BAB V

IMPLEMENTASI SISTEM

5.1 Implementasi

Implementasi *prototype* sistem informasi pelayanan kesehatan apotek ikhwan soreang dilakukan menggunakan bahasa dan lingkungan pemrograman Borland Delphi7, basis data yang digunakan ialah SQL Server 7. Implementasi dan pengujian sepenuhnya hanya dilakukan di perangkat PC (*personal computer*) dengan sistem operasi *Microsoft Windows XP*.

5.1.1 Batasan Implementasi

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

1. Aplikasi sistem informasi ini bisa aktif bila server juga aktif.
2. Tampilan aplikasi menggunakan bahasa indonesia, kecuali bagian tertentu yang dianggap penulis lebih baik menggunakan bahasa asing.
3. Sistem informasi pelayanan kesehatan apotek ikhwan ini hanya sebatas pengelolaan data pasien, data rekammedik, dan data penjualan obat.

5.1.2 Implementasi Perangkat Lunak

Untuk pengembangan perangkat lunak ini digunakan Borland Delphi 7 sebagai perangkat lunak pengembang karena menyediakan fasilitas yang memadai untuk membuat aplikasi, MySQL sebagai perangkat lunak pengembang dalam pembuatan basis data.

5.1.3 Implementasi Perangkat Keras

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

Server :

1. Processor yang digunakan adalah *intel pentium 4 CPU 2.60 Ghz.*
   Rekomendasi minimum adalah processor intel pentium 4 CPU 1.60 Ghz.
2. Harddisk terpasang 40 GB, rekomendasi minimun adalah 20 GB.
3. Memory terpasang 256 MB, rekomendasi minimum adalah 8 MB.
4. VGA Card terpasang 32 MB, rekomendasi minimum adalah 1 MB.
5. Mouse, Keyboard, dan Monitor sebagai peralatan antarmuka.

Client :

1. Processor yang digunakan adalah *intel pentium 4 CPU 2.60 Ghz.*
   Rekomendasi minimum adalah processor intel pentium 4 CPU 1.60 Ghz.
2. Harddisk terpasang 40 GB, rekomendasi minimun adalah 20 GB.
3. Memory terpasang 256 MB, rekomendasi minimum adalah 8 MB.
4. VGA Card terpasang 32 MB, rekomendasi minimum adalah 1 MB.
5. Mouse, Keyboard, dan Monitor sebagai peralatan antarmuka.

5.1.4 Implementasi Basis Data

Pembuatan basis data dilakukan dengan mengunakan bahasa MySQL dimana aplikasi pemrograman yang digunakan adalah MySQL.

```
# HeidiSQL Dump
#
# --------------------------------------------------------
# Host:                 127.0.0.1
# Database:             dbapotek
# Server version:       4.0.0-alpha-nt
# Server OS:
# Target-Compatibility: Standard ANSI SQL
# HeidiSQL version:     3.2 Revision: 1129
# --------------------------------------------------------

/*!40100 SET CHARACTER SET binary;*/
`/*!40101              SET              @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ANSI';*/
/*!40014SET
@OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;*/
#
# Database structure for database 'dbapotek'
#
CREATE DATABASE /*!32312 IF NOT EXISTS*/ "dbapotek";

USE "dbapotek";
```

```
#
# Table structure for table 'dokter'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "dokter" (
  "NIP_Dokter" varchar(20) NOT NULL default '',
  "Nama_Dokter" varchar(50) default NULL,
  "NoTelp" varchar(15) default NULL,
  "Alamat" varchar(255) default NULL,
  "Spesialisasi" varchar(20) default NULL,
  PRIMARY KEY  ("NIP_Dokter")) YPE=MyISAM;

#
# Dumping data for table 'dokter'
#

LOCK TABLES "dokter" WRITE;
/*!40000 ALTER TABLE "dokter" DISABLE KEYS;*/
REPLACE INTO "dokter" ("NIP_Dokter", "Nama_Dokter", "NoTelp",
"Alamat", "Spesialisasi") VALUES
        ('DP0002','Yudhi','92950527','Jl. Soekarno Hatta','Kulit');
REPLACE INTO "dokter" ("NIP_Dokter", "Nama_Dokter", "NoTelp",
"Alamat", "Spesialisasi") VALUES
```

```
('DP0001','Trisna Kusmawanto','081573982727','Ir. H. Juanda 12','Paru-paru');
REPLACE INTO "dokter" ("NIP_Dokter", "Nama_Dokter", "NoTelp", "Alamat", "Spesialisasi") VALUES
('DP0003','Hadian Sugandi','93752509','Jl. Pasteur no 99','Kelamin');
REPLACE INTO "dokter" ("NIP_Dokter", "Nama_Dokter", "NoTelp", "Alamat", "Spesialisasi") VALUES
('DP0004','Jerry Kristianty','95016827','Pasirr Koja no 234','UMUM');
REPLACE INTO "dokter" ("NIP_Dokter", "Nama_Dokter", "NoTelp", "Alamat", "Spesialisasi") VALUES
('DP0005','Dinda Dwi Apriadi','022-92167225','Cijerah 108','Internist');
/*!40000 ALTER TABLE "dokter" ENABLE KEYS;*/
UNLOCK TABLES;

#
# Table structure for table 'obat'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "obat" (
  "Kode_obat" varchar(7) NOT NULL default '',
```

```
  "Nama_Obat" varchar(30) default NULL,
  "Harga_Obat" int(10) default NULL,
  "satuan" varchar(10) default NULL,
  "Tipe_obat" varchar(10) default NULL,
  PRIMARY KEY  ("Kode_obat")) YPE=MyISAM;

#
# Dumping data for table 'obat'
#

LOCK TABLES "obat" WRITE;
/*!40000 ALTER TABLE "obat" DISABLE KEYS;*/
REPLACE INTO "obat" ("Kode_obat", "Nama_Obat", "Harga_Obat",
"satuan", "Tipe_obat") VALUES
        ('COB002','dexamethasone',3000,'strip','tablet');
REPLACE INTO "obat" ("Kode_obat", "Nama_Obat", "Harga_Obat",
"satuan", "Tipe_obat") VALUES
        ('COB001','Nichostan',2500,'strip','tablet');
REPLACE INTO "obat" ("Kode_obat", "Nama_Obat", "Harga_Obat",
"satuan", "Tipe_obat") VALUES
        ('COB003','ibuprofen',1750,'strip','tablet');
REPLACE INTO "obat" ("Kode_obat", "Nama_Obat", "Harga_Obat",
"satuan", "Tipe_obat") VALUES
```

```
('COB004','intunal f',4000,'strip','tablet');
REPLACE INTO "obat" ("Kode_obat", "Nama_Obat", "Harga_Obat",
"satuan", "Tipe_obat") VALUES
('COB005','xiltrop',3500,'strip','tablet');
REPLACE INTO "obat" ("Kode_obat", "Nama_Obat", "Harga_Obat",
"satuan", "Tipe_obat") VALUES
('COB006','dextral',5200,'strip','tablet');
REPLACE INTO "obat" ("Kode_obat", "Nama_Obat", "Harga_Obat",
"satuan", "Tipe_obat") VALUES
('COB007','salbutamol',1500,'strip','tablet');
/*!40000 ALTER TABLE "obat" ENABLE KEYS;*/
UNLOCK TABLES;

#
# Table structure for table 'pasien'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "pasien" (
  "KodePasien" varchar(5) NOT NULL default '',
  "NamaPasien" varchar(30) default NULL,
  "TempatLahir" varchar(30) default NULL,
  "TglLahir" date default NULL,
  "Kelamin" varchar(10) default NULL,
```

```
  "Gol_darah" char(2) default NULL,
  "Alamat_pasien" varchar(80) default NULL,
  "Telp_pasien" varchar(20) default NULL,
  "TglDaftar" date default NULL,
  PRIMARY KEY  ("KodePasien")) YPE=MyISAM;

#
# Dumping data for table 'pasien'
#

LOCK TABLES "pasien" WRITE;
/*!40000 ALTER TABLE "pasien" DISABLE KEYS;*/
REPLACE INTO "pasien" ("KodePasien", "NamaPasien", "TempatLahir",
"TglLahir", "Kelamin", "Gol_darah", "Alamat_pasien", "Telp_pasien",
"TglDaftar") VALUES
        ('0001P','Eka Mayang Sari','Kendari','1987-05-
09','Perempuan','A','Kp. Papakmanggu Rt 01 Rw 13 Kec. Pasir
Jambu','085624230277','2010-05-27');
REPLACE INTO "pasien" ("KodePasien", "NamaPasien", "TempatLahir",
"TglLahir", "Kelamin", "Gol_darah", "Alamat_pasien", "Telp_pasien",
"TglDaftar") VALUES
        ('0002P','Furi Defiq','Bandung','1999-05-28','Perempuan','O','Desa
Cibodas Rt 03 Rw 03 Kec. Ciwidey','081394202377','2010-05-28');
```

```
REPLACE INTO "pasien" ("KodePasien", "NamaPasien", "TempatLahir",
"TglLahir", "Kelamin", "Gol_darah", "Alamat_pasien", "Telp_pasien",
"TglDaftar") VALUES
        ('0001L','Fikri','Bandung','2000-05-01','Laki - Lak','B','Desa
Cisondari Rt 05 Rw 01','085222202728','2010-05-28');
REPLACE INTO "pasien" ("KodePasien", "NamaPasien", "TempatLahir",
"TglLahir", "Kelamin", "Gol_darah", "Alamat_pasien", "Telp_pasien",
"TglDaftar") VALUES
        ('0002L','Pian','Bandung','1989-06-01','Laki -
Lak','O','Ciwidey','08122338889','2010-06-11');
REPLACE INTO "pasien" ("KodePasien", "NamaPasien", "TempatLahir",
"TglLahir", "Kelamin", "Gol_darah", "Alamat_pasien", "Telp_pasien",
"TglDaftar") VALUES
('0003L','Arief','Bandung','1990-06-14','Laki
Lak','A','Ciwastra','085572453',NULL);
/*!40000 ALTER TABLE "pasien" ENABLE KEYS;*/
UNLOCK TABLES

#
# Table structure for table 'pemeriksaan'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "pemeriksaan" (
```

```
  "No_pemeriksaan" varchar(7) NOT NULL default '',
  "Tgl_pemeriksaan" date default NULL,
  "No_resep" varchar(10) default NULL,
  "Kode_pasien" varchar(5) default NULL,
  "No_rekamMedis" varchar(7) default NULL,
  "BiayaDokter" int(10) default NULL,
  "TotalBayar" int(10) default NULL,
  PRIMARY KEY  ("No_pemeriksaan")) YPE=MyISAM;

#
# Dumping data for table 'pemeriksaan'
#

LOCK TABLES "pemeriksaan" WRITE;
/*!40000 ALTER TABLE "pemeriksaan" DISABLE KEYS;*/
REPLACE INTO "pemeriksaan" ("No_pemeriksaan", "Tgl_pemeriksaan",
"No_resep", "Kode_pasien", "No_rekamMedis", "BiayaDokter",
"TotalBayar") VALUES
        ('02652','2010-05-28','100611001','84845','001',50000,NULL);
/*!40000 ALTER TABLE "pemeriksaan" ENABLE KEYS;*/
UNLOCK TABLES;
```

```
#
# Table structure for table 'pendaftaran'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "pendaftaran" (
  "Kode_Pendaftaran" varchar(7) NOT NULL default '',
  "tgl_pendaftaran" date default NULL,
  "Kode_pasien" varchar(7) default NULL,
  PRIMARY KEY  ("Kode_Pendaftaran")) YPE=MyISAM;

#
# Dumping data for table 'pendaftaran'
#

# (No data found.)

#
# Table structure for table 'rekammedik'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "rekammedik" (
  "No_Rekammedis" varchar(10) NOT NULL default '',
  "NIP_Dokter" varchar(20) default NULL,
```

```
  "Keluhan" varchar(100) default NULL,
  "diagnosis" varchar(50) default NULL,
  "Kode_pasien" varchar(5) default NULL,
  PRIMARY KEY  ("No_Rekammedis")) YPE=MyISAM;

#
# Dumping data for table 'rekammedik'
#

LOCK TABLES "rekammedik" WRITE;
/*!40000 ALTER TABLE "rekammedik" DISABLE KEYS;*/
REPLACE INTO "rekammedik" ("No_Rekammedis", "NIP_Dokter",
"Keluhan", "diagnosis", "Kode_pasien") VALUES
        ('0000001','DP0002','sesak nafas','asma
','0001P');
REPLACE INTO "rekammedik" ("No_Rekammedis", "NIP_Dokter",
"Keluhan", "diagnosis", "Kode_pasien") VALUES
        ('0000002','DP0002','kulit pecah-pecah
','kapalan','0002P');
REPLACE INTO "rekammedik" ("No_Rekammedis", "NIP_Dokter",
"Keluhan", "diagnosis", "Kode_pasien") VALUES
        ('0000003','DP0004','dada sakit, sulit bernapas, batuk berdahak
','batuk','0001L');
```

```
REPLACE INTO "rekammedik" ("No_Rekammedis", "NIP_Dokter",
"Keluhan", "diagnosis", "Kode_pasien") VALUES
        ('0000004','DP0005','pusing
','anemia','0002L');
/*!40000 ALTER TABLE "rekammedik" ENABLE KEYS;*/
UNLOCK TABLES;

#
# Table structure for table 'resep'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "resep" (
  "No_Resep" varchar(10) NOT NULL default '',
  "Tgl_Resep" date default NULL,
  "Total_Resep" int(10) default NULL,
  PRIMARY KEY  ("No_Resep")) YPE=MyISAM;

#
# Dumping data for table 'resep'
#

LOCK TABLES "resep" WRITE;
/*!40000 ALTER TABLE "resep" DISABLE KEYS;*/
```

```
REPLACE INTO "resep" ("No_Resep", "Tgl_Resep", "Total_Resep")
VALUES
        ('100611001','2010-06-11',92000);
REPLACE INTO "resep" ("No_Resep", "Tgl_Resep", "Total_Resep")
VALUES
        ('001','2010-05-27',8000);
REPLACE INTO "resep" ("No_Resep", "Tgl_Resep", "Total_Resep")
VALUES
        ('100611002','2010-06-11',27500);
/*!40000 ALTER TABLE "resep" ENABLE KEYS;*/
UNLOCK TABLES;

#
# Table structure for table 'resep_dtl'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "resep_dtl" (
  "No_resep" varchar(10) NOT NULL default '',
  "Kode_Obat" varchar(7) NOT NULL default '',
  "Jml_Beli" int(5) default NULL,
  "SubTotal" int(10) default NULL,
  PRIMARY KEY  ("No_resep","Kode_Obat")) YPE=MyISAM;
```

```
#
# Dumping data for table 'resep_dtl'
#

LOCK TABLES "resep_dtl" WRITE;
/*!40000 ALTER TABLE "resep_dtl" DISABLE KEYS;*/
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
        ('100527001','001',2,4000);
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
        ('001','COB002',1,3000);
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
        ('001','COB005',1,3500);
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
        ('001','COB007',1,1500);
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
        ('100611001','COB004',10,40000);
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
```

```
        ('100611001','COB006',10,52000);
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
        ('100611002','COB004',5,20000);
REPLACE INTO "resep_dtl" ("No_resep", "Kode_Obat", "Jml_Beli",
"SubTotal") VALUES
        ('100611002','COB007',5,7500);
/*!40000 ALTER TABLE "resep_dtl" ENABLE KEYS;*/
UNLOCK TABLES;

#
# Table structure for table 'user'
#

CREATE TABLE /*!32312 IF NOT EXISTS*/ "user" (
  "KodeUser" varchar(10) NOT NULL default '',
  "Pass" varchar(15) default NULL,
  PRIMARY KEY  ("KodeUser")) YPE=MyISAM;

#
# Dumping data for table 'user'
#
```

```
LOCK TABLES "user" WRITE;
/*!40000 ALTER TABLE "user" DISABLE KEYS;*/
REPLACE INTO "user" ("KodeUser", "Pass") VALUES
    ('admin','1234');
/*!40000 ALTER TABLE "user" ENABLE KEYS;*/
UNLOCK TABLES;
/*!40101 SET SQL_MODE=@OLD_SQL_MODE;*/
/*!40014SET
FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;*/
```

5.1.5 Implementasi Antar Muka

Dalam bahasa delphi 7 implementasi antarmuka dengan sebuah tampilan form yang berekstensi form dalam sebuah project. Semua form dikumpulkan dalam sebuah project.

5.1.5.1 Implementasi Halaman Utama

Pada implementasi halaman utama akan dibentuk *file unit* dan beberapa file form dimana setiap form membentuk antarmuka pemakai. Untuk lebih jelasnya dapat dilihat sebagai berikut :

Tabel 5.1 Implementasi Menu Utama

No	Nama Form	Deskripsi
1.	File	menu program yang menyediakan submenu untuk melakukan proses login user, logout user, dan keluar dari aplikasi.
2.	Data master	Menu program yang menyediakan submenu untuk menginputkan, mencari, dan menghapus data dokter, dan data obat.

3.	Transaksi	Menu program yang menyediakan submenu untuk melakukan transaksi pencatatan pasien dan pelayanan.
4.	Laporan	Menu program yang menyediakan submenu untuk menampilkan laporan-laporan transaksi.
5.	Help	Menu program yang menyediakan submenu untuk meminta bantuan, dan melihat sekilas tentang pembuat aplikasi.

5.1.5.2 Implementasi Menu File

Pada implementasi submenu file akan dibentuk file unit dan beberapa file form dimana setiap form membentuk antarmuka pemakai. Untuk lebih jelasnya dapat dilihat pada tabel berikut :

Tabel 5.2 Implementasi Menu File

No	Nama Form	Deskripsi
1.	Login	Sub menu untuk masuk ke aplikasi
2.	Logout	Sub menu untuk ganti login user ke aplikasi
3.	Keluar	Sub menu untuk keluar dari aplikasi

5.1.5.3 Implementasi Menu Data Master

Pada implementasi submenu file akan dibentuk file unit dan beberapa file form dimana setiap form membentuk antarmuka pemakai. Untuk lebih jelasnya dapat dilihat pada tabel berikut :

Tabel 5.3 Implementasi Menu Data Master

No	Nama Form	Deskripsi
1.	Dokter	Sub menu untuk masuk menambah, merubah, dan menghapus data dokter
2.	Obat	Sub menu untuk masuk menambah, merubah, dan menghapus data obat

5.1.5.4 Implementasi Menu Transaksi

Pada implementasi submenu file akan dibentuk file unit dan beberapa file form dimana setiap form membentuk antarmuka pemakai. Untuk lebih jelasnya dapat dilihat pada tabel berikut :

Tabel 5.4 Implementasi Menu Transaksi

No	Nama Form	Deskripsi
1.	Pencatatan Data Pasien	Sub menu untuk masuk menangani pendaftaran pasien.
2.	Pelayanan	Sub menu untuk menangani transaksi pelayanan kesehatan.

5.1.5.5 Implementasi Menu Laporan

Pada implementasi submenu file akan dibentuk file unit dan beberapa file form dimana setiap form membentuk antarmuka pemakai. Untuk lebih jelasnya dapat dilihat pada tabel berikut :

Tabel 5.5 Implementasi Menu Laporan

No	Nama Form	Deskripsi
1.	Laporan Registrasi Harian	Sub menu untuk memberikan laporan kunjungan pasien sesuai dengan tanggal periode yang ditentukan
2.	Laporan Obat Keluar	Sub menu untuk memberikan laporan data obat yang keluar tiap hari.

5.2 Pengujian Sistem

Pengujian merupakan bagian yang penting dalam siklus pengembangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal.

5.2.1 Rancangan Pengujian

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 istem atau perangkat lunak memberikan keluaran seperti yang kita harapkan atau tidak.

5.2.1.1 Pengujian Black-Box

Pengujian black box berfokus pada peryaratan fungsional perangkat lunak. Dengan demikian, pengujian black box memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian black box bukan merupakan alternatif dari teknik white-box, tetapi merupakan pendekatan komplementer yang kemungkinan besar mampu mengungkap kelas kesalahan dari pada metode white-box. Pengujian black box berusha menemukan kesalahan dalam kategori sebagai berikut :

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

Tidak seperti pengujian white box, yang dilakukan pada saat awal proses pengujian, pengujian black box cenderung diaplikasikan selama tahap akhir

pengujian, karena pengujian black box memperhatikan struktur kontrol maka perhatian berfokus pada domain informasi.

5.2.2 Rencana Pengujian

Pengujian sistem informasi pelayanan kesehatan berikut menggunakan data uji berupa sebuah data masukan dari kepala apotek.

Tabel 5.6 Rencana Penujian Sistem Informasi Pelayanan Kesehatan

Kelas Uji	Butir Uji	Tinkat pengujian	Jenis Pengujian
Login User	Pemasukan nama pengguna, bagian, dan kata sandi	Sistem	Black Box
Pegujian	Memasukan data pengguna	Sistem	Black Box
Pemasukan Data			

5.2.3 Kasus dan Hasil Pengujian

Penyusun akan mengambil sampel sebagai kasus untuk pengujian sistem.

1. **Pengujian login**

   Pengujian login ini untuk aministrator. Berikut ini adalah tabel pengujian login administrator Apotek Ikhwan Soreang.

Tabel 5.7 Pengujian Login administrator

Kasus dan Hasil Uji			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pengguna: Insan, Nama Bagian : Admin Kata sandi : 1234	Tercantum pada textbox nama pengguna, combobox nama bagian, textbox kata sandi	Dapat mengisi login admin yang diharapkan	[X] diterima [ ] ditolak
Klik masuk	Dapat masuk ke semua menu aplikasi	Tombol masuk dapat berfungsi sesuai yang diharapkan	[X] diterima [ ] ditolak
Klik keluar	Dapat keluar dari form login dan tidak masuk ke	Tombol keluar dapat berfungsi sesuai yang	[X] diterima [ ] ditolak

	menu semua aplikasi	diharapkan	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pengguna: Insan, Nama Bagian : Admin Kata sandi : 1234	Tidak dapat login	Login gagal	[X] diterima [ ] ditolak

2. Pengujian Pemasukan Data

Berikut ini adalah tabel pengujian pemasukan data :

Tabel 5.8 Pengujian Pemasukan Data

Kasus dan Hasil Uji			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan Kode Dokter	Data bisa masuk ke *database*	Data ditolak	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih Kode dokter 1234	Data masuk ke *database*	Data ditolak dengan konfirmasi "data kode dokter salah"	[X] diterima [ ] ditolak

5.2.4 Kesimpulan dan Hasil Pengujian

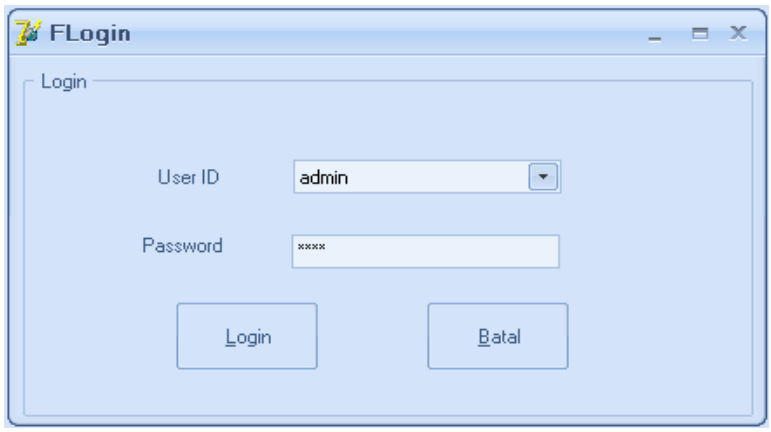
Berdasarkan hasil pengujian dengan kasus uji sample diatas dapat ditarik kesimpulan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional mengeluarkan fungsi sesuai dengan yang diharapkan.

5.3 Tata Cara Penggunaan Program

Dalam tata cara penggunaan program aplikasi ini, terdapat beberapa form yang digunakan yaitu sebagai berikut :

5.3.1 Form Login

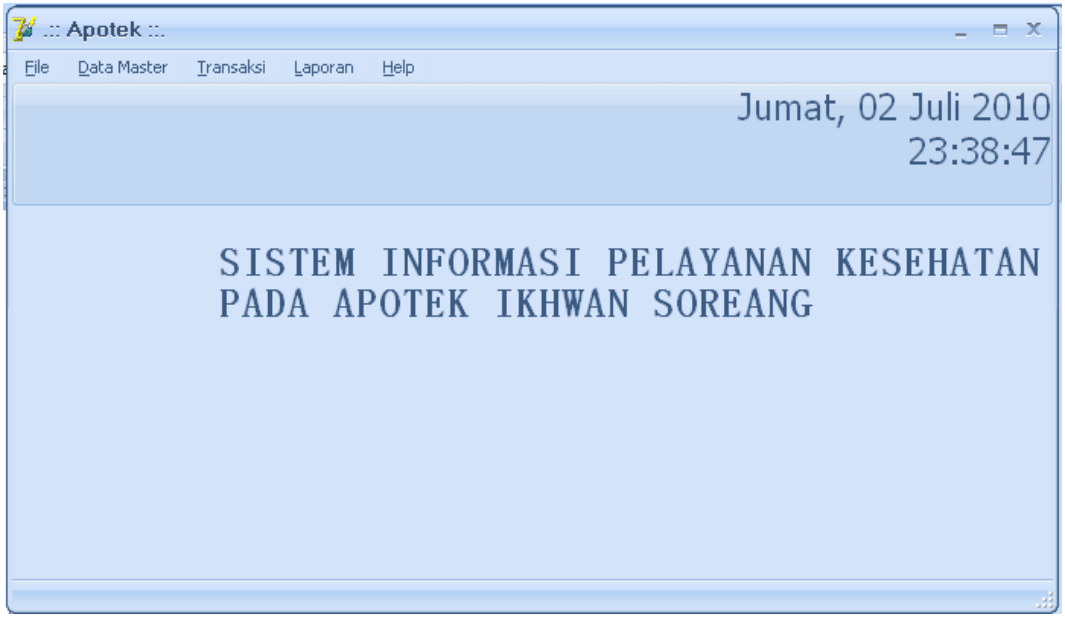
Form ini muncul seseorang ingin mengakses data yang sifatnya *private*. Dapat dilihat di gambar berikut :



Gambar 5.1 Tampilan Menu Login

5.3.2 Form Menu Utama

Form halaman utama muncul ketika program mulai digunakan. Semua orang bisa mengakses data yang ada didalam halaman utama tanpa harus login terlebih dahulu



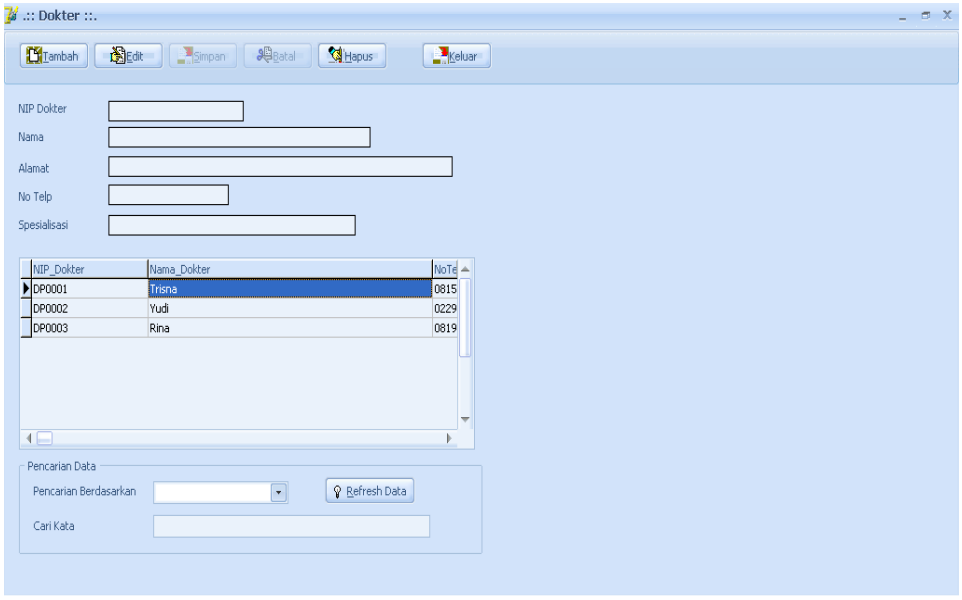
Gambar 5.2 Form Utama

Tabel 5.9 Keterangan Form Menu Utama

No	Nama Objek	Keterangan
1.	File	Berisi sub menu logout, dan keluar
2.	Data Master	Berisi sub menu data dokter, dan data obat
3.	Transaksi	Berisi sub menu pecatatan data pasien dan pelayanan
4.	Laporan	Berisi sub menu laporan data registrasi harian dan laporan data obat keluar
5.	Help	Berisi data bantuan penggunaan dan data pembuat aplikasi

5.3.3 Form Data Dokter

Submenu dokter berisi data dokter yang ditampilkan pada table dokter dan beberapa tombol yang berfungsi untuk memanipulasi data yang terdapat pada tabel dokter seperti tombol tambah, simpan, batal. Ketika pengguna memilih menu data master dokter, maka akan tampil form dokter seperti gambar 5.3



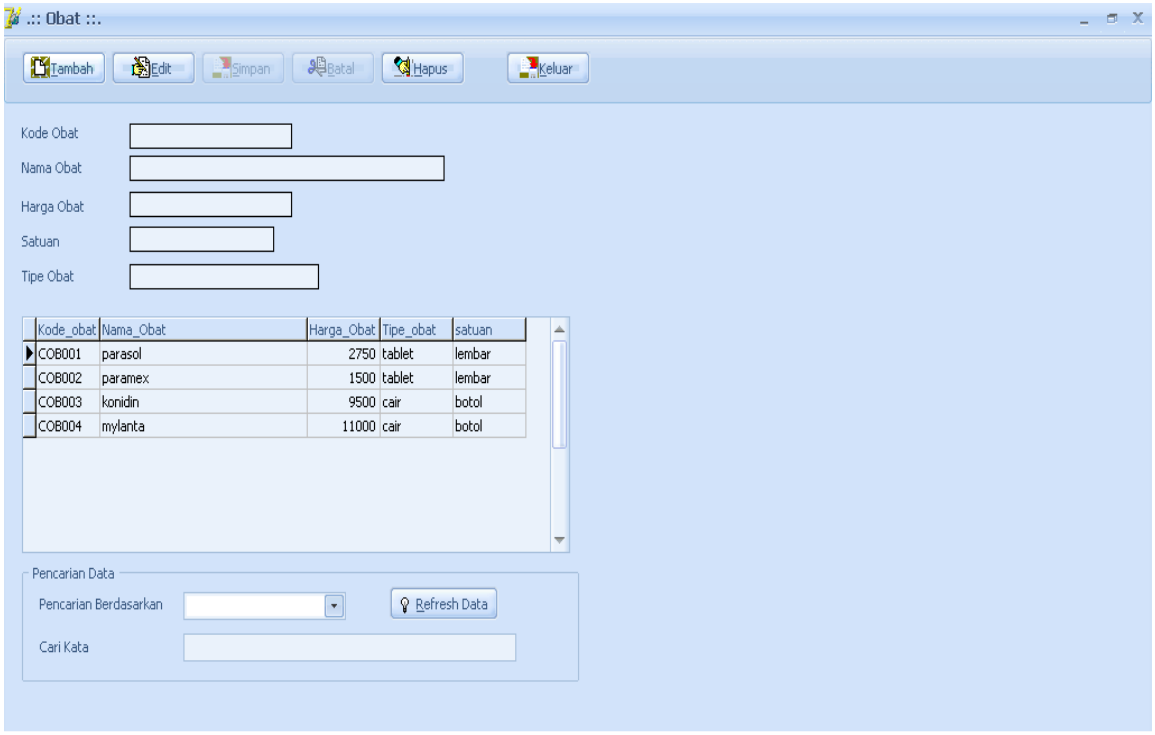
Gambar 5.3 Form Data Dokter

Tabel 5.10 Keterangan Form Dokter

No	Nama Objek	Keterangan
1.	Tombol tambah	Memasukan data dokter yang baru
2.	Tombol edit	Merubah data dokter yang sudah ada
3.	Tombol hapus	Menghapus data dokter yang sudah ada
4.	Tombol keluar	Keluar dari form dokter

5.3.4 Form Data Obat

Submenu obat berisi data obat yang ditampilkan pada table obat dan beberapa tombol yang berfungsi untuk memanipulasi data yang terdapat pada tabel obat seperti tombol tambah, simpan, batal. Ketika pengguna memilih menu data master obat, maka akan tampil form obat seperti gambar 5.4



Gambar 5.4 Form Data Obat

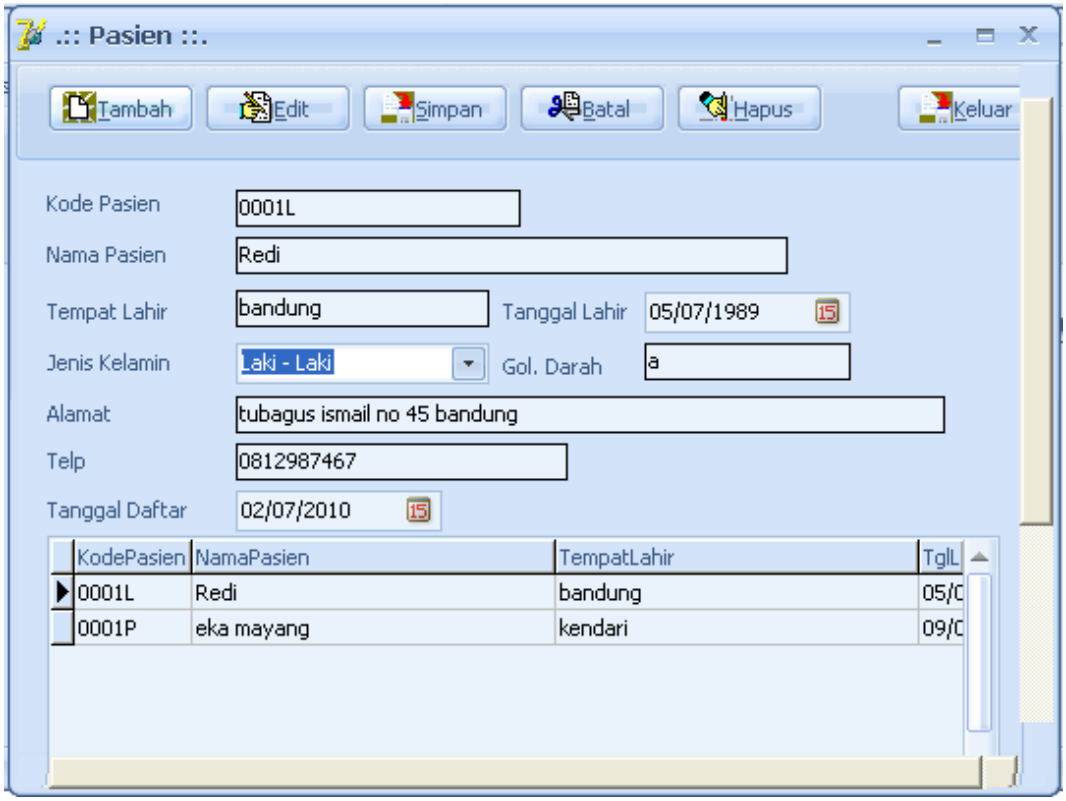
Tabel 5.11 Keterangan Form Obat

No	Nama Objek	Keterangan
1.	Tombol tambah	Memasukan data obat yang baru
2.	Tombol edit	Merubah data obat yang sudah ada
3.	Tombol hapus	Menghapus data obat yang sudah ada
4.	Tombol keluar	Keluar dari form obat

5.3.5 Form Menu Pendaftaran Pasien

Submenu pasien berisi data pasien yang ditampilkan pada table pasien dan beberapa tombol yang berfungsi untuk memanipulasi data yang terdapat pada

tabel pasien seperti tombol tambah, simpan, batal. Ketika pengguna memilih menu pendaftaran pasien, maka akan tampil form pasien seperti gambar 5.5



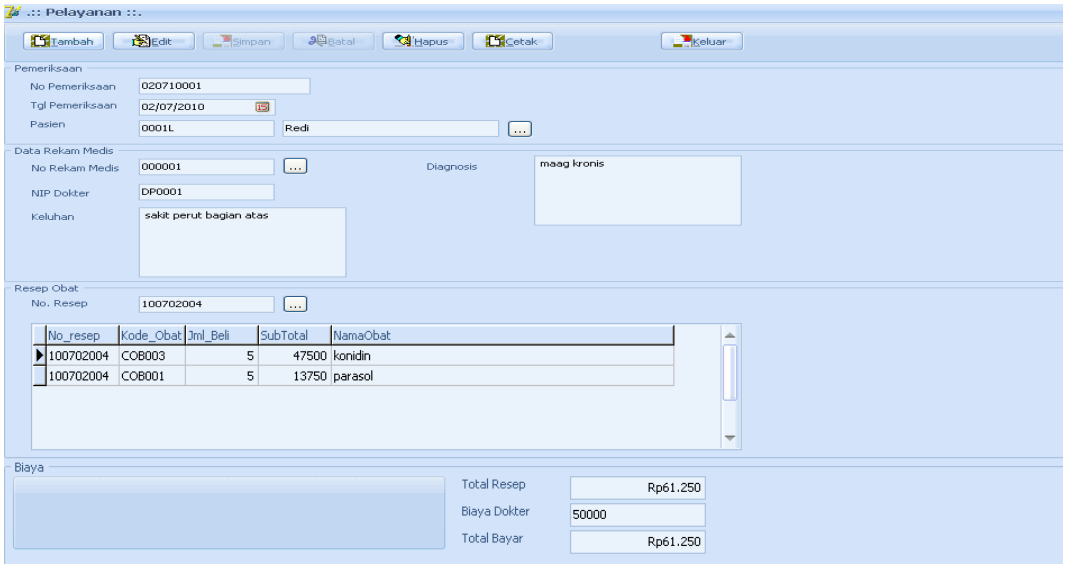
Gambar 5.5 Form Pendaftaran Pasien

Tabel 5.12 Keterangan Form Pendaftaran Pasien

No	Nama Objek	Keterangan
1.	Tombol tambah	Memasukan data pasien yang baru
2.	Tombol edit	Merubah data pasien yang sudah ada
3.	Tombol hapus	Menghapus data pasien yang sudah ada
4.	Tombol keluar	Keluar dari form pasien

5.3.6 Form Menu Transaksi Pelayanan Rekammedik

Submenu rekammedik berisi data rekammedik yang ditampilkan pada table rekammedik dan beberapa tombol yang berfungsi untuk memanipulasi data yang terdapat pada tabel rekammedik seperti tombol tambah, simpan, batal. Ketika pengguna memilih menu pelayanan, maka akan tampil form rekammedik seperti gambar 5.6



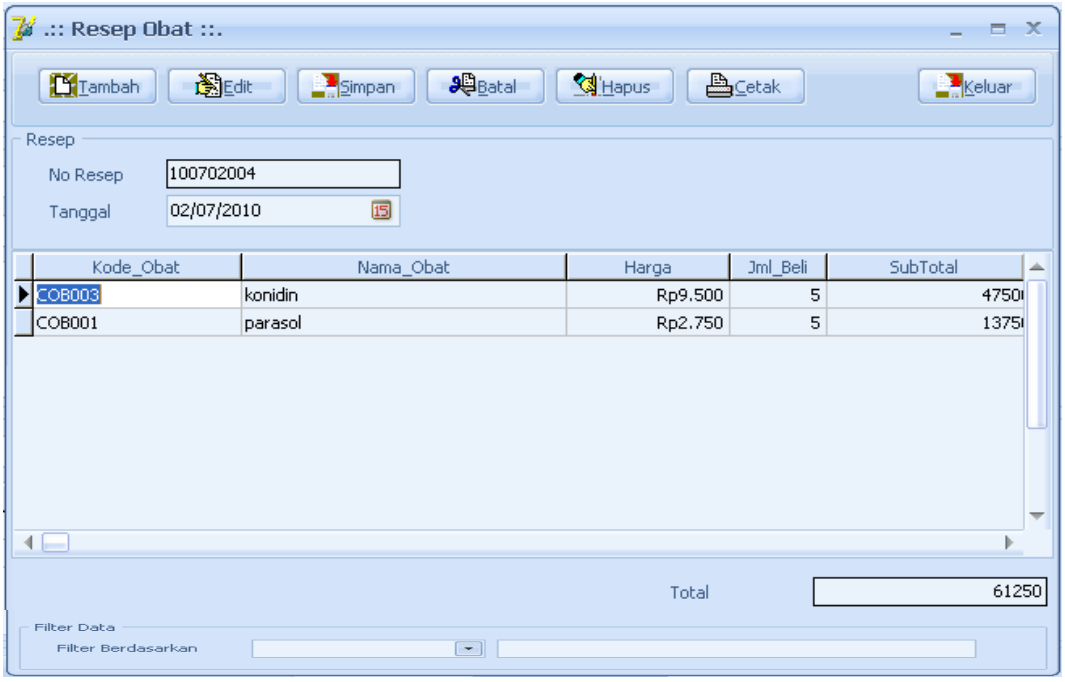
Gambar 5.6 Form Pelayanan Rekammedik

Tabel 5.13 Keterangan Form Pelayanan Rekammedik

No	Nama Objek	Keterangan
1.	Tombol simpan	Memasukan data rekammedik yang baru
2.	Tombol batal	Untuk membatalkan masukan data

5.3.7 Form Menu Transaksi Pelayanan Resep

Submenu resep berisi data resep yang ditampilkan pada table resep dan beberapa tombol yang berfungsi untuk memanipulasi data yang terdapat pada tabel resep seperti tombol tambah, simpan, batal. Ketika pengguna memilih menu pelayanan resep, maka akan tampil form resep seperti gambar 5.7



Gambar 5.7 Form Pelayanan Resep

Tabel 5.14 Keterangan Form Pelayanan Resep

No	Nama Objek	Keterangan
1.	Tombol simpan	Memasukan data resep
2.	Tombol cetak	Untuk mencetak hasil resep