BAB V

PENGUJIAN DAN IMPLEMENTASI SISTEM

5.1 Implementasi Sistem

Tahap implementasi adalah tahapan penerapan sistem untuk dapat dioperasikan. Pada tahap ini dijelaskan mengenai sistem yang dirancang dan bagaimana cara penggunaanya. Kegiatan implementasi sistem ini meliputi kebutuhan Perangkat Lunak, Perangkat Keras, Basis Data, dan implementasi antar muka serta pengujian yang telah dirancang sesuai dengan kebutuhan sistem diApotek Permata Agung.

Tujuan dari implementasi ini adalah untuk menerapapkan system agar dapat dioperasikan sesuai dengan kebutuhan.

5.1.1 Batasan Implementasi

Dalam mengimplementasikan perangkat lunak perangkat lunak system informasi persediaan obat pada Apotek Permata Agung terdapat beberapa hal yang menjadi batasan implementasi system yaitu :

1. Transaksi penjualan dalam Sistem Informasi ini hanya sebatas pada pembayaran tunai.
2. Sistem ini tidak menyediakan sistem return obat yang kadaluarsa dan rusak.

3. Sistem ini hanya membahas obat yang bukan racikan walaupun dari resep dokter.
4. Sistem yang dibuat hanya membahas mengenai persediaan, penjualan dan pembelian obat.

5.1.2 Implementasi Perangkat Lunak

Perangkat lunak dalam sistem komputer adalah program-program yang berfungsi untuk menjalankan perangkat kerasnya dengan berbagai perintah. Yang termasuk perangkat lunak adalah aplikasi pembuatan *database*, bahasa pemrograman, sistem operasi dan program aplikasi. Perangkat lunak yang harus dipersiapkan dalam perancangan sistem informasi persediaanobat pada aptek Permata Agung ini adalah menggunakan :

1. Sistem Operasi Windows 7.
2. Software Program Java NetBeans IDE 7.2.
3. Software Database MySQL Server.
4. Software iReport 4.0.2.

5.1.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi untuk dapat menjalankan sistem aplikasi ini yaitu:

1. Processor minimal Intel Pentium 4
2. RAM Minimal 1GB

3. Hard Disk sebagai media penyimpanan.

4. *Mouse, keyboard,* Monitor sebagai *inputdevice*.

5. *Printer,* sebagai alat cetak.

5.1.4 Implementasi Basis Data

1. Tabel detail penjualan

```
CREATE TABLE IF NOT EXISTS `detail_penjualan` (
  `id_detail_penjualan` varchar(36) NOT NULL,
  `id_penjualan` varchar(36) NOT NULL,
  `id_obat` varchar(36) NOT NULL,
  `harga_berlaku` decimal(19,4) NOT NULL,
  `qty` int(11) NOT NULL,
  PRIMARY KEY (`id_detail_penjualan`),
  KEY `id_penjualan` (`id_penjualan`),
  KEY `id_obat` (`id_obat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel detail pembelian

```
CREATE TABLE IF NOT EXISTS `detail_pembelian` (
```

```
  `id_detail_pembelian` varchar(36) NOT NULL,
  `id_pembelian` varchar(36) NOT NULL,
  `id_obat` varchar(36) NOT NULL,
  `qty` int(11) NOT NULL,
  `exp_date` date NOT NULL,
  `harga_beli` decimal(19,4) NOT NULL,
  PRIMARY KEY (`id_detail_pembelian`),
  KEY `id_pembelian` (`id_pembelian`),
  KEY `id_obat` (`id_obat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel detail penjualan_2

```
CREATE TABLE IF NOT EXISTS `detail_penjualan_2` (
  `id_penjualan` varchar(36) NOT NULL,
  `nama_pasien` varchar(30) NOT NULL,
  `alamat_pasien` text NOT NULL,
  `umur_pasien` int(11) NOT NULL DEFAULT '0',
  `nama_dokter` varchar(30) NOT NULL,
```

```
  PRIMARY KEY (`id_penjualan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel golongan

```
CREATE TABLE IF NOT EXISTS `golongan` (
  `id_golongan_obat` int(2) NOT NULL,
  `nama_golongan` varchar(20) NOT NULL,
  PRIMARY KEY (`id_golongan_obat`),
  UNIQUE KEY `nama_golongan` (`nama_golongan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Golongan
obat';
```

5. Tabel obat

```
 CREATE TABLE IF NOT EXISTS `obat` (
   `id_obat` varchar(36) NOT NULL,
   `kodefikasi` varchar(13) NOT NULL,
   `nama_obat` varchar(100) NOT NULL,
   `dose` int(5) NOT NULL,
   `satuan_dose` enum('ml','mg') NOT NULL,
   `id_golongan_obat` int(2) NOT NULL,
   `id_satuan_obat` int(2) NOT NULL,
```

```
  `harga_obat` decimal(19,4) NOT NULL,
  `id_produsen_obat` varchar(36) NOT NULL,
  `min_stok` int(9) NOT NULL DEFAULT '20',
  PRIMARY KEY (`id_obat`),
  UNIQUE KEY `nama_obat` (`nama_obat`,`id_produsen_obat`),
  KEY `id_golongan_obat` (`id_golongan_obat`),
  KEY `id_satuan_obat` (`id_satuan_obat`),
  KEY `id_produsen_obat` (`id_produsen_obat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel obat_peringatan_map

```
 CREATE TABLE IF NOT EXISTS `obat_peringatan_map` (
   `id_obat` varchar(36) NOT NULL,
   `id_peringatan_obat` int(2) NOT NULL,
   PRIMARY KEY (`id_obat`),
   KEY `id_peringatan_obat` (`id_peringatan_obat`)
   ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Tabel pembelian

```
  CREATE TABLE IF NOT EXISTS `pembelian` (
    `id_pembelian` varchar(36) NOT NULL,
    `id_supplier` varchar(36) NOT NULL,
    `tgl_transaksi` date NOT NULL,
    `id_user` varchar(36) NOT NULL,
```

```
  PRIMARY KEY (`id_pembelian`),
  KEY `id_supplier` (`id_supplier`),
  KEY `id_user` (`id_user`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

8. Tabel penjualan

```
CREATE TABLE IF NOT EXISTS `penjualan` (
  `id_penjualan` varchar(36) NOT NULL,
  `waktu_transaksi` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP,
  `id_user` varchar(36) NOT NULL,
  `tax` int(3) NOT NULL DEFAULT '10',
  PRIMARY KEY (`id_penjualan`),
  KEY `id_user` (`id_user`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Table peringatan

```
CREATE TABLE IF NOT EXISTS `peringatan` (
  `id_peringatan_obat` int(2) NOT NULL,
  `peringatan` varchar(100) NOT NULL,
  PRIMARY KEY (`id_peringatan_obat`),
  UNIQUE KEY `peringatan` (`peringatan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. Table produsen

```
CREATE TABLE IF NOT EXISTS `produsen` (
 `id_produsen_obat` varchar(36) NOT NULL,
 `nama_produsen_obat` varchar(100) NOT NULL,
 PRIMARY KEY (`id_produsen_obat`),
 UNIQUE KEY `nama_produsen_obat` (`nama_produsen_obat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
COMMENT='Produsen yang membuat/menjual obat';
```

11. Table satuan

```
CREATE TABLE IF NOT EXISTS `satuan` (
 `id_satuan_obat` int(2) NOT NULL,
 `nama_satuan_obat` varchar(11) NOT NULL,
 PRIMARY KEY (`id_satuan_obat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Satuan
jual obat';
```

12. Table stok

```
 `id_stok` varchar(36) NOT NULL,
 `id_obat` varchar(36) NOT NULL,
 `exp_date` date NOT NULL,
 `jumlah_stok` int(11) NOT NULL,
 PRIMARY KEY (`id_stok`),
 UNIQUE KEY `id_obat` (`id_obat`,`exp_date`)
```

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

13. Table stok_id

```
CREATE TABLE IF NOT EXISTS `stok_id` (
  `stok_id` varchar(5) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

14. Table supplier

```
  `id_supplier` varchar(36) NOT NULL,
  `nama_supplier` varchar(100) NOT NULL,
  `alamat_supplier` text NOT NULL,
  `telepon_supplier` varchar(20) NOT NULL,
  `kota_supplier` varchar(30) NOT NULL,
  PRIMARY KEY (`id_supplier`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

15. Table user

```
  `id_user` varchar(36) NOT NULL,
  `username` varchar(32) NOT NULL,
  `nama` varchar(60) NOT NULL,
  `password` varchar(60) NOT NULL,
  `type` enum('god','apoteker','kasir') NOT NULL,
  PRIMARY KEY (`id_user`),
  UNIQUE KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.5 Implemetasi Antar Muka

1. form utama



Gambar5.1. Form Utama Sistem Informasi Apotek Permata Agung

Form Utama ini merupakan form yang akan ditampilkan saat pertama kali masuk kedalam sistem .Kemudian , untuk dapat menjalankan fungsi- fungsi lain dari sistem kita terlebih dahulu melakukan login.

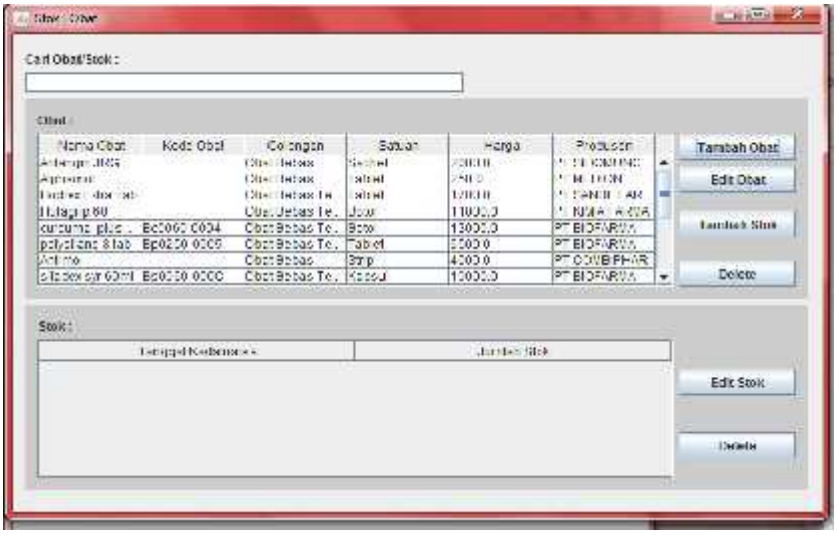
2. Form LOGIN



Gambar5.2 Form LOGIN

Form login ini adalah form yang harus dijalankan terlebih dahulu dengan memasukan *username* dan *password* untuk menentukan batasan hak akses yang akan ditampilkan oleh system.

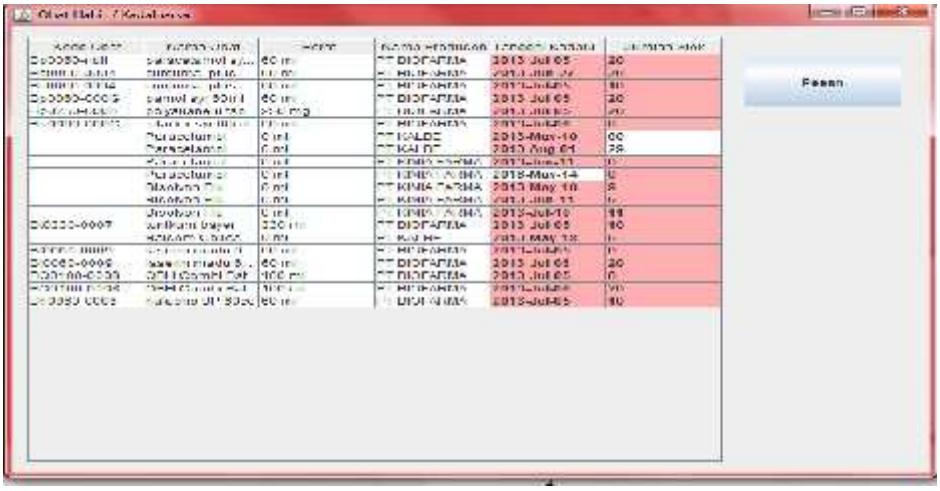
3. Stok & obat



Gambar 5.3 Form Stok & Obat

Form ini adalah form untuk mencari data obat dan melihat tanggal kadaluarsa dari data- data obat yang telah diinputkan.

4. Obat habis / kadaluarsa



Gambar 5.4 Form obat habis / kadaluarsa

Form ini adalah form untuk melihat kode,tanggal kadaluarsa dan jumlah stok yang sudah diinputkan di data obat sebelumnya.

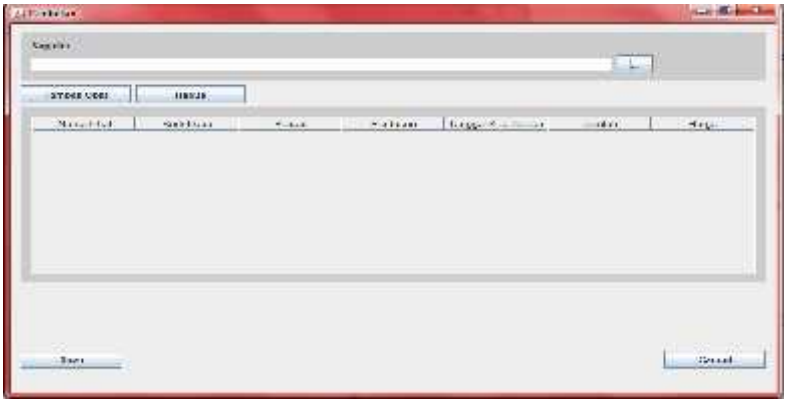
5. Form Penjualan



Gambar 5.5 Form penjualan

Form ini adalah form transaksi untuk mengetahui obat yang keluar karna permintaan konsumen.

6. Form Barang Masuk



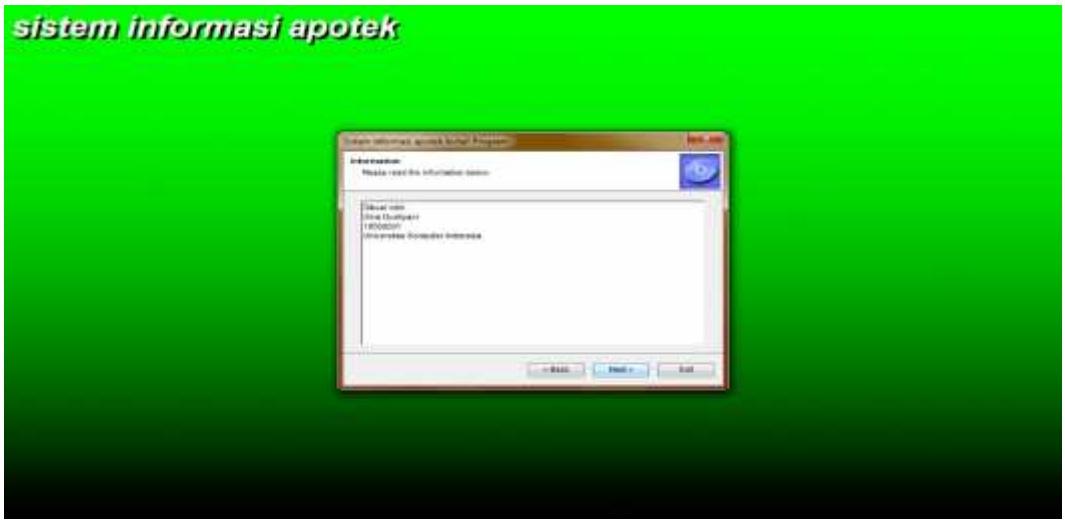
Gambar 5.6 Form pembelian

Form ini adalah form untuk mengetahui jumlah obat yang akan dibeli ke supplier dari pihak apotek.

5.1.6 Implementasi Instalasi Program



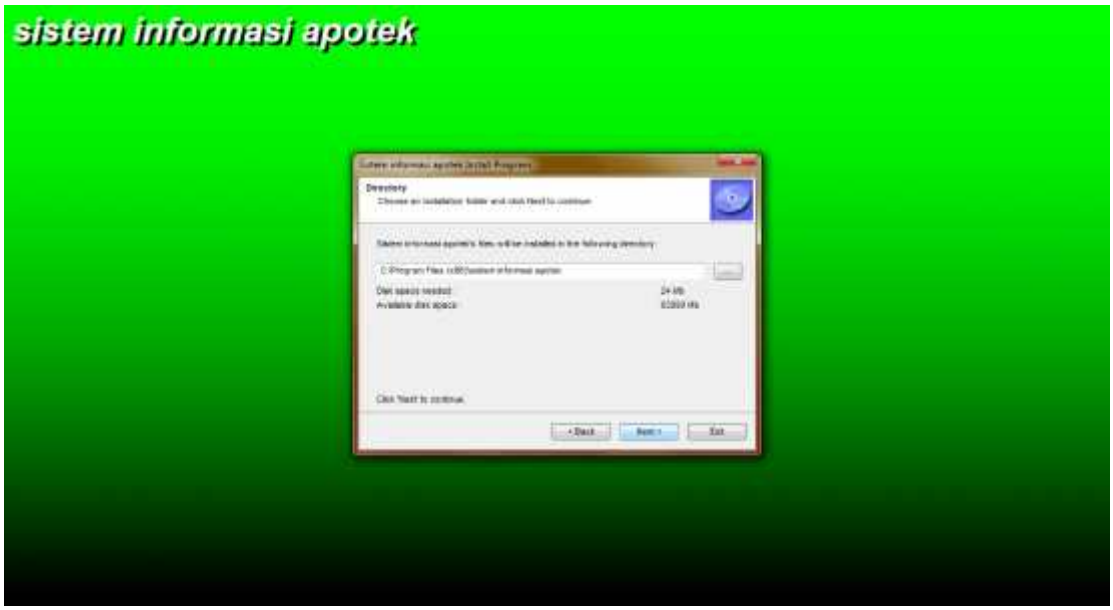
Gambar 5.7 Proses Instalasi pertama



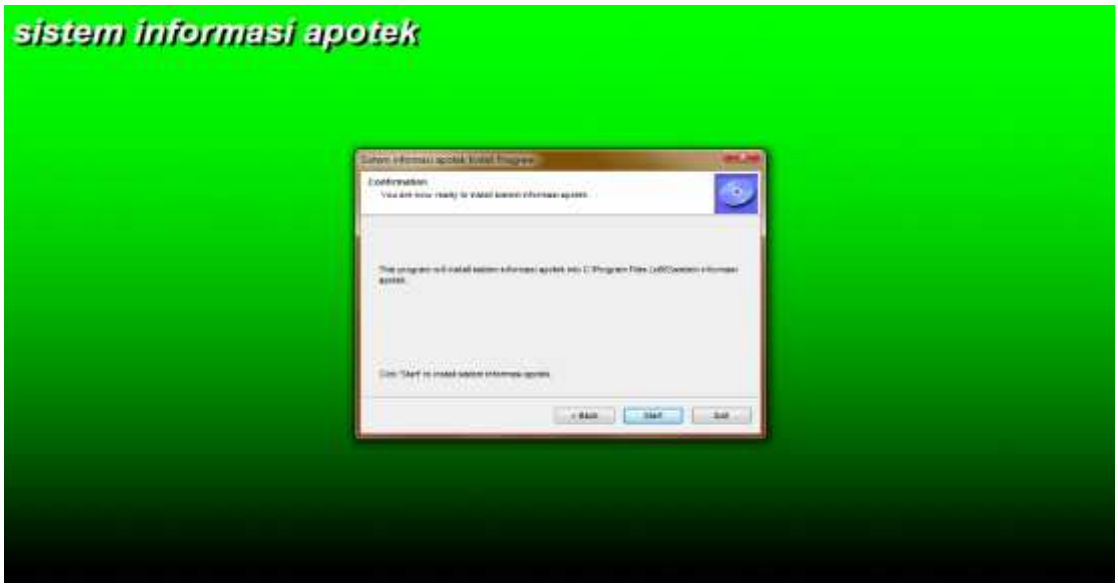
Gambar 5.8 Proses Instalasi kedua



Gambar 5.9 Proses Instalasi ketiga



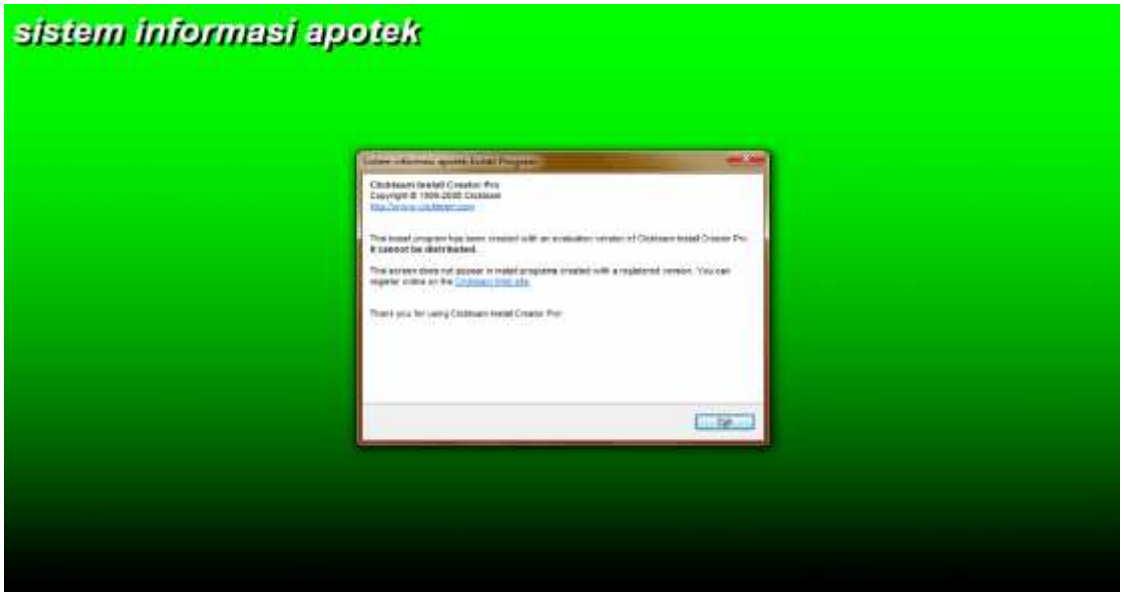
Gambar 5.10 Proses Instalasi keempat



Gambar 5.11 Proses Instalasi kelima



Gambar 5.12 Proses Instalasi keenam



Gambar 5.13 Proses Instalasi ketujuh

5.2 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak *(software)*. Tujuan dari pengujian adalah untuk menjamin perangkat lunak yang di bangun memilki kualitas yang handal, yaitu mampu melakukan kajian pokok dari spesifikasi, analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri. Pengujian perangkat lunak ini menggunakan metode pengujian *black box.* Pengujian *black box* ini tidak perlu tahu apa yang sesungguhnya terjadi dalam sistem atau perangkat lunak, yang diuji adalah masukan serta keluarannya. Dengan berbagai masukan yang diberikan, apakah sistem atau perangkat lunak memberikan keluaran seperti yang kita harapkan atau tidak.

5.2.1 Rencana Pengujian

Table 5.1 rencana pengujian

Kelas Uji	Butir Uji	Tingkat Pengujian	Jenis Pengujian
User LOGIN	Pemasukan Username dan Password	Sistem	*Black Box*
Input	Data obat	Sistem	*Black Box*
Input	Data Stok & obat	Sistem	*Black Box*
Input	Data obat habis /	Sistem	*Black Box*

	kadaluarsa		
Input	Produsen	Sistem	*Black Box*
Input	Supplier	Sistem	*Black Box*
Proses	Penjualan	Sistem	*Black Box*
Proses	Form barang masuk	Sistem	*Black Box*

5.2.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian dibuat untuk mengetahui apakah Sistem Informasi persediaan pada apotek Permata Agung dapat berjalan baik atau tidak.

5.2.2.1 Pengujian LOGIN

1. Pengujian Login *User*

Tabel 5.2 Pengujian LOGIN

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Menginputkan Username dan Password	*User* berhasil melakukan Login dan masuk ke	Masuk ke halaman Utama Menu sistem dan	[ ] Diterima [ ] Ditolak

dengan benar	tampilan utama menu sistem.	muncul *Message Dialog* "*Sistem Informasi Apotik*"	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Menginputkan Username dan Password dengan salah atau tidak lengkap	User tidak berhasil melakukan login dan gagal masuk ke tampilan utama Menu sistem	Tidak dapat masuk ke tampilan utama menu sistem , dan muncul *Message Dialog* ada password salah	[ ] Diterima [ ] Ditolak

5.2.2.2 Pengujian Input

1. Pengujian input Data obat

Tabel 5.3 Pengujian data obat

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data obat yang	Data obat dapat	Data obat yang	[ ] Diterima

akan dicari diisi dengan lengkap	ditampilkan pada daftar Tabel obat	telah diupdate berhasil masuk kedalam database obat	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data obat yang akan diedit tidak diisi dengan lengkap	Data obat tidak terdapat pada daftar Tabel obat	Data obat yang telah diupdate tidak berhasil masuk kedalam database obat	[ ] Diterima [ ] Ditolak

2. Pengujian Input Data Stok & Obat

Tabel 5.4 Pengujian Input Data Stok & Obat

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Stok & Obat diinputkan dengan lengkap	Data Stok & Obat dapat ditampilkan pada	Data yang telah diinputkan berhasil masuk	[ ] Diterima [ ] Ditolak

	Tabel daftar Stok & Obat	kedalam database stok_id	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Stok & Obat tidak diinputkan dengan lengkap	Data Stok&Obatdapat ditampilkan pada Tabel daftar Stok & Obat	Data yang telah diinputkan tidak berhasil masuk kedalam database Stok_id	[ ] Diterima [ ] Ditolak

3. Pengujian Input Data obat habis / kadaluarsa

Tabel 5.5 Pengujian Input Data obat habis / kadaluarsa

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data obat habis / kadaluarsa diinputkan	Data obat habis / kadaluarsa dapat ditampilkan pada	Data yang telah diinputkan berhasil masuk	[ ] Diterima [ ] Ditolak

dengan lengkap	Tabel daftar obat habis / kadaluarsa	kedalam database stok	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data obat habis / kadaluarsa tidak diinputkan dengan lengkap	Data obat habis / kadaluarsadapat ditampilkan pada Tabel daftar obat habis / kadaluarsa	Data yang telah diinputkan tidak berhasil masuk kedalam database stok	[ ] Diterima [ ] Ditolak

4. Pengujian Input Data Produsen

Tabel 5.6 Pengujian Input Data Produsen

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data produsen diinputkan	Data produsen dapat ditampilkan pada	Data yang telah diinputkan berhasil masuk	[ ] Diterima [ ] Ditolak

dengan lengkap	Tabel daftar produsen	kedalam database produsen	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data produsen tidak diinputkan dengan lengkap	Data produsendapat ditampilkan pada Tabel daftar produsen	Data yang telah diinputkan tidak berhasil masuk kedalam database produsen	[ ] Diterima [ ] Ditolak

5. Pengujian Input Data supplier

Tabel 5.7 Pengujian Input Data Supplier

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data supplier diinputkan dengan lengkap	Data supplier dapat ditampilkan pada Tabel daftar	Data yang telah diinputkan berhasil masuk kedalam	[ ] Diterima [ ] Ditolak

	supplier	database supplier	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data supplier tidak diinputkan dengan lengkap	Data supplierdapat ditampilkan pada Tabel daftar supplier	Data yang telah diinputkan tidak berhasil masuk kedalam database supplier	[ ] Diterima [ ] Ditolak

5.2.2.3 Pengujian Proses

1. Pengujian Proses Transaksi penjualan

Tabel 5.8 Pengujian Proses Transaksi *Penjualan*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Mengenai Detail Penjualan diinputkan dengan lengkap	Detail Penjualan yang telah diinputkan ada dalam daftar tampilan transaksi penjualan	Detail Penjualan masuk kedalam database penjualan dan detail penjualan	[] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Mengenai Detail Penjualan tidak diinputkan dengan lengkap	Detail Penjualan yang telah diinputkan tidak ada dalam daftar tampilan transaksi penjualan	Detail Penjualan masuk kedalam database penjualan dan detail penjualan dan muncul	[ ] Diterima [ ] Ditolak

2. Pengujian Proses Form Barang Masuk

Tabel 5.9 Pengujian Proses Form Barang Masuk

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Mengenai Form Barang Masukdiinputkan dengan lengkap	Form Barang masuk yang telah diinputkan ada dalam daftar pembelian	Detail Data form barang masuk kedalam database pembelian	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Mengenai Form Barang masuktidak diinputkan dengan lengkap	Form barang masuk yang telah diinputkan tidak ada dalam daftar pembelian	Detail Data form barang masuk tidak masuk kedalam database pembelian	[ ] Diterima [ ] Ditolak

5.2.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji *sample* diatas dapat ditarik kesimpulan bahwa perangkat lunak Sistem informasi Persediaan obat pada Apotek

Permata Agung bebas dari kesalahan *sintaks* dan secara fungsional mengeluarkan fungsi sesuai dengan yang diharapkan.