime('yyMM',date)+ copy('00000',1,5-
length(inttostr(strtoint(RightStr(Fieldbyname('No_Bayar').AsString,5)) +1))) +
```

```
inttostr(strtoint(Rightstr(Fieldbyname('No_Bayar').AsString,5))+1)
```

```
    else
```

```
        result:='BP-' + Formatdatetime('yyMM',date)+ '00001';
```

```
    end;
```

```
end;
```

```
procedure TBayar_Pesanan.Save_BayarPesanan(no_Bayar1, tgl_Bayar1,
```

```
    No_Pesan1, Total_Bayar1: string);
```

```
begin
```

```
    Sqlcmd:= 'INSERT INTO T_BAYAR_PESAN VALUES('+ Quotedstr(no_Bayar1) +','+
Quotedstr(Tgl_Bayar1) +',' +
```

```
    '+ Quotedstr(No_Pesan1) +','+ total_Bayar1 +)';
```

```
    FDatabase.ADOConn.ExecuteDirect(Sqlcmd);
```

```
end;
```

```
end.
```

---