BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi Sistem

Implementasi sistem merupakan pengaplikasian perangkat lunak, perangkat keras serta *database* yang digunakan. Pada tahapan ini akan dilakukan implementasi meliputi lingkungan implementasi, batasan implementasi, implemetasi kelas serta konfigurasi dari kebutuhan sistem untuk dapat menjalankan aplikasi perangkat lunak yang telah dibangun.

5.1.1 Batasan Implementasi

Batasan implementasi sistem menyangkut beberapa perihal di bawah ini:

1. Hanya admin terdaftar yang dapat mengakses masing-masing bagian dalam sistem sesuai modulnya.
2. Data *input* merupakan masukan berupa data *real*, yaitu *string* dan angka.
3. Sementara sebagai *output*, akan dihasilkan data berupa laporan.
4. Konfigurasi basis data harus dilakukan secara manual dengan memanfaatkan *file Dump* SQL.
5. Implementasi dilakukan pada komputer lokal di mana *server* basis data dan *server* aplikasi berada pada komputer yang sama.

5.1.2 Lingkungan Implementasi

Lingkungan implementasi merupakan suatu kondisi dan persyaratan agar perangkat lunak yang dibangun mampu diaplikasikan dengan baik. Berikut ini

merupakan lingkungan implementasi yang terbagi menjadi dua, yaitu Lingkungan Perangkat Lunak dan Lingkungan Perangkat Keras.

5.1.2.1 Implementasi Perangkat Lunak

Spesifikasi lingkungan perangkat lunak yang digunakan dalam pengembangan adalah sebagai berikut:

1. Sistem Operasi Microsoft Windows XP *Professional Service Pack* 2
2. Bahasa Pemrograman *Software Development Kit* (SDK) Java 2 *Standard Edition* 6 (JDK – JRE 1.6.0_24)
3. Perangkat Pemrograman Netbeans 6.9.1-ml-windows
4. DBMS MySQL *Version* 5.0.51b
5. *Database Server* AppServer *Version* 2.5.10 for Windows
6. Laporan menggunakan iReport-3.7.5 *for* Windows
7. Java *Libraries*

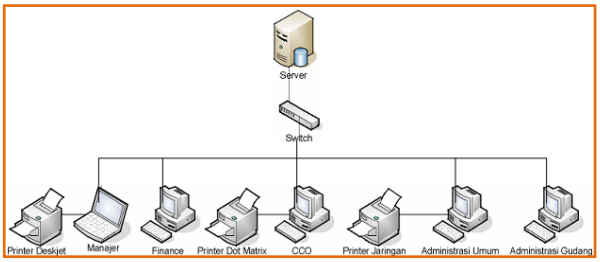
5.1.2.2 Implementasi Perangkat Keras

Sedangkan lingkungan perangkat keras yang digunakan dalam pengembangan perangkat lunak, baik *Client* maupun *Server* memiliki spesifikasi sebagai berikut:

1. *Processor* Intel Core i3 2.4 GHz (*minimum* Intel Dual Core 2.2. GHz)
2. *Memori* DDR3 1 GB
3. *Harddisk* SATA 160 GB
4. VGA ATI Radeon 512 MB
5. Perangkat keluaran berupa *Monitor* 14”
6. Perangkat masukan berupa *mouse* dan *keyboard*

5.1.2.3 Implementasi Jaringan

Jaringan yang dipakai adalah jaringan yang berarsitektur *Client-Server*, dimana *server* dapat diakses oleh masing-masing bagian pada komputer yang berbeda. Selain itu juga dilakukan pen-*share*-an *printer* dan perangkat lainnya.



Gambar 5.1 Skema Jaringan dan Pengaksesan Sistem

Langkah-langkah untuk mengkonfigurasi IP komputer yang akan bertindak sebagai SERVER di Windows XP:

1 ***Start*** > ***Control Panel***

   a. Pilih *Network Connection*

2 Akan tampil *window Network Connections*

   a. Klik kanan pada *icon* ***Local Area Connection***

   b. Kemudian klik ***Properties***

3 Akan tampil *window Local Area Connection Properties*

   a. Klik *Internet Protocol* (TCP/IP)

   b. Klik tombol *Properties*

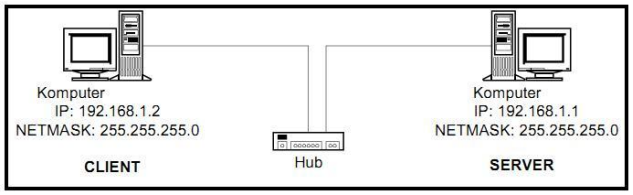
4 Akan tampil *window Internet Protocol* (TCP/IP) *Properties*

   a. Klik *Use the following IP Address*

   b. Misal, Ketikan IP *address*: 192.168.1.1

   c. Ketikan *Subnet mask*: 255.255.255.0

d. Klik tombol OK



Gambar 5.2 Konfigurasi *Client-Server*

Lakukan langkah-langkah yang sama terhadap komputer *Client*, yang membedakan adalah *IP Address*-nya.

5.1.3 Implementasi Kelas

Hasil perancangan kelas selanjutnya diimplementasikan secara fisik ke dalam file berekstensi .java dengan menggunakan konsep pemrograman berorientasi objek dengan bahasa pemrograman Java.

Tabel 5.1 Implementasi Kelas

No.	Kelas	Nama File Fisik
1	*User Interface*	InManagementView.java
2	*About Box*	InManagementAboutBox.java
3	Fungsi Aplikasi	InManagementApp.java
4	*Login*	LoginForm.java
5	Pilih Periode	PilihPeriode,java
6	*Admin*	Admin.java
7	Barang	Barang,java
8	Gudang	Gudang.java
9	Kategori	Kategori.java
10	Pencarian	Pencarian.java
11	Pelanggan	Pelanggan.java
12	Pemasok	Pemasok.java
13	Penjualan	Penjualan.java
14	Pencarian Transaksi	PencarianTrans.java
15	Pembelian	Pembelian.java
16	Stok	IsiGudang.java
17	Penyesuaian	Penyesuaian.java
18	Transfer Barang	Transfer.java
19	*Database Package*	SettingUI.java

Adapun *libraries* tambahan yang digunakan untuk membangun perangkat lunak ini adalah sebagai berikut:

1. DBPaket-5.0

   Library ini digunakan untuk melakukan setting *database.*

2. MySQL *Connector* Java *version* 5.1.6

   Library ini digunakan untuk melakukan koneksi (konektor) antara bahasa pemrograman Java dengan basis data MySQL.

3. JCalendar 1.3.2

   Library ini digunakan untuk melakukan *input* dan pemilihan tanggal, bulan dan tahun.

4. AppFramework 1.0.3

   Library ini digunakan untuk membangun program menggunakan prinsip pemodelan MVC (*Model, View, and Control*)

5. JasperReports-3.7.5

   Library ini digunakan untuk membuat laporan dalam tampilan PDF.

5.1.4 Implementasi Basisdata

Pada pembangunan aplikasi ini dibangun *database* agar tercipta sistem informasi yang terintegrasi dan dinamis. Selain itu pembuatan dan implementasi *database* merupakan langkah yang memperlihatkan kesesuaian perancangan pada tahapan analisis dan perancangan terhadap pembangunan aplikasi. Berikut adalah basis data yang dibuat menggunakan bahasa SQL (MySQL) dalam Aplikasi Manajemen Inv*entory Multi Warehouse* Menggunakan Metode *Economic Order Quantity* (EOQ):

```
-- Database: `inventory`
-- --------------------------------------------------------
CREATE TABLE `admin` (
  `kd_user` varchar(4) NOT NULL,
  `user_name` varchar(12) NOT NULL,
  `user_pass` varchar(12) NOT NULL,
  `bagian` varchar(1) NOT NULL COMMENT '0=master, 1=gudang,
        2=pembelian, 3=penjualan',
  PRIMARY KEY  (`kd_user`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `barang` (
  `kd_brg` varchar(4) NOT NULL,
  `nm_brg` varchar(25) NOT NULL,
  `kd_kat` varchar(3) NOT NULL,
  `harga_beli` double NOT NULL,
  `harga_jual` double NOT NULL,
  `qty_tahun` double NOT NULL default '0',
  `biy_pesan` double NOT NULL default '0',
  `biy_simpan` double NOT NULL default '0',
  `qty_hari_mak` double NOT NULL default '0',
  `qty_eoq` double NOT NULL default '0',
  `lead_time` double NOT NULL default '0',
  `rop_safety` double NOT NULL default '0',
  PRIMARY KEY  (`kd_brg`),
  KEY `kd_kat` (`kd_kat`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `barang_digudang` (
  `kd_brg` varchar(4) NOT NULL,
  `kd_gd` varchar(2) NOT NULL,
  `jumlah` double NOT NULL,
  PRIMARY KEY  (`kd_brg`,`kd_gd`),
  KEY `kd_gd` (`kd_gd`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `gudang` (
  `kd_gd` varchar(2) NOT NULL,
  `nm_gd` varchar(25) NOT NULL,
  `jenis` char(1) NOT NULL COMMENT '0=pokok, 1=pengecer, 2=distribusi',
  PRIMARY KEY  (`kd_gd`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `kategori` (
  `kd_kat` varchar(3) NOT NULL,
  `nm_kat` varchar(25) NOT NULL,
```

```
  PRIMARY KEY  (`kd_kat`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `pelanggan` (
  `kd_plg` varchar(4) NOT NULL,
  `nm_plg` varchar(25) NOT NULL,
  `telp` varchar(12) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  PRIMARY KEY  (`kd_plg`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `pemasok` (
  `kd_pemasok` varchar(4) NOT NULL,
  `nama` varchar(25) NOT NULL,
  `telp` varchar(12) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  PRIMARY KEY  (`kd_pemasok`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `pembelian` (
  `no_beli` varchar(4) NOT NULL,
  `tgl` date NOT NULL,
  `kd_pemasok` varchar(4) NOT NULL,
  `kd_gd` varchar(2) NOT NULL,
  PRIMARY KEY  (`no_beli`),
  KEY `kd_pemasok` (`kd_pemasok`),
  KEY `kd_gd` (`kd_gd`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `pembelian_detil` (
  `no_beli` varchar(4) NOT NULL,
  `kd_brg` varchar(4) NOT NULL,
  `harga_beli` double NOT NULL,
  `jumlah` double NOT NULL,
  PRIMARY KEY  (`no_beli`,`kd_brg`),
  KEY `kd_brg` (`kd_brg`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `penjualan` (
  `no_jual` varchar(4) NOT NULL,
  `tgl` date NOT NULL,
  `kd_plg` varchar(4) NOT NULL,
  `kd_gd` varchar(4) NOT NULL,
  PRIMARY KEY  (`no_jual`),
  KEY `kd_plg` (`kd_plg`),
  KEY `kd_gd` (`kd_gd`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `penjualan_detil` (
  `no_jual` varchar(4) NOT NULL,
  `kd_brg` varchar(4) NOT NULL,
  `harga_jual` double NOT NULL,
  `jumlah` double NOT NULL,
  PRIMARY KEY  (`no_jual`,`kd_brg`),
  KEY `kd_brg` (`kd_brg`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `penyesuaian` (
  `no_pen` varchar(4) NOT NULL,
  `tgl` date NOT NULL,
  `kd_gd` varchar(2) NOT NULL,
  `status` char(1) NOT NULL COMMENT '0=rusak, 1=Hilang, 2=Expire, 3=Lain',
  PRIMARY KEY  (`no_pen`),
  KEY `kd_gd` (`kd_gd`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `penyesuaian_detil` (
  `no_pen` varchar(4) NOT NULL,
  `kd_brg` varchar(4) NOT NULL,
  `jumlah_data` double NOT NULL,
  `jumlah_fisik` double NOT NULL,
  PRIMARY KEY  (`no_pen`,`kd_brg`),
  KEY `kd_brg` (`kd_brg`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `transfer_barang` (
  `no_transfer` varchar(4) NOT NULL,
  `tgl` date NOT NULL,
  `kd_gd_from` varchar(2) NOT NULL,
  `kd_gd_to` varchar(2) NOT NULL,
  PRIMARY KEY  (`no_transfer`),
  KEY `kd_gd_from` (`kd_gd_from`),
  KEY `kd_gd_to` (`kd_gd_to`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
-- --------------------------------------------------------
CREATE TABLE `transfer_barang_detil` (
  `no_transfer` varchar(4) NOT NULL,
  `kd_brg` varchar(4) NOT NULL,
  `jumlah` double NOT NULL,
  PRIMARY KEY  (`no_transfer`,`kd_brg`),
  KEY `kd_brg` (`kd_brg`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

5.1.5 Implementasi Antar Muka

Antarmuka dalam sebuah aplikasi adalah hal yang sangat penting, aplikasi *desktop* yang menggunakan bahasa JAVA ini, implementasi antarmuka dilakukan dengan sebuah halaman yang berekstensi .java. Implementasi antarmuka dari halaman yang dibuat dapat dilihat pada tabel di bawah ini.

Tabel 5.2 Implementasi Antar Muka

No.	Nama	Tipe	Keterangan
1	InManagementView.java	*Form*	Tampilan Menu Utama *Master* Admin
2	InventoryView.java	*Form*	Tampilan Utama Bagian Gudang
3	TransaksiView.java	*Form*	Tampilan Utama Bagian Pembelian dan Penjualan
4	ManagementView.java	*Form*	Tampilan Proses Transfer Barang dan Penyesuaian
5	LoginForm.java	*Form*	Menu *Login*
6	DB Setting	Paket *Library*	Fungsi koneksi *Database*
7	ReportFactory.java	*Form*	Tampilan Laporan
8	EOQFactory.java	Fungsi	Menghitung pemesanan berdasarkan EOQ Method

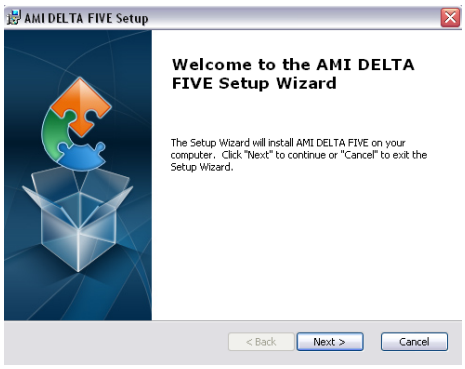
5.1.6 Implementasi Program

Dalam mengimplementasikan program ke dalam komputer, maka sebelumnya harus dipersiapkan segala kebutuhan yang mendukung agar perangkat lunak dapat berjalan dengan baik pada *platform* teknologi yang kita pakai. Berikut ini adalah beberapa hal yang harus dipersiapkan terlebih dahulu, yaitu:

1. Instalasi AppServer atau XAMPP (disarankan untuk menggunakan AppServer) terbaru untuk aplikasi *database*
2. Lakukan *Dump* SQL untuk mengimport *database* ke dalam AppServer

3. Lalu lakukan instalasi SetUp AMI DELTA FIVE V1.0, sebagai berikut:

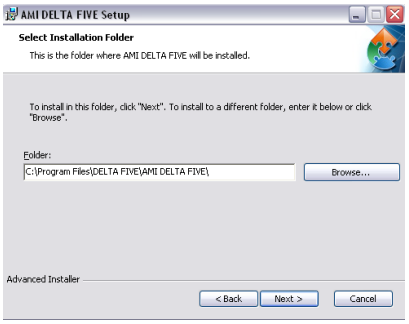
   a. Klik ganda *File Setup* AMI DELTA FIVE, maka akan tampak halaman "*Welcome*", seperti gambar di bawah ini.



Gambar 5.3 Tampilan *Welcome*

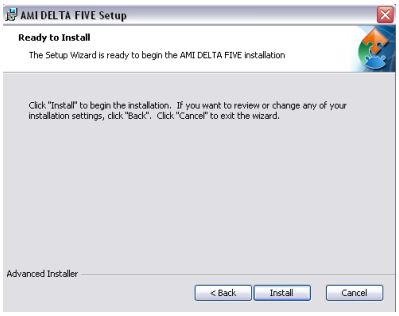
   b. Kemudian klik *Next*.

   c. Maka akan muncul tampilan pemilihan lokasi instalasi program, klik "*Browse*" untuk menentukan lokasi. Kali ini kita biarkan saja pada kondisi *default*. Kemudian klik *Next*



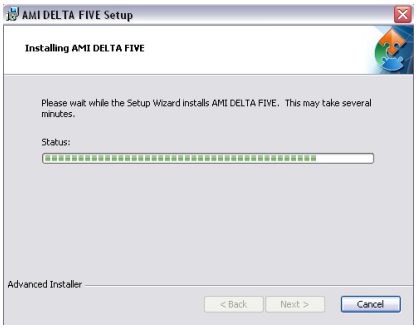
Gambar 5.4 Lokasi Instalasi

   d. Jika tampilan berikut muncul, berarti Program telah siap untuk di install. Klik *install*.



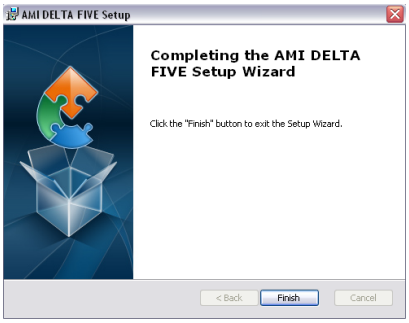
Gambar 5.5 *Ready to Install*

e. Tunggulah proses instalasi sampai selesai



Gambar 5.6 *Setup* AMI Delta Five

f. Klik *Finish*.

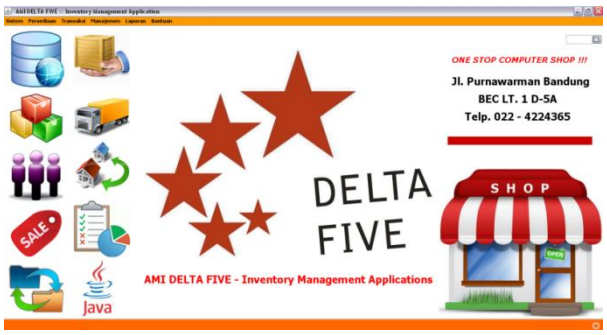


Gambar 5.7 *Finish*

5.1.7 Penggunaan Program

1. Menu Utama

Merupakan menu utama yang mengakses semua fungsi dan modul yang ada pada aplikasi manajemen *inventory multi warehouse* ini.



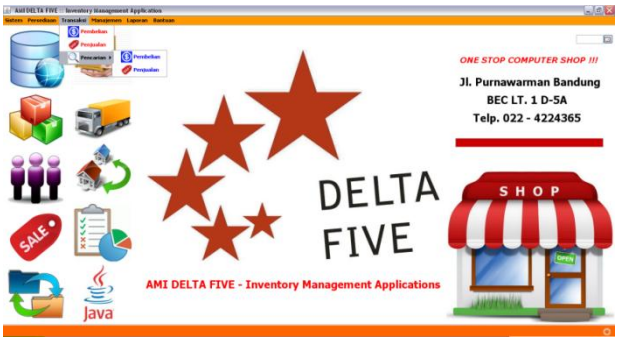
Gambar 5.8 Menu Utama *Master* Admin

Merupakan tampilan menu dari modul persediaan/inventori. Dimana didalamnya terdapat 6 sub menu, yaitu Admin, Gudang, Kategori, Barang, Pemasok dan Pelanggan.



Gambar 5.9 Menu Persediaan

Merupakan tampilan menu dari modul transaksi. Dimana didalamnya terdapat 3 sub menu, yaitu Pembelian, Penjualan dan Pencarian Transaksi.



Gambar 5.10 Menu Transaksi

Merupakan tampilan menu dari modul management. Dimana didalamnya terdapat 4 sub menu, yaitu Penyesuaian Stok, Transfer Barang, Stok, dan Pencarian Manajemen Barang.



Gambar 5.11 Menu Manajemen

Merupakan tampilan menu dari modul laporan, yang didalamnya terdapat semua laporan-laporan baik persediaan barang, transaksi maupu manajemen barang.



Gambar 5.12 Menu Laporan

Merupakan *form* untuk menginputkan, menambah, mengubah dan menghapus serta melihat isi data *admin* atau *user*.



Gambar 5.13 *Form Administrator*

2. Menu Bagian Gudang

Merupakan *form* untuk menambah, menghapus, mengubah dan menyimpan serta melihat data gudang.



Gambar 5.14 *Form* Gudang

Merupakan form untuk menambah, mengubah dan menghapus serta melihat isi data kategori barang.



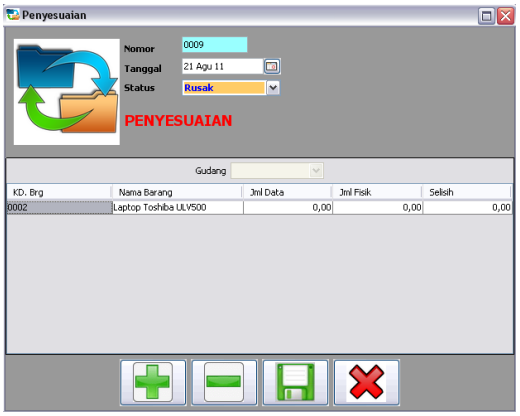
Gambar 5.15 *Form* Kategori Barang

Merupakan *form* untuk menambah, mengubah, menghapus dan melihat isi data barang serta perhitungan EOQ.



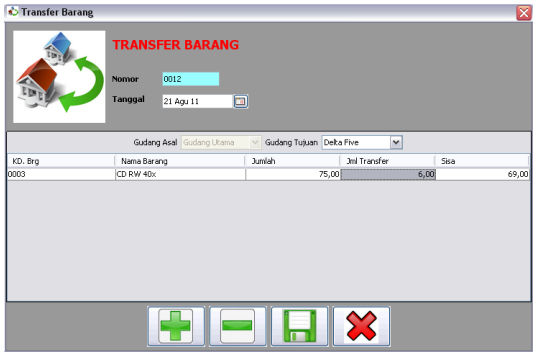
Gambar 5.16 *Form* Barang dan EOQ

Merupakan *form* untuk menambah, mengubah dan menghapus data penyesuaian stok barang yang ada di dalam gudang.



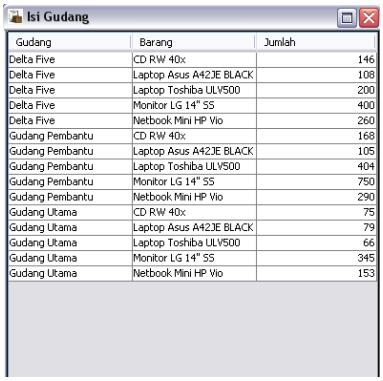
Gambar 5.17 *Form* Penyesuaian

Merupakan *form* untuk menambah, mengubah dan menghapus data transfer barang antar gudang.



Gambar 5.18 *Form* Transfer Barang

Merupakan *form* yang menampilkan data stok barang, stok barang ditampilkan dalam bentuk perkategori, yaitu berdasarkan gudang.



Gambar 5.19 Stok

3. Menu Bagian Pembelian

Merupakan *form* untuk menambah, mengubah dan menghapus serta melihat isi data mengenai pemasok.



Gambar 5.20 *Form* Pemasok

Merupakan *form* untuk menambah, mengubah dan menghapus serta melihat data transaksi pembelian barang kepada *supplier*.



Gambar 5.21 *Form* Pembelian

4. Menu Bagian Penjualan

Merupakan *form* untuk menambah, mengubah dan menghapus serta melihat data pelanggan dari Delta Five.



Gambar 5.22 *Form* Pelanggan

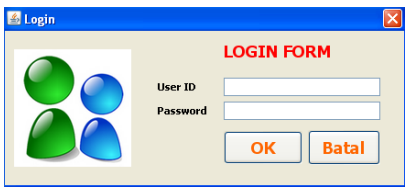
Merupakan *form* untuk menambah, mengubah dan menghapus data transaksi penjualan yang dilakukan.



Gambar 5.23 *Form* Penjualan

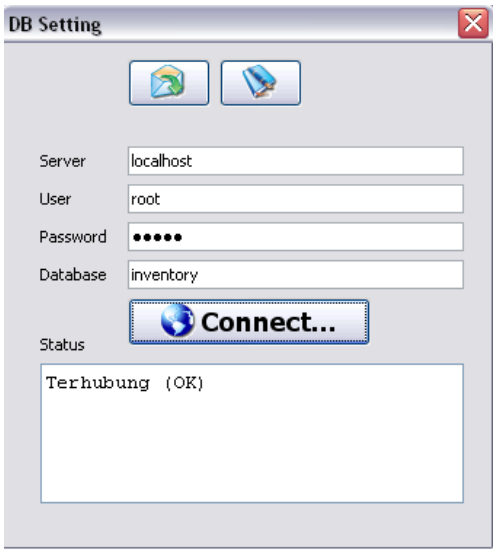
5. Menu Akses dan *Setting*

Merupakan *form* untuk melakukan *login* ke dalam sistem, untuk dapat mengakses sistem silahkan masukkan *username* dan *password*: *master*, pembelian, penjualan, dan gudang.



Gambar 5.24 *Form Login*

Merupakan *form* untuk melakukan *setting* konfigurasi *Database*, inputkan *localhost* atau IP *Address* dari *server*.



Gambar 5.25 DB *Setting*

6. Laporan

Merupakan tampilan dari laporan Daftar Barang, laporan ditampilkan berdasarkan kategori barang.



Gambar 5.26 Laporan Daftar Barang

Merupakan tampilan dari laporan Stok Barang, ditampilkan secara total pergudang.



DELTA FIVE
Jl. Purnawarman Bandung
BEC LT. 1 D-5A
Telp. 022 - 4224365

Laporan Stok
Per Gudang

Minggu 21 Agustus 2011

Delta Five

Kode Barang	Nama Barang	Stok
0001	Laptop Asus A42JE	108,00
0002	Laptop Toshiba ULV500	200,00
0003	CD RW 40x	146,00
0004	Netbook Mini HP Vio	260,00
0006	Monitor LG 14" SS	400,00

Gambar 5.27 Laporan Stok

Merupakan tampilan laporan untuk memperlihatkan laporan semua Pelanggan tetap dari Delta Five



DELTA FIVE
Jl. Purnawarman Bandung
BEC LT. 1 D-5A
Telp. 022 - 4224365

Daftar Pelanggan
Semua pelanggan

Minggu 21 Agustus 2011

Kode	Nama Pelanggan	Telp	Alamat
0001	Dyna Comp	0224537284	Jl. ABC No. 112
0002	Kurniawan	081736483748	Jl. Pahlawan 8
0003	PT. Anortasi	0225407756	Jl. Lengkong Besar No. 56
0004	Starcom	022450676	Jl. Otto Iskandardinata No. 370
0005	Esson Sihombing	085722800971	Jl. Kopo Cirangrang Timur II No. 11A/48
0006	SysCom	0227569003	Jl. Wahid Hasyim (Kopo) 105 A Bandung

Gambar 5.28 Laporan Pelanggan

Merupakan tampilan untuk memperlihatkan Laporan Pemasok barang kepada Delta Five



DELTA FIVE
Jl. Purnawarman Bandung
BEC LT. 1 D-5A
Telp. 022 - 4224365

Daftar Pemasok
Semua pemasok

Minggu 21 Agustus 2011

Kode	Nama Pemasok	Telp	Alamat
0001	ASUS Center	0222507678	Jl. Lengkong Besar No. 3A Bandung
0002	TreComp	02274385836	BEC LT.2
0003	Plaza Komputer	085722800971	Plaza IBCC LT. 3
0004	ASUStek Service Center	0216126555	Mangga Dua Mal Ruko No. 25 Jakarta
0005	Setra Komputer	0226776901	Plaza IBCC

Gambar 5.29 Laporan Pemasok

Merupakan tampilan laporan transaksi pembelian per nomor transaksinya, dimana laporan disajikan dalam bentuk per-periode.



DELTA FIVE

DELTA FIVE Jl. Purnawarman Bandung
BEC LT. 1 D-5A
Telp. 022 - 4224365

Laporan Pembelian

Minggu 21 Agustus 2011

Periode: 01/07/2011 sampai 31/07/2011

No Beli : 0006 Tanggal : 09/07/2011 Pemasok : ASUS Center

Detil barang

KD Barang	Nama Barang	Harga	Jumlah	Subtotal
0001	Laptop Asus A42JE BLACK	1.000.000,00	5,00	5.000.000,00
			Total	5.000.000,00

No Beli : 0007 Tanggal : 14/07/2011 Pemasok : TreComp

Detil barang

KD Barang	Nama Barang	Harga	Jumlah	Subtotal
0001	Laptop Asus A42JE BLACK	5.045.000,00	3,00	15.135.000,00
			Total	15.135.000,00

Gambar 5.30 Laporan Pembelian

5.2 Pengujian Sistem

Pengujian ini dilakukan dengan cara menggunakan perangkat lunak untuk menguji performansi terhadap beberapa parameter uji, dengan menggunakan beberapa *dataset* yang telah disusun sedemikian rupa sesuai dengan skenario yang akan dibahas kemudian.

5.2.1 Tujuan Pengujian

Terdapat beberapa hal yang menjadi tujuan pelaksanaan pengujian terhadap perangkat lunak, yaitu:

1. Memeriksa kesesuaian hasil implementasi perangkat lunak dengan spesifikasi kebutuhannya
2. Memeriksa kesesuaian hasil implementasi dengan hasil analisis dan perancangan perangkat lunak.
3. Melakukan analisis terhadap hasil pengujian
4. Melakukan uji terhadap unit/modul, integrasi, validitas dan kesesuaian sistem dengan perancangan dan implementasi.

5.2.2 Rencana Pengujian

Pengujian dilakukan melalui menggunakan data uji yang dimasukkan oleh admin (*master*, gudang, pembelian, dan penjualan). Berikut adalah rencana pengujian yang dilakukan:

Tabel 5.3 Rencana Pengujian

Item Uji	Deskripsi	Form	Jenis Pengujian
Login	Melakukan *login* ke dalam sistem	LoginForm.java	*Black Box*
Pendaftaran *User*	Melakukan penambahan *User*	Admin.java	*Black Box Form*
Pembelian	Melakukan transaksi pembelian	Pembelian.java	*Black Box*
Penjualan	Melakukan transaksi penjualan	Penjualan.java	*Black Box*
EOQ	Melakukan cek perhitungan dan komparasi dengan perhitungan manual	Barang.java	*Black Box*
Transfer Barang	Melakukan transfer antar gudang	Transfer.java	*Black Box*

5.2.3 Kasus dan Hasil Pengujian

Pada bagian ini, *item-item* yang telah ditentukan di atas akan diujikan. Kasus-kasus dalam pengujian akan dipaparkan berikut dengan hasil pengujian tersebut.

5.2.3.1 Pengujian Validitas *Login*

Pada pengujian *login*, data antara *username* dan *password* harus sesuai, untuk pengujian diasumsikan semua pengguna dengan data *username* dan *password* inputan “master” adalah benar, selain dari itu salah. Pada pengujian ini akan diperlihatkan pengujian pada tingkatan Validitas dari *Black Box*.

Tabel 5.4 Pengujian *Login*

Data Masukan	Skenario	Pengamatan	Kesimpulan
Username: master *Password*: master	Ketikkan ke dalam *Textbox Username* dan *Textbox Password*	Inputan sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Klik OK	Masuk ke halaman selanjutnya	Tombol OK sesuai	[X] Diterima [ ] Ditolak
Data *Login User* salah *Username* : master *Password*: 123	Tidak dapat *Login* dan menampilkan pesan "Gagal Login!"	*User* tidak dapat *login* dan menampilkan pesan "Gagal Login!"	[X] Diterima [ ] Ditolak

5.2.3.2 Pengujian *Form* Pendaftaran *User*

Pada pengujian tingkat *Form*, yang perlu diperhatikan adalah apakah *form* yang ditampilkan sesuai dengan apa yang kita rancang dan harapkan. Tentunya data yang diinputkan sesuai atau tidak.

Tabel 5.5 Pengujian Pendaftaran *User*

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan lengkap	*Form* tambah *user* tampil	*Form* tampil sesuai rancangan awal	[X] Diterima [ ] Ditolak
Masukan lengkap	Data tersimpan	Data tersimpan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan tidak lengkap	Keluar pesan peringatan	Keluar pesan peringatan dan data tidak tersimpan	[X]Diterima [ ] Ditolak

5.2.3.3 Pengujian Pembelian

Pada pengujian pembelian, data inputan harus sesuai, untuk pengujian diasumsikan semua inputan adalah benar. Pada pengujian ini akan diperlihatkan pengujian pada tingkatan Integrasi data dari *Black Box*.

Tabel 5.6 Pengujian Pembelian

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan lengkap	Data tersimpan	Data tersimpan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan tidak lengkap	Keluar pesan peringatan	Keluar pesan peringatan dan data tidak tersimpan	[X]Diterima [ ] Ditolak

5.2.3.4 Pengujian Penjualan

Pada pengujian penjualan, data inputan harus sesuai, untuk pengujian diasumsikan semua inputan adalah benar. Pada pengujian ini akan diperlihatkan pengujian pada tingkatan Integrasi data dari *Black Box*.

Tabel 5.7 Pengujian Penjualan

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan lengkap	Data tersimpan	Data tersimpan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan tidak lengkap	Keluar pesan peringatan	Keluar pesan peringatan dan data tidak tersimpan	[X]Diterima [ ] Ditolak

5.2.3.5 Pengujian Transfer Barang

Pada pengujian transfer barang, data inputan harus sesuai, untuk pengujian diasumsikan semua inputan adalah benar. Pada pengujian ini akan diperlihatkan pengujian pada tingkatan Integrasi data dari *Black Box*.

Tabel 5.8 Pengujian Transfer

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan lengkap	Data tersimpan	Data tersimpan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Skenario**	**Pengamatan**	**Kesimpulan**
Masukan tidak lengkap	Keluar pesan peringatan	Keluar pesan peringatan dan data tidak tersimpan	[X]Diterima [ ] Ditolak

5.2.3.6 Pengujian Perhitungan EOQ

Pengujian sistem dilakukan dengan cara menghitung pemesanan bahan baku yang paling ekonomis dan periode pemesanannya melalui perhitungan dengan metode EOQ (*Economic Order Quantity*) yang telah dilakukan oleh program.

Contoh:

> Delta Five membutuhkan Harddisk 1000 Paket/tahun, dimana biaya untuk tiap unitnya adalah Rp. 100,-. Biaya pemesanan yang dibebankan Rp. 1000,-/pesanan. Sementara itu untuk biaya penyimpanan dihitung 40% dari nilai persediaan. Waktu yang dibutuhkan untuk melakukan pengiriman barang adalah 5 hari. Dengan ketentuan Toko buka setiap hari selama setahun (365 hari). Dan bagaimana jika ternyata kebutuhan akan barang bukan 2,7

akan tetapi 5. Maka berapa nilai pemesanan paling ekonomis (EOQ), ROP, dan ROPs?

Diketahui:

A : Kebutuhan = 1000 paket/tahun

S : Biaya Pemesanan = Rp. 1000,-

i : Biaya Simpan = 40% = 0.4

LT : 5 hari

C : Biaya/Unit = Rp. 100,-

1 tahun = 365 hari

Ditanya :

1. Berapa Nilai EOQ?
2. ROP?
3. ROPs?

Jawab:

H = i x C = 0.4 x 100 = 40

Maka, $Q = \sqrt{2AS/H}$

$Q = \sqrt{2 \times 1000 \times 1000 / 40}$

$Q = \sqrt{25000}$

Q = 158.11 (pembulatan keatas 158) unit/pesan

Q = 1000 / 365 = 2,74

→ **ROP** = 2,74 x 5 = 13,7 = **14 unit** (periode pesan)

Jika ternyata kebutuhan akan barang bukan 2,7 akan tetapi 5, maka akan terjadi yang disebut dengan kehabisan stok pada hari selanjutnya. Oleh karena itu perhitungan ROP yang baru adalah sebagai berikut:

ROPs = ROP + (Qmaks – Qr) x LT

ROPs = 13,7 + (5 – 2,7) x 5

ROPs = 13,7 + 11,5 = **25,2 (25) unit**

Kesimpulan:

Apabila jumlah stok sudah mencapai jumlah 25 unit, maka saatnya untuk melakukan pengadaan kembali terhadap barang, yaitu sejumlah 158 unit, sesuai dengan kalkulasi Nilai EOQ.

5.2.4 Kesimpulan Hasil Pengujian

Pengujian yang telah dilakukan merupakan beberapa proses yang terdapat dalam aplikasi. Berdasarkan hasil pengujian diatas dapat ditarik kesimpulan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil yang diharapkan. Namun demikian pengujian tersebut dapat dikatakan belum sempurna, dikarenakan hanya dilakukan pada satu sisi pengujian. Dari semua yang telah dilakukan dalam pengujian ini diharapkan dapat mewakili pengujian fungsi yang lainnya. Secara garis besar dari pengujian dan hasil implementasi perangkat lunak yang dibangun sudah sesuai dengan spesifikasi kebutuhan dan hasil perancangan perangkat lunak.