# BAB V

# IMPLEMENTASI DAN PENGUJIAN SISTEM

## 5.1 Implementasi

Implementasi prototipe aplikasi *inventory* obat ini dilakukan menggunakan bahasa pemrograman Java dengan penggunaan *Java Development Kit* 6.9 (JDK6.9) dan menggunakan *tools* pembangun *Netbeans IDE 7.1.2*. Basis data yang digunakan adalah *MySQL*. Implementasi dan pengujian sepenuhnya hanya dilakukan di perangkat keras PC (*personal computer*) dengan sistem operasi *Microsoft Windows 7*.

### 5.1.1 Batasan Implementasi

Dalam implementasi perangkat lunak, pemantauan dan pengendalian aplikasi ini ada beberapa hal yang menjadi batasan, yaitu :

1. Basis data yang digunakan dalam Implementasi sistem ini adalah *mySQL* sebagai basis data utama.
2. Dalam pembuatan aplikasi ini menggunakan bahasa pemrograman java dengan penggunaan *Java Development Kit* 6.9 (JDK6.9) dan menggunakan *tools Netbeans IDE 7.1.2* sebagai editor.
3. System yang dibuat bertujuan untuk membantu proses jual beli dan pengaturan data stok obat pada toko obat Hikmah Jaya.

### 5.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan program aplikasi *inventory* ini adalah sebagai berikut :

1. *MySQL* dipilih sebagai database yang digunakan dengan pertimbangan bahwa jumlah data yang diolah cukup banyak, dan applikasi yang dibangun merupakan applikasi *client-server.*
2. Perangkat lunak yang digunakan sebagai *tools* pembangun applikasi adalah *Netbeans IDE 7.1.2.*
3. Bahasa pemrograman yang digunakan untuk membangun applikasi ini adalah bahasa pemrograman java dengan *Java Development Kit* 6.9 (JDK6.9).
4. Sistem operasi yang digunakan dalam pembangunan dan pengujian applikasi ini adalah *Microsoft Windows 7.*

### 5.1.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus dipenuhi adalah sebagai berikut :

1. Client
    a. Processor Intel P4 2.4 atau diatasnya.
    b. Memory DDR RAM 512 MB.
    c. Harddisk 40 GB.
    d. Monitor, mouse dan keyboard.

### 5.1.4 Implementasi Basis Data

Pembahasan pembuatan basis data akan dibahas dengan menampilkan desain *query* dari basis data, dimana aplikasi pemrograman yang dipakai adalah *mySQL,* Implementasi basis data dalam *mySQL* adalah sebagai berikut:

```
-- phpMyAdmin SQL Dump
-- version 3.1.1
-- http://www.phpmyadmin.net
--
-- Host: localhost
-- Generation Time: Jun 10, 2013 at 02:36 AM
-- Server version: 5.1.30
-- PHP Version: 5.2.8

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

--
-- Database: `db_obat`
--

-- --------------------------------------------------------

--
-- Table structure for table `t_belibarang`
--

CREATE TABLE IF NOT EXISTS `t_belibarang` (
  `noNota` varchar(15) NOT NULL,
  `idObat` int(11) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `kadaluarsa` date NOT NULL,
  `HargaBeli` int(11) NOT NULL,
  PRIMARY KEY (`noNota`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
```

```
-- Dumping data for table `t_belibarang`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_jenis`
--

CREATE TABLE IF NOT EXISTS `t_jenis` (
  `idJenis` int(11) NOT NULL,
  `namaJenis` varchar(25) NOT NULL,
  `keterangan` text,
  PRIMARY KEY (`idJenis`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_jenis`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_jualbarang`
--

CREATE TABLE IF NOT EXISTS `t_jualbarang` (
  `noNota` varchar(15) NOT NULL,
  `idObat` int(11) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `kadaluarsa` date NOT NULL,
  `hargaJual` int(11) NOT NULL,
  PRIMARY KEY (`noNota`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
```

```
-- Dumping data for table `t_jualbarang`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_kategori`
--

CREATE TABLE IF NOT EXISTS `t_kategori` (
  `idKategori` int(11) NOT NULL,
  `namaKategori` varchar(50) NOT NULL,
  `keterangan` text,
  PRIMARY KEY (`idKategori`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_kategori`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_obat`
--

CREATE TABLE IF NOT EXISTS `t_obat` (
  `idObat` int(11) NOT NULL,
  `idKategori` int(11) NOT NULL,
  `idJenis` int(11) NOT NULL,
  `namaObat` varchar(50) NOT NULL,
  `kadaluarsa` date NOT NULL,
  `jumlah` int(11) NOT NULL,
  `satuan` varchar(10) NOT NULL,
  `keterangan` text NOT NULL,
  `Harga` bigint(20) NOT NULL,
```

```
  PRIMARY KEY (`idObat`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_obat`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_pembelian`
--

CREATE TABLE IF NOT EXISTS `t_pembelian` (
  `noNota` varchar(15) NOT NULL,
  `tanggalTransaksi` date NOT NULL,
  `idSupplier` int(11) NOT NULL,
  `totalHarga` bigint(20) NOT NULL,
  PRIMARY KEY (`noNota`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_pembelian`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_penjualan`
--

CREATE TABLE IF NOT EXISTS `t_penjualan` (
  `noNota` varchar(15) NOT NULL,
  `tanggalTransaksi` date NOT NULL,
  `totalHarga` bigint(20) NOT NULL,
  PRIMARY KEY (`noNota`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_penjualan`
--


-- --------------------------------------------------------

--
-- Table structure for table `t_supplier`
--

CREATE TABLE IF NOT EXISTS `t_supplier` (
  `idSupplier` int(11) NOT NULL,
  `namaSupplier` varchar(50) NOT NULL,
  `alamat` text NOT NULL,
  `noTelp` bigint(20) NOT NULL,
  `keterangan` text NOT NULL,
  PRIMARY KEY (`idSupplier`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `t_supplier`
--
```

### 5.1.5 Implementasi Antar Muka

Dalam *Netbeans IDE* , implementasi antarmuka dilakukan dengan sebuah *form* yang berekstensi form. Setiap halaman dalam perangkat lunak dibuka *file program* yang ditulis dengan berektensi form. Berikut ini adalah implementasi dari setiap halaman yang dibuat :

#### 5.1.5.1 Implementasi Halaman Utama

Form halaman utama merupakan sentral dari semua halaman form. Form ini berisi menu-menu utama yang dapat mengakses dan menjadi penghubung antara pengguna dengan sistem. Berikut ini implementasi struktur menu pada halaman utama Sistem Penggajian:

**Tabel 5. 1 Keterangan Halaman Utama (Menu)**

| Menu | Deskripsi | Nama File |
|---|---|---|
| Maintain Data Jenis | Menu untuk mencari, mengubah ataupun menambahkan data jenis obat yang baru pada database | JenisUI |
| Maintain Data Kategori | Menu untuk mencari, mengubah ataupun menambahkan data Kategori obat yang baru pada database | KategoriUI |
| Maintain Data Supplier | Menu untuk mencari, mengubah ataupun menambahkan data supplier yang baru pada database | SupplierUI |
| Maintain Data Obat | Menu untuk mencari, mengubah ataupun menambahkan data obat yang baru pada database | ObatUI |

| | | |
|---|---|---|
| Transaksi Penjualan | Menu untuk mencari ataupun menambahkan data penjualan yang baru pada database | PenjualanUI, ObatJualUI |
| Transaksi Pembelian | Menu untuk mencari ataupun menambahkan data transaksi pembelian yang baru pada database | PembelianUI, ObatBeliUI |
| Laporan Data | Menu yang digunakan untuk mencetak laporan, baik laporan penjualan, laporan pembelian, ataupun laporan rencana pembelian | FormTransaksiReport |
| Exit | Menu untuk keluar dari applikasi | MainFormUI |

### 5.1.6 Penggunaan Program

Kegiatan ini bertujuan untuk menerangkan secara singkat penggunaan program *Inventory* Obat yang sudah dibangun. Adapun cara pengoperasiannya adalah sebagai berikut :

**1. *Form* Utama**

Ketika pertama kali program applikasi ini dijalankan maka yang pertama kali muncul adalah form form utama seperti di bawah ini. Halaman ini merupakan halaman yang berisikan menu-menu yang dapat diakses.

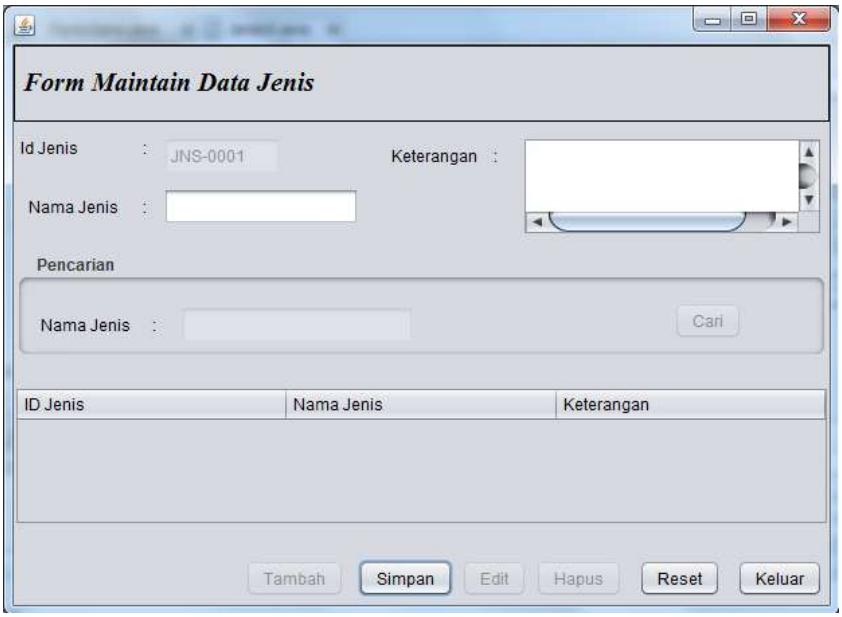



**Gambar 5. 1 Tampilan form utama**

**2. *Form Entry* Data Jenis**

Pengguna dapat melakukan pencarian data jenis dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data pengguna dapat memilih data jenis yang ingin di ubah dari list data jenis dan kemudian menekan tombol Edit. Setelah data keterangan jenis muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data

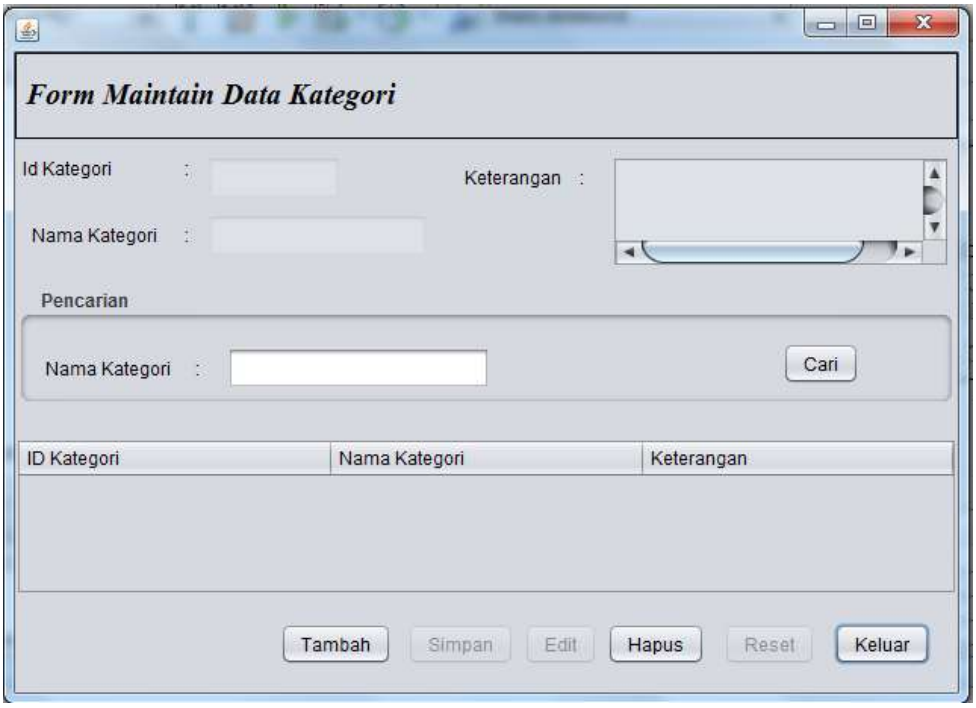
jenis yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.




**Gambar 5. 2 Tampilan *form Entry* Data Jenis**

### 3. *Form Maintain* Data Kategori

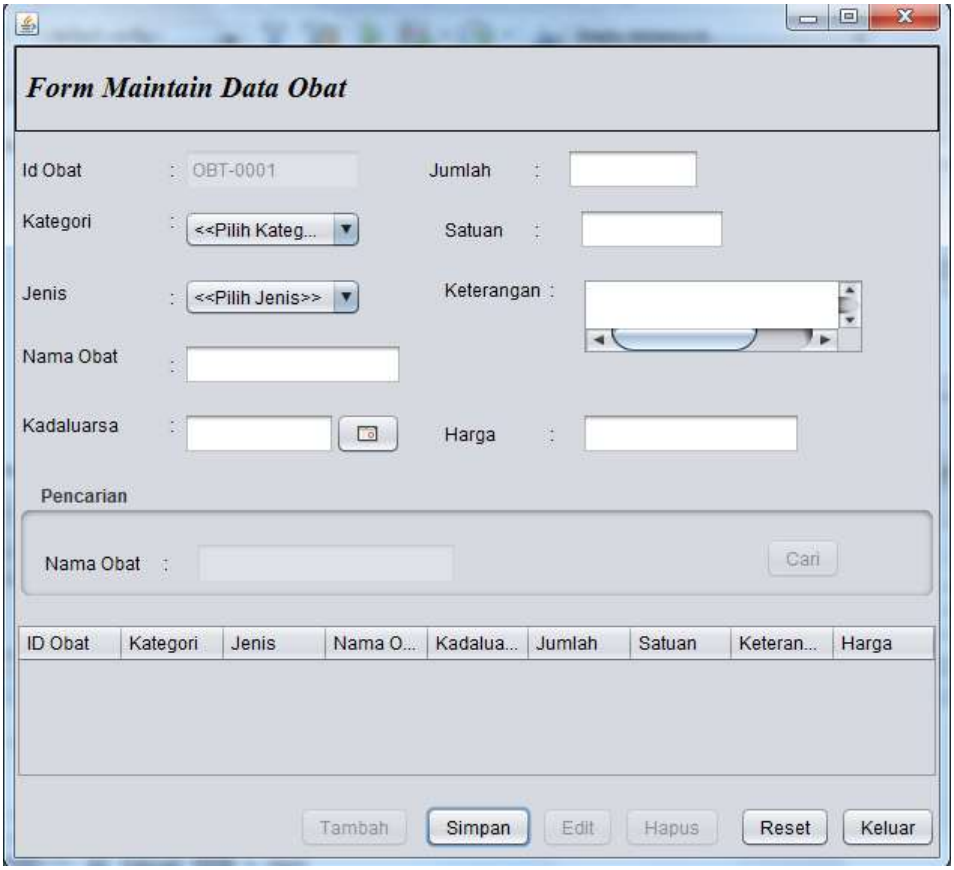
Pengguna dapat melakukan pencarian data kategori dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data pengguna dapat memilih data kategori yang ingin di ubah dari *list* data kategori dan kemudian menekan tombol Edit. Setelah data keterangan kategori muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data kategori yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.




**Gambar 5. 3 Tampilan *Form Maintain* Data Kategori**

**4. *Form Maintain* Data Obat**

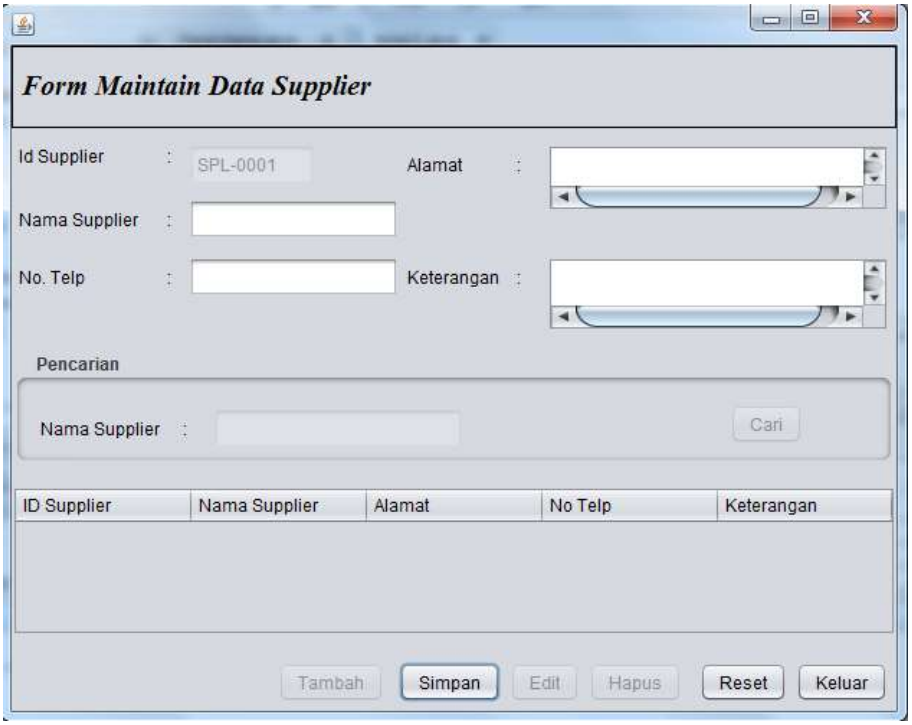
Pengguna dapat melakukan pencarian data obat dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data, pengguna dapat memilih data obat yang ingin di ubah dari *list* data obat dan kemudian menekan tombol Edit. Setelah data keterangan obat muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data obat yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.




**Gambar 5. 4 Tampilan *Form Maintain* Data Obat**

### 5. *Form Entry* Data *Supplier*

Pengguna dapat melakukan pencarian data *supplier* dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data pengguna dapat memilih data *supplier* yang ingin di ubah dari *list* data *supplier* dan kemudian menekan tombol Edit. Setelah data keterangan *supplier* muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data *supplier* yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



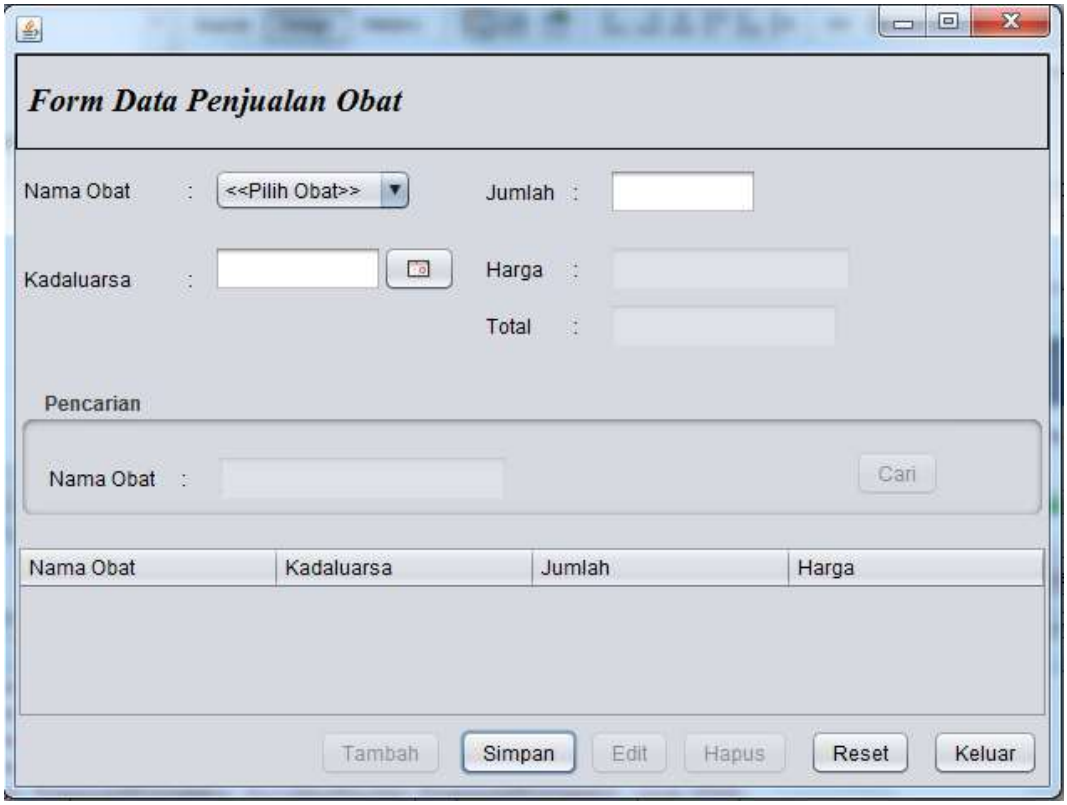

**Gambar 5. 5 Tampilan Form *Maintain* Data *Supplier***

### 6. Form Penjualan

Pengguna dapat melakukan penambahan data penjualan dengan menekan tombol tambah, kemudian menekan tombol tambah obat untuk menambahkan obat yang dijual.




**Gambar 5. 6 Tampilan Form Transaksi Penjualan**



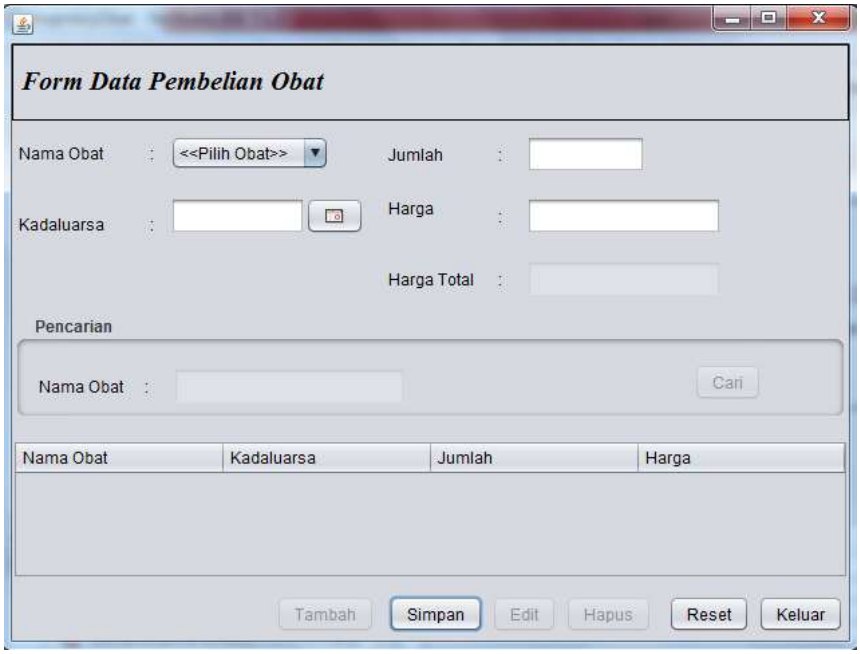

**Gambar 5. 7 Tampilan Form Penambahan data penjualan obat**

## 7. Form Transaksi Pembelian

Pengguna dapat melakukan penambahan data pembelian dengan menekan tombol tambah, kemudian menekan tombol tambah obat untuk menambahkan obat yang dibeli.



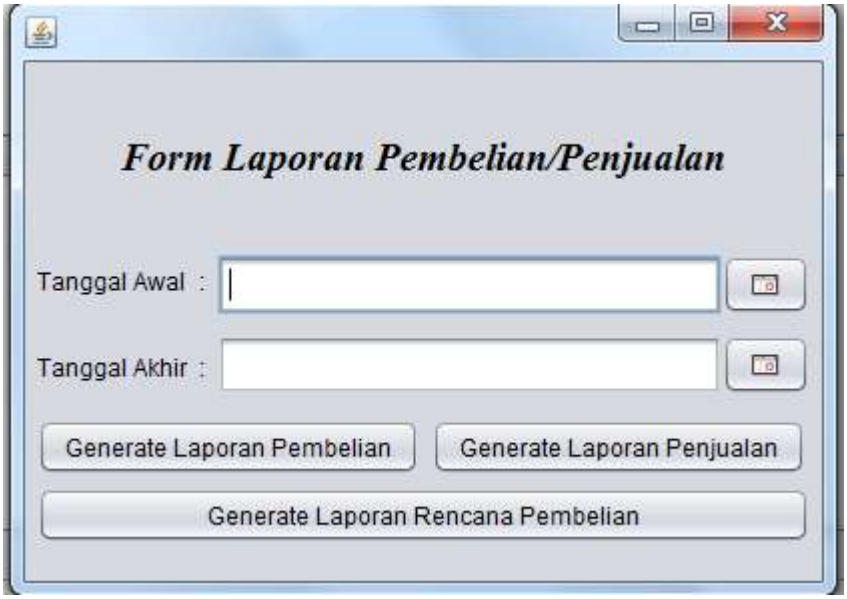

**Gambar 5. 8 Tampilan Form Transaksi Pembelian**




**Gambar 5. 9 Tampilan Form Data Pembelian Obat**

## 8. Form Laporan Pembelian/Penjualan

Pada menu laporan terdapat form yang berfungsi untuk melakukan pencetakan laporan sesuai dengan periode yang diinginkan.




**Gambar 5. 10 Tampilan Form Laporan Pembelian/Penjualan**

## 5.2 Pengujian

Pengujian dapat berarti proses untuk memeriksa apakah suatu perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai dengan standar tertentu. Metode pengujian adalah cara atau teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.

### 5.2.1 Rencana Pengujian

Adapun hal-hal yang akan diuji melalui teknik pengujian *black box* adalah sebagai berikut :

**Tabel 5. 2 Rencana Pengujian Sistem *Inventory* obat**

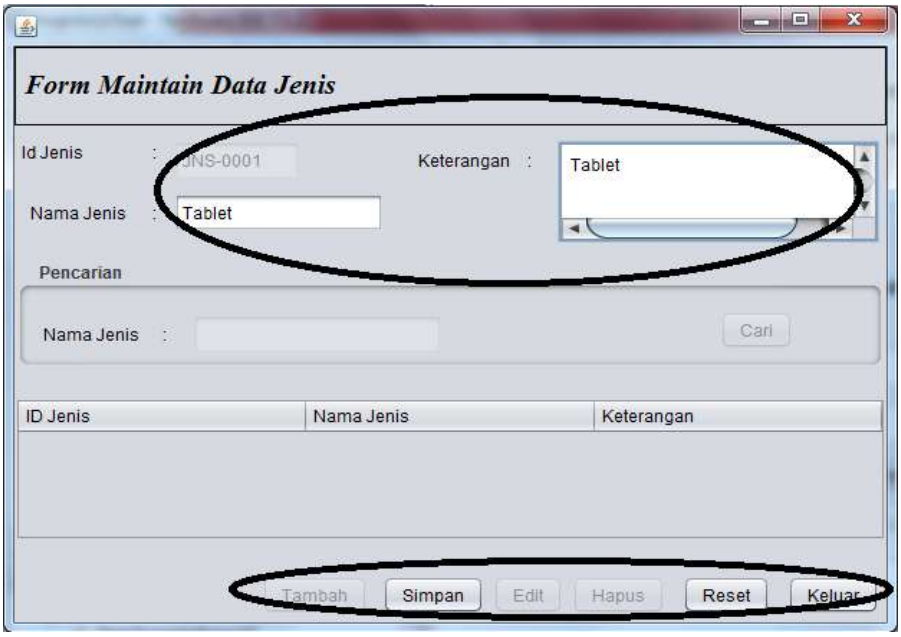
| Kelas Uji | Butir Uji | Halaman Pengujian |
|---|---|---|
| Penambahan data jenis | Penambahan data jenis obat | Form Maintain Data Jenis |
| Penambahan data kategori | Penambahan data kategori obat | Form Maintain Data Kategori |
| Penambahan data obat | Penambahan data obat | Form Maintain Data Obat |
| Penambahan data supplier | Penambahan data supplier | Form Maintain Data Supplier |
| Penambahan data | Penambahan data transaksi | a. Form Transaksi |

| | | |
|---|---|---|
| Transaksi Pembelian | pembelian obat | Pembelian<br>b. Form Data Pembelian Obat |
| Penambahan data Transaksi Penjualan | Penambahan data transaksi penjualan obat | c. Form Transaksi Penjualan<br>d. Form Data Penjualan Obat |
| Pencetakan Pelaporan | Pencetakan laporan | Form Laporan Pembelian/Penjualan |

### 5.2.2 Kasus Hasil Pengujian

1. Penambahan data jenis

   Pengguna dapat melakukan pencarian data jenis dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data pengguna dapat memilih data jenis yang ingin di ubah dari *list* data jenis dan kemudian menekan tombol Edit. Setelah data keterangan jenis muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data jenis yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



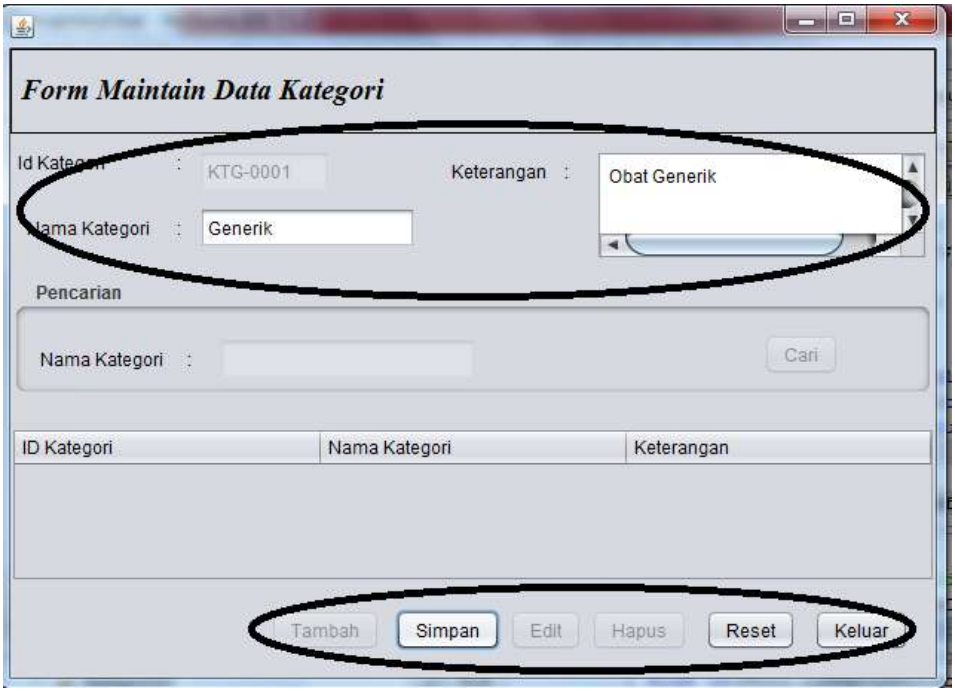

**Gambar 5. 11 Form Maintain Data Jenis**

**Tabel 5. 3 Pengujian pengolahan data jenis**

| Kasus dan hasil pengujian | | | |
|---|---|---|---|
| Data Masukan | Yang Diharapkan | Pengamatan | Kesimpulan |
| Masukan data pencarian untuk pengubahan data | Data pada table muncul sesuai pencarian | Data pada table sesuai pencarian | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data jenis yang baru lalu tekan simpan | Data jenis yang baru tersimpan pada database | Data jenis yang baru tersimpan pada database | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data jenis yang sama atau data kosong | Muncul peringatan pada form sesuai kesalahan | Muncul peringatan pada form sesuai kesalahan | [ ] Diterima<br>[ ] Ditolak |

2. Pengolahan data kategori

   Pengguna dapat melakukan pencarian data kategori dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data pengguna dapat memilih data kategori yang ingin di ubah dari *list* data kategori dan kemudian menekan tombol Edit. Setelah data keterangan kategori muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data kategori yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



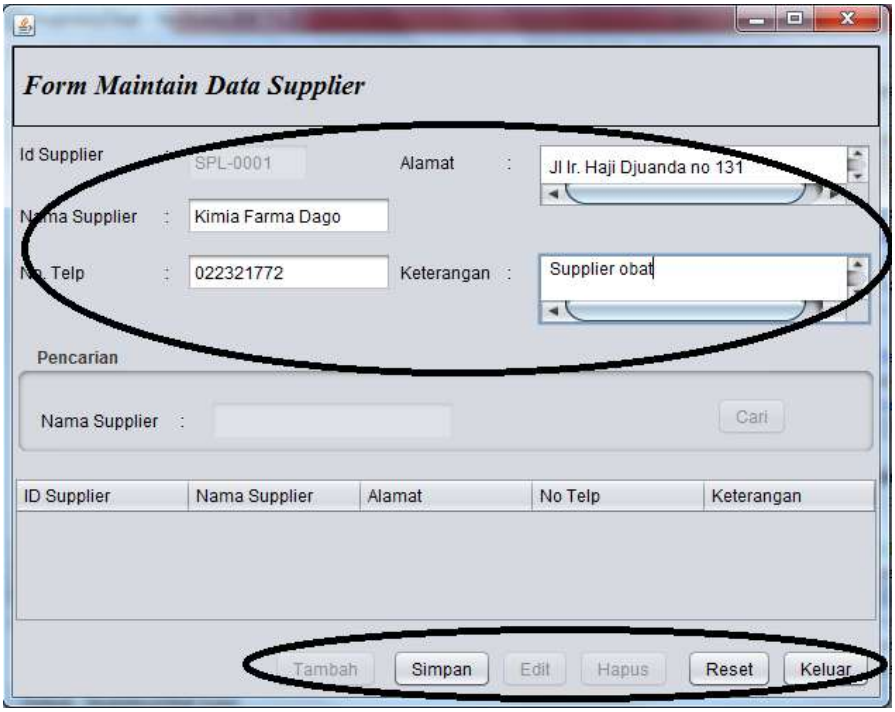

**Gambar 5. 12 Form Maintain Data Kategori**

**Tabel 5. 4 Pengujian pengolahan data jenis**

| Kasus dan hasil pengujian | | | |
|---|---|---|---|
| Data Masukan | Yang Diharapkan | Pengamatan | Kesimpulan |
| Masukan data pencarian untuk pengubahan data | Data pada table muncul sesuai pencarian | Data pada table sesuai pencarian | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data kategori yang baru lalu tekan simpan | Data kategori yang baru tersimpan pada database | Data kategori yang baru tersimpan pada database | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data kategori yang sama atau data kosong | Muncul peringatan pada form sesuai kesalahan | Muncul peringatan pada form sesuai kesalahan | [ ] Diterima<br>[ ] Ditolak |

3. Pengolahan data *supplier*

Pengguna dapat melakukan pencarian data *supplier* dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data pengguna dapat memilih data *supplier* yang ingin di ubah dari *list* data *supplier* dan kemudian menekan tombol Edit. Setelah data keterangan *supplier* muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data *supplier* yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.



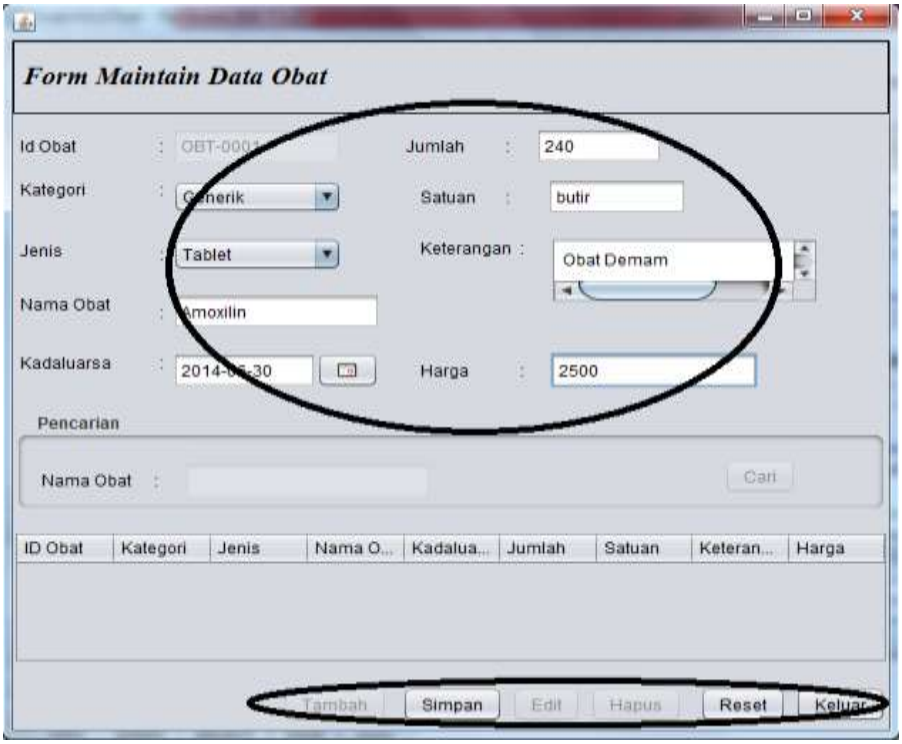

**Gambar 5. 13 Form Maintain Data *Supplier***

**Tabel 5. 5 Pengujian pengolahan data supplier**

| Kasus dan hasil pengujian | | | |
|---|---|---|---|
| Data Masukan | Yang Diharapkan | Pengamatan | Kesimpulan |
| Masukan data pencarian untuk pengubahan data | Data pada table muncul sesuai pencarian | Data pada table sesuai pencarian | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data supplier yang baru lalu tekan simpan | Data supplier yang baru tersimpan pada database | Data kategori yang baru tersimpan pada database | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data supplier yang sama atau data kosong atau data salah | Muncul peringatan pada form sesuai kesalahan | Muncul peringatan pada form sesuai kesalahan | [ ] Diterima<br>[ ] Ditolak |

4. Pengolahan data obat

   Pengguna dapat melakukan pencarian data obat dengan mengisi data sesuai dengan yang ingin dicari, kemudian menekan tombol Cari. Untuk melakukan pengubahan data, pengguna dapat memilih data obat yang ingin di ubah dari *list* data obat dan kemudian menekan tombol Edit. Setelah data keterangan obat muncul, pengguna dapat mengubah detail yang ada dan kemudian menekan tombol Simpan. Atau pengguna juga dapat menambah data obat yang baru dengan mengisi *field* yang ada dan kemudian menekan tombol Simpan.




**Gambar 5. 14 Form Maintain Data Obat**

**Tabel 5. 6 Pengujian pengolahan data obat**

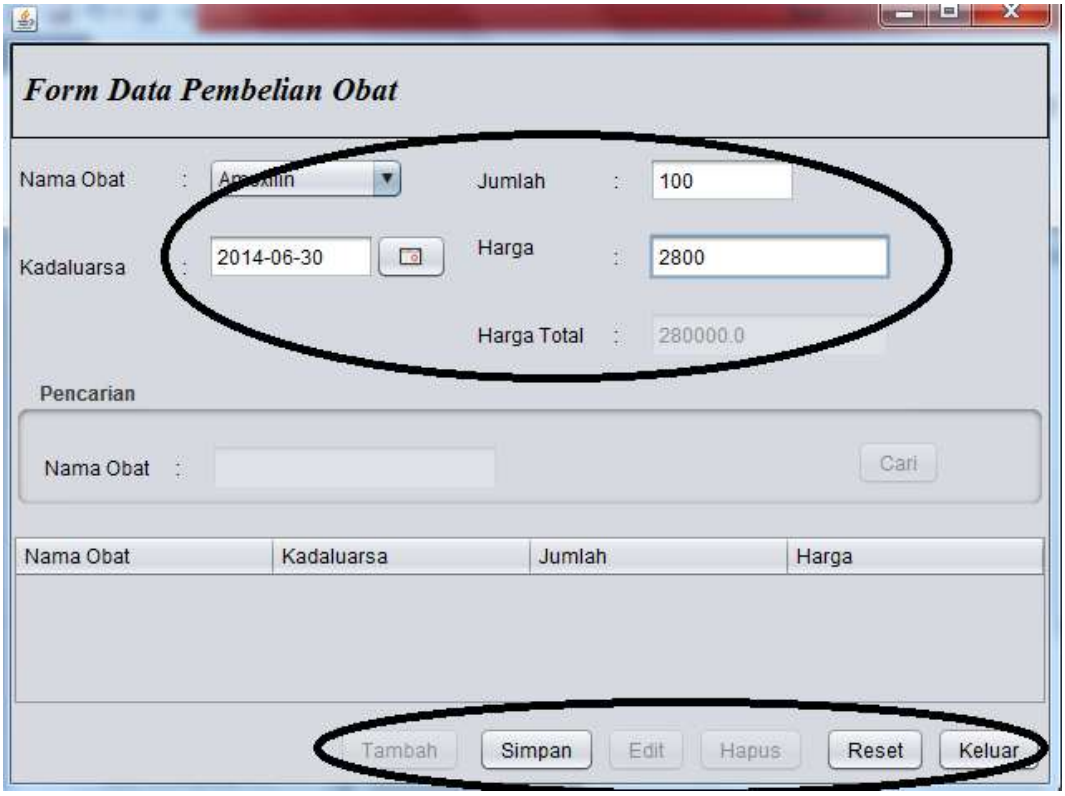
| Kasus dan hasil pengujian | | | |
|---|---|---|---|
| Data Masukan | Yang Diharapkan | Pengamatan | Kesimpulan |
| Masukan data pencarian untuk pengubahan data | Data pada table muncul sesuai pencarian | Data pada table sesuai pencarian | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data obat yang baru lalu tekan simpan | Data obat yang baru tersimpan pada database | Data obat yang baru tersimpan pada database | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data obat yang sama atau data kosong atau data salah | Muncul peringatan pada form sesuai kesalahan | Muncul peringatan pada form sesuai kesalahan | [ ] Diterima<br>[ ] Ditolak |

5. Penambahan data pembelian

Pengguna dapat melakukan penambahan data pembelian dengan menekan tombol tambah, kemudian menekan tombol tambah obat untuk menambahkan obat yang dibeli.




**Gambar 5. 15 Pengujian Form Transaksi Pembelian**




**Gambar 5. 16 Pengujian Form Data Pembelian Obat**

**Tabel 5. 7 Pengujian pengolahan data pembelian obat**

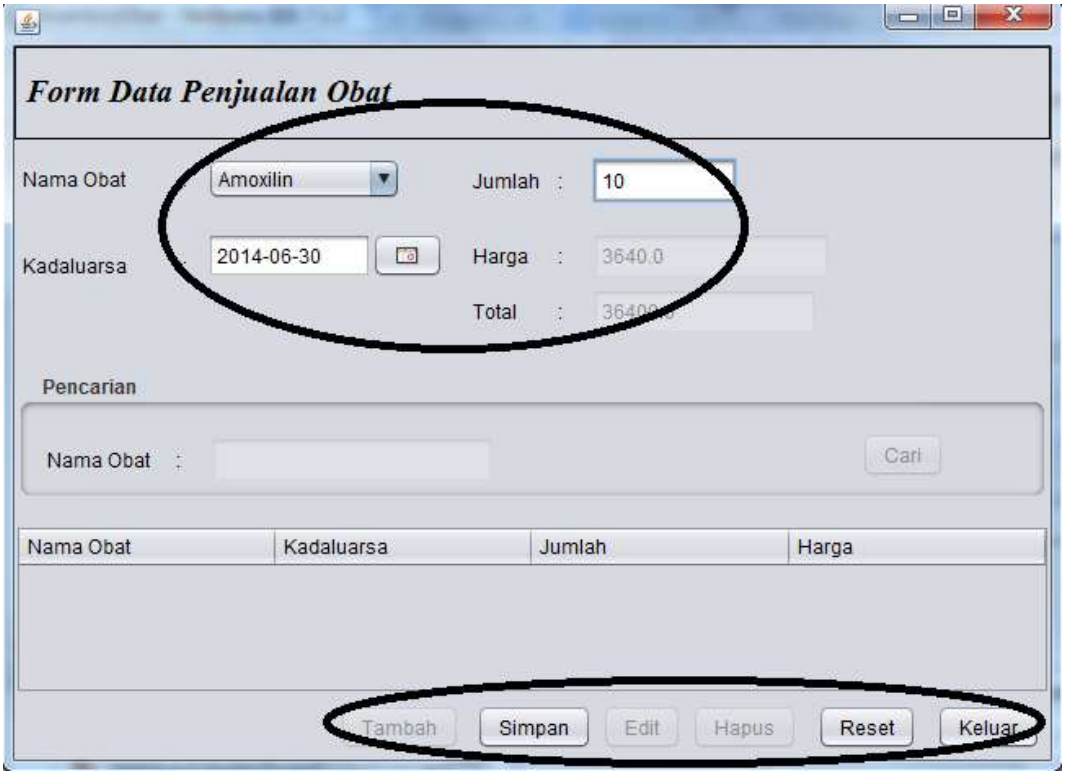
| Kasus dan hasil pengujian | | | |
|---|---|---|---|
| Data Masukan | Yang Diharapkan | Pengamatan | Kesimpulan |
| Masukan data pencarian untuk pengubahan data | Data pada table muncul sesuai pencarian | Data pada table sesuai pencarian | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data obat yang ingin dibeli yang baru lalu tekan simpan | Data pembelian obat yang baru tersimpan pada database | Data obat yang baru tersimpan pada database | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data obat yang sama atau data kosong atau data salah | Muncul peringatan pada form sesuai kesalahan | Muncul peringatan pada form sesuai kesalahan | [ ] Diterima<br>[ ] Ditolak |

6. Penambahan data penjualan

Pengguna dapat melakukan penambahan data penjualan dengan menekan tombol tambah, kemudian menekan tombol tambah obat untuk menambahkan obat yang dijual.




**Gambar 5. 17 Pengujian Form Transaksi Penjualan**




**Gambar 5. 18 Pengujian Form Data Penjualan Obat**

**Tabel 5. 8 Pengujian pengolahan data penjualan obat**

| Kasus dan hasil pengujian | | | |
|---|---|---|---|
| Data Masukan | Yang Diharapkan | Pengamatan | Kesimpulan |
| Masukan data pencarian untuk pengubahan data | Data pada table muncul sesuai pencarian | Data pada table sesuai pencarian | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data obat yang ingin dijual yang baru lalu tekan simpan | Data penjualan obat yang baru tersimpan pada database | Data obat yang baru tersimpan pada database | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data obat yang sama atau data kosong atau data salah | Muncul peringatan pada form sesuai kesalahan | Muncul peringatan pada form sesuai kesalahan | [ ] Diterima<br>[ ] Ditolak |

7. Pengujian Form Data Pelaporan

Pengguna dapat mencetak laporan dengan memasukkan data periode pelaporan yang diinginkan dan kemudian memilih jenis laporan.




**Gambar 5. 19 Form Laporan Pembelian/Penjualan**

**Tabel 5. 9 Pengujian pengolahan data pembelian obat**

| Kasus dan hasil pengujian | | | |
|---|---|---|---|
| Data Masukan | Yang Diharapkan | Pengamatan | Kesimpulan |
| Masukan data periode pelaporan yang diinginkan | Data laporan muncul sesuai periode | Data laporan muncul sesuai periode | [ ] Diterima<br>[ ] Ditolak |
| Masukkan data periode yang transaksinya tidak ada | Data laporan menampilkan data kosong | Data laporan menampilkan data kosong | [ ] Diterima<br>[ ] Ditolak |

### 5.2.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian uji *sample* di atas dapat disimpulan bahwa perangkat lunak Sistem Informasi *inventory* obat dapat berjalan dengan baik dan mengeluarkan hasil sesuai dengan yang diharapkan.