

**DESAIN SISTEM INFORMASI PENJUALAN OBAT PADA APOTEK RUMAH  
SAKIT UMUM DISTRICT LIQUIÇA TIMOR LESTE**

**SKRIPSI**



**Disusun oleh :**

**Mariza Fátima Dos Santos Da Cruz  
NIM. 0512731**

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

RECORDS SECTION  
STANDARD FORM NO. 64

1954

1954

RECORDS SECTION  
STANDARD FORM NO. 64

RECORDS SECTION  
STANDARD FORM NO. 64

1954

**LEMBAR PERSETUJUAN**

**DESAIN SISTEM INFORMASI PENJUALAN OBAT PADA APOTEK RUMAH  
SAKIT UMUM DISTRICT LIQUIÇA TIMOR LESTE**

**SKRIPSI**

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

**Disusun oleh :**  
**Mariza Fátima Dos Santos Da Cruz**  
**NIM: 0512731**

Mengetahui,

Ketua Program Studi Teknik Elektro S-1



*[Signature]*  
Ir. Yusuf Ismail Nakhoda, MT  
NIP. Y. 1018800189

Diperiksa dan disetujui,

Dosen Pembimbing I

Dosen Pembimbing II

*[Signature]*  
Joseph Dedy Irawan, ST. MT  
NIP. 197404162005011002

*[Signature]*  
Ahmad Faisol, ST  
NIP.P. 1031000431

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

## **ABSTRAK**

### **DESAIN SISTEM INFORMASI PENJUALAN OBAT PADA RUMAH SAKIT UMUM DISTRICT LIQUIÇA TIMOR LESTE**

**Mariza Fátima Dos Santos Da Cruz, NIM 0512731**

**Dosen Pembimbing: Joseph Dedy Irawan, ST. MT dan Ahmad Faisol, ST**

Komputer merupakan suatu perangkat yang dibutuhkan untuk proses penyajian data, agar data yang diolah dapat memberikan suatu informasi yang diperlukan oleh pimpinan ataupun suatu perusahaan yang membutuhkan. Keunggulan komputer dalam memproses data akan meningkatkan efektivitas, produktivitas, serta efisiensi suatu sistem informasi. Teknologi dan informatika semakin mendorong Apotek Rumah Sakit Umum District Liquiça Timor Leste sebagai satu-satunya apotek di kabupaten Liquiça Timor Leste untuk melakukan pembenahan guna meningkatkan pelayanannya.

Alasan dilakukannya penelitian ini adalah karena banyaknya kendala yang ditemui dalam pelayanannya sebagai penyalur dan penyedia obat bagi masyarakat. Kendala-kendala itu adalah: sistem pelaporan data berkala yang masih manual mengakibatkan seringnya terjadi kesalahan; kendala kedua yaitu pelayanan terhadap konsumen yang kurang efektif dan efisien, mengakibatkan adanya pengaruh terhadap kinerja pelayanan.

Untuk mengatasi masalah dan kendala ini maka penulis merasa perlu untuk membuat suatu aplikasi sistem informasi penjualan obat untuk menyajikan informasi secara tepat, akurat dan cepat. Diharapkan dengan sistem informasi ini dapat digunakan sebagai sarana untuk mengolah data-data informasi khususnya di Apotek Rumah Sakit Umum District Liquiça Timor Leste.

**Kata kunci :** sistem informasi, Apotek Rumah Sakit Umum District Liquiça Timor Leste.

## **KATA PENGANTAR**

Dengan memanjatkan puji syukur kehadirat Tuhan Yang Maha Kuasa, yang telah memberikan rahmat dan hidayah-Nya sehingga dapat terselesaikan laporan Skripsi ini dengan baik dan lancar.

Laporan skripsi ini merupakan salah satu persyaratan akademik dalam menyelesaikan program Strata 1 jurusan Teknik Elektro, Konsentrasi Teknik Komputer dan Informatika, Institut Teknologi Nasional Malang. Adapun judul Laporan Skripsi ini adalah :

### **DESAIN SISTEM INFORMASI PENJUALAN OBAT PADA APOTEK RUMAH SAKIT UMUM DISTRICT LIQUIÇA TIMOR LESTE**

Oleh karena itu pada kesempatan ini dengan segala kerendahan hati perkenankan untuk mengucapkan terima kasih kepada :

1. Bapak Prof.Dr.Eng. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang
2. Bapak Ir. Sidik Noertjahjono, MT, selaku Dekan Fakultas Teknologi Industri ITN Malang
3. Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku Ketua Jurusan Teknik Elektro
4. Bapak Joseph Dedy Irawan, ST. MT, selaku Dosen Pembimbing I yang telah memberikan bimbingan dalam pembuatan skripsi ini
5. Ahmad Faisol, ST, selaku Dosen Pembimbing II yang telah memberikan bimbingan dalam pembuatan skripsi ini
6. Semua pihak yang telah membantu dalam penulisan dan penyusunan skripsi ini.

Penulis berharap agar buku laporan skripsi ini dapat memberikan banyak manfaat bagi semua pihak yang membutuhkan, terlebih bagi rekan-rekan mahasiswa. Penulis menyadari bahwa hasil dari penyusunan laporan skripsi ini masih jauh dari sempurna,

untuk itu kritik dan saran dari pembaca sangat penulis harapkan demi perbaikan buku laporan skripsi ini.

Malang, Februari 2011

Penyusun

## DAFTAR ISI

<b>Lembar Persetujuan .....</b>	<b>i</b>
<b>Abstrak.....</b>	<b>ii</b>
<b>Kata Pengantar .....</b>	<b>iii</b>
<b>Daftar Isi.....</b>	<b>vi</b>
<b>Daftar Tabel.....</b>	<b>x</b>
<b>Daftar Gambar.....</b>	<b>Xi</b>

### **BAB I PENDAHULUAN**

1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	2
1.4 Tujuan .....	2
1.5 Metode Penelitian .....	3
1.6 Sistematika Penulisan.....	5

### **BAB II DASAR TEORI**

2.1 Pengertian Dasar Sistem.....	6
2.1.1 Komponen Dasar .....	6
2.1.2 Komponen Tambahan .....	6
2.2 Pengertian Dasar Informasi .....	7
2.3 Pengertian Sistem Informasi.....	7
2.4 Sistem Informasi Penjualan.....	10
2.5 Pengertian Database dan RDBMS.....	11
2.5.1 Definisi Database.....	11
2.5.2 Relational Database dan Managemen Sistem.....	12

2.6	Microsoft SQL.....	12
2.6.1	SQL.....	12
2.6.2	Microsoft SQL server 2000.....	13
2.7	Visual Studio.NET .....	14
2.8	Perangkat Analisis dan Perancangan.....	15
2.8.1	Diagram Aliran Data .....	15
2.8.2	Pemodelan Data.....	16
2.9	Metode Pengembangan Sistem <i>Waterfall</i> .....	17

### **BAB III ANALISA DAN PERANCANGAN SISTEM**

3.1	Sistem Informasi Apotek.....	19
3.1.1	Pengguna Sistem .....	19
3.1.2	Spesifikasi Sistem.....	20
3.2	Perancangan Sistem.....	21
3.2.1	Data Flow Diagram .....	21
3.2.2	DFD level 1 .....	22
3.2.3	Desain Data Base.....	23
3.2.3.1	Relasi Antar Tabel .....	23
3.2.3.2	Struktur Tabel Yang Di Gunakan.....	25
3.2.4	Desain Antar Muka Aplikasi.....	32
3.2.4.1	Desain Form Login .....	32
3.2.4.2	Desain Halaman Internal.....	32
3.2.4.3	Desain Halaman Entry Data Jenis Obat.....	33
3.2.4.4	Desain Halaman Entry Data Konsumen .....	33
3.2.4.5	Desain Halaman Entry Data Obat .....	34



3.2.4.6 Desain Halaman Entry Data Supplier .....	34
3.2.4.7 Desain Halaman Entry Data User .....	35
3.2.4.8 Desain Halaman Entry Data Pembelian.....	36
3.2.4.9 Desain Halaman Entry Data Penjualan .....	36
3.2.4.10 Desain Form Laporan.....	37

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

4.1 Implementasi Sistem .....	40
4.1.1 Menu Pendataan .....	41
4.1.1.1 Sub Menu Ganti Pasword .....	41
4.1.1.2 Sub Menu Data Jenis Obat.....	42
4.1.1.3 Sub Menu Konsumen.....	42
4.1.1.4 Sub Menu Data Obat.....	43
4.1.1.5 Sub Menu Supplier.....	43
4.1.1.6 Sub Menu User.....	44
4.1.1.7 Sub Menu Data Opname Obat.....	44
4.1.2 Menu Transaksi .....	45
4.1.2.1 Sub Menu Transaksi Pembelian Obat .....	45
4.1.2.2 Sub Menu Transaksi Pembelian Umum .....	45
4.1.2.3 Sub Menu Transaksi Penjualan Obat .....	46
4.1.2.4 Sub Menu Transaksi Penjualan Umum.....	47
4.1.3 Menu Laporan .....	47
4.1.3.1 Laporan Data Konsumen .....	47
4.1.3.2 Laporan Data Supplier .....	48
4.1.3.3 Laporan Data Obat .....	49
4.1.3.4 Laporan Pembelian .....	49

4.1.3.5 Laporan Penjualan.....	50
4.1.3.6 Laporan Data Stock Opname Obat.....	51
4.1.3.7 Laporan Rugi.....	51
4.1.3.8 Laporan Laba.....	52
4.1.4 Menu Nota.....	52
4.1.3.1 Nota Penjualan .....	52
4.1.3.2 Nota Pembelian.....	53
4.2 Perbandingan Sistem Lama dengan Sistem yang Sekarang .....	54

## **BAB V PENUTUP**

5.1 Kesimpulan.....	56
5.2 Saran .....	56

## **DAFTAR PUSTAKA**

## **LAMPIRAN**

## DAFTAR TABEL

Table 2.1 : Daftar Sejumlah Pernyataan SQL.....	13
Table 3.1 : Tabel T_Supplier .....	25
Table 3.2 : Tabel T_User .....	25
Table 3.3 : Tabel T_Penjualan .....	26
Table 3.4 : Tabel T_Konsumen.....	26
Table 3.5 : Tabel T_Pembelian .....	27
Table 3.6 : Tabel T_Detail_Penjualan .....	27
Table 3.7 : Tabel T_Retjual .....	28
Table 3.8 : Tabel T_Jenis Obat .....	28
Table 3.9 : Tabel T_Retbeli .....	29
Table 3.10 : Tabel T_Detail_Pembelian .....	29
Table 3.11 : Tabel T_Obat .....	30
Table 3.12 : Tabel T_Tbopname.....	31
Table 4.1 : Perbandingan Sistem Lama Dengan Sistem Yang Sekarang.....	54

## DAFTAR GAMBAR

Gambar 2.1 : Sistem Informasi Yang Sederhana .....	8
Gambar 2.2 : Sistem Informasi Berhubungan Dengan Berbagai Pemakai .....	9
Gambar 2.3 : Sistem Informasi Dan Teknologi Informasi Mempunyai Interaksi Terhadap Komponen-Komponen Penting Dalam Perusahaan .....	10
Gambar 2.4 : Simbol Utama DFD .....	15
Gambar 2.5 : Metode Air Terjun( <i>Waterfall</i> ).....	18
Gambar 3.1 : <i>Contex Diagram</i> Sistem Informasi Apotek.....	21
Gambar 3.2 : DFD Level 1 Sistem Informasi Apotek .....	22
Gambar 3.3 : CDM Basis Data Sistem Informasi Apotek .....	23
Gambar 3.4 : PDM Basis Data Sistem Informasi Apotek.....	24
Gambar 3.5 : Desain Form Login .....	32
Gambar 3.6 : Desain Form Menu Utama.....	32
Gambar 3.7 : Desain Form Data Jenis Obat.....	33
Gambar 3.8 : Desain Form Data Konsumen .....	34
Gambar 3.9 : Desain Form Data Obat.....	34
Gambar 3.10 : Desain Form Data Konsumen .....	35
Gambar 3.11 : Desain Form Data User.....	35
Gambar 3.12 : Desain Form Data Pembelian .....	36
Gambar 3.13 : Desain Form Data Penjualan.....	36
Gambar 3.14 : Desain Form Laporan.....	37
Gambar 3.15 : Desain Form Data Konsumen Pada Crystal Report.....	37
Gambar 3.16 : Desain Form Data Supplier Pada Crystal Report.....	38
Gambar 3.17 : Desain Form Data Obat Pada Crystal Report .....	38
Gambar 3.18 : Desain Laporan Pembelian Pada Crystal Report .....	39

Gambar 3.19 : Desain Laporan Penjualan Pada Crystal Report .....	39
Gambar 4.1 : Form Login .....	40
Gambar 4.2 : Menu Utama.....	41
Gambar 4.3 : Menu Ganti Password .....	41
Gambar 4.4 : Entry Data Jenis Obat .....	42
Gambar 4.5 : Entry Data Konsumen .....	42
Gambar 4.6 : Entry Data Obat .....	43
Gambar 4.7 : Entry Data Supplier.....	43
Gambar 4.8 : Entry Data User.....	44
Gambar 4.9 : Entry Data Opname Obat.....	44
Gambar 4.10 : Transaksi Pembelian Obat .....	45
Gambar 4.11 : Transaksi Pembelian Umum .....	46
Gambar 4.12 : Transaksi Penjualan Obat.....	46
Gambar 4.13 : Transaksi Penjualan Umum .....	47
Gambar 4.14 : Laporan Data Konsumen .....	48
Gambar 4.15 : Laporan Data Supplier .....	48
Gambar 4.16 : Laporan Data Obat .....	49
Gambar 4.17 : Laporan Pembelian .....	50
Gambar 4.18 : Laporan Penjualan.....	50
Gambar 4.19 : Laporan Data Stock Opname Obat .....	51
Gambar 4.20 : Laporan Rugi.....	51
Gambar 4.21 : Laporan Laba .....	52
Gambar 4.22 : Nota Penjualan .....	52
Gambar 4.23 : Nota Pembelian .....	53

# **BAB I**

## **PENDAHULUAN**

### **1.1 LATAR BELAKANG**

Komputer merupakan suatu perangkat yang sangat dibutuhkan untuk proses penyajian data, agar data yang diolah dapat memberikan suatu informasi yang diperlukan oleh pimpinan ataupun suatu perusahaan yang membutuhkan. Perkembangan komputer mengalami perubahan yang sangat pesat, kebutuhan akan informasi yang akurat, tepat dan terkini juga semakin dibutuhkan agar dapat tetap bertahan dalam menghadapi segala tantangan di era globalisasi dan persaingan bebas.

Perkembangan teknologi dan ilmu informatika yang sangat pesat mendorong masyarakat baik kelompok maupun perorangan, instansi baik pemerintah maupun swasta untuk memanfaatkan perkembangan teknologi dan ilmu informatika tersebut. Keunggulan komputer dalam memproses data akan meningkatkan efektivitas, produktivitas, serta efisiensi suatu sistem informasi. Teknologi dan Informatika semakin mendorong Apotek Rumah Sakit Umum Distric Liquiça Timor Leste sebagai satu-satunya apotek di kabupaten Liquiça Timor Leste untuk melakukan pembenahan guna meningkatkan pelayanannya.

Mengingat selama ini banyak kendala ditemui dalam menjalankan tugasnya sebagai penyedia dan penyalur obat yang seharusnya mampu memberikan pelayanan terbaik kepada masyarakat. Kendala yang dihadapi oleh Rumah Sakit Umum Distric Liquiça Timor Leste banyak berdampak terhadap pihak internal maupun external. Laporan bulanan yang disampaikan kepada pimpinan sering terjadi kesalahan karena dalam pengolahan datanya masih manual, dan hal ini sangat tidak efektif terhadap kinerja karyawan itu merupakan kendala internal.

Kendala eksternal dihadapi saat proses transaksi penjualan obat terhadap pihak konsumen, dimana mereka harus lama mengantri untuk mendapat giliran mendapatkan pelayanan. Sistem informasi penjualan sangat penting bagi apotek Rumah Sakit Umum Distric Liquiça Timor Leste dalam proses penyajian dan pengolahan data sehingga memberikan informasi yang dibutuhkan pimpinan apotek itu sendiri. Dari latar belakang tersebut, peneliti mengambil suatu kesimpulan bagaimana memberikan yang terbaik bagi apotek tersebut, terutama dalam hal mengolah data maka peneliti merasa perlu mengadakan

penelitian dengan judul “**Desain Sistem Informasi Penjualan Obat Pada Apotek Rumah Sakit Umum Distric Liquica Timor Leste**”.

## **1.2 Rumusan Masalah**

Berdasarkan hal diatas maka timbul suatu permasalahan bagaimana mendesain suatu aplikasi Sistem Informasi Penjualan Obat untuk meningkatkan efektifitas pengolahan data dan efisiensi dalam pelayanan pada Apotek Rumah Sakit Umum Distric Liquica Timor Leste.

## **1.3 Tujuan**

Pembuatan skripsi ini bertujuan untuk Membangun Sistem Informasi Penjualan Obat Pada Apotek Rumah Sakit Umum Distric Liquica Timor Leste. Dimana aplikasi yang dibangun dapat digunakan untuk mengelola data obat, termasuk dengan transaksi barang masuk penjualan dan pembelian retail obat, stok obat , dan data konsumen pada apotek tersebut.

## **1.4 Batasan Masalah**

Pembahasan skripsi ini dibatasi oleh hal – hal sebagai berikut:

1. Aplikasi sistem informasi Apotek ini tidak terhubung dengan internet.
2. Sistem dibangun dengan menggunakan software Visual Basic.NET , sistem basis data server menggunakan Microsoft SQL Server 2000, dan Crystal Reports.
3. Pengguna aplikasi ini hanya kalangan tertentu yang memiliki hak akses terhadap program.
4. Tidak membahas data racikan obat oleh apoteker.
5. Program aplikasi ini hanya digunakan pada RSUD Distric Liquica Timor Leste.

## **1.5 METODOLOGI PENELITIAN**

### **a. Metode Pengumpulan Data**

Data merupakan sumber atau bahan mentah yang sangat berharga bagi proses menghasilkan informasi. Oleh sebab itu, dalam pengambilan data perlu dilakukan penanganan secara cermat dan hati-hati, sehingga data yang diperoleh dapat bermanfaat dan berkualitas.

Dalam pengumpulan data, penyusun menggunakan metode sebagai berikut :

#### **1. Study Lapangan**

Dengan metode ini data – data diperoleh secara langsung dari sumber yang bersangkutan, dimana penyusun berhadapan langsung dengan obyek yang diteliti, yang dilakukan dengan cara :

##### **a. Survey**

Teknik pengumpulan data dengan terjun secara langsung dan mencatat secara sistematis terhadap obyek masalah.

##### **b. Wawancara (Interview)**

Teknik pengumpulan data dengan jalan mengadakan komunikasi langsung dengan pimpinan atau pegawai perusahaan tentang sistem yang diterapkan.

#### **2. Studi Pustaka / Literature**

Pengumpulan data ini dilakukan dengan cara mencari bahan – bahan kepustakaan sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan obyek penelitian.

### **b. Metode Pengembangan Sistem**

Metode yang digunakan dalam pengembangan Sistem Informasi Penjualan Obat pada Apotek adalah :

#### **1. Rekayasa Sistem.**

Tahapan awal yang dilakukan adalah merumuskan sistem yang akan dibuat. Rekayasa sistem ini bertujuan agar programmer benar benar mengerti sistem yang akan dibangun dan langkah – langkah serta kebijaksanaan apa saja yang berkaitan dengan pembangunan sistem.



## 2. Analisa Sistem

Tahapan kedua adalah analisa yang berkaitan dengan proses dan data yang diperlukan oleh sistem serta keterkaitannya. Pemodelan yang digunakan pada analisis ini adalah dengan menggunakan *Data Flow Oriented* dengan *tool Data Flow Diagram(DFD)*.

## 3. Desain Sistem

Tahapan yang ketiga adalah desain sistem, dilakukan setelah mendapat gambaran yang jelas dari sistem yang akan dibangun. Tahapan desain sistem ini dilakukan untuk memberikan gambaran umum yang jelas kepada pengguna dan rancangan bangun yang lengkap terhadap sistem yang akan dibangun.

## 4. Implementasi Sistem

Setelah mendapat gambaran yang jelas tentang rancang bangun sistem, kemudian dilakukan implementasi rancangan sistem kedalam kode – kode dalam bahasa pemrograman. Pada tahap ini dilakukan pembuatan komponen – komponen sistem yang meliputi implementasi modul – modul program, *interface*, dan basis data.

## 5. Pengujian Sistem.

Tujuan dilakukan tahapan pengujian ini adalah untuk mendapatkan perangkat lunak yang benar – benar valid dan sesuai dengan kebutuhan yang sudah dideskripsikan. Setelah dilakukan pengujian dan sistem diyakini benar – benar memenuhi persyaratan, selanjutnya sistem tersebut didistribusikan kepada pengguna. Pada tahap ini juga dilakukan evaluasi terhadap sistem yang baru untuk melihat apakah sistem ini telah memenuhi tujuan yang ingin dicapai. Dari hasil evaluasi ini memungkinkan dilakukan perubahan – perubahan yang perlu terhadap sistem yang ada.

## **1.6 SISTEMATIKA PENULISAN**

Penulisan skripsi ini terbagi dalam lima bab dengan sistematika sebagai berikut :

### **BAB I : PENDAHULUAN**

Berisi tentang latar belakang, tujuan, permasalahan, batasan masalah, dan sistematika pembahasan dari skripsi ini.

### **BAB II : TINJAUAN PUSTAKA**

Berisi tentang penjelasan tentang teori – teori yang mendukung dalam perancangan sistem yang meliputi teori Visual Basic.NET , sistem basis data server menggunakan Microsoft SQL Server 2000.

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

Berisi tentang membangun Sistem Informasi Penjualan dan Pembelian Obat.

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

Berisi tentang implementasi serta pengujian hasil mengenai cara kerja dari sistem.

### **BAB V : PENUTUP**

Berisi tentang kesimpulan dari saran pada hasil pembahasan skripsi ini.

## BAB II LANDASAN TEORI

### 2.1 Pengertian Dasar Sistem

Secara sederhana suatu sistem dapat diartikan kumpulan dari elemen – elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Teori sistem mengatakan bahwa setiap unsur pembentuk organisasi adalah penting dan harus mendapat perhatian yang utuh.

Unsur atau komponen pembentuk organisasi di sini bukan hanya bagian – bagian yang tampak secara fisik, tetapi juga hal – hal yang mungkin bersifat abstrak atau konseptual, seperti misi, pekerjaan, kegiatan, kelompok informal dan lain sebagainya.

Komponen sistem :

#### 2.1.1 Komponen dasar

- **Input** : meliputi elemen yang *dicapture*, dirakit yang masuk ke sistem untuk diproses. Contoh : *raw material*, energi, data.
- **Proses** : meliputi proses transformasi yang mengubah input menjadi output. Contoh : proses manufaktur, kalkulasi matematika.
- **Output** : meliputi transfer elemen yang dihasilkan oleh proses ke tujuan. Contoh : sistem manufaktur menerima *raw material* sebagai input dan menghasilkan barang sebagai output. SI menerima *resource* / data sebagai input dan memprosesnya menjadi produk (informasi) sebagai output.

#### 2.1.2 Komponen tambahan

- **Feedback** : data yang menyatakan performansi sistem. Contoh : data tentang performansi penjualan adalah *feedback* bagi manager penjualan.
- **Control** : meliputi *monitoring* dan evaluasi *feedback* untuk menentukan apakah sistem mencapai tujuan. Fungsinya adalah membuat penyesuaian untuk input sistem dan pemrosesan komponen untuk memastikan sistem output yang tepat

## 2.2 Pengertian Dasar Informasi

Informasi sangat penting artinya bagi suatu sistem yang akan dibuat dalam organisasi. Informasi adalah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat.

Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang.

Kualitas informasi terkadang juga dipakai untuk menyatakan informasi yang baik. Kualitas dari informasi tergantung dari 3 (tiga) hal, yaitu :

### a. Akurat

Informasi harus bebas dari kesalahan dan tidak bias atau menyesatkan.

Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya.

### b. Tepat waktu

Informasi yang sampai pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam pengambilan keputusan. Bila pengambilan keputusan terlambat maka dapat berakibat fatal bagi organisasi. Dewasa ini informasi mahal karena harus cepat dikirim dan didapatkan sehingga memerlukan teknologi untuk mendapatkan, mengolah dan mengirimnya.

### c. Relevan

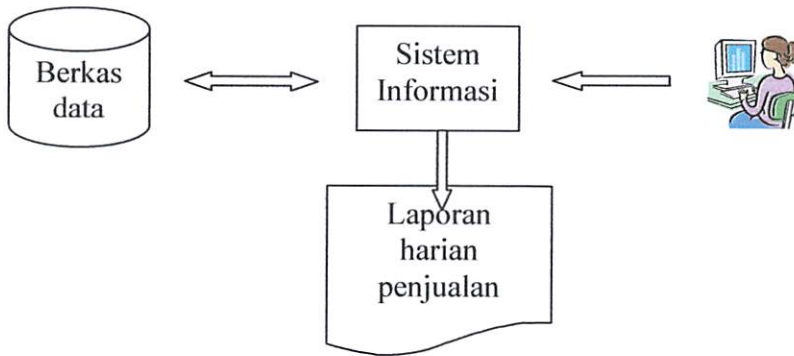
Informasi tersebut mempunyai manfaat untuk pemakaiannya. Relevansi informasi untuk setiap orang, satu dengan yang lain adalah berbeda.

## 2.3 Sistem Informasi

Dalam artian umum, sistem informasi adalah sebuah sistem terintegrasi atau sistem manusia-mesin, untuk menyediakan informasi untuk mendukung operasi, manajemen dalam suatu organisasi.

Perlu diketahui bahwa sistem informasi tidak harus selalu berbentuk kompleks. Gambar 2.1 memperlihatkan sebuah sistem informasi yang bersifat sederhana. Sistem tersebut hanya digunakan untuk mencatat transaksi penjualan dan melibatkan satu orang saja, melalui sebuah komputer, pemakai memasukkan data penjualan dan saat setelah toko

ditutup, laporan harian penjualan dicetak. Selanjutnya, laporan digunakan untuk melakukan analisis tentang barang-barang yang laku yang berguna untuk pengambilan keputusan pembelian barang.



Gambar 2.1 Sistem informasi yang sederhana

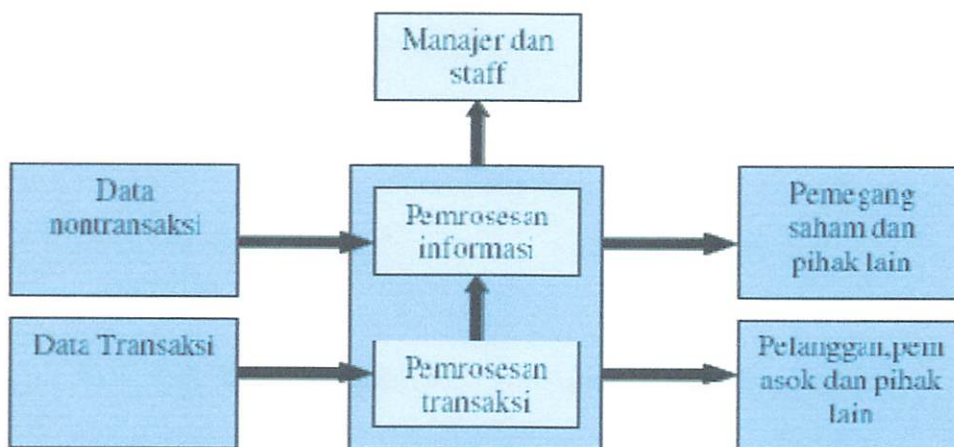
Hal-hal yang bisa dikerjakan oleh sistem informasi tentu saja terkait dengan kemampuan yang dapat dilakukannya ( Turban, Mclean, dan Wetherbe,1999) adalah sebagai berikut :

- Melaksanakan komputasi numeric, bervolume besar dan dengan kecepatan tinggi
- Menyediakan komunikasi dalam organisasi atau antar organisasi yang murah, akurat, dan cepat.
- Menyimpan informasi dalam jumlah yang besar dalam ruang yang kecil tetapi mudah diakses.
- Memungkinkan pengaksesan informasi diseluruh dunia dengan cepat dan murah.
- Meningkatkan efektifitas dan efisiensi orang-orang yang bekerja dalam kelompok dalam suatu tempat atau pada beberapa lokasi.
- Menyajikan informasi yang jelas yang menggugah pikiran manusia.
- Mengotomatisasi proses-proses bisnis yang semiotomatis dan tugas-tugas yang dikerjakan secara manual.
- Mempercepat pengetikan dan penyuntingan.
- Pembiayaan yang jauh lebih murah daripada pengerjaan secara manual.

Kemampuan-kemampuan ini mendukung sasaran bisnis yang mencakup :

- Peningkatan produktivitas.
- Pengurangan biaya.
- Peningkatan pengambilan keputusan.
- Peningkatan layanan ke pelanggan.
- Pengembangan aplikasi-aplikasi strategis lainnya.

Dengan kemampuannya yang memudahkan pengaksesan informasi, komputer menjadi sangat berguna bagi siapa saja, tidak terbatas pada manajer atau staf dalam suatu organisasi, tetapi juga bagi para pelanggan yang ikut menikmati hasilnya.

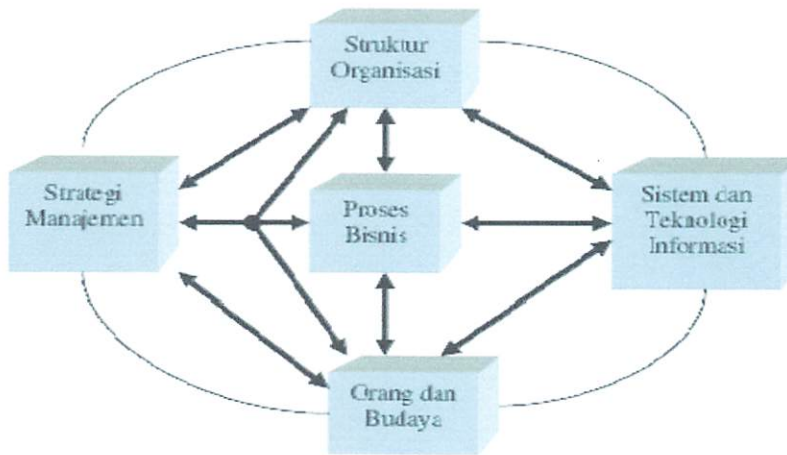


Gambar 2.2 Sistem informasi berhubungan dengan berbagai pemakai

Oleh karena itu tidaklah mengherankan jika perusahaan-perusahaan bisnis masa kini melibatkan komponen perilaku dan teknologi yang berinteraksi di dalam lingkungan sosioteknologi (O'Brien, 1996). Gambar 2.2 memperlihatkan keadaan bahwa data terbagi dua yaitu: Data nontransaksi dan data transaksi. Data non transaksi misalnya yang berhubungan dengan lokasi penelitian yaitu data penyediaan obat sedangkan data transaksi contohnya data pembelian dan penjualan.

Kedua macam data tersebut akan mengalami pemrosesan yang berbeda. Data non transaksi langsung mengalami pemrosesan informasi untuk kemudian dilanjutkan ke pemegang saham dan pihak lain yang berkepentingan serta untuk diolah oleh manajer dan staf. Sedangkan data transaksi harus mengalami pemrosesan transaksi untuk dapat diubah

menjadi informasi dan dilanjutkan kepada pelanggan, pemasok, dan pihak lain yang berkepentingan serta untuk diolah lebih lanjut oleh manajer dan staf.



Gambar 2.3 Sistem informasi dan teknologi informasi mempunyai interaksi terhadap komponen- komponen penting dalam perusahaan

Ada empat peranan penting dalam sistem informasi dalam organisasi (Alter,1992), yaitu :

1. Berpartisipasi dalam pelaksanaan tugas-tugas.
2. Mengaitkan perencanaan, pengerjaan dan pengendali dalam sebuah subsistem.
3. Mengkoordinasikan subsistem-subsistem.
4. Mengintegrasikan subsistem-subsistem.

#### 2.4 Sistem Informasi Penjualan

Sistem Informasi Penjualan adalah Sistem Informasi yang dikembangkan khusus untuk bidang usaha apotik. Aplikasi ini sudah mendukung proses integrasi aplikasi dari front sampai backend.

Aplikasi ini di develop berbasis desktop dan berjalan di sistem operasi windows, sehingga dapat memudahkan user dalam penggunaannya. Aplikasi ini juga sudah di design untuk ruang lingkup Local Area Network (LAN).

Dalam aplikasi ini antara menu dan sub menu sudah terintegrasi, sehingga setiap pelaporan dan data transaksi dapat dilihat secara langsung, detail dari menu, sub menu dan report aplikasi.

Aplikasi ini mengadministrasi kegiatan operasional dari front sampai backend antara lain memmanage data obat, termasuk dengan transaksi barang masuk, penjualan dan pembelian retail obat, stok obat, dan data konsumen pada apotek tersebut. Diharapkan dengan adanya aplikasi ini dapat membantu management dan karyawan dalam menjalankan operasional dan juga berfungsi untuk monitoring dan dapat meningkatkan layanan apotik.

## **2.5 Pengertian Database dan RDBMS**

### **2.5.1 Database**

Basis data (*database*) adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi.

Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas. Sebuah konsep database memiliki beberapa hal sebagai berikut :

- 3 Entitas : merupakan tempat informasi direkam, dapat berupa orang, tempat, kejadian dan lain-lain. Sebagai contoh dalam kasus administrasi siswa maka terdapat entity siswa, mata pelajaran, guru, pembayaran.
- 4 Atribut : disebut juga data elemen, data field, atau data item yang digunakan untuk menerangkan suatu entitas dan mempunyai harga tertentu, misalnya atribut dari entitas siswa diterangkan oleh, nama, tanggal lahir, alamat.
- 5 Data value : informasi atau data actual yang disimpan pada tiap data, elemen, atau atribut.
- 6 File/Table : kumpulan record sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda nilai datanya.
- 7 Record/Tuple : kumpulan elemen- elemen yang saling berkaitan menginformasikan tentang suatu entitas secara lengkap. Satu record mewakili satu data atau satu informasi.

Untuk mengelola basis data diperlukan perangkat lunak yang disebut DBMS. DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol, dan mengakses basis data dengan cara yang praktis dan efisien.



## 2.5.2 Relational Database dan Management Sistem

Relational Database adalah kumpulan data yang saling berelasi yang dipakai/ada dalam suatu lingkup tertentu, misalnya instansi, perusahaan dan lain-lain atau kasus tertentu. RDBMS (Relational Database Management System) merupakan koleksi atau kumpulan data yang di dalamnya memiliki suatu sistem yang mengatur relasi di dalamnya bersama dengan satu set program yang berfungsi untuk melakukan manajemen sistem terhadap data tersebut.

Selanjutnya dalam RDBMS semua data disimpan dalam tabel-tabel, dimana sebuah tabel menyimpan informasi mengenai sebuah subjek tertentu. Dengan RDBMS, sebuah database akan dengan mudah dikelola walaupun jumlah datanya banyak dan kompleks, seperti pendefinisian data, mana data yang akan dimuat dalam sebuah database, bagaimana mengelolanya, serta bagaimana membagi data. Ide RDBMS ini yaitu menggunakan konsep matematika aljabar relasional untuk membagi data dalam beberapa himpunan (set) yang saling berhubungan dalam subset. Dalam model relasional, data dipisahkan dalam beberapa set yang parallel dengan struktur tabel. Struktur tabel ini mengandung elemen data individual yang disebut kolom atau *field*. Satu set kumpulan kolom disebut

## 2.6 Microsoft SQL

### 2.6.1 SQL

SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengakses basis data yang tergolong relasional.

Sesungguhnya SQL tidak terbatas hanya untuk mengambil data (*query*), tetapi juga dapat dipakai untuk menciptakan tabel, menghapus tabel, menambahkan data ke tabel, menghapus data pada tabel, mengganti data pada tabel, dan berbagai operasi lainnya.

<b>Pernyataan</b>	<b>keterangan</b>
SELECT	Untuk mengambil data
INSERT	Untuk menambahkan data
UPDATE	Untuk mengganti data
DELETE	Untuk menghapus data
CREATE TABLE	Untuk menciptakan tabel

Tabel 2.1 Daftar sejumlah pernyataan SQL

### 2.6.2 Microsoft SQL Server 2000

MS SQL Server adalah salah satu produk Relational Database Management System (RDBMS). Fungsi utamanya adalah sebagai database server yang mengatur semua proses penyimpanan data dan transaksi suatu aplikasi.

SQL Server merupakan suatu Database Engine terkemuka walaupun sederhana namun dapat diandalkan dan juga sangat mudah digunakan, terbukti memiliki kemampuan yang baik dalam perluasan database, failover, auto generate XML, replikasi, auto alert dan messaging, full text searching, analisis service, dan sebagainya. Serta kemudahan dalam pengoperasian.

Alasan kemudahan dalam pengoperasian dan integrasi dengan MS Windows yang menjadi faktor utama dalam pemilihan SQL Server sebagai penyimpan data pada komputer. Dan juga SQL Server sangat mudah untuk dipahami baik bagi pemula untuk pengoperasiannya.

SQL Server 2000 memberikan bahasa dan antarmuka (interface) yang baik untuk pemrograman dan komunikasi pada server. Transact-SQL merupakan bahasa pemrograman server yang merupakan superset dari ANSI-SQL. ANSI-SQL mendefinisikan empat perintah dasar untuk manipulasi data yaitu : SELECT, INSERT, UPDATE, DELETE dan sejumlah perintah untuk mendefinisikan struktur database.

Salah satu alasan peneliti menggunakan SQL Server 2000 karena SQL Server 2000 kompatibel dengan beberapa data access interface yang digunakan dalam Development Tool seperti pada Visual Basic, Visual C++, Power Builder, Delphi, Visual

FoxPro dan sebagainya. Database SQL Server dapat diakses dengan menggunakan Microsoft Jet Engine dan Data Access Object (DAO), Remote Data Object (RDO), ActiveX Data Object (ADO), OLEDB, ODBC, SQL Server Built-in Library and interface dari third party lainnya.

SQL Server juga dapat digunakan pada aplikasi yang membutuhkan database stand alone (*Desktop Database System*) dan tersimpan secara local pada komputer client. SQL Server dapat mengkoordinasi sendiri secara dinamis untuk berjalan efektif dengan resource yang tersedia pada client, tanpa memerlukan administrator database sendiri. Pembuat aplikasi dapat menyertakan SQL Server sebagai komponen data storage pada aplikasi mereka.

## 2.7 Visual Basic.NET

Microsoft Visual Basic.NET 2005 merupakan bahasa pemrograman yang dibangun secara spesifik untuk *developer* pemula atau pengalaman yang ingin memperoleh kehebatan piranti lunak untuk pengembangan aplikasi. Bekal ilmu pemrograman Java dan C++ atau VB.NET sangat mendukung sekali untuk mempelajari bahasa ini dengan cepat. Visual C#.NET lebih kuat, produktif, dan stabil. VB 2005 memiliki berbagai perbedaan tampilan dibandingkan VB 2003. Berikut beberapa alasan penting lainnya untuk melakukan migrasi ke VB 2005, antara lain :

1. Visual Basic.Net 2005 mengatasi semua masalah yang sulit di sekitar pengembangan aplikasi berbasis *Windows* dan mengurangi penggunaan aplikasi lainnya serta versi komponen, bahkan mewarisi sifat C++ dan berbau Java.
2. Visual Basic.Net 2005 memiliki fasilitas penanganan bug yang hebat dan *real time background compiler* yang mengakibatkan developer visual C# dapat mengetahui kesalahan kode yang terjadi secara *up-to-date*.
3. *Windows form designer* memungkinkan developer memperoleh aplikasi dekstop dalam waktu singkat.
4. Bagi *developer*, Visual Basic.Net 2005 menyediakan model pemrograman data akses ActiveX Data Object (ADO) yang sudah dikenal dan diminati, ditambah dengan XML baru yang berbasis Microsoft ADO.NET. dengan ADO.NET,

developer akan memperoleh akses ke komponen yang lebih *powerfull*, seperti control DataSet.

5. Visual Basic.Net 2005 menghasilkan Visual Basic 2005 untuk web. Menggunakan *form* web yang baru, dapat dengan mudah membangun *thin-client* aplikasi berbasis web yang secara cerdas berjalan di *browser* dan *platform* mana pun.

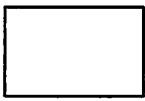
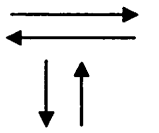
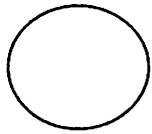
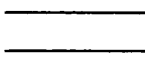
Mendukung pembangunan aplikasi *client-server*, terdistribusi, serta berupa aplikasi yang berbasiskan *Windows* serta web.

## 2.8 Perangkat Analisis dan Perancangan

### 2.8.1 Diagram Aliran Data

Diagram Aliran Data atau yang biasa disebut dengan *DFD (Data Flow Diagram)* merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh professional sistem kepada pemakai maupun pembuat program.

Ada beberapa simbol DFD yang banyak dipakai, yaitu :

No.	Simbol	Penjelasan
1		Kesatuan luar adalah lingkungan luar sistem yang dapat berupa orang, departemen, atau sistem lain yang memberikan input ataupun menerima output dari sistem.
2		Arus data adalah aliran data yang mengalir diantara proses, simpanan data dan kesatuan luar.
3		Proses adalah kerja atau kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.
4		Penyimpanan data adalah merupakan data berupa file.

Gambar 2.4 Simbol Utama DFD

### 2.8.2 Pemodelan Data

Model data adalah sekumpulan cara / peralatan / *tool* untuk mendeskripsikan data-data, hubungannya satu sama lain, semantiknya, serta batasan konsistensi.

Ada dua model data, yaitu : *Entity Relationship Diagram* (ERD) dan model relasional. Keduanya menyediakan cara untuk mendeskripsikan perancangan basis data pada peringkat logika.

Model ERD atau *Conceptual Data Model* (CDM) adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (*entity*) serta hubungan (*relationship*) antara entitas-entitas itu.

Model Relasional atau *Physical Data Model* (PDM) adalah model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom dimana setiap kolom memiliki nama yang unik.

Di dalam ERD atau CDM maupun PDM. Relasi (hubungan) setiap entitas mempunyai derajat hubungan (kardinalitas) yang menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas yang lain. Relasi kardinalitas yang terjadi di antara dua himpunan entitas dapat berupa :

- a. 1 ke 1 (*one to one*), setiap entitas pada suatu himpunan entitas berhubungan paling banyak satu entitas pada himpunan entitas yang lain, begitu sebaliknya.
- b. 1 ke N (*one to many*), setiap entitas berhubungan dengan banyak entitas pada himpunan entitas yang lain, tetapi tidak sebaliknya.
- c. N ke 1 (*many to one*), setiap entitas berhubungan dengan paling banyak satu entitas pada himpunan entitas yang lain, tetapi tidak sebaliknya.
- d. N ke N (*many to many*), setiap entitas pada suatu himpunan dapat berhubungan dengan entitas pada himpunan entitas yang lain, demikian juga sebaliknya.

## 2.9 Metode Pengembangan Sistem Waterfall

Model air terjun (*waterfall*), merupakan salah satu metode pengembangan sistem yang memiliki beberapa langkah, yaitu :

### 1. Rekayasa Sistem.

Tahapan awal yang dilakukan adalah merumuskan sistem yang akan dibuat. Rekayasa sistem ini bertujuan agar programmer benar benar mengerti sistem yang akan dibangun dan langkah – langkah serta kebijaksanaan apa saja yang berkaitan dengan pembangunan sistem.

### 2. Analisa Sistem

Tahapan kedua adalah analisa yang berkaitan dengan proses dan data yang diperlukan oleh sistem serta keterkaitannya. Pemodelan yang digunakan pada analisis ini adalah dengan menggunakan *Data Flow Oriented* dengan *tool Data Flow Diagram(DFD)*.

### 3. Desain Sistem

Tahapan yang ketiga adalah desain sistem, dilakukan setelah mendapat gambaran yang jelas dari sistem yang akan dibangun. Tahapan desain sistem ini dilakukan untuk memberikan gambaran umum yang jelas kepada pengguna dan rancangan bangun yang lengkap terhadap sistem yang akan dibangun.

### 4. Implementasi Sistem

Setelah mendapat gambaran yang jelas tentang rancang bangun sistem, kemudian dilakukan implementasi rancangan sistem kedalam kode – kode dalam bahasa pemrograman. Pada tahap ini dilakukan pembuatan komponen – komponen sistem yang meliputi implementasi modul – modul program, *interface*, dan basis data.

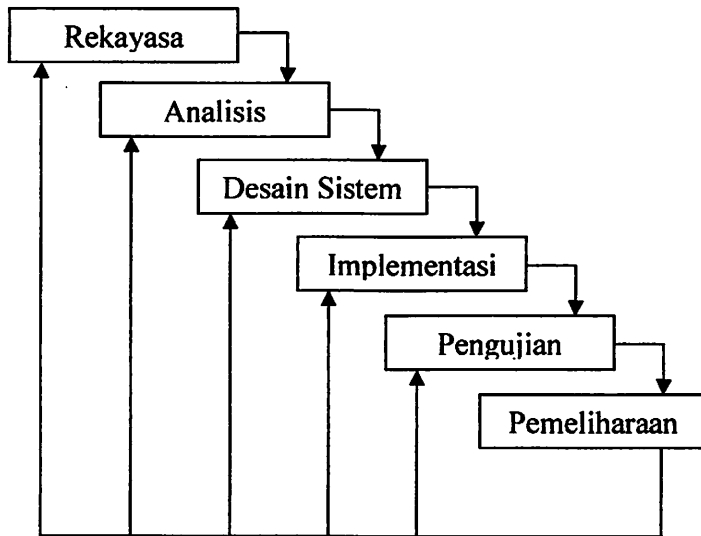
### 5. Pengujian Sistem

Tujuan dilakukan tahapan pengujian ini adalah untuk mendapatkan perangkat lunak yang benar – benar valid dan sesuai dengan kebutuhan yang sudah dideskripsikan.

### 6. Pemeliharaan Sistem

Setelah dilakukan pengujian dan sistem diyakini benar – benar memenuhi persyaratan, selanjutnya sistem tersebut didistribusikan kepada pengguna. Pada tahap ini juga dilakukan evaluasi terhadap sistem yang baru untuk melihat apakah sistem ini telah memenuhi tujuan yang ingin dicapai. Dari hasil evaluasi ini

memungkinkan dilakukan perubahan – perubahan yang perlu terhadap sistem yang ada.



Gambar 2.5 Model Air Terjun (*Waterfall*)

## **BAB III**

### **ANALISA DAN PERANCANGAN SISTEM**

Pada bab ini dijelaskan mengenai analisis dan perancangan sistem aplikasi. Analisis ditujukan untuk memberikan gambaran secara umum terhadap aplikasi. Hal ini berguna untuk menunjang perancangan aplikasi yang akan dikembangkan sehingga kebutuhan akan aplikasi tersebut dapat diketahui sebelumnya. Kemudian hasil analisis akan menjadi dasar untuk melakukan perancangan atau desain aplikasi sesuai kebutuhan sistem.

Dalam merancang aplikasi pada proyek akhir ini terlebih dahulu dilakukan pembuatan desain proses, desain data, serta desain antar muka aplikasi. Desain proses berguna untuk mengintegrasikan semua proses yang terjadi dalam aplikasi yang akan dibuat. Desain data berguna untuk mengetahui data apa saja yang dibutuhkan dalam proses yang akan dikerjakan. Sedangkan perancangan antarmuka berfungsi sebagai antar muka interaksi antara pengguna dengan sistem aplikasi yang dibuat, sehingga pengguna dapat mengoperasikan aplikasi yang dibuat.

#### **3.1. Sistem Informasi Apotek**

Sistem Informasi Apotek pada RSUD Liquiça Timor Leste menghendaki pengelolaan data tentang pelayanan transaksi penjualan obat yang efektif dan efisien. Efektif dalam arti tingkat keberhasilan penanganan terhadap konsumen yang membutuhkan obat cukup tinggi dan efisien berarti optimal dalam penggunaan sumber daya yang ada. Data yang diolah pada aplikasi ini mengambil masukan (*input*) data dari beberapa sumber yaitu data supplier, data obat, dan data konsumen yang bekerja sama yang diproses oleh seorang petugas sehingga menjadi informasi yang berguna bagi apotek.

Adapun pengertian Sistem Informasi Apotek adalah suatu sistem berbasis komputer yang dikembangkan khusus untuk bidang apotek. Hasil informasi berupa laporan dan digunakan oleh pengguna dalam mengambil keputusan atau peningkatan upaya pelayanan.

##### **3.1.1. Pengguna Sistem**

Pengguna sistem dibedakan berdasarkan *role ID* atau hak akses yang telah diberikan oleh admin, hal ini berfungsi agar dapat menjaga keutuhan data pada sistem informasi apotek ini. Pengguna dari sistem informasi kepegawaian ini adalah :



- a) Konsumen, yaitu pembeli yang memiliki hak akses untuk membeli obat pada Apotek tersebut.
- b) Apoteker, yaitu Tempat penyimpanan obat yang digunakan untuk menjual obat pada konsumen yang memiliki hak akses untuk membeli obat pada Apotek tersebut.
- c) Supplier, yaitu penyuplai obat pada apotek yang kehabisan stok.
- d) Logistik & inventaris, yaitu memiliki hak akses mengecek obat yang udah kadaluarsa, stock obat, dan penataan obat pada sistem informasi apotek.
- e) Plano e Financas, yaitu pengguna yang memiliki hak akses untuk data pembayaran obat.
- f) Director, yaitu pengguna yang memiliki hak akses untuk menerima laporan - laporan.

### 3.1.2. Spesifikasi Sistem

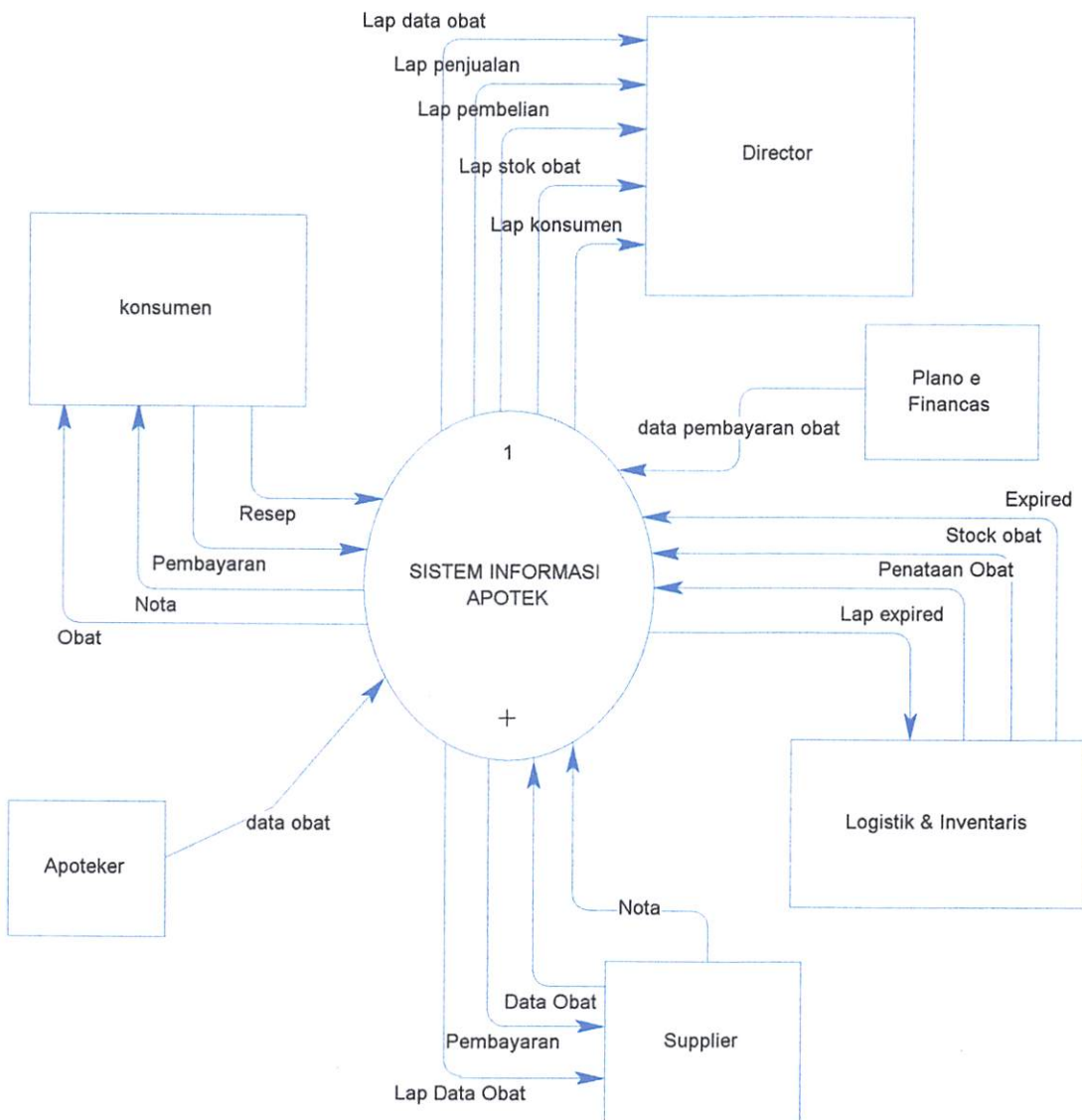
Sistem Informasi sesuai dengan tujuan dari pengembangan sistem informasi apotek, maka spesifikasi dari sistem informasi apotek adalah sebagai berikut :

- a) Aplikasi ini di design berbasis desktop dan berjalan di sistem operasi windows, sehingga dapat memudahkan user dalam penggunaannya. Aplikasi ini juga dapat digunakan dalam ruang lingkup Local Area Network (LAN).
- b) Sistem basis data *server* menggunakan Microsoft SQL Server 2000.
- c) Aplikasi ini nantinya akan berjalan di lingkungan Apotek RSUD Liquica Timor Leste, dengan pembatasan akses sesuai dengan hak akses yang telah ditentukan.
- d) Sistem informasi ini diharapkan dapat mengelola data yang berhubungan dengan transaksi obat – obatan dalam lingkungan RSUD Liquica Timor Leste.
- e) Modul laporan serta rekapan hasil dari proses pengolahan data yang dibutuhkan sebagai arsip data dari sistem informasi apotek itu sendiri.

## 3.2. Perancangan Sistem

### 3.2.1. Data Flow Diagram (DFD)

DFD atau biasa disebut juga Diagram Alir Data (DAD), merupakan salah satu tahap dari perancangan sistem yang menggambarkan proses dari aliran data secara keseluruhan dari sistem. Diagram tersebut memperlihatkan dari mana data tersebut dimasukkan dan apa saja yang dihasilkan dari setiap proses. Hal tersebut dibutuhkan untuk melihat detail dari setiap proses pada sistem tersebut.



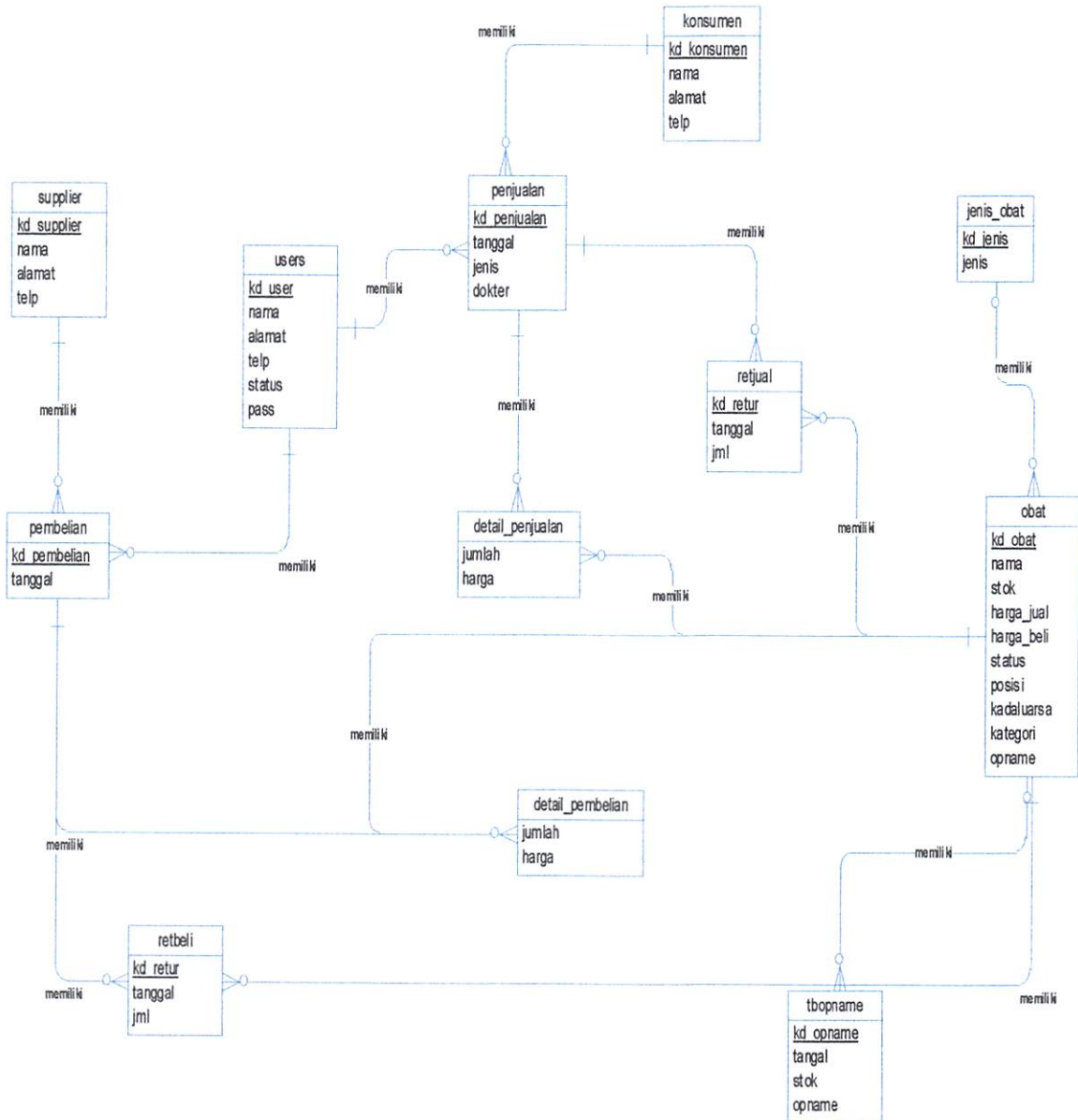
**Gambar 3.1.** Context Diagram Sistem Informasi Apotek



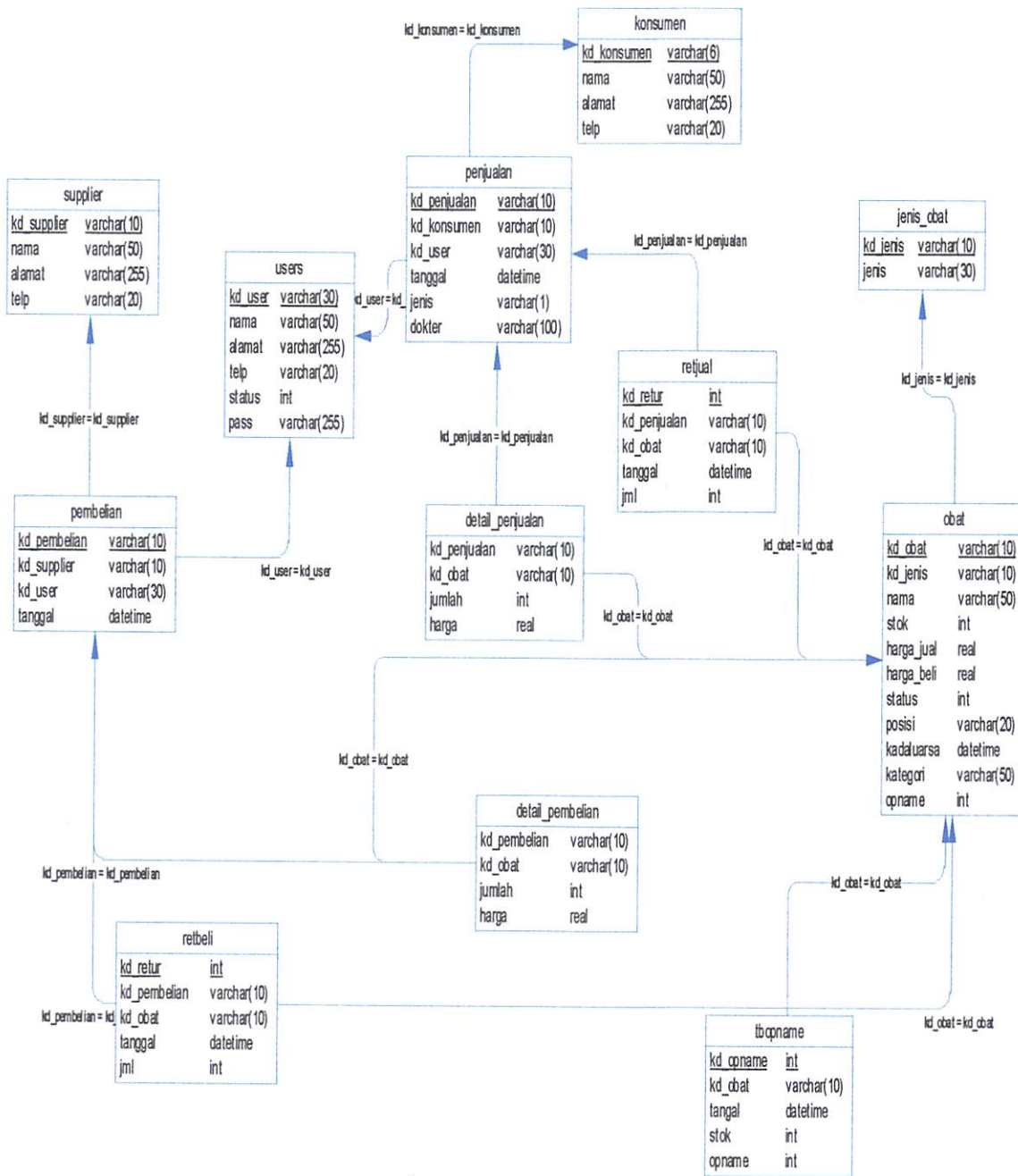
### 3.2.3. Desain Data Base

#### 3.2.3.1. Relasi Antar Tabel

Relasi antar tabel pada sistem informasi apotek ini digambarkan dalam bentuk konsep atau *Conceptual Data Model (CDM)* dan fisik atau *Physical Data Model (PDM)* seperti ditunjukkan pada gambar dibawah ini.



**Gambar 3.3.** CDM Basis Data Sistem Informasi Apotek



Gambar 3.4. PDM Basis Data Sistem Informasi Apotek

### 3.2.3.2. Struktur Tabel yang Digunakan

Berdasarkan *Conceptual Data Model (CDM)* diatas, maka tabel – tabel yang digunakan antara lain :

#### 1. Tabel T\_SUPPLIER

Tabel ini berfungsi untuk menyimpan data supplier yang memasok obat ke apotek.

Nama Tabel : Tabel T\_SUPPLIER

Primary Key : Kd\_Supplier

No	Nama Kolom	Type Data	Keterangan
1	Kd_Supplier	Varchar(10)	<i>Primary Key (PK)</i> , Kode supplier
2	Nama	Varchar(50)	Nama supplier
3	Alamat	Varchar(255)	Alamat supplier
4	Telp	Varchar(20)	No Telepon Supplier

Tabel 3.1 Tabel T\_Supplier

#### 2. Tabel T\_USERS

Tabel ini berfungsi untuk menuser data obat ke apotek.

Nama Tabel : Tabel T\_USERS

Primary Key : Kd\_User

No	Nama Kolom	Type Data	Keterangan
1	Kd_User	Varchar(30)	<i>Primary Key (PK)</i> , Kode User
2	Nama	Varchar(50)	Nama User
3	Alamat	Varchar(255)	Alamat User
4	Telp	Varchar(20)	No Telepon User
5	Status	Int	Status User
6	Pass	Varchar(255)	Pass User

Tabel 3.2 Tabel T\_Users

### 3. Tabel T\_PENJUALAN

Tabel ini menyimpan informasi penjualan obat yang terjadi dalam apotek.

Nama Tabel : Tabel T\_PENJUALAN

Primary Key : Kd\_Penjualan

No	Nama Kolom	Type Data	Keterangan
1	Kd_Penjualan	Varchar(10)	<i>Primary Key (PK)</i> , Kode Penjualan
2	Kd_Konsumen	Varchar(10)	Kode Konsumen
3	Kd_User	Varchar(30)	Kode User
4	Tanggal	Datetime	Tanggal Penjualan
5	Jenis	Varchar(1)	Jenis Penjualan
6	Dokter	Varchar(100)	Nama Dokter

Tabel 3.3 Tabel T\_Penjualan

### 4. Tabel T\_KONSUMEN

Tabel ini berfungsi untuk menyimpan data konsumen yang melakukan transaksi dengan apotek.

Nama Tabel : Tabel T\_KONSUMEN

Primary Key : Kd\_Konsumen

No	Nama Kolom	Type Data	Keterangan
1	Kd_Konsumen	Varchar(6)	<i>Primary Key (PK)</i> , Kode Konsumen
2	Nama	Varchar(50)	Nama Konsumen
3	Alamat	Varchar(255)	Alamat Konsumen
4	Telp	Varchar(20)	No Telepon Konsumen

Tabel 3.4 Tabel T\_Konsumen

## 5. Tabel T\_PEMBELIAN

Tabel ini berfungsi untuk menyimpan informasi pembelian obat yang dilakukan dari supplier.

Nama Tabel : Tabel T\_PEMBELIAN

Primary Key : Kd\_Pembelian

No	Nama Kolom	Type Data	Keterangan
1	Kd_Pembelian	Varchar(10)	<i>Primary Key (PK)</i> , Kode Pembelian
2	Kd_Supplier	Varchar(10)	Kode Supplier
3	Kd_User	Varchar(30)	Kode User
4	Tanggal	Datetime	Tanggal pembelian

Tabel 3.5 Tabel T\_Pembelian

## 6. Tabel T\_DETAIL\_PENJUALAN

Tabel ini menyimpan informasi detail penjualan obat yang terjadi dalam apotek

Nama Tabel : Tabel T\_DETAIL\_PENJUALAN

Primary Key : Kd\_Penjualan

No	Nama Kolom	Type Data	Keterangan
1	Kd_Penjualan	Varchar(10)	<i>Primary Key (PK)</i> , Kode Penjualan
2	Kd_Obat	Varchar(10)	Kode Obat
3	Jumlah	Int	Jumlah Penjualan
4	Harga	Real	Harga Penjualan

Tabel 3.6 Tabel T\_Detail\_Penjualan



## 7. Tabel T\_RETJUAL

Tabel ini menyimpan informasi retur obat yang terjadi dalam apotek.

Nama Tabel : Tabel T\_RETJUAL

Primary Key : Kd\_Retur

No	Nama Kolom	Type Data	Keterangan
1	Kd_Retur	Int	<i>Primary Key (PK)</i> , Kode Retur
2	Kd_Penjualan	Varchar(10)	Kode Penjualan
3	Kd_Obat	Varchar(10)	Kode Obat
4	Tanggal	Datetime	Tanggal Retur
5	Jumlah	Int	Jumlah Retur

Tabel 3.7 Tabel T\_Retjual

## 8. Tabel T\_JENIS OBAT

Tabel ini menyimpan informasi jenis obat yang terjadi dalam apotek.

Nama Tabel : Tabel T\_JENIS OBAT

Primary Key : Kd\_Jenis

No	Nama Kolom	Type Data	Keterangan
1	Kd_Jenis	Varchar(10)	<i>Primary Key (PK)</i> , Kode Jenis
2	Jenis	Varchar(30)	Jenis Obat

Tabel 3.8 Tabel T\_Jenis obat

## 9. Tabel T\_RETBELI

Tabel ini menyimpan informasi retbeli obat yang terjadi dalam apotek.

Nama Tabel : Tabel T\_RETBELI

Primary Key : Kd\_Retur

No	Nama Kolom	Type Data	Keterangan
1	Kd_Retur	Int	<i>Primary Key (PK)</i> , Kode Retur
2	Kd_Pembelian	Varchar(10)	Kode Pembelian
3	Kd_Obat	Varchar(10)	Kode Obat
4	Tanggal	Datetime	Tanggal Retbeli
5	Jumlah	Int	Jumlah Retbeli

Tabel 3.9 Tabel T\_Retbeli

## 10. Tabel T\_DETAIL\_PEMBELIAN

Tabel ini menyimpan informasi pembelian detail obat yang terjadi dalam apotek.

Nama Tabel : Tabel T\_DETAIL\_PEMBELIAN

Primary Key : Kd\_Pembelian

No	Nama Kolom	Type Data	Keterangan
1	Kd_Pembelian	Varchar(10)	<i>Primary Key (PK)</i> , Kode Pembelian
2	Kd_Obat	Varchar(10)	Kode Obat
3	Jumlah	Int	Jumlah Detail Pembelian
4	Harga	Real	Harga Detail Pembelian

Tabel 3.10 Tabel T\_Detail\_Pembelian

## 11. Tabel T\_OBAT

Tabel ini digunakan untuk menyimpan data-data yang berhubungan dengan obat.

Nama Tabel : Tabel T\_OBAT

Primary Key : Kd\_Obat

No	Nama Kolom	Type Data	Keterangan
1	Kd_Obat	Varchar(10)	<i>Primary Key (PK)</i> , Kode Obat
2	Kd_Jenis	Varchar(10)	Kode Jenis Obat
3	Nama	Varchar(50)	Nama Obat
4	Stok	Int	Stok Obat
5	Harga_Jual	Real	Harga Jual Obat
6	Harga_Beli	Real	Harga Beli Obat
7	Status	Int	Status Obat
8	Posisi	Varchar(10)	Posisi Obat
9	Kadaluarsa	Datetime	Kadaluarsa Obat
10	Kategori	Varchar(50)	Kategori Obat
11	Opname	Int	Opname Obat

Tabel 3.11 Tabel T\_Obat

## 12. Tabel T\_TBOPNAME

Tabel ini digunakan untuk menyimpan data-data tbopname yang berhubungan dengan obat.

Nama Tabel : Tabel T\_TBOPNAME

Primary Key : Kd\_Opname

No	Nama Kolom	Type Data	Keterangan
1	Kd_Opname	Int	<i>Primary Key (PK)</i> , Kode Opname
2	Kd_Obat	Varchar(10)	Kode Obat
3	Tanggal	Datetime	Tanggal Opname
4	Stok	Int	Stok Opname
5	Opname	Int	Opname

Tabel 3.12 Tabel T\_Tbopname

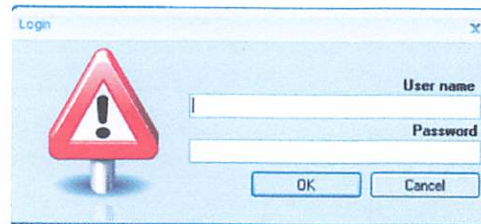
### 3.2.4. Desain Antarmuka Aplikasi

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

Ada empat macam desain antarmuka pada aplikasi ini baik halaman untuk admin maupun halaman sistem informasi itu sendiri, yaitu desain form login, form utama, form entri data, dan form laporan. Sedangkan desain untuk menu ditampilkan berdasarkan hak akses setiap pengguna sistem.

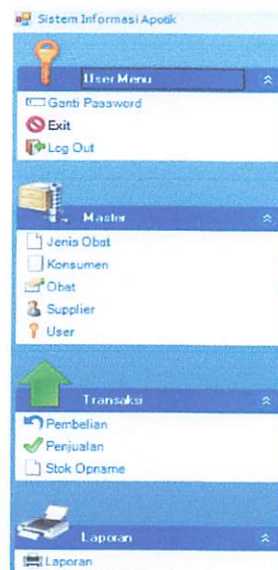
#### 3.2.4.1. Desain Form Login

Desain halaman *login* merupakan tampilan antar muka saat *user* pertama kali mengakses sistem informasi apotik ini. Skema tampilan halaman *login* dapat dilihat pada gambar dibawah ini.



Gambar 3.5. Desain Form Login

#### 3.2.4.2 Desain Halaman Internal

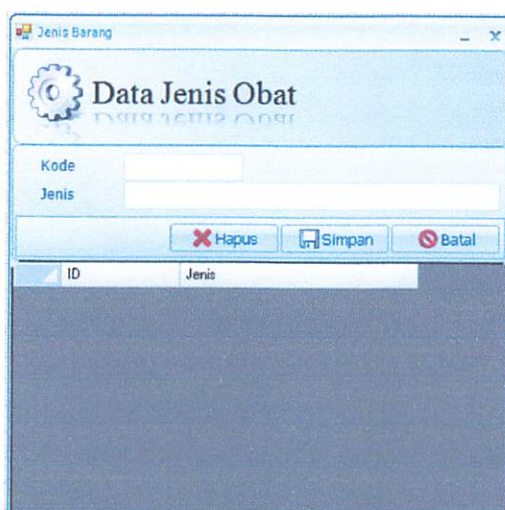


Gambar 3.6. Desain Form Menu Utama

Desain form pada gambar diatas merupakan skema desain halaman utama beserta menu aplikasinya.

#### 3.2.4.3. Desain Halaman Entry Data Jenis Obat

Desain halaman entry data jenis obat merupakan halaman yang ditujukan untuk proses pengentrian data ke dalam database. Proses pengentrian data umumnya hampir semua sama untuk menu aplikasi, hanya dibedakan dari fungsi masing-masing data yang diinputkan. Berikut adalah contoh skema penginputan data jenis obat.



The image shows a screenshot of a web application window titled "Jenis Barang". The main heading is "Data Jenis Obat". Below the heading, there are two input fields: "Kode" and "Jenis". To the right of these fields are three buttons: "Hapus" (with a red X icon), "Simpan" (with a floppy disk icon), and "Batal" (with a red circle and slash icon). Below the buttons is a table with two columns: "ID" and "Jenis". The table is currently empty.

Gambar 3.7. Desain Form Data Jenis Obat

#### 3.2.4.4 Desain Halaman Entry Konsumen

Desain halaman entry data konsumen merupakan halaman yang ditujukan untuk proses pengentrian data ke dalam database. Proses pengentrian data umumnya hampir semua sama untuk menu aplikasi, hanya dibedakan dari fungsi masing-masing data yang diinputkan. Berikut adalah contoh skema penginputan data konsumen.

**Gambar 3.8.** Desain Form Data Konsumen

#### 3.2.4.5 Desain Halaman Entry Data Obat

Desain halaman entry data obat merupakan halaman yang ditujukan untuk proses pengentrian data ke dalam database. Proses pengentrian data umumnya hampir semua sama untuk menu aplikasi, hanya dibedakan dari fungsi masing-masing data yang diinputkan. Berikut adalah contoh skema penginputan data obat.

**Gambar 3.9.** Desain Form Data Konsumen

#### 3.2.4.6 Desain Halaman Entry Data Supplier

Desain halaman entry data supplier merupakan halaman yang ditujukan untuk proses pengentrian data ke dalam database. Proses pengentrian data umumnya hampir

semua sama untuk menu aplikasi, hanya dibedakan dari fungsi masing-masing data yang diinputkan. Berikut adalah contoh skema penginputan data supplier.

**Gambar 3.10.** Desain Form Data Konsumen

#### 3.2.4.7 Desain Halaman Entry Data User

Desain halaman entry data user merupakan halaman yang ditujukan untuk proses pengentrian data ke dalam database. Proses pengentrian data umumnya hampir semua sama untuk menu aplikasi, hanya dibedakan dari fungsi masing-masing data yang diinputkan. Berikut adalah contoh skema penginputan data user.

**Gambar 3.11.** Desain Form Data User



### 3.2.4.8 Desain Halaman Entry Data Pembelian

Desain halaman entry data pembelian merupakan halaman yang ditujukan untuk proses pengentrian data ke dalam database. Proses pengentrian data umumnya hampir semua sama untuk menu aplikasi, hanya dibedakan dari fungsi masing-masing data yang diinputkan. Berikut adalah contoh skema penginputan data pembelian.

**Gambar 3.12.** Desain Form Data Pembelian

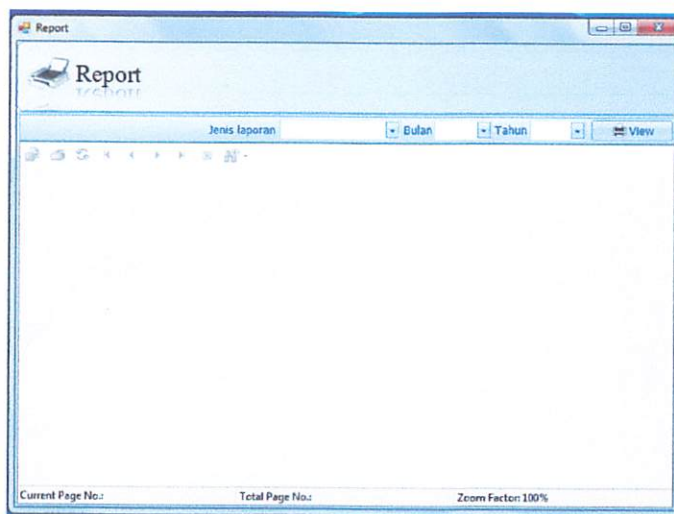
### 3.2.4.9 Desain Halaman Entry Data Penjualan

Desain halaman entry data penjualan merupakan halaman yang ditujukan untuk proses pengentrian data ke dalam database. Proses pengentrian data umumnya hampir semua sama untuk menu aplikasi, hanya dibedakan dari fungsi masing-masing data yang diinputkan. Berikut adalah contoh skema penginputan data penjualan.

**Gambar 3.13.** Desain Form Data Penjualan

### 3.2.4.10 Desain Form Laporan

Desain halaman laporan diperlukan sebagai proses *output* dari segala informasi yang telah dikelola di sistem informasi ini. Berbagai macam laporan bisa dibuat sesuai kebutuhan yang diperlukan. Berikut adalah skema desain halaman laporan.



**Gambar 3.14.** Desain Form Laporan

Untuk tampilan laporan pada form laporan, dibuat dengan menggunakan Crystal Reports. Berikut ini desain tampilan laporan pada Crystal Reports.

Sistem pelaporan yang digunakan untuk sistem informasi apotek terdiri dari beberapa laporan, antara lain :

#### 1. Laporan Data Konsumen

ID	Nama	Alamat	Telp
C00000	Tomas	Jl. Jombang Malang	(031) 555-7876
C00001	Yono	Malang	(0397) 333-3832
C00002	Iqo	Jl. Mentan	(0851) 335-1419
C00003	Dm	Papua	(0251) 888-8871

**Gambar 3.15.** Desain Laporan Data Konsumen pada Crystal Report

Laporan konsumen berisi data-data konsumen yang melakukan transaksi dengan apotik.

## 2. Laporan Data Supplier

Report

Jenis laporan: Supplier    Bulan:    Tahun:    View

Main Report

Laporan Data Supplier

2/12/2011

ID	Nama	Alamat	Telp
S00001	Kalbe Farma	Jl surabaya 50	(081) 105-0888
S00002	Bode	Jakarta	(221) 555-0989
S00003	Hulusa	Jogja	(081) 333-1419
S00004	Zoon	Bandung	(071) 7375911

Current Page No.:1    Total Page No.:1    Zoom Factor:100%

**Gambar 3.16.** Desain Laporan Data Supplier pada Crystal Report

Laporan Supplier akan berisi data supplier yang melakukan suplai obat kepada apotik

## 3. Laporan Data Obat

Report

Jenis laporan: Obat    Bulan:    Tahun:    View

Main Report

Laporan Data Obat

2/12/2011

Kode	Nama	Jenis	Stok	Harga
CE0000000	Obat Panu	Kapsul	90	6.000,00
CE0000001	Ecodriks	Jamu	150	6.000,00
CE0000002	Panadol	Kapsul	90	90.000,00
CE0000003	dlopril	Tablet	60	5.000,00
CE0000004	parasetamol	Kapsul	80	6.000,00
CE0000005	ibu	Syalak	80	9.000,00
CE0000006	Konidin	Kapsul	100	7.000,00

Current Page No.:1    Total Page No.:1    Zoom Factor:100%

**Gambar 3.17.** Desain Laporan Data Obat pada Crystal Report

Laporan data obat ini akan berisi data tentang obat yang di beli dari supplier beserta harga jual obat tersebut yang akan dijual kepada konsumen.

#### 4. Laporan Pembelian

Nota	Supplier	Tgl
0000000004	Kalbe Farma	02/05/2011
	Obat	Jual
	Obat Panu	50
	Total	8.000,00
0000000005	Kalbe Farma	02/05/2011
	Obat	Jual
	Obat	11
	Panadol	1.111
	Total	134.444,00
0000000006	hatuasa	02/05/2011
	Obat	Jual
	Panadol	1
	Obat	111
	Total	111.112,00
0000000007	hatuasa	02/05/2011
	Obat	Jual
	Panadol	111
	Botolnya	12.121
	Total	154.444,00
	Total	405.000,00

Gambar 3.18. Desain Laporan Pembelian pada Crystal Report

Berisi data-data pembelian yang dilakukan apotik untuk membeli obat dari supplier.

#### 5. Laporan Penjualan

Nota	Konsumen	Tgl
0000000003	Tomasa	02/05/2011
	Obat	Jual
	Panadol	1.111
	Obat	111
	Obat Panu	50
	Obat	40
	Botolnya	1.000
	Total	116.000,00

Gambar 3.19. Desain Laporan Penjualan pada Crystal Report

Berisi data-data pembelian yang dilakukan oleh konsumen pada apotik tersebut.

## BAB IV

### IMPLEMENTASI DAN PENGUJIAN SISTEM

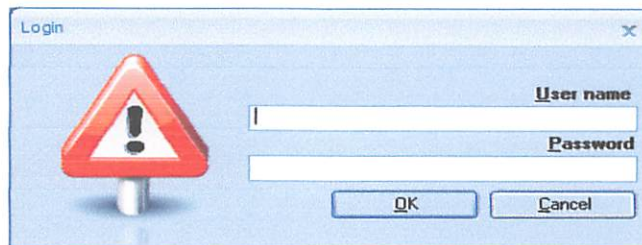
#### 4.1. Implementasi Sistem

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

Dalam aplikasi sistem informasi apotek ini, program yang digunakan dalam pengimplementasian adalah menggunakan Visual Basic.NET 2005. Sebagai media penyimpanan data, aplikasi ini menggunakan basis data Microsoft SQL Server 2000, yang didalamnya terdapat beberapa tabel yang saling berhubungan.

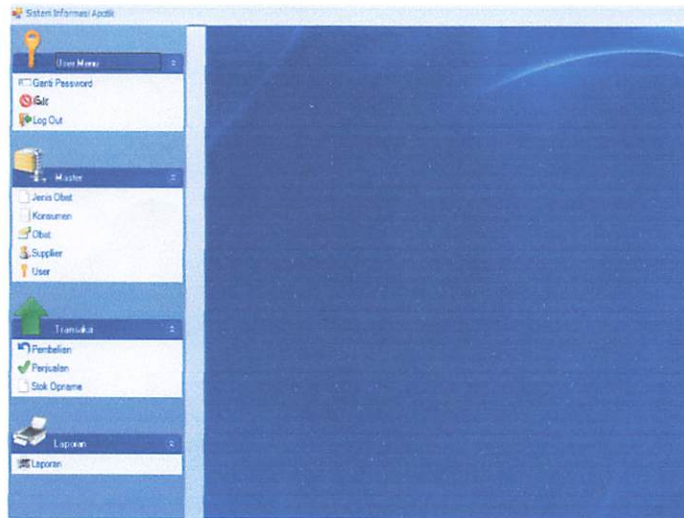
Hasil implementasi dari sistem informasi apotek mencakup data supplier, data obat, data konsumen, data pembelian, data penjualan, data pembayaran obat dan laporan – laporan yang diperlukan. Tampilan form menu utama sistem informasi apotek seperti berikut :

Pada saat aplikasi pertama dijalankan, akan muncul form login yang menentukan hak akses atau wewenang terhadap menu aplikasi yang dapat dijalankan oleh user tentunya setelah memasukan user dan password yang dibuat dan disetujui oleh administrator. Bentuk form login dapat dilihat pada gambar 4.1



Gambar 4.1. Form Login

Pada saat aplikasi pertama dijalankan, form menu utama akan menjeleskan bagian-bagian implementasi sistem yang terbagi menjadi beberapa form aplikasi yaitu : User menu terdiri dari; ganti password, exit, logout. Master terdiri dari; jenis obat, konsumen, obat, supplier dan user. Transaksi terdiri dari; penjualan dan pembelian, dan sub menu laporan berisi semua laporan.



**Gambar 4.2.** Menu Utama

#### 4.1.1. Menu Pendataan

Pada menu ini terdapat, sub menu ganti password, sub menu data jenis obat, sub menu konsumen, sub menu supplier, dan sup menu data user.

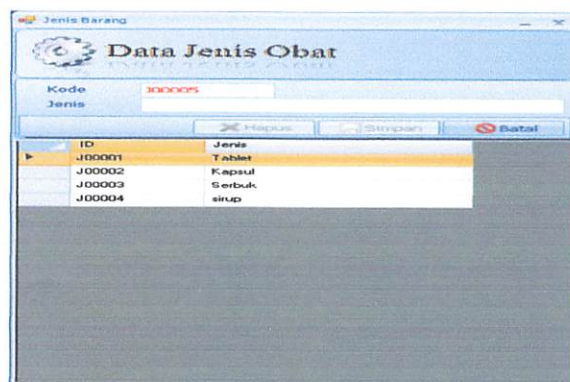
##### 4.1.1.1. Sub Menu Ganti password

Sub menu ini berfungsi untuk mengganti password lama dan memasukkan password baru. Adapun tampilannya seperti pada gambar dibawah ini.

**Gambar 4.3.** Menu Ganti Password

#### 4.1.1.2. Sub Menu Data Jenis Obat

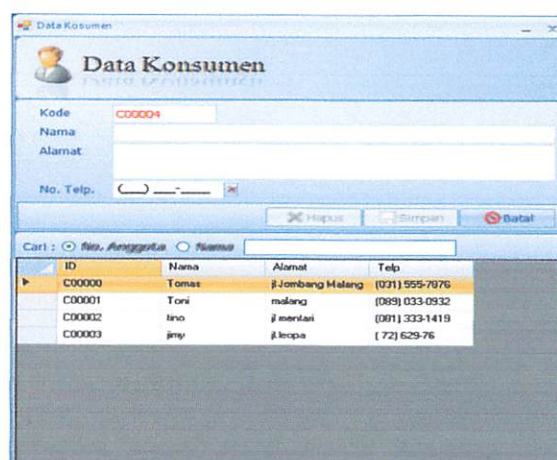
Sub menu ini berfungsi sebagai inputan data jenis obat yang akan di jual pada apotek ini. Adapun tampilannya seperti pada gambar dibawah ini.



Gambar 4.4 Entry data jenis obat

#### 4.1.1.3. Sub Menu Konsumen

Sub menu ini berfungsi sebagai inputan data konsumen yang melakukan transaksi pada apotek ini. Adapun tampilannya seperti pada gambar dibawah.



Gambar 4.5 Entry data konsumen

#### 4.1.1.4. Sub Menu Data Obat

Sub menu ini berfungsi sebagai inputan data-data obat yang akan di jual pada apotek ini. Adapun tampilannya seperti pada gambar dibawah ini.

The screenshot shows the 'Data Obat' application window. It features a header with the application name and logo. Below the header is a form with fields for 'Kode' (containing '080000005'), 'Jenis Obat' (a dropdown menu), 'Nama', 'Stok', and 'Harga'. There are buttons for 'Hapus', 'Simpan', and 'Batal'. Below the form is a search bar with radio buttons for 'No. Obat' and 'Nama'. A table displays the following data:

ID	Nama	Stok	Harga
080000001	Sakatonk	40	6000
080000002	Neuralgin	30	5000
080000003	Klorokin	20	10000
080000004	Heocin	50	12000

Gambar 4.6 Entry data obat

#### 4.1.1.5. Sub Menu Supplier

Sub menu ini berfungsi sebagai inputan data-data supplier yang mensuplai obat ke apotek ini. Adapun tampilannya seperti pada gambar dibawah ini.

The screenshot shows the 'Data Supplier' application window. It features a header with the application name and logo. Below the header is a form with fields for 'Kode' (containing 'S00005'), 'Nama', 'Alamat', and 'No. Telp.'. There are buttons for 'Hapus', 'Simpan', and 'Batal'. Below the form is a search bar with radio buttons for 'No. Anggota' and 'Nama'. A table displays the following data:

ID	Nama	Alamat	Telp
S00001	Kalbe Farma	Jl surabaya 90	(081) 099-0888
S00002	Bode	Jakarta	(221) 555-0989
S00003	heluae	Jl melati	(08) 133-3141
S00004	ry	Jl catur	(0) 812-2233

Gambar 4.7 Entry data supplier



#### 4.1.1.6. Sub Menu Data User

Sub menu ini berfungsi sebagai inputan data-data user pada apotek ini. Adapun tampilannya seperti pada gambar dibawah ini.

The 'Data User' window contains a form with the following fields: ID, Jenis Account (dropdown), Nama, Alamat, No. Telp. (with a country code dropdown), and Password. Below the form are buttons for 'Hapus', 'Simpan', and 'Batal'. A table below the form lists user data:

ID	Nama	Alamat	Telp
102	Jeka	Jl.batu	[ 0] 341-5609
111	Tety	jl.bunga	(03) 415-1960
123	Maiza	jl.kembang	(11) 034-7898
222	Tino	Jl.Saturnus	(72) 564-5011

Gambar 4.8 Entry data user

#### 4.1.1.7. Sub Menu Data Opname Obat

Sub menu ini berfungsi untuk memperbaruhi stock opname obat pada apotek ini. Adapun tampilannya seperti gambar di bawah ini.

The 'Data Opname Obat' window displays a table of drug stock data and a form for updating it. The table below the form shows the current stock levels:

ID	Nama	Stok	Harga	Stok Opname
CE00000001	Obat Panu	90	6000	50
CE00000002	Boatreks	150	6000	0
CE00000003	Paradol	90	90000	70
CE00000004	clapret	60	5000	0
CE00000005	paracetamol	200	3000	0

The form below the table has a 'Jml.' label and a 'Stok Opname' input field with the value '50'. It also includes 'Simpan' and 'Batal' buttons.

Gambar 4.9 Entry data opname obat

## 4.1.2. Menu Transaksi

Dalam menu ini terdapat beberapa sub menu antara lain pembelian obat, pembelian umum, penjualan obat dan penjualan umum.

### 4.1.2.1.Sub Menu Transaksi Pembelian Obat

Sub menu ini berfungsi sebagai inputan data transaksi pembelian obat dari supplier. Adapun tampilannya seperti pada gambar dibawah ini.

The screenshot shows a software window titled 'FormPembelian' with a sub-header 'Data Pembelian'. The interface includes several input fields and a data table. The 'Kode' field contains '000000007'. Below it are 'ID Supplier' and 'ID Obat' fields, each with a 'Baru' button. A table lists drug items with columns for 'ID', 'Nama', 'Jumlah', 'Harga', and 'Jml.'. The table contains four rows: Sakatonik, Neuralgin, klorokin, and Herocin. At the bottom, there are buttons for 'Cetak Nota', 'Hapus', 'Simpan', and 'Batal'.

No.	Kode	Nama.	Jml.	Harga.
	0800000001	Sakatonik		
	0800000002	Neuralgin		
	0800000003	klorokin		
	0800000004	Herocin		

Gambar 4.10 Transaksi Pembelian Obat

### 4.1.2.2.Sub Menu Transaksi Pembelian umum

Sub menu ini berfungsi sebagai inputan data transaksi pembelian obat tanpa menggunakan resep yang dilakukan konsumen pada apotek ini. Adapun tampilannya seperti pada gambar dibawah ini.

Form Pembelian

**Data Pembelian**

Kode: 000000007

ID Supplier: S00001

- Nama : Kalbe Farma  
- Alamat : Jl Surabaya 90  
- No. Telp : (081) 099-0888

ID Obat:

ID Obat	Nama
080000001	Sakatonik
080000002	Neuragin
080000003	klorokin
080000004	Herochin

Jumlah: 0    Harga: 0,00   

No.	Kode	Nama.	Jml.	Harga.
1	080000002	Neuragin	50	3000
2	080000001	Sakatonik	50	9000
3	080000003	klorokin	70	100000

**Gambar 4.11 Transaksi Pembelian Umum**

#### 4.1.2.3. Sub Menu Transaksi Penjualan Obat

Sub menu ini berfungsi sebagai inputan data transaksi penjualan obat dari supplier. Adapun tampilannya seperti pada gambar dibawah ini.

Data Penjualan

**Data Penjualan**

Kode: 000000003

ID Customer:

ID Obat:

ID Obat	Nama
080000001	Sakatonik
080000002	Neuragin
080000003	klorokin
080000004	Herochin

Jumlah:    Harga: 6.000,00   

No.	Kode	Nama.	Jml.	Harga.
-----	------	-------	------	--------

**Gambar 4.12 Transaksi Penjualan Obat**

#### 4.1.2.4. Sub Menu Transaksi Penjualan umum

Sub menu ini berfungsi sebagai inputan data transaksi penjualan obat tanpa menggunakan resep yang dilakukan konsumen pada apotek ini. Adapun tampilannya seperti pada gambar dibawah ini.

Gambar 4.13 Transaksi Penjualan Umum

### 4.1.3 Menu Laporan

Menu laporan terdiri dari sub menu laporan yang berisi report-report hasil dari penyimpanan data.

#### 4.1.3.1 Laporan Data Konsumen

Laporan ini berisi data konsumen

Form data konsumen berguna untuk mengetahui nama konsumen yang melakukan transaksi di apotek dalam kurung waktu tertentu.

31/03/2000

ID	Nama	Alamat	Telp
C00000	Tomas	Jl Jombang Malang	(031) 555-7876
C00001	Toni	Malang	(089) 033-0932
C00002	tino	Jl mentan	(081) 333-1419
C00003	jmy	Jl leopa	(72) 629-76

Current Page No.: 1      Total Page No.: 1      Zoom Factor: 100%

**Gambar 4.14 Laporan Data Konsumen**

#### 4.1.3.2 Laporan Data Supplier

Laporan ini berisi data supplier

Form data supplier berguna untuk mengetahui nama supplier yang menyuplai obat di apotek tersebut dalam kurung waktu tertentu.

31/03/2000

ID	Nama	Alamat	Telp
S00001	Kalbe Farma	Jl Surabaya 90	(081) 099-0898
S00002	Bode	Jakarta	(221) 555-0999
S00003	hatuae	Jl melati	(08) 133-3141
S00004	ny	Jl caty	(0) 812-2233

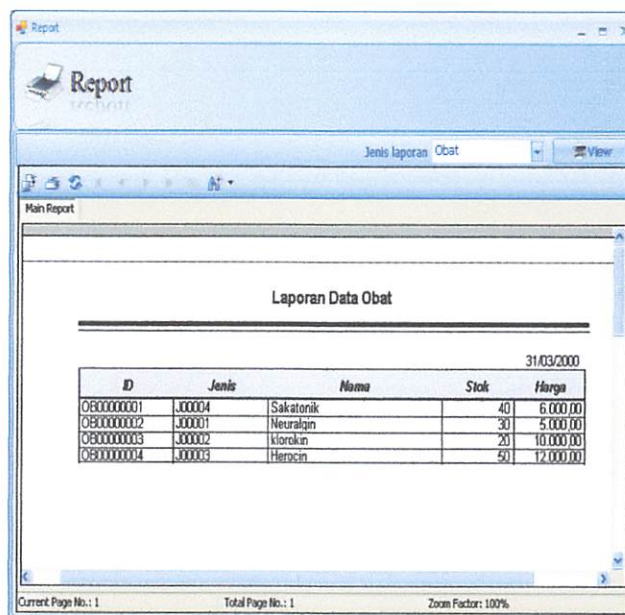
Current Page No.: 1      Total Page No.: 1      Zoom Factor: 100%

**Gambar 4.15 Laporan Data Supplier**

### 4.1.3.3 Laporan Data Obat

Laporan ini berisi data obat

Form data obat berguna untuk mengetahui jumlah jenis obat yang tersedia di apotek tersebut.



The screenshot shows a software window titled "Report" with a sub-window titled "Main Report". The report content is titled "Laporan Data Obat" and includes a date "31.03.2000". Below the title is a table with the following data:

ID	Jenis	Nama	Stok	Harga
0800000001	J00004	Sakatonik	40	6.000,00
0800000002	J00001	Neuralgin	30	5.000,00
0800000003	J00002	klorokin	20	10.000,00
0800000004	J00003	Herocin	50	12.000,00

At the bottom of the window, it displays "Current Page No.: 1", "Total Page No.: 1", and "Zoom Factor: 100%".

**Gambar 4.16 Laporan Data Obat**

### 4.1.3.4 Laporan Pembelian

Laporan ini berisi data pembelian

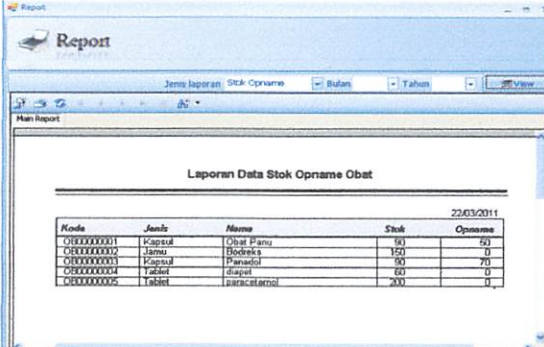
Form pembelian berguna untuk mengetahui transaksi pembelian dan kemudian melaporkan data jumlah obat yang telah mengalami transaksi.

The screenshot shows a report window titled "Laporan Pembelian" with a table containing purchase data. The table has three columns: "No", "ID Supplier", and "Tgl". The data is grouped into several sections, each with a "Total" row. The status bar at the bottom indicates "Current Page No.: 1", "Total Page No.: 1", and "Zoom Factor: 100%".

No	ID Supplier	Tgl
00000001	S00001	11/07/2010
00000002	S00002	11/07/2010
00000003	S00003	11/07/2010
00000004	S00004	11/07/2010
00000005	S00005	11/07/2010
00000006	S00006	11/07/2010
00000007	S00007	11/07/2010
00000008	S00008	11/07/2010
00000009	S00009	11/07/2010
00000010	S00010	11/07/2010
00000011	S00011	11/07/2010
00000012	S00012	11/07/2010
00000013	S00013	11/07/2010
00000014	S00014	11/07/2010
00000015	S00015	11/07/2010
00000016	S00016	11/07/2010
00000017	S00017	11/07/2010
00000018	S00018	11/07/2010
00000019	S00019	11/07/2010
00000020	S00020	11/07/2010
00000021	S00021	11/07/2010
00000022	S00022	11/07/2010
00000023	S00023	11/07/2010
00000024	S00024	11/07/2010
00000025	S00025	11/07/2010
00000026	S00026	11/07/2010
00000027	S00027	11/07/2010
00000028	S00028	11/07/2010
00000029	S00029	11/07/2010
00000030	S00030	11/07/2010
00000031	S00031	11/07/2010
00000032	S00032	11/07/2010
00000033	S00033	11/07/2010
00000034	S00034	11/07/2010
00000035	S00035	11/07/2010
00000036	S00036	11/07/2010
00000037	S00037	11/07/2010
00000038	S00038	11/07/2010
00000039	S00039	11/07/2010
00000040	S00040	11/07/2010
00000041	S00041	11/07/2010
00000042	S00042	11/07/2010
00000043	S00043	11/07/2010
00000044	S00044	11/07/2010
00000045	S00045	11/07/2010
00000046	S00046	11/07/2010
00000047	S00047	11/07/2010
00000048	S00048	11/07/2010
00000049	S00049	11/07/2010
00000050	S00050	11/07/2010
00000051	S00051	11/07/2010
00000052	S00052	11/07/2010
00000053	S00053	11/07/2010
00000054	S00054	11/07/2010
00000055	S00055	11/07/2010
00000056	S00056	11/07/2010
00000057	S00057	11/07/2010
00000058	S00058	11/07/2010
00000059	S00059	11/07/2010
00000060	S00060	11/07/2010
00000061	S00061	11/07/2010
00000062	S00062	11/07/2010
00000063	S00063	11/07/2010
00000064	S00064	11/07/2010
00000065	S00065	11/07/2010
00000066	S00066	11/07/2010
00000067	S00067	11/07/2010
00000068	S00068	11/07/2010
00000069	S00069	11/07/2010
00000070	S00070	11/07/2010
00000071	S00071	11/07/2010
00000072	S00072	11/07/2010
00000073	S00073	11/07/2010
00000074	S00074	11/07/2010
00000075	S00075	11/07/2010
00000076	S00076	11/07/2010
00000077	S00077	11/07/2010
00000078	S00078	11/07/2010
00000079	S00079	11/07/2010
00000080	S00080	11/07/2010
00000081	S00081	11/07/2010
00000082	S00082	11/07/2010
00000083	S00083	11/07/2010
00000084	S00084	11/07/2010
00000085	S00085	11/07/2010
00000086	S00086	11/07/2010
00000087	S00087	11/07/2010
00000088	S00088	11/07/2010
00000089	S00089	11/07/2010
00000090	S00090	11/07/2010
00000091	S00091	11/07/2010
00000092	S00092	11/07/2010
00000093	S00093	11/07/2010
00000094	S00094	11/07/2010
00000095	S00095	11/07/2010
00000096	S00096	11/07/2010
00000097	S00097	11/07/2010
00000098	S00098	11/07/2010
00000099	S00099	11/07/2010
00000100	S00100	11/07/2010
00000101	S00101	11/07/2010
00000102	S00102	11/07/2010
00000103	S00103	11/07/2010
00000104	S00104	11/07/2010
00000105	S00105	11/07/2010
00000106	S00106	11/07/2010
00000107	S00107	11/07/2010
00000108	S00108	11/07/2010
00000109	S00109	11/07/2010
00000110	S00110	11/07/2010
00000111	S00111	11/07/2010
00000112	S00112	11/07/2010
00000113	S00113	11/07/2010
00000114	S00114	11/07/2010
00000115	S00115	11/07/2010
00000116	S00116	11/07/2010
00000117	S00117	11/07/2010
00000118	S00118	11/07/2010
00000119	S00119	11/07/2010
00000120	S00120	11/07/2010
00000121	S00121	11/07/2010
00000122	S00122	11/07/2010
00000123	S00123	11/07/2010
00000124	S00124	11/07/2010
00000125	S00125	11/07/2010
00000126	S00126	11/07/2010
00000127	S00127	11/07/2010
00000128	S00128	11/07/2010
00000129	S00129	11/07/2010
00000130	S00130	11/07/2010
00000131	S00131	11/07/2010
00000132	S00132	11/07/2010
00000133	S00133	11/07/2010
00000134	S00134	11/07/2010
00000135	S00135	11/07/2010
00000136	S00136	11/07/2010
00000137	S00137	11/07/2010
00000138	S00138	11/07/2010
00000139	S00139	11/07/2010
00000140	S00140	11/07/2010
00000141	S00141	11/07/2010
00000142	S00142	11/07/2010
00000143	S00143	11/07/2010
00000144	S00144	11/07/2010
00000145	S00145	11/07/2010
00000146	S00146	11/07/2010
00000147	S00147	11/07/2010
00000148	S00148	11/07/2010
00000149	S00149	11/07/2010
00000150	S00150	11/07/2010
00000151	S00151	11/07/2010
00000152	S00152	11/07/2010
00000153	S00153	11/07/2010
00000154	S00154	11/07/2010
00000155	S00155	11/07/2010
00000156	S00156	11/07/2010
00000157	S00157	11/07/2010
00000158	S00158	11/07/2010
00000159	S00159	11/07/2010
00000160	S00160	11/07/2010
00000161	S00161	11/07/2010
00000162	S00162	11/07/2010
00000163	S00163	11/07/2010
00000164	S00164	11/07/2010
00000165	S00165	11/07/2010
00000166	S00166	11/07/2010
00000167	S00167	11/07/2010
00000168	S00168	11/07/2010
00000169	S00169	11/07/2010
00000170	S00170	11/07/2010
00000171	S00171	11/07/2010
00000172	S00172	11/07/2010
00000173	S00173	11/07/2010
00000174	S00174	11/07/2010
00000175	S00175	11/07/2010
00000176	S00176	11/07/2010
00000177	S00177	11/07/2010
00000178	S00178	11/07/2010
00000179	S00179	11/07/2010
00000180	S00180	11/07/2010
00000181	S00181	11/07/2010
00000182	S00182	11/07/2010
00000183	S00183	11/07/2010
00000184	S00184	11/07/2010
00000185	S00185	11/07/2010
00000186	S00186	11/07/2010
00000187	S00187	11/07/2010
00000188	S00188	11/07/2010
00000189	S00189	11/07/2010
00000190	S00190	11/07/2010
00000191	S00191	11/07/2010
00000192	S00192	11/07/2010
00000193	S00193	11/07/2010
00000194	S00194	11/07/2010
00000195	S00195	11/07/2010
00000196	S00196	11/07/2010
00000197	S00197	11/07/2010
00000198	S00198	11/07/2010
00000199	S00199	11/07/2010
00000200	S00200	11/07/2010
00000201	S00201	11/07/2010
00000202	S00202	11/07/2010
00000203	S00203	11/07/2010
00000204	S00204	11/07/2010
00000205	S00205	11/07/2010
00000206	S00206	11/07/2010
00000207	S00207	11/07/2010
00000208	S00208	11/07/2010
00000209	S00209	11/07/2010
00000210	S00210	11/07/2010
00000211	S00211	11/07/2010
00000212	S00212	11/07/2010
00000213	S00213	11/07/2010
00000214	S00214	11/07/2010
00000215	S00215	11/07/2010
00000216	S00216	11/07/2010
00000217	S00217	11/07/2010
00000218	S00218	11/07/2010
00000219	S00219	11/07/2010
00000220	S00220	11/07/2010
00000221	S00221	11/07/2010
00000222	S00222	11/07/2010
00000223	S00223	11/07/2010
00000224	S00224	11/07/2010
00000225	S00225	11/07/2010
00000226	S00226	11/07/2010
00000227	S00227	11/07/2010
00000228	S00228	11/07/2010
00000229	S00229	11/07/2010
00000230	S00230	11/07/2010
00000231	S00231	11/07/2010
00000232	S00232	11/07/2010
00000233	S00233	11/07/2010
00000234	S00234	11/07/2010
00000235	S00235	11/07/2010
00000236	S00236	11/07/2010
00000237	S00237	11/07/2010
00000238	S00238	11/07/2010
00000239	S00239	11/07/2010
00000240	S00240	11/07/2010
00000241	S00241	11/07/2010
00000242	S00242	11/07/2010
00000243	S00243	11/07/2010
00000244	S00244	11/07/2010
00000245	S00245	11/07/2010
00000246	S00246	11/07/2010
00000247	S00247	11/07/2010
00000248	S00248	11/07/2010
00000249	S00249	11/07/2010
00000250	S00250	11/07/2010
00000251	S00251	11/07/2010
00000252	S00252	11/07/2010
00000253	S00253	11/07/2010

#### 4.1.3.6 Laporan Data Stock Opname Obat

Laporan ini berisi data stock opname obat  
Form stock opname obat berguna untuk memperbaharui jumlah stock obat pada apotek tersebut.

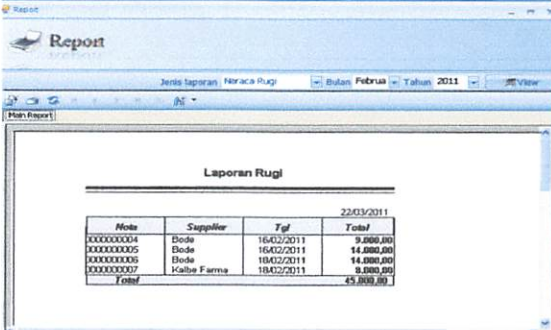


Kode	Jumlah	Nama	Stok	Opname
000000001	Kontrol	Obat Pains	50	60
000000002	Jamu	Bodrex	150	0
000000003	Kapsul	Parasetol	90	70
000000004	Tablet	Clavit	60	0
000000005	Tablet	Lasacatanol	200	0

Gambar 4.19 Laporan Data Stock Opname Obat

#### 4.1.3.7 Laporan Rugi

Laporan ini berisi data penjualan obat ke customer dikelompokkan perbulan.



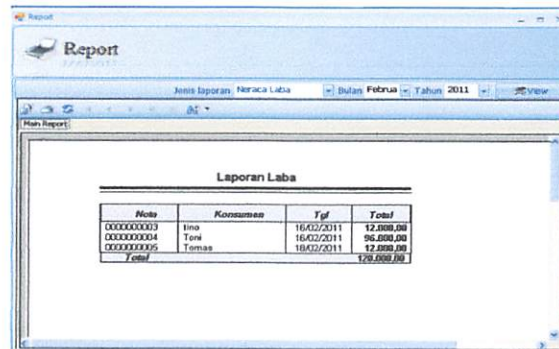
Kode	Supplier	Tgl	Total
000000004	Bode	16/02/2011	9.000,00
000000005	Bode	16/02/2011	14.000,00
000000006	Bode	18/02/2011	14.000,00
000000007	Kabir Farma	18/02/2011	8.000,00
<b>Total</b>			<b>45.000,00</b>

Gambar 4.20 Laporan Rugi



### 4.1.3.8 Laporan Laba

Laporan ini berisi data supplier dari perusahaan apa saja yang menyuplai obat pada apotek tersebut, dikelompokkan perbulan.



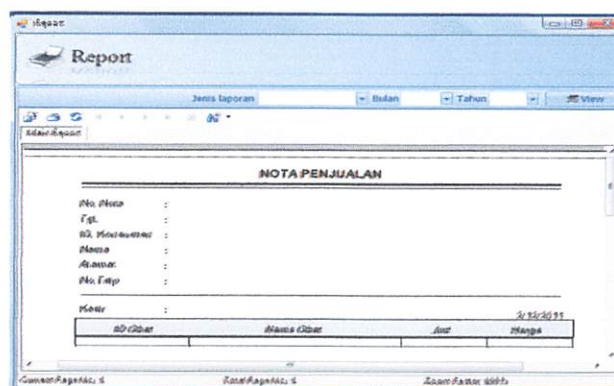
Nota	Konsumsi	Tgl	Total
000000003	Iris	16/02/2011	12.000,00
000000004	Toni	16/02/2011	96.000,00
000000005	Tomas	18/02/2011	12.000,00
<b>Total</b>			<b>120.000,00</b>

Gambar 4.21 Laporan Laba

### 4.1.4 Menu Nota

#### 4.1.3.1 Nota Penjualan

Form nota penjualan ini berguna bagi konsumen untuk mengetahui transaksi yang dilakukan pada apotek ini.



NOTA PENJUALAN

No. Nota :  
 Tgl. :  
 No. Revisi/Retur :  
 Nama :  
 Alamat :  
 No. Telp. :

Kode : Rp 12.500,00

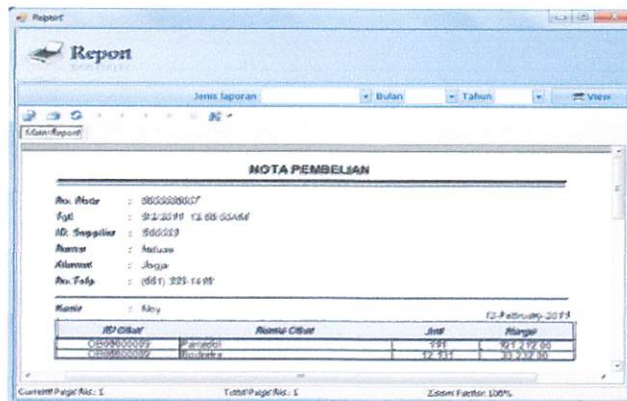
No. Obat	Nama Obat	Jml	Harga

Column Page: 4      Total Page: 4      Zoom Factor: 100%

Gambar 4.22 Nota Penjualan

### 4.1.3.2 Nota Pembelian

Form nota pembelian ini berguna untuk mengetahui transaksi yang dilakukan pada apotek ini dan kemudian dilaporkan pada pimpinan apotek yang dilakukan secara berkala dalam periode tertentu.



Report

Jenis laporan: Bulan Tahun View

1:GanReport

**NOTA PEMBELIAN**

No. Nota : 000000007  
Tgl : 02-2019 12 00 00AM  
No. Supplier : 000000  
Nama : Natura  
Alamat : Jorja  
No. Telp : (061) 333 14 00  
Nomor : 000

12-Februari-2019

ID Obat	Nama Obat	Jml	Harga
000000000	Parasetamol	100	21.700,00
000000000	Amoxicillin	12.500	93.720,00

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

Gambar 4.23 Nota Pembelian

## 4.2 Perbandingan Sistem Lama Dengan Sistem Yang Sekarang

Dari pengujian proses yang telah dilakukan diatas, maka dapat dibandingkan hasilnya dengan kinerja sistem yang lama. Berikut adalah tabel perbandingan hasil pengujian sistem.

Tabel 4.1

Tabel Perbandingan Hasil Pengujian Sistem

No.	Proses	Sistem Lama	Sistem Yang Dibuat
1	Data Supplier	Semua data supplier ditulis dalam sebuah buku.	Untuk sistem yang sudah dirancang,data supplier tercatat dan tersimpan dalam database.
2	Obat	Sejak obat diterima dari supplier, petugas mencatat obat berdasarkan jenis obat, nama obat, jlh stok obat, harga pembelian obat, dalam buku catatan.	Dengan sistem yang sudah dirancang, data obat yang telah dibeli dari supplier, tercatat dan tersimpan dalam database.
3	Data konsumen	Belum ada proses pencatatan data konsumen yang melakukan transaksi pada apotek.	Data konsumen disimpan dalam sistem ini untuk melihat sejauh mana pelayanan apotek ini yang diberikan kepada para konsumen.
4.	Transaksi pembelian obat	Pencatatan dilakukan dengan menggunakan sebuah buku khusus, yang disediakan untuk mencatat transaksi pembelian obat dari	Pencatatan dilakukan dengan menggunakan sebuah buku khusus, yang disediakan untuk mencatat transaksi pembelian obat dari supplier.

		supplier.	
5.	Transaksi pembelian umum	Pencatatan dilakukan dengan menggunakan sebuah buku khusus, yang disediakan untuk mencatat transaksi pembelian umum	Data transaksi tersimpan dalam sebuah database yang terintegrasi dengan sistem.
6.	Transaksi penjualan obat	Pencatatan dilakukan dengan menggunakan sebuah buku khusus, yang disediakan untuk mencatat transaksi penjualan obat.	Pencatatan dilakukan dengan menggunakan sebuah buku khusus, yang disediakan untuk mencatat transaksi penjualan obat dari supplier.
7.	Transaksi penjualan umum	Pencatatan dilakukan dengan menggunakan sebuah buku khusus, yang disediakan untuk mencatat transaksi pembelian umum.	Data transaksi tersimpan dalam sebuah database yang terintegrasi dengan sistem.
8.	Laporan	Fasilitas laporan yang disediakan belumlah maksimal karena laporan yang tersedia hanya untuk menampilkan data obat, data supplier, data pembelian obat dan pembayaran obat saja	Fasilitas laporan yang disediakan lebih memudahkan pengguna dalam mengelola data informasi, karena laporan terbagi dalam beberapa kategori laporan.

## **BAB V**

### **PENUTUP**

#### **5.1. Kesimpulan**

Dari hasil analisis dan perancangan aplikasi sistem informasi penjualan obat pada apotek Rumah Sakit Umum Distric Liquiça Timor Leste serta implementasi dan pengujian yang telah dilakukan dapat diambil beberapa kesimpulan sebagai berikut :

1. Dengan sistem informasi apotek ini, bertujuan untuk membangun system informasi penjualan obat, dimana aplikasi yang dibangun dapat digunakan untuk mengelola data obat, transaksi barang masuk penjualan dan pembelian retail obat, stok obat, dan data konsumen pada apotek tersebut.
2. Dengan fasilitas yang ada pada aplikasi ini, dapat meningkatkan pelayanan kepada para konsumen yang membutuhkan obat-obatan.
3. Semua data baik data transaksi maupun non-transaksi tidak lagi membutuhkan banyak ruang penyimpanan dengan adanya sistem informasi ini.

#### **5.2. Saran**

Untuk pengembangan lebih lanjut dari aplikasi ini, penulis menyarankan agar sistem selanjutnya sudah disertakan dengan perhitungan keuntungan dan kerugian yang terjadi dalam sistem informasi apotek ini serta agar dikembangkan menjadi sistem informasi manajemen.

## DAFTAR PUSTAKA

- [1] Komputer, Wahana, 007, Membangun Aplikasi Database dengan Visual Basic.NET, Yogyakarta:Penerbit Andi.
- [2] Prasetyo, Didik Dwi, 2006, Pemrograman Aplikasi Database dengan Visual Basic.NET 2005, Jakarta:PT. Elex Media Komputindo
- [3] Subari. Yuswanto. 2005. Mengolah Database dengan SQL Server 2005, Surabaya:Prestasi Pustaka.
- [4] Utomo, Eko Priyo. 2006. Membuat Aplikasi Database dengan Visual Basic.NET, Bandung:Yrama Widya.
- [5] Umar, M, (2004), “Manajemen Apotik Praktis”, cetakan I. penerbit Ar Rahman, Solo.
- [6] H.S, Suryadi D., & Bunawan. 1995. *Pengantar Metodologi Pengembangan Sistem Informasi Penjualan*. Jakarta: Gunadarma.
- [7] <http://duniapancer.wordpress.com/2008/12/04/sim-apotik-mudah-kah/> (23 nov 09 )
- [8] Sutabri, Tata. 2004. *Analisa Sistem Informasi*, Yogyakarta:Penerbit Andi.
- [9] Salemba Infotek , 008, Pembuatan Aplikasi Profesional dengan Visual Basic.NET, Jakarta: Edward Tanujaya.

# LAMPIRAN



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, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

**BERITA ACARA UJIAN SKRIPSI**

**FAKULTAS TEKNOLOGI INDUSTRI**

Nama : MARIZA FÁTIMA DOS SANTOS DA CRUZ  
NIM : 05.12.731  
Jurusan : T.Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika  
Judul Skripsi : DESAIN SISTEM INFORMASI PENJUALAN OBAT PADA APOTEK RUMAH SAKIT UMUM DISTRICT LIQUIÇA TIMOR LESTE

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

Hari : **Senin**  
Tanggal : **21 Februari 2011**  
Dengan Nilai : **73,75 (B+) *10***

**Panitia Ujian Skripsi**

**Ketua Majelis Penguji**

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

**Dosen Penguji I**

(Sotyohadi, ST)  
NIP.Y. 1039700309

**Sekretaris Majelis Penguji**

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

**Dosen Penguji II**

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





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

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

PT. BNI (PERSERO) MALANG  
 BANK NIAGA MALANG

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

**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 : Mariza Fátima Dos Santos Da Cruz  
 NIM : 05.12.731  
 Jurusan : Teknik Elektro S-1  
 Konsentrasi : Teknik Komputer & Informatika  
 Masa Bimbingan : 09 Desember 2010 s/d 09 April 2011  
 Judul Skripsi : Desain Sistem Informasi Penjualan Obat Pada Apotek Rumah Sakit Umum Distric Liquica Timor Leste.

Penguji/Tanggal	Uraian	Paraf
Penguji II 21 Februari 2011	Perbaiki tinjauan pustaka/landasan teori, perbaiki relasi data base/CDM(relasi harus diberi nama/keterangan), perbaiki kesimpulan, stock oupname belum ada(pada akhir bulan memanfaatkan crystal report), hapus batasan masalah no 5 karena secara otomatis membahas keuntungan, tambahkan laporan/neraca rugi laba.	

**Penguji I**

**(Sotyohadi, ST)**  
 NIP.Y. 1039700309

**Dosen Penguji**

**Penguji II**

**(Sandy Nataly Mantja, S.Kom)**  
 NIP.P. 1030800418

**Mengetahui**

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**(Joseph Dedy Irawan, ST. MT)**  
 NIP. 197404162005011002

**(Ahmad Faisol, ST)**  
 NIP.P. 1031000431



**FORMULIR BIMBINGAN SKRIPSI**

Nama : Mariza Fatima dos Santos da Cruz  
Nim : 05.12.731  
Masa Bimbingan : 09 Desember 2010 s/d 09 Juni 2011 *BA*  
Judul Skripsi : Desain Sistem Informasi Apotek pada Rumah Sakit Umum Distric  
Liquiça – Timor Leste

No	Tanggal	Uraian	Paraf Pembimbing
1	11/01/2011	Acc BAB I	
2	14/01/2011	Acc BAB II	
3	17/01/2011	Acc BAB III	
4	19/01/2011	Acc BAB IV	
5	22/01/2011	Acc BAB V	
6			
7			
8			
9			
10			

Malang, 2011

Dosen pembimbing I

**Joseph Dedy Irawan, ST. MT**  
NIP. 19740416200501 1002



**FORMULIR BIMBINGAN SKRIPSI**

Nama : Mariza Fatima dos Santos da Cruz  
Nim : 05.12.731  
Masa Bimbingan : 09 Desember 2010 s/d 09 Juni 2011 <sup>34</sup>  
Judul Skripsi : Desain Sistem Informasi Apotek pada Rumah Sakit Umum Distric  
Liquiça – Timor Leste

No	Tanggal	Uraian	Paraf Pembimbing
1	03/01/2011	ACC Bab I, II	
2	10/01/2011	Rengi Bab II - Perbaiki DFD	
3			
4			
5			
6			
7			
8			
9			
10			

Malang, 2011

Dosen pembimbing II

Ahmad Fajol, ST

## Source Code

### • Form Jenis Obat

```
Public Class FormJenis
    Inherits DevComponents.DotNetBar.Office2007Form

    Private Sub get_code()
        Dim i As Integer = 1
        Dim found As Boolean = False
        While Not found And i < Integer.MaxValue
            If Not found And Me.Jenis_obatTableAdapter.IsExist("J" +
Format(i, "00000")) = 0 Then
                found = True
                kode.Text = "J" + Format(i, "00000")
            End If
            i = i + 1
        End While
    End Sub

    Private Sub FormJenis_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the 'Ds_Con.jenis_obat'
table. You can move, or remove it, as needed.
        Me.Jenis_obatTableAdapter.Fill(Me.Ds_Con.jenis_obat)
        get_code()
    End Sub

    Private Sub BtnBatal_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBatal.Click
        Me.Jenis_obatTableAdapter.Fill(Me.Ds_Con.jenis_obat)
        get_code()
        nama.Clear()
        BtnHapus.Enabled = False
        BtnSimpan.Text = "Simpan"
        BtnSimpan.Enabled = False
    End Sub

    Private Sub kode_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles kode.TextChanged, nama.TextChanged
        BtnSimpan.Enabled = (nama.Text.Trim <> "")
    End Sub

    Private Sub data_CellContentClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
data.CellContentClick, data.CellClick
        BtnSimpan.Text = "Edit"
        BtnHapus.Enabled = True
        BtnSimpan.Enabled = True
        get_code()
        nama.Clear()
    End Sub

    Private Sub BtnSimpan_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnSimpan.Click
```

```

Try
    If BtnSimpan.Text = "Edit" Then
        Dim dt As DataTable
        Dim dtDta As New
Ds_ConTableAdapters.jenis_obatTableAdapter()

        dt = dtDta.GetDataByKode(tempKode.Text)
        If dt.Rows.Count > 0 Then
            Me.nama.Text = dt.Rows(0).Item("jenis")
            Me.kode.Text = tempKode.Text
            Me.BtnSimpan.Text = "Update"
        End If
    ElseIf BtnSimpan.Text = "Simpan" Then
        If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menyimpan Data?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
            Me.Jenis_obatTableAdapter.InsertQuery(kode.Text.Trim,
nama.Text.Trim)
            DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil disimpan.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            BtnBatal_Click(sender, e)
        End If
    ElseIf BtnSimpan.Text = "Update" Then
        If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Melakukan Update Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
            Me.Jenis_obatTableAdapter.UpdateQuery(nama.Text.Trim,
tempKode.Text)
            DevComponents.DotNetBar.MessageBoxEx.Show("Data
Berhasil DiUpdate.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            BtnBatal_Click(sender, e)
        End If
    End If
Catch
    DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
End Sub

Private Sub BtnHapus_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnHapus.Click
    Try
        If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menghapus Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
            Me.Jenis_obatTableAdapter.DeleteQuery(tempKode.Text)
            DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil Dihapus.", "Konfirmasi", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            BtnBatal_Click(sender, e)
        End If
    Catch ex As Exception
        DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)

```

```

        End Try
    End Sub

    Private Sub ReflectionLabell_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ReflectionLabell.Click

        End Sub
    End Class

```

## • Form Konsumen

```

Public Class FormKonsumen
    Inherits DevComponents.DotNetBar.Office2007Form

    Private Sub get_code()
        Dim i As Integer = 1
        Dim found As Boolean = False
        While Not found And i < Integer.MaxValue
            If Not found And Me.KonsumenTableAdapter.IsExist("C" +
Format(i, "00000")) = 0 Then
                found = True
                kode.Text = "C" + Format(i, "00000")
            End If
            i = i + 1
        End While
    End Sub

    Private Sub FormKonsumen_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the 'Ds_Con.konsumen'
table. You can move, or remove it, as needed.
        Me.KonsumenTableAdapter.Fill(Me.Ds_Con.konsumen)
        get_code()
    End Sub

    Private Sub ButtonX2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnSimpan.Click
        Try
            If BtnSimpan.Text = "Edit" Then
                Dim dt As DataTable
                Dim dtPasien As New
Ds_ConTableAdapters.konsumenTableAdapter()

                dt = dtPasien.GetDataByKode(tempKode.Text)
                If dt.Rows.Count > 0 Then
                    Me.nama.Text = dt.Rows(0).Item("nama")
                    Me.alamat.Text = dt.Rows(0).Item("alamat")
                    Me.telp.Text = dt.Rows(0).Item("telp")
                    Me.kode.Text = tempKode.Text
                    Me.BtnSimpan.Text = "Update"
                End If
            ElseIf BtnSimpan.Text = "Simpan" Then

```

```

        If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menyimpan Data?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
            Me.KonsumenTableAdapter.InsertQuery(kode.Text.Trim,
nama.Text.Trim, alamat.Text.Trim, telp.Text.Trim)
            DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil disimpan.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            ButtonX1_Click(sender, e)
        End If
    ElseIf BtnSimpan.Text = "Update" Then
        If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Melakukan Update Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
            Me.KonsumenTableAdapter.UpdateQuery(nama.Text.Trim,
alamat.Text.Trim, telp.Text.Trim, tempKode.Text, tempKode.Text)
            DevComponents.DotNetBar.MessageBoxEx.Show("Data
Berhasil DiUpdate.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
            ButtonX1_Click(sender, e)
        End If
    End If
Catch
    DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
End Sub

Private Sub ButtonX1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBatal.Click
    get_code()
    telp.Text = ""
    alamat.Clear()
    nama.Clear()
    BtnHapus.Enabled = False
    BtnSimpan.Text = "Simpan"
    BtnSimpan.Enabled = False
    Me.KonsumenTableAdapter.Fill(Me.Ds_Con.konsumen)
End Sub

Private Sub kode_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles kode.TextChanged, telp.TextChanged,
nama.TextChanged, alamat.TextChanged
    BtnSimpan.Enabled = (nama.Text.Trim <> "") And (alamat.Text.Trim
<> "") And (telp.Text.Trim <> "( ) -")

End Sub

Private Sub nama_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs)

End Sub

Private Sub DataGridViewX1_CellContentClick(ByVal sender As
System.Object, ByVal e As System.Windows.Forms.DataGridViewCellEventArgs)
Handles data.CellContentClick, data.CellClick
    BtnSimpan.Text = "Edit"

```

```

        BtnHapus.Enabled = True
        BtnSimpan.Enabled = True
        get_code()
        telp.Text = ""
        alamat.Clear()
        nama.Clear()
        BtnJual.Visible = Not panelisi.Visible
    End Sub

    Private Sub TxtCari_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TxtCari.TextChanged
        If pill.Checked Then
            Me.KonsumenTableAdapter.FillByLikeKode(Me.Ds_Con.konsumen, TxtCari.Text.Trim)
        Else
            Me.KonsumenTableAdapter.FillByLikeNama(Me.Ds_Con.konsumen, TxtCari.Text.Trim)
        End If
        data.Refresh()
    End Sub

    Private Sub BtnHapus_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BtnHapus.Click
        Try
            If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk Menghapus Data?", "Konfirmasi.", MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                Me.KonsumenTableAdapter.DeleteQuery(tempKode.Text)
                DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah Berhasil Dihapus.", "Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
                ButtonX1_Click(sender, e)
            End If
        Catch ex As Exception
            DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End Sub

    Private Sub BtnJual_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BtnJual.Click
        If FormPenjualan.Visible Then
            FormPenjualan.CariData.Text = tempKode.Text
            Me.Close()
        End If
    End Sub

    Private Sub ReflectionLabel1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ReflectionLabel1.Click

    End Sub

    Private Sub PanelEx1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles PanelEx1.Click

    End Sub

```



End Class

- **Form Obat**

```
Public Class FormObat
    Inherits DevComponents.DotNetBar.Office2007Form

    Private Sub get_code()
        Dim i As Integer = 1
        Dim found As Boolean = False
        While Not found And i < Integer.MaxValue
            If Not found And Me.ObatTableAdapter.IsExist("OB" + Format(i,
"00000000")) = 0 Then
                found = True
                kode.Text = "OB" + Format(i, "00000000")
            End If
            i = i + 1
        End While
    End Sub

    Private Sub data_CellContentClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
data.CellContentClick, data.CellClick, data.CellContentDoubleClick,
data.CellDoubleClick
        BtnSimpan.Text = "Edit"
        BtnHapus.Enabled = True
        BtnSimpan.Enabled = True
        get_code()
        stok.Text = 0
        harga.Text = 0
        nama.Clear()
    End Sub

    Private Sub FormObat_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the 'Ds_Con.jenis_obat'
table. You can move, or remove it, as needed.
        Me.Jenis_obatTableAdapter.Fill(Me.Ds_Con.jenis_obat)
        'TODO: This line of code loads data into the 'Ds_Con.obat' table.
You can move, or remove it, as needed.
        Me.ObatTableAdapter.Fill(Me.Ds_Con.obat)
        'TODO: This line of code loads data into the 'Ds_Con.obat' table.
You can move, or remove it, as needed.
        Me.ObatTableAdapter.Fill(Me.Ds_Con.obat)
        get_code()

    End Sub

    Private Sub kode_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles kode.TextChanged, stok.TextChanged,
nama.TextChanged, jenis.TextChanged, harga.TextChanged,
stok.ValueChanged, harga.ValueChanged
        BtnSimpan.Enabled = (nama.Text.Trim <> "") And (harga.Text.Trim
<> "") And (stok.Text.Trim <> "")

    End Sub
```

```

Private Sub BtnSimpan_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnSimpan.Click
    Try
        If BtnSimpan.Text = "Edit" Then
            Dim dt As DataTable
            Dim dtPasien As New
Ds_ConTableAdapters.obatTableAdapter()

            dt = dtPasien.GetDataByKode(tempKode.Text)
            If dt.Rows.Count > 0 Then
                Me.nama.Text = dt.Rows(0).Item("nama")
                Me.stok.Text = dt.Rows(0).Item("stok")
                Me.harga.Text = dt.Rows(0).Item("harga_jual")
                Me.jenis.SelectedValue = dt.Rows(0).Item("kd_jenis")
                Me.kode.Text = tempKode.Text
                Me.BtnSimpan.Text = "Update"
            End If
            ElseIf BtnSimpan.Text = "Simpan" Then
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menyimpan Data?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                    Me.ObatTableAdapter.InsertQuery(kode.Text.Trim,
jenis.SelectedValue, nama.Text.Trim, stok.Value, harga.Value, 0, 0)
                    DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil disimpan.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                    BtnBatal_Click(sender, e)
                End If
            ElseIf BtnSimpan.Text = "Update" Then
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Melakukan Update Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                    Me.ObatTableAdapter.UpdateQuery(jenis.SelectedValue,
nama.Text.Trim, stok.Value, harga.Value, 0, 0, tempKode.Text)
                    DevComponents.DotNetBar.MessageBoxEx.Show("Data
Berhasil DiUpdate.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                    BtnBatal_Click(sender, e)
                End If
            End If
        Catch
            DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End Sub

```

```

Private Sub BtnBatal_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBatal.Click
    get_code()
    harga.Text = 0
    stok.Text = 0
    nama.Clear()
    BtnHapus.Enabled = False
    BtnSimpan.Text = "Simpan"
    BtnSimpan.Enabled = False
    Me.Jenis_obatTableAdapter.Fill(Me.Ds_Con.jenis_obat)

```

'TODO: This line of code loads data into the 'Ds\_Con.obat' table.  
You can move, or remove it, as needed.

```
Me.ObatTableAdapter.Fill(Me.Ds_Con.obat)
End Sub
```

```
Private Sub BtnHapus_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnHapus.Click
```

```
Try
    If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menghapus Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
        Me.ObatTableAdapter.DeleteQuery(tempKode.Text)
        DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil Dihapus.", "Konfirmasi", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
        BtnBatal_Click(sender, e)
    End If
```

```
Catch ex As Exception
    DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
End Sub
```

```
Private Sub TxtCari_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles TxtCari.TextChanged
```

```
    If pill.Checked Then
        Me.ObatTableAdapter.FillByLikeKode(Me.Ds_Con.obat,
TxtCari.Text.Trim)
    Else
        Me.ObatTableAdapter.FillByLikeNama(Me.Ds_Con.obat,
TxtCari.Text.Trim)
    End If
    data.Refresh()
End Sub
```

```
Private Sub PanelEx1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PanelEx1.Click
```

```
End Sub
End Class
```

## • Form Supplier

```
Public Class FormSupplier
    Inherits DevComponents.DotNetBar.Office2007Form

    Private Sub get_code()
        Dim i As Integer = 1
        Dim found As Boolean = False
        While Not found And i < Integer.MaxValue
            If Not found And Me.SupplierTableAdapter.IsExist("S" +
Format(i, "00000")) = 0 Then
                found = True
                kode.Text = "S" + Format(i, "00000")
            End If
            i = i + 1
        End While
    End Sub

    Private Sub FormSupplier_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the 'Ds_Con.supplier'
table. You can move, or remove it, as needed.
        Me.SupplierTableAdapter.Fill(Me.Ds_Con.supplier)
        get_code()
    End Sub

    Private Sub kode_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles kode.TextChanged, telp.TextChanged,
nama.TextChanged, alamat.TextChanged
        BtnSimpan.Enabled = (nama.Text.Trim <> "") And (alamat.Text.Trim
<> "") And (telp.Text.Trim <> "( ) -")

    End Sub

    Private Sub BtnHapus_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnHapus.Click
        Try
            If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menghapus Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                Me.SupplierTableAdapter.DeleteQuery(tempKode.Text)
                DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil Dihapus.", "Konfirmasi", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                BtnBatal_Click(sender, e)
            End If

            Catch ex As Exception
                DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
            End Try
        End Sub

    Private Sub BtnBatal_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBatal.Click
        get_code()
        telp.Text = ""
    End Sub
End Class
```

```

        alamat.Clear()
        nama.Clear()
        BtnHapus.Enabled = False
        BtnSimpan.Text = "Simpan"
        BtnSimpan.Enabled = False
        Me.SupplierTableAdapter.Fill(Me.Ds_Con.supplier)
    End Sub

    Private Sub BtnSimpan_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnSimpan.Click
        Try
            If BtnSimpan.Text = "Edit" Then
                Dim dt As DataTable
                Dim dtdata As New
Ds_ConTableAdapters.supplierTableAdapter()

                dt = dtdata.GetDataByKode(tempKode.Text)
                If dt.Rows.Count > 0 Then
                    Me.nama.Text = dt.Rows(0).Item("nama")
                    Me.alamat.Text = dt.Rows(0).Item("alamat")
                    Me.telp.Text = dt.Rows(0).Item("telp")
                    Me.kode.Text = tempKode.Text
                    Me.BtnSimpan.Text = "Update"
                End If
            ElseIf BtnSimpan.Text = "Simpan" Then
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menyimpan Data?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                    Me.SupplierTableAdapter.InsertQuery(kode.Text.Trim,
nama.Text.Trim, alamat.Text.Trim, telp.Text.Trim)
                    DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil disimpan.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                    BtnBatal_Click(sender, e)
                End If
            ElseIf BtnSimpan.Text = "Update" Then
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Melakukan Update Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                    Me.SupplierTableAdapter.UpdateQuery(nama.Text.Trim,
alamat.Text.Trim, telp.Text.Trim, tempKode.Text)
                    DevComponents.DotNetBar.MessageBoxEx.Show("Data
Berhasil DiUpdate.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                    BtnBatal_Click(sender, e)
                End If
            End If
        End If
        Catch
            DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End Sub

    Private Sub data_CellContentClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
data.CellContentClick, data.CellDoubleClick, data.CellContentDoubleClick,
data.CellClick

```

```

        BtnSimpan.Text = "Edit"
        BtnHapus.Enabled = True
        BtnSimpan.Enabled = True
        get_code()
        telp.Text = ""
        alamat.Clear()
        nama.Clear()
        BtnBeli.Visible = Not panelisi.Visible
    End Sub

    Private Sub TxtCari_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles TxtCari.TextChanged
        If pill.Checked Then
            Me.SupplierTableAdapter.FillByLikeKode(Me.Ds_Con.supplier, TxtCari.Text.Trim)
        Else
            Me.SupplierTableAdapter.FillByLikeNama(Me.Ds_Con.supplier, TxtCari.Text.Trim)
        End If
        data.Refresh()

    End Sub

    Private Sub BtnBeli_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles BtnBeli.Click
        If FormPembelian.Visible Then
            FormPembelian.CariData.Text = tempKode.Text
            Me.Close()
        End If
    End Sub

    Private Sub PanelEx1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles PanelEx1.Click

    End Sub
End Class

```

- **Form User**

```

Public Class FormUsers
    Inherits DevComponents.DotNetBar.Office2007Form

    Private Sub kode_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles kode.TextChanged, telp.TextChanged,
pass.TextChanged, nama.TextChanged, jenis.TextChanged, alamat.TextChanged
        BtnSimpan.Enabled = (nama.Text.Trim <> "") And (alamat.Text.Trim
<> "") And (telp.Text.Trim <> "") And (kode.Text.Trim <> "") And
(pass.Text.Trim <> "") And (jenis.SelectedIndex <> -1)
    End Sub

    Private Sub BtnSimpan_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnSimpan.Click

        Try
            If BtnSimpan.Text = "Edit" Then
                Dim dt As DataTable
                Dim dtPasien As New
Ds_ConTableAdapters.usersTableAdapter()

                dt = dtPasien.GetDataByKode(tempKode.Text)
                If dt.Rows.Count > 0 Then
                    Me.nama.Text = dt.Rows(0).Item("nama")
                    Me.alamat.Text = dt.Rows(0).Item("alamat")
                    Me.telp.Text = dt.Rows(0).Item("telp")
                    Me.kode.Text = tempKode.Text
                    Me.jenis.SelectedIndex = dt.Rows(0).Item("status")
                    Me.kode.ReadOnly = True

                    Me.BtnSimpan.Text = "Update"
                End If
            ElseIf BtnSimpan.Text = "Simpan" Then
                If (Me.UsersTableAdapter.IsExist(kode.Text.Trim) > 0)
Then
                    DevComponents.DotNetBar.MessageBoxEx.Show("ID Sudah
Digunakan.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
                    Exit Sub
                End If
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menyimpan Data?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                    Me.UsersTableAdapter.InsertQuery(kode.Text.Trim,
nama.Text.Trim, alamat.Text.Trim, telp.Text.Trim, jenis.SelectedIndex,
pass.Text.Trim)

                    DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil disimpan.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                    BtnBatal_Click(sender, e)
                End If
            ElseIf BtnSimpan.Text = "Update" Then
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Melakukan Update Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then

```

```
Me.UsersTableAdapter.UpdateQuery(nama.Text.Trim,
alamat.Text.Trim, telp.Text.Trim, jenis.SelectedIndex, pass.Text.Trim,
kode.Text.Trim)
```

```
DevComponents.DotNetBar.MessageBoxEx.Show("Data
Berhasil DiUpdate.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
```

```
BtnBatal_Click(sender, e)
```

```
End If
```

```
End If
```

```
Catch
```

```
DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
```

```
End Try
```

```
End Sub
```

```
Private Sub FormUsers_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
```

```
'TODO: This line of code loads data into the 'Ds_Con.users'
table. You can move, or remove it, as needed.
```

```
Me.UsersTableAdapter.Fill(Me.Ds_Con.users)
```

```
End Sub
```

```
Private Sub BtnBatal_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBatal.Click
```

```
kode.ReadOnly = False
```

```
telp.Text = ""
```

```
alamat.Clear()
```

```
nama.Clear()
```

```
pass.Clear()
```

```
kode.Clear()
```

```
BtnHapus.Enabled = False
```

```
BtnSimpan.Text = "Simpan"
```

```
BtnSimpan.Enabled = False
```

```
Me.UsersTableAdapter.Fill(Me.Ds_Con.users)
```

```
End Sub
```

```
Private Sub data_CellContentClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
data.CellContentClick, data.CellContentDoubleClick, data.CellClick
```

```
BtnSimpan.Text = "Edit"
```

```
BtnHapus.Enabled = True
```

```
BtnSimpan.Enabled = True
```

```
telp.Text = ""
```

```
alamat.Clear()
```

```
nama.Clear()
```

```
nama.Clear()
```

```
pass.Clear()
```

```
kode.Clear()
```

```
End Sub
```

```
Private Sub BtnHapus_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnHapus.Click
```

```
Try
```

```
If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menghapus Data?", "Konfirmasi.", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
```



```

        Me.UsersTableAdapter.DeleteQuery(tempKode.Text)
        DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil Dihapus.", "Konfirmasi", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
        BtnBatal_Click(sender, e)
    End If

    Catch ex As Exception
        DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
End Sub

Private Sub ReflectionLabel1_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ReflectionLabel1.Click

    End Sub
End Class

```

- **FormReport**

```

Public Class FormReport
    Inherits DevComponents.DotNetBar.Office2007Form
    Public status, kode As String

    Private Sub BtnView_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnView.Click
        If status = "1" Then

            Dim Nota As notaBeli
            Nota = New notaBeli
            Dim dt As DataTable
            dt = Me.Nota_BeliTableAdapter1.GetDataByNota(kode)
            Nota.SetDataSource(dt)
            Me.CrystalReportViewer1.ReportSource = Nota

            Exit Sub
        ElseIf status = "2" Then
            Dim Nota As notaJual
            Nota = New notaJual
            Dim dt As DataTable
            dt = Me.Nota_JualTableAdapter1.GetDataByNota(kode)
            Nota.SetDataSource(dt)
            Me.CrystalReportViewer1.ReportSource = Nota
            CrystalReportViewer1.PrintReport()
            Exit Sub
        Else
            Select (CbJenis.SelectedIndex)
                Case 0
                    Dim lap As lapkonsumen
                    lap = New lapkonsumen
                    Dim dt As DataTable
                    dt = Me.KonsumenTableAdapter1.GetData()
                    lap.SetDataSource(dt)

```

```

        Me.CrystalReportViewer1.ReportSource = lap
    Case 1
        Dim lap As lapsupplier
        lap = New lapsupplier
        Dim dt As DataTable
        dt = Me.SupplierTableAdapter1.GetData()
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case 2
        Dim lap As lapobat
        lap = New lapobat
        Dim dt As DataTable
        dt = Me.LapObatTableAdapter1.GetData()
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case 3
        Dim lap As lapbeli
        lap = New lapbeli
        Dim dt As DataTable
        dt =
Me.LaporanJualTableAdapter1.GetData(bulan.SelectedIndex + 1,
tahun.SelectedIndex + 2010)
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case 4
        Dim lap As lapJual
        lap = New lapJual
        Dim dt As DataTable
        dt =
Me.LaporanBeliTableAdapter1.GetData(bulan.SelectedIndex + 1,
tahun.SelectedIndex + 2010)
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case Else
        DevComponents.DotNetBar.MessageBoxEx.Show("Pilih
Jenis Laporan.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Select

    End If
End Sub

Private Sub FormReport_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
    CbJenis.Items.Add("Konsumen")
    CbJenis.Items.Add("Supplier")
    CbJenis.Items.Add("Obat")
    CbJenis.Items.Add("Pembelian")
    CbJenis.Items.Add("Penjualan")

End Sub

Private Sub CbJenis_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
CbJenis.SelectedIndexChanged
    bulan.Enabled = False
    Select Case (CbJenis.Text)

```

```

        Case "Penjualan"
            bulan.Enabled = True
        Case "Pembelian"
            bulan.Enabled = True
        Case Else
            bulan.Enabled = False
    End Select
    tahun.Enabled = bulan.Enabled
End Sub

Private Sub PanelEx1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PanelEx1.Click

    End Sub
End Class

```

### • Form Pembelian

```

Public Class FormReport
    Inherits DevComponents.DotNetBar.Office2007Form
    Public status, kode As String

    Private Sub BtnView_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnView.Click
        If status = "1" Then

            Dim Nota As notaBeli
            Nota = New notaBeli
            Dim dt As DataTable
            dt = Me.Nota_BeliTableAdapter1.GetDataByNota(kode)
            Nota.SetDataSource(dt)
            Me.CrystalReportViewer1.ReportSource = Nota

            Exit Sub
        ElseIf status = "2" Then
            Dim Nota As notaJual
            Nota = New notaJual
            Dim dt As DataTable
            dt = Me.Nota_JualTableAdapter1.GetDataByNota(kode)
            Nota.SetDataSource(dt)
            Me.CrystalReportViewer1.ReportSource = Nota
            CrystalReportViewer1.PrintReport()
            Exit Sub
        Else
            Select (CbJenis.SelectedIndex)
                Case 0
                    Dim lap As lapkonsumen
                    lap = New lapkonsumen
                    Dim dt As DataTable
                    dt = Me.KonsumenTableAdapter1.GetData()
                    lap.SetDataSource(dt)
                    Me.CrystalReportViewer1.ReportSource = lap
                Case 1
                    Dim lap As lapsupplier

```

```

        lap = New lapsupplier
        Dim dt As DataTable
        dt = Me.SupplierTableAdapter1.GetData()
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case 2
        Dim lap As lapobat
        lap = New lapobat
        Dim dt As DataTable
        dt = Me.LapObatTableAdapter1.GetData()
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case 3
        Dim lap As lapbeli
        lap = New lapbeli
        Dim dt As DataTable
        dt =
Me.LaporanJualTableAdapter1.GetData(bulan.SelectedIndex + 1,
tahun.SelectedIndex + 2010)
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case 4
        Dim lap As lapJual
        lap = New lapJual
        Dim dt As DataTable
        dt =
Me.LaporanBeliTableAdapter1.GetData(bulan.SelectedIndex + 1,
tahun.SelectedIndex + 2010)
        lap.SetDataSource(dt)
        Me.CrystalReportViewer1.ReportSource = lap
    Case Else
        DevComponents.DotNetBar.MessageBoxEx.Show("Pilih
Jenis Laporan.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Select

    End If
End Sub

```

```

Private Sub FormReport_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
    CbJenis.Items.Add("Konsumen")
    CbJenis.Items.Add("Supplier")
    CbJenis.Items.Add("Obat")
    CbJenis.Items.Add("Pembelian")
    CbJenis.Items.Add("Penjualan")

```

```
End Sub
```

```

Private Sub CbJenis_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
CbJenis.SelectedIndexChanged
    bulan.Enabled = False
    Select Case (CbJenis.Text)
        Case "Penjualan"
            bulan.Enabled = True
        Case "Pembelian"

```

```

        bulan.Enabled = True
    Case Else
        bulan.Enabled = False
    End Select
    tahun.Enabled = bulan.Enabled
End Sub

Private Sub PanelEx1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PanelEx1.Click

    End Sub
End Class

```

### • Form penjualan

```

Public Class FormPenjualan
    Inherits DevComponents.DotNetBar.Office2007Form
    Dim idx As Integer = -1
    Private Sub get_code()
        Dim i As Integer = 1
        Dim found As Boolean = False
        While Not found And i < Integer.MaxValue
            If Not found And Me.PenjualanTableAdapter1.IsExist("'" +
Format(i, "0000000000")) = 0 Then
                found = True
                kode.Text = "'" + Format(i, "0000000000")
            End If
            i = i + 1
        End While
    End Sub

    Private Sub FormPenjualan_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
        'TODO: This line of code loads data into the 'Ds_Con.obat' table.
        You can move, or remove it, as needed.
        Me.ObatTableAdapter.Fill(Me.Ds_Con.obat)
        LvData.Columns.Add("No.")
        LvData.Columns.Add("Kode", 100)
        LvData.Columns.Add("Nama.", 220)
        LvData.Columns.Add("Jml.")
        LvData.Columns.Add("Harga.")

        get_code()

    End Sub

    Private Sub BtnCari_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnCari.Click
        If FormKonsumen.Visible = False Then
            FormKonsumen.Show()
            FormKonsumen.panelisi.Hide()
        End If
    End Sub

```

```

Private Sub CariData_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles CariData.TextChanged
    Dim dt As DataTable
    Dim dtcari As New Ds_ConTableAdapters.konsumenTableAdapter()

    dt = dtcari.GetDataByKode(CariData.Text)
    If dt.Rows.Count > 0 Then
        Me.DataCari.Text = "- Nama : " + dt.Rows(0).Item("nama") +
Chr(13) + Chr(10) _
        + "- Alamat : " + dt.Rows(0).Item("alamat") + Chr(13) +
Chr(10) _
        + "- No. Telp : " + dt.Rows(0).Item("telp")
        BtnSimpan.Enabled = LvData.Items.Count > 0 And
CariData.Text.Trim <> ""
    End If
End Sub

Private Sub BtnBaru_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBaru.Click
    If FormKonsumen.Visible = False Then
        FormKonsumen.Show()

    End If
End Sub

Private Sub cariobat_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles cariobat.TextChanged
    Me.ObatTableAdapter.FillByLikeKode(Me.Ds_Con.obat,
cariobat.Text.Trim)
    dataObat.Refresh()
End Sub

Private Sub BtnOatBaru_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles BtnOatBaru.Click
    If FormObat.Visible = False Then
        FormObat.Show()

    End If
End Sub

Private Sub jml_ValueChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles jml.ValueChanged, jml.TextChanged,
harga.TextChanged
    Btnpesan.Enabled = jml.Value.ToString <> "0" And
harga.Value.ToString <> "0"
End Sub

Private Sub Btnpesan_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Btnpesan.Click
    Dim item As ListViewItem
    item = New ListViewItem
    item.Text = LvData.Items.Count + 1
    item.SubItems.Add(kodeObat.Text.Trim)
    item.SubItems.Add(namaObat.Text.Trim)
    item.SubItems.Add(jml.Value.ToString)
    item.SubItems.Add(harga.Value.ToString)
    LvData.Items.Add(item)
    jml.Text = "0"

```

```

        harga.Text = "0"
        BtnSimpan.Enabled = LvData.Items.Count > 0 And CariData.Text.Trim
<> ""
    End Sub

    Private Sub LvData_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
LvData.SelectedIndexChanged

    End Sub

    Private Sub LvData_ItemSelectionChanged(ByVal sender As
System.Object, ByVal e As
System.Windows.Forms.ListViewItemSelectionChangedEventArgs) Handles
LvData.ItemSelectionChanged
        idx = e.ItemIndex
        BtnHapus.Enabled = True
    End Sub

    Private Sub BtnHapus_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnHapus.Click
        LvData.Items.RemoveAt(idx)
        idx = -1
        BtnHapus.Enabled = False
        BtnSimpan.Enabled = LvData.Items.Count > 0 And CariData.Text.Trim
<> ""
    End Sub

    Private Sub BtnSimpan_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnSimpan.Click
        Try
            If BtnSimpan.Text = "Simpan" Then
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menyimpan Data?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
                    Me.PenjualanTableAdapter1.InsertQuery(kode.Text.Trim,
CariData.Text.Trim, "1", CDate(Format(Now, "MM/dd/yyyy")))

                    For i = 0 To LvData.Items.Count - 1

Detail_penjualanTableAdapter1.InsertQuery(kode.Text.Trim,
LvData.Items.Item(i).SubItems(1).Text,
(LvData.Items.Item(i).SubItems(3).Text),
(LvData.Items.Item(i).SubItems(4).Text))

                    Next i

                    DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil disimpan.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)
                    If Cetak.Checked Then
                        'FormReport.Jenis.Visible = False
                        'FormReport.status = "2"
                        'FormReport.kode = kode.Text.Trim
                        'FormReport.ShowDialog()
                        With FormReport

```

```

        Dim Nota As notaBeli
        Nota = New notaBeli
        Dim dt As DataTable
        dt =
.Nota_BeliTableAdapter1.GetDataByNota(kode.Text.Trim)
        Nota.SetDataSource(dt)
        .CrystalReportViewer1.ReportSource = Nota
        .CrystalReportViewer1.PrintReport()

        End With

    End If
    BtnBatal_Click(sender, e)
End If

    End If
    Catch
        DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End Try
End Sub

    Private Sub BtnBatal_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBatal.Click
        LvData.Items.Clear()
        CariData.Clear()
        DataCari.Clear()
        get_code()
    End Sub

    Private Sub harga_ValueChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles harga.ValueChanged

    End Sub

    Private Sub dataObat_CellContentClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
dataObat.CellContentClick

    End Sub

    Private Sub PanelEx1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PanelEx1.Click

    End Sub
End Class

```



- **Form Dialogpass**

```
Imports System.Windows.Forms
Imports DevComponents.DotNetBar

Public Class DialogPass
    Inherits DevComponents.DotNetBar.Office2007Form
    Private Sub OK_Button_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles OK_Button.Click

        If UsersTableAdapter1.Login(LoginForm1.UsernameTextBox.Text.Trim,
lama.Text.Trim) > 0 Then
            UsersTableAdapter1.GantiPass(baru.Text.Trim,
LoginForm1.UsernameTextBox.Text.Trim)
            MessageBoxEx.Show("Password Berhasil Diganti..",
"Konfirmasi", MessageBoxButtons.OK, MessageBoxIcon.Information)
            FormMain.logout_Click(sender, e)
        Else
            MessageBoxEx.Show("Password Salah..", "Error",
MessageBoxButtons.OK, MessageBoxIcon.Error)
        End If
        Me.Close()

    End Sub

    Private Sub Cancel_Button_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Cancel_Button.Click
        Me.DialogResult = System.Windows.Forms.DialogResult.Cancel
        Me.Close()
    End Sub

    Private Sub lama_TextChanged(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles lama.TextChanged, baru.TextChanged
        OK_Button.Enabled = lama.Text.Trim <> "" And baru.Text.Trim <> ""
    End Sub

    Private Sub DialogPass_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load

    End Sub
End Class
```

- **Form Main**

```
Public Class FormMain
    Inherits DevComponents.DotNetBar.Office2007Form
    Private listForm As List(Of Form)
    Private Sub FormShow(ByVal Frm As Form)
        ' Frm.MdiParent = Me
        Frm.Show()
        '    Frm.BringToFront()

        If Frm.WindowState = FormWindowState.Minimized Then
            Frm.WindowState = FormWindowState.Normal
        End If

        If Not listForm.Contains(Frm) Then
            listForm.Add(Frm)
        Else
            Frm.Close()

        End If
    End Sub

    Private Sub FormMain_FormClosing(ByVal sender As Object, ByVal e As
System.Windows.Forms.FormClosingEventArgs) Handles Me.FormClosing
        If listForm.Count = 0 Then
            LoginForm1.Close()
            Exit Sub
        End If

        e.Cancel = True
        For Each frm In listForm
            frm.Close()
        Next
        listForm.Clear()
        LoginForm1.Close()
    End Sub

    Private Sub ButtonItem2_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)

    End Sub

    Private Sub FormMain_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        listForm = New List(Of Form)
    End Sub

    Private Sub ButtonItem5_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)

    End Sub

    Private Sub ButtonItem15_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)

    End Sub
```

```
Private Sub jenis_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles jenis.Click
    FormShow(FormJenis)
End Sub
```

```
Private Sub Exits_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Exits.Click
    If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk Menutup Program?", "Konfirmasi", MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then
        Close()
        If Not IsDBNull(LoginForm1) Then
            LoginForm1.Close()
        End If
    End If
End Sub
```

```
Private Sub konsumens_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles konsumens.Click
    FormShow(FormKonsumen)
End Sub
```

```
Private Sub obats_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles obats.Click
    FormShow(FormObat)
End Sub
```

```
Private Sub suppliers_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles suppliers.Click
    FormShow(FormSupplier)
End Sub
```

```
Private Sub users_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles users.Click
    FormShow(FormUsers)
End Sub
```

```
Private Sub pembelians_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles pembelians.Click
    FormShow(FormPembelian)
End Sub
```

```
Private Sub penjualans_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles penjualans.Click
    FormShow(FormPenjualan)
End Sub
```

```
Private Sub laporan_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles laporan.Click
    FormShow(FormReport)
End Sub
```

```
Public Sub logout_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles logout.Click
    If listForm.Count = 0 Then
        Hide()
    End If
End Sub
```

```

        LoginForm1.Show()
    Exit Sub
End If

For Each frm In listForm
    frm.Close()
Next
listForm.Clear()

Hide()
LoginForm1.Show()
End Sub

Private Sub ganti_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ganti.Click
    FormShow(DialogPass)
End Sub
End Class

```

- **LoginForm**

```

Imports DevComponents.DotNetBar

Public Class LoginForm1
    Inherits DevComponents.DotNetBar.Office2007Form
    ' TODO: Insert code to perform custom authentication using the
    provided username and password
    ' (See http://go.microsoft.com/fwlink/?LinkId=35339).
    ' The custom principal can then be attached to the current thread's
    principal as follows:
    '     My.User.CurrentPrincipal = CustomPrincipal
    ' where CustomPrincipal is the IPrincipal implementation used to
    perform authentication.
    ' Subsequently, My.User will return identity information encapsulated
    in the CustomPrincipal object
    ' such as the username, display name, etc.

    Private Sub OK_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles OK.Click
        Try
            Dim grant As Boolean = False
            If UsernameTextBox.Text.Trim.ToLower = "admin" And
PasswordTextBox.Text.Trim.ToLower = "admin" Then

                PasswordTextBox.Clear()
                FormMain.Show()
                Me.Hide()
                Exit Sub
            End If
            Dim jabatan As String
            If UsersTableAdapter1.Login(UsernameTextBox.Text.Trim,
PasswordTextBox.Text.Trim) > 0 Then

```

```

        grant = True
        Dim dt As DataTable
        Dim dtusers As New
Ds_ConTableAdapters.usersTableAdapter()

        dt = dtusers.GetDataByKode(UsernameTextBox.Text.Trim)
        If dt.Rows.Count > 0 Then
            ' Me.nama.Text = dt.Rows(0).Item("nama")

            jabatan = dt.Rows(0).Item("status")
        End If
    Else
        MessageBoxEx.Show("Login Gagal..", "Error",
        MessageBoxButtons.OK, MessageBoxIcon.Error)
        Exit Sub
    End If

    If grant = True Then
        PasswordTextBox.Clear()
        FormMain.Show()
        Me.Hide()
    End If
Catch
    MessageBoxEx.Show("Koneksi Database Errot", "Error",
    MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try

End Sub

Private Sub Cancel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Cancel.Click
    If Not IsDBNull(FormMain) Then FormMain.Close()
    Me.Close()
End Sub

Private Sub UsernameLabel_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles UsernameLabel.Click

End Sub
End Class

```

## • Form Opname Obat

```
Public Class FormOpname
    Inherits DevComponents.DotNetBar.Office2007Form
    Private Sub FormOpnae_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        'TODO: This line of code loads data into the 'Ds_Con.obat1'
table. You can move, or remove it, as needed.
        Me.Obat1TableAdapter.Fill(Me.Ds_Con.obat1)
        'TODO: This line of code loads data into the 'Ds_Con.obat' table.
You can move, or remove it, as needed.
        Me.ObatTableAdapter.Fill(Me.Ds_Con.obat)

        'TODO: This line of code loads data into the 'Ds_Con.obat' table.
You can move, or remove it, as needed.
        Me.ObatTableAdapter.Fill(Me.Ds_Con.obat)

        'TODO: This line of code loads data into the 'Ds_Con.obat' table.
You can move, or remove it, as needed.
        Me.ObatTableAdapter1.Fill(Me.Ds_Con.obat)
        'TODO: This line of code loads data into the 'Ds_Con.obat' table.
You can move, or remove it, as needed.

    End Sub

    Private Sub cariobat_TextChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles cariobat.TextChanged
        Me.ObatTableAdapter.FillByLikeKode(Me.Ds_Con.obat,
cariobat.Text.Trim)
    End Sub

    Private Sub BtnSimpan_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnSimpan.Click
        Try
            If BtnSimpan.Text = "Simpan" Then
                If DevComponents.DotNetBar.MessageBoxEx.Show("Yakin Untuk
Menyimpan Data?", "Konfirmasi", MessageBoxButtons.YesNo,
MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes Then

                    Me.ObatTableAdapter1.UpdateOpname(CInt(opname.Text.Trim()),
CInt(jml.Text.Trim()), tempKode.Text, tempKode.Text)
                    DevComponents.DotNetBar.MessageBoxEx.Show("Data Telah
Berhasil disimpan.", "Confirm", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation)

                    BtnBatal_Click(sender, e)
                End If
            End If
        Catch
            DevComponents.DotNetBar.MessageBoxEx.Show("Telah Terjadi
Error.", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
        End Try
    End Sub
End Class
```

```

End Sub

Private Sub BtnBatal_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles BtnBatal.Click
    opname.Value = 0
    jml.Value = 0
    cariobat.Text = ""
    BtnSimpan.Enabled = False

    Me.ObatTableAdapter.Fill(Me.Ds_Con.obat)
    Me.Obat1TableAdapter.Fill(Me.Ds_Con.obat1)
    'TODO: This line of code loads data into the 'Ds_Con.obat' table.
    You can move, or remove it, as needed.

End Sub

Private Sub tempKode_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles tempKode.Click

End Sub

Private Sub opname_ValueChanged(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles opname.ValueChanged
    BtnSimpan.Enabled = (opname.Value <> 0) And (tempKode.Text.Trim
<> "")
End Sub

Private Sub DataGridViewX1_CellClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
DataGridViewX1.CellClick, DataGridViewX1.CellContentDoubleClick,
DataGridViewX1.CellContentClick

End Sub

Private Sub dataObat_CellContentClick(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles
dataObat.CellContentClick

End Sub

Private Sub ReflectionLabell1_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ReflectionLabell1.Click

End Sub
End Class

```