BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Implementasi merupakan tahapan penerapan sistem yang telah dirancang dan dapat diorientasikan secara optimal sesuai dengan kebutuhan. Implementasi yang dilakukan meliputi implementasi perangkat lunak, perangkat keras, Basis data, Antar muka serta Instalasi Program. Sedangkan untuk pengujian meliputi pengujian terhadap perangkat lunak hasil implementasi. Bahasa pemograman yang digunakan dalam Sistem Informasi Pelayanan Kesehatan pada Poliklinik Apotek Batupermata ini menggunakan Java.

5.1.1. Batasan Implementasi

Dalam pengimplementasian perangkat lunak yang dibangun ada beberapa hal yang menjadi batasan implementasi dianataranya :

1. Sistem informasi ini menggunakan database MySQL.
2. Sistem informasi ini hanya dapat menampilakn informasi tentang data pasien, data dokter, data rekam medis, data poli, data penyakit, data obat, data faktur.
3. Implementasi perangkat lunak ini menggunakan topologi jaringan bus untuk menghubungkan satu komputer dengan komputer yang lainnya.

5.1.2. Implementasi Perangkat Lunak

Peragkat lunak yang dibutuhkan oleh server diantaranya adalah sebagai berikut :

1. Sistem operasi : Microsoft windows 7 ultimate.
2. Database : MySQL.
3. Program Aplikasi : Jdk7, Jre7, XAMPP 1.7.0.

Sedangkan perangkat lunak yang dibutuhkan oleh *Client* adalah sebagai berikut :

1. Sistem operasi : Microsoft windows 7 ultimate
2. Program Aplikasi : Jdk7, Jre7.

5.1.3. Implementasi Perangkat Keras

Spesifikasi untuk implementasi perangkat keras yang diusulkan untuk server diantaranya adalah sebagai berikut :

1. Komputer Server
    a. Processor : Intel Core I3
    b. Ram : 4GB
    c. Harddisk : 500GB
    d. VGA : 1GB
    e. Perangkat input dan output : mouse, keyboard, monitor, printer.
2. Komputer Client
    a. Processor : Intel Core2 Duo
    b. Ram : 2GB

c. Harddisk : 80GB

d. VGA : 512MB

e. Perangkat Input dan Output : mouse, keyboard, monitor, printer

5.1.4. Implementasi Basis Data

Berikut adalah implementasi basis data dengan menggunakan MySQL :

```
CREATE TABLE IF NOT EXISTS `daftar_penyakit_pasien` (
  `no` int(10) NOT NULL AUTO_INCREMENT,
  `no_diagnosa` varchar(10) NOT NULL,
  `id_penyakit` varchar(10) NOT NULL,
  `tanggal` date NOT NULL,
  PRIMARY KEY (`no`),
  KEY `id_penyakit` (`id_penyakit`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=72 ;

CREATE TABLE IF NOT EXISTS `daftar_resep_pasien` (
  `no` int(10) NOT NULL AUTO_INCREMENT,
  `no_resep` varchar(10) NOT NULL,
  `id_obat` varchar(10) NOT NULL,
  `jumlah` varchar(20) NOT NULL,
```

```
  PRIMARY KEY (`no`),
  KEY `id_obat` (`id_obat`),
  KEY `no_resep` (`no_resep`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=144 ;

CREATE TABLE IF NOT EXISTS `daftar_tindakan_pasien` (
  `no` int(10) NOT NULL AUTO_INCREMENT,
  `no_tindakan` varchar(10) NOT NULL,
  `id_tindakan` varchar(10) NOT NULL,
  `tanggal` date NOT NULL,
  PRIMARY KEY (`no`),
  KEY `id_tindakan` (`id_tindakan`),
  KEY `no_tindakan` (`no_tindakan`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=128 ;

CREATE TABLE IF NOT EXISTS `data_dokter` (
  `id_dokter` varchar(10) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `id_poli` varchar(10) NOT NULL,
  `jenis` varchar(20) NOT NULL,
  `tgl_lahir` date NOT NULL,
```

```
  `tempat_lahir` varchar(30) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `no_ponsel` varchar(15) NOT NULL DEFAULT '',
  `status` varchar(20) NOT NULL DEFAULT '',
  PRIMARY KEY (`id_dokter`),
  KEY `id_poli` (`id_poli`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `data_ft` (
  `no_ft` int(10) NOT NULL AUTO_INCREMENT,
  `id_pasien` varchar(10) NOT NULL,
  `no_tindakan` varchar(10) NOT NULL,
  `no_resep` varchar(10) NOT NULL,
  `no_diagnosa` varchar(10) NOT NULL,
  `id_dokter` varchar(10) NOT NULL,
  `tanggal` date NOT NULL,
  `status` varchar(20) NOT NULL,
  `keluhan` varchar(40) NOT NULL,
  `alergi` varchar(40) NOT NULL,
  PRIMARY KEY (`no_ft`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=16 ;
```

```
CREATE TABLE IF NOT EXISTS `data_karyawan` (
  `username` varchar(20) NOT NULL,
  `bagian` varchar(20) NOT NULL,
  `password` varchar(20) NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `data_kunjungan` (
  `no_kunjungan` int(10) NOT NULL AUTO_INCREMENT,
  `id_pasien` varchar(10) NOT NULL,
  `id_dokter` varchar(10) NOT NULL,
  `tgl_kunjungan` date NOT NULL,
  `status` varchar(20) NOT NULL,
  PRIMARY KEY (`no_kunjungan`),
  KEY `id_pasien` (`id_pasien`),
  KEY `id_dokter` (`id_dokter`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
AUTO_INCREMENT=96 ;

CREATE TABLE IF NOT EXISTS `data_obat` (
  `id_obat` varchar(10) NOT NULL,
  `nama` varchar(30) NOT NULL,
  `satuan` varchar(20) NOT NULL,
```

```
  `stok` int(10) NOT NULL,
  `harga` int(10) NOT NULL,
  `tgl_exp` date NOT NULL,
  PRIMARY KEY (`id_obat`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `data_pasien` (
  `id_pasien` varchar(10) NOT NULL,
  `nama` varchar(30) NOT NULL,
  `jenis` varchar(20) NOT NULL,
  `tgl_lahir` date NOT NULL,
  `umur` int(3) NOT NULL,
  `tempat_lahir` varchar(50) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `no_ponsel` varchar(15) NOT NULL,
  `tgl_daftar` datetime NOT NULL,
  PRIMARY KEY (`id_pasien`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `data_penyakit` (
  `id_penyakit` varchar(10) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `kategori` varchar(5) NOT NULL,
```

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

CREATE TABLE IF NOT EXISTS `data_poli` (
  `id_poli` varchar(10) NOT NULL,
  `nama` varchar(20) NOT NULL,
  `status` varchar(20) NOT NULL DEFAULT '',
  PRIMARY KEY (`id_poli`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `data_rs` (
  `id_rs` varchar(10) NOT NULL,
  `nama_rs` varchar(30) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  `no_kontak` varchar(12) NOT NULL,
  PRIMARY KEY (`id_rs`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `data_tindakan` (
  `id_tindakan` varchar(10) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `kategori` varchar(5) NOT NULL,
  `harga` int(10) NOT NULL,
```

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

5.1.5. Implementasi Antar Muka

Implementasi antar muka dibuat dari dalam form yang telah dibuat di NetBeans 7.3. setiap halaman *Frame* mempunyai ekstensi .java. *Frame* tersebut merupakan sarana penghubung sistem dengan user.

1. Implementasi halaman utama

   Halaman utama merupakan halaman yang akan muncul setelah user melakukan login sesuai bagiannya masing-masing. Menu utama ini akan menjadikan penghubung antara user dengan sistem.

   Berikut ini merupakan struktur menu pada halaman utama sistem informasi pelayanan kesehatan pada poliklinik apotek batupermata.

Tabel 5.1 Implementasi halaman utama

Sub Menu	Keterangan	Nama File
File	Sub menu yang digunakan untuk memanggil form logout dan exit	Menu.java

Bagian Pendaftaran	Sub menu yang digunakan untuk memanggil form pendaftaran, input data rekam medis, data dokter, data poli, data penyakit.	Menu.java
Apoteker	Sub menu yang berfungsi untuk memanggil form cetak faktur resep, tindakan, data obat, lihat status faktur	Menu.java
Administrasi	Sub menu yang berfungsi untuk memanggil form input pembayaran, surat rujukan, surat sakit	Menu.java

Laporan	Sub menu yang berfungsi untuk memanggil form laporan data pasien, laporan data kunjungan pasien, laporan data faktur, laporan data obat, laporan data penyakit	Menu.java

2. Implementasi halaman utama berdasarkan File

Tabel 5.2 Implementasi halaman utama berdasarkan file

Sub Menu	Keterangan	Nama File
Logout	Sub menu ini berfungsi untuk keluar dari sistem user yang sudah melakukan login, namun masih berada dalam sistem.	Menu.java
Exit	Sub menu ini berfungsi untu	Menu.java

	keluar dari sistem	

3. Implementasi halaman utama berdasarkan Bagian pendaftaran

Tabel 5.3 Implementasi halaman utama berdasarkan bagian pendaftaran

Sub menu	Keterangan	Nama File
Pendaftaran	Sub menu ini berfungis untuk menagani pasien yang ingin melakukan pendftaran ke poliklinik apotek batupermata	Data_pasien.java
Input data rekam medis	Sub menu ini berfungsi untuk menginputkan semua data yang ada didalam kartu rekam medis manual berdasarkan catatan dari dokter yang	Catatan_medis.java

	bersangkutan	
Data dokter	Sub menu ini berfungsi untuk menangani data dokter	Tabel_dokter.java
Data poli	Sub menu ini berfungsi untuk menangani data poli	Data_poli.java
Data penyakit	Sub menu ini berfungsi untuk menangani data penyakit	Data_penyakit.java

4. Implementasi halaman utama berdasarkan Apoteker

Tabel 5.4 Implementasi halaman utama berdasarkan apoteker.

Sub menu	**Keterangan**	**Nama File**
Cetak	Sub menu ini berfungsi untuk memanggil form cetak faktur resep	Menu.java

	dan tindakan	
Lihat Status Faktur	Sub menu ini berfungsi untuk melihat status data faktur yang telah buat.	Status_pembayaran_resep.java
Data_obat	Sub menu ini berfungsi untuk menangani data obat.	Data_obat.java

5. Implementasi halaman utama berdasarkan Administrasi

Tabel 5.5 Implementasi halaman utama berdasarkan administrasi

Sub menu	**Keterangan**	**Nama File**
Input pembayaran	Berfungsi untuk mengubah status pembayaran resep obat.	Input_pembayaran.java
Surat rujukan	Sub menu ini berfungsi untuk membuat surat rujukan	Surat_rujukan.java

	berdasarkan rujukan dokter	
Surat sakit	Sub menu ini berfungsi untuk membuat surat keterangan sakit berdesarkan keterangan sakit dokter	Surat_sakit.java

6. Implementasi halaman utama berdasarkan Laporan

Tabel 5.6 Implementasi halaman utama berdasarkan laporan

Sub menu	**Keterangan**	**Nama File**
Laporan data pasien	Sub menu ini berfungsi untuk menampilkan laporan seluruh data pasien	Laporan_data_pasien.java
Laporan data	Sub menu ini berfungsi	Laporan_kunjungan_pasien.java

kunjungan pasien	untuk menampilkan seluruh data kunjungan pasien	
Laporan data faktur	Sub menu ini berfungsi untuk menampilkan laporan data faktur	Menu.java
Laporan data obat	Sub menu ini berfungsi untuk menampilkan data obat yang terpakai	Data_obat_keluar.java
Laporan data penyakit	Sub menu ini berfungsi untuk menampilkan data penyakit	Data_penyakit_keluar.java

5.1.6. Implementasi Instalasi Program

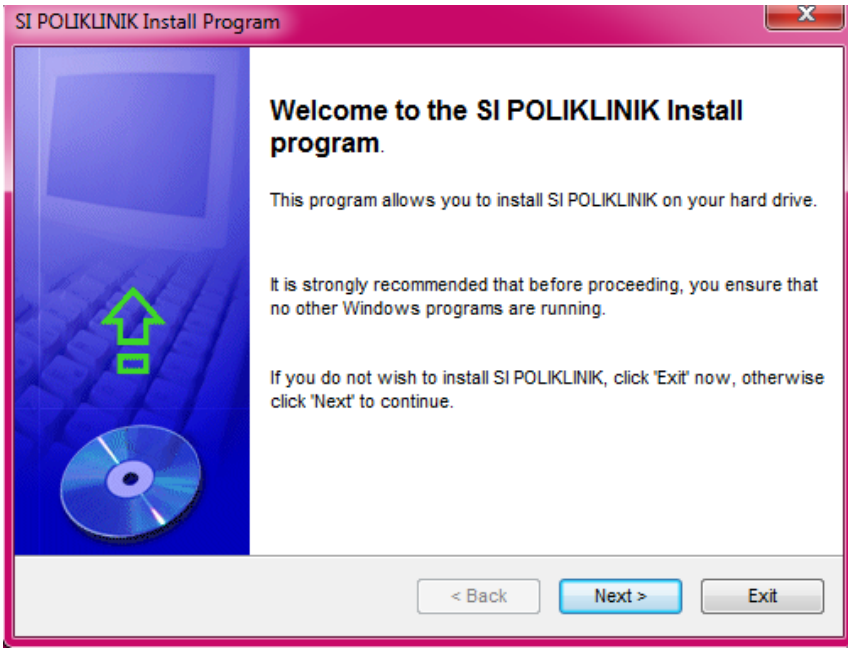
Berikut ini merupakan tahapan-tahapan sebelum menggunakan program sistem informasi pelayanan kesehatan pada poliklinik apotek batupermata adalah sebagai berikut :

1. Klik pada ikon setup yang ada pada gambar dibawah ini



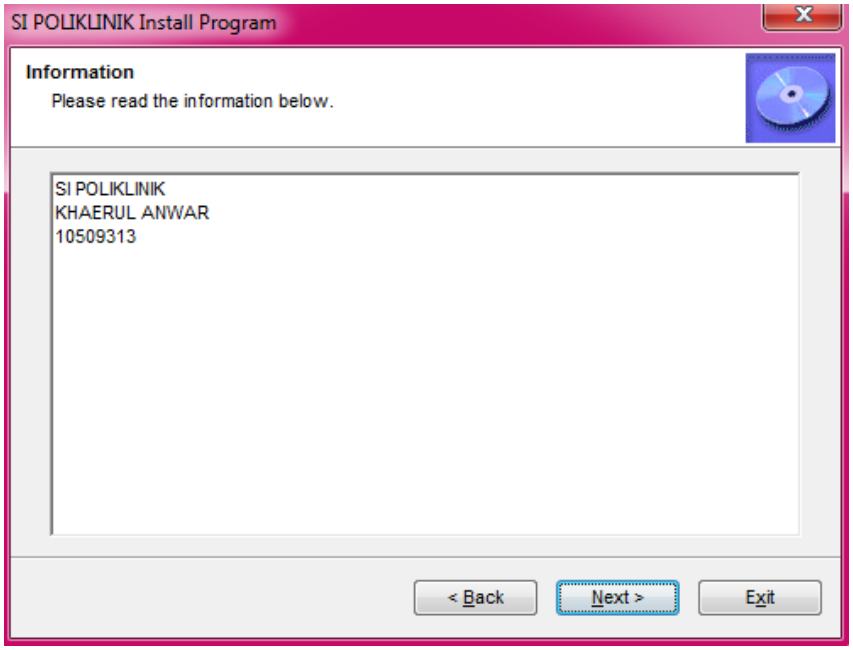
Gambar 5.1 Ikon Setup SI klinik

2. Kemudian akan tampil kotak dialog seperti dibawah ini dan klik tombol next sehingga akan melanjutkan ke proses selanjutnya.



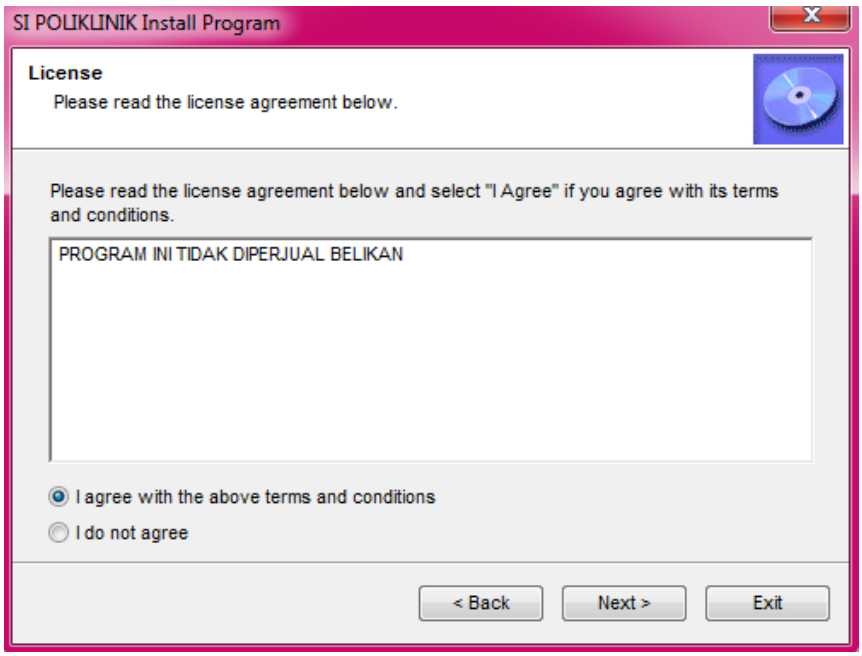
Gambar 5.2 instalasi proses 1

3. Kemudia akan tampil informasi kotak dialog dari sistem informasi poliklinik ini dan kemudian klik tombol next



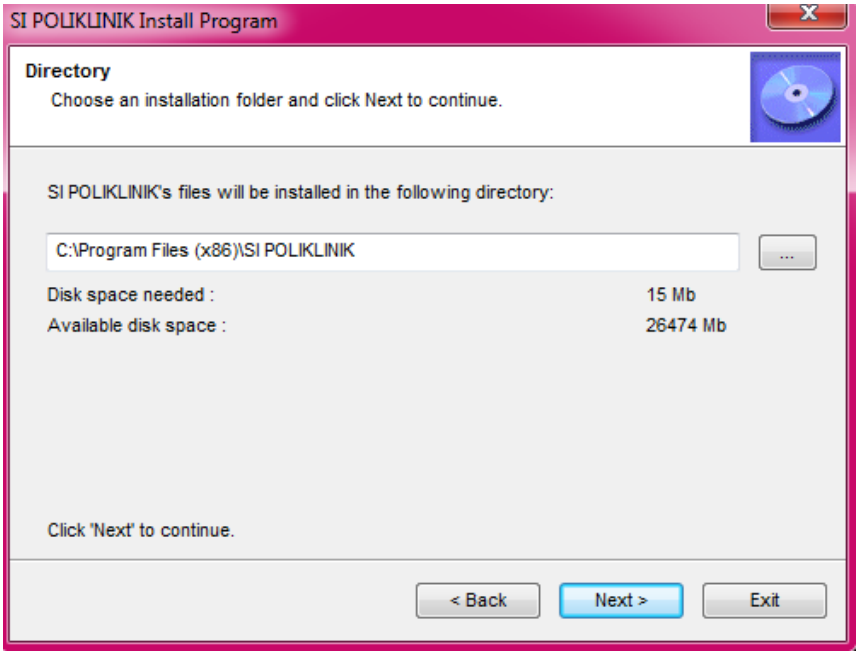
Gambar 5.3 instalasi proses 2

4. Tampil kotak dialog license yang menerangkan tentang software sistem informasi poliklinik ini dan poli pada bagian " I agree with the above terms and conditions" dan klik tombol next.



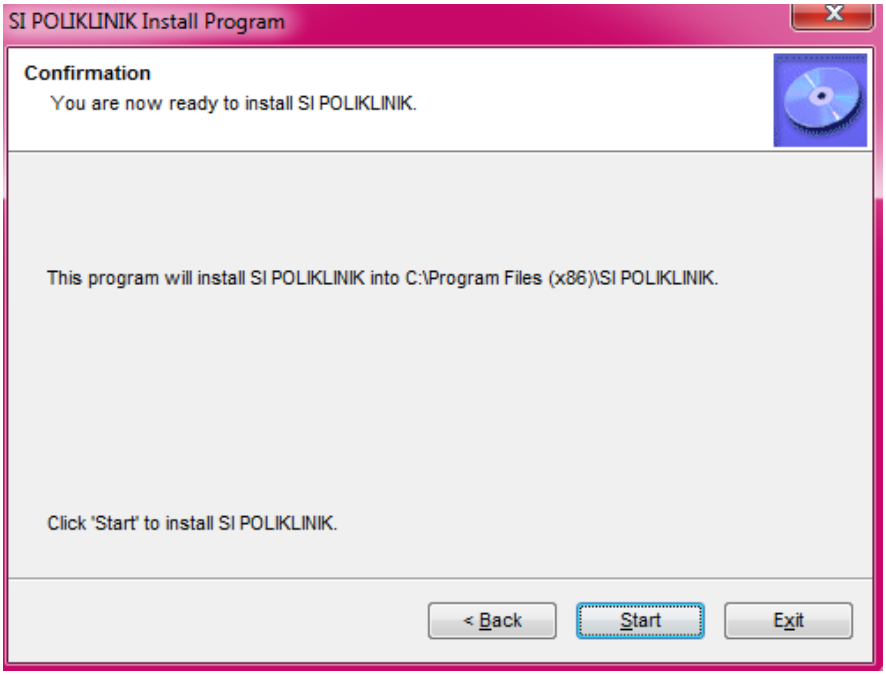
Gambar 5.4 instalasi proses 3

5. Kemudian akan tampil kotak dialog diamana bisa memilih tempat tujuan untuk instalasi program ini atau bisa memilih secara *default* dari program. Ketika sudah menentukan tempat pilihan instalasi kemudian klik tombol next.



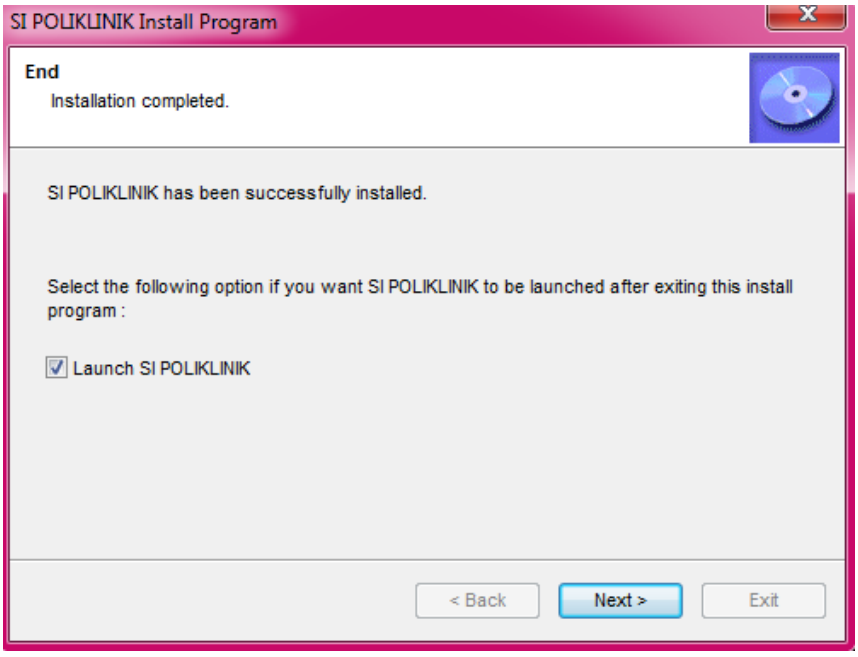
Gambar 5.5 instalasi proses 4

6. Kemudia akan muncul keterangan berhasil dalam instalasi program dan klik start



Gambar 5.6 instalasi proses 5

7. Kemudian akan tampil pilihan cek box pada kotak dialog seperti dibawah ini dan pilih cek box tersebut ketika ingin menjalankan program setelah instalasi selesai.



Gambar 5.7 instalasi proses 6

5.1.7. Penggunaan Program

1. Form Login

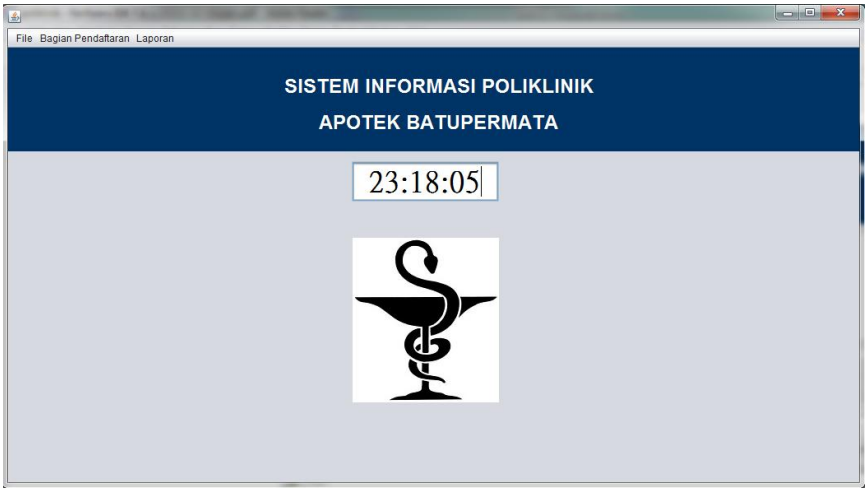
   Form login ini berfungsi untuk masuk kedalam sistem berdasarkan hak akses yang telah diberikan kepada setiap bagian user.



Gambar 5.8 Form login

2. Menu utama pada bagian pendaftaran

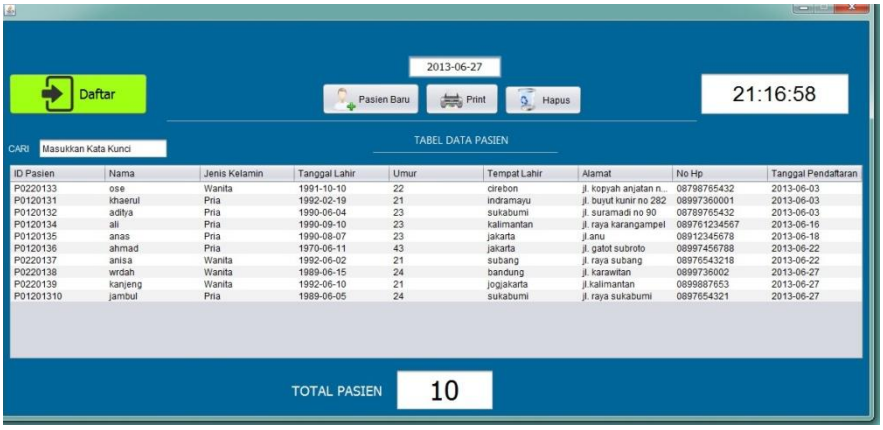
   Pada bagian menu pendafaran ini terdapat tiga sub menu yaitu file, pendaftaran dan laporan.



Gambar 5.9 Menu bagian pendaftaran

Didalam menu pendaftaran ini user bagian pendaftaran dapa mengakses data-data sebagai berikut :

a. Form pendaftaran Pasien

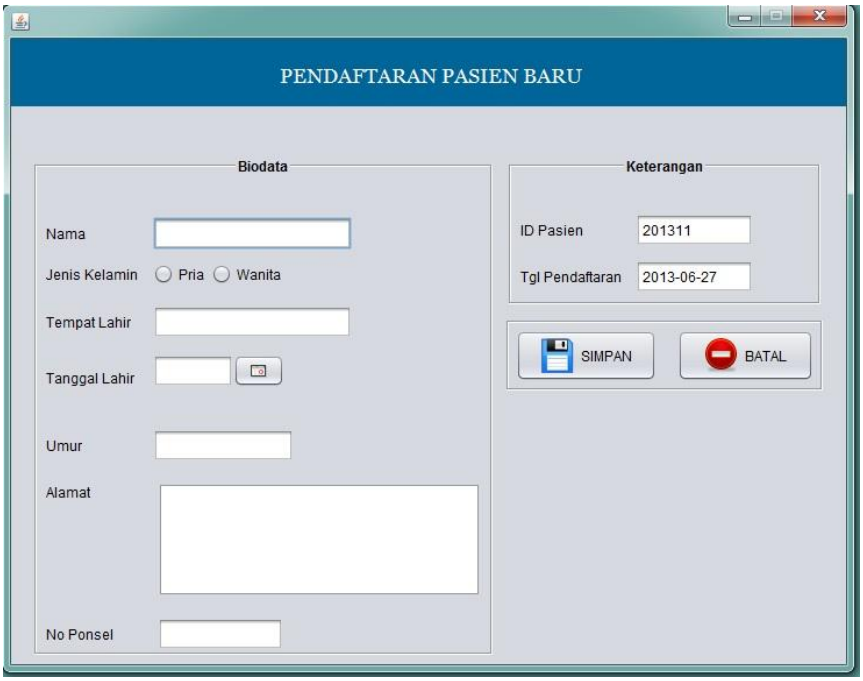


Gambar 5.10 Form pendaftaran

Form ini berfungsi untuk menampilkan semua data pasien, terdapat empat tombol, yaitu tombol daftar, pasien baru, print dan hapus. Tombol daftar akan berfungsi setelah user melakukan pemilihan data pasien yang ditampilkan didalam tabel data pasien, fungsinya untuk memasukkan data pasien

kedalam data kunjungan pasien. Tombol Pasien baru berfungsi untuk memanggil form untuk penambahan pasien baru, tombol print akan berfungsi setelah user melakukan pemilihan data pasien yang ditampilkan didalam tabel data pasien, fungsinya untuk mengambil data untuk dicetak dan tombol hapus berfungsi untuk menghapus data yang ada didalam tabel data pasien setelah user memilih data pasien yang akan dihapus.

b. Form pendaftaran pasien baru



Gambar 5.11 form pendaftaran pasien baru

Tombol simpan berfungsi untuk menyimpan data kedalam database setelah data-data yang telah dimasukkan kedalam form pendaftaran pasien ini secara lengkap.

c. Form cetak kartu



Gambar 5.12 Form cetak kartu

Gambar kartu yang ada disebelah kiri berfungsi untuk mencetak kartu pasien dan gambar kertas yang ada disebelah kanan berfungsi untuk mencetak kartu pasien berdasarkan data yang sudah diilih oleh user bagian pendaftaran pada form pendaftaran pasien.

d. Form memilih poli



Gambar 5.13 form memilih poli

Form ini berfungsi untuk memilih poli dan dokter siapa yang akan memeriksa pasien. terdapat tombol daftar dan tombol batal, tombol daftar untuk menyimpan data dari hasil pemilihan poli dan dokter yang telah dipilih kedalam database. Tombol batal untuk keluar dari sistem.

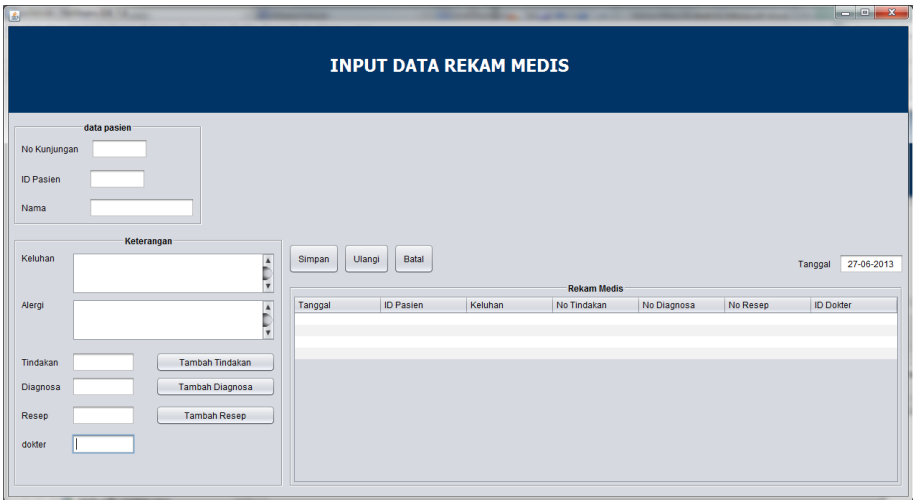
e. Form daftarr registrasi pasien



Gambar 5.14 form daftar registrasi pasien

Form ini berfungsi setelah user melakukan penginputan data pasien kedalam data kunjungan pasien, terdapat tombol diagnosa yang berfungsi untuk mengakses form input data rekam medis setelah memilih data yang ada didalam tabel daftar registrasi pasien, namun setelah data rekam medis pasien telah diinputkan maka data pasien yang ada didalam daftar pasien tersebut akan hilang.

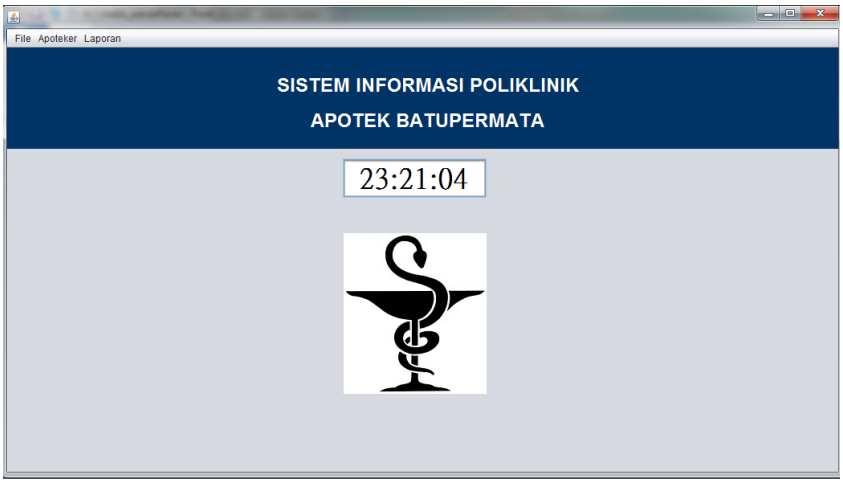
f. Form input data rekam medis



Gambar 5.15 form input data rekam medis

Form ini berfungsi untuk memasukkan data rekam medis berdasarkan data rekam medis yang telah diisi oleh dokter yang bersangkutan, terdapat tombol tambah tindakan yang berfugsi untuk menambahkan tindakan-tindakan apa saja yang ada didalam kartu rekam medis, tombol tamah diagnosa yang berfugsi untuk menambahkan diagnosa penyakit apa saja yang ada didalam kartu rekam medis, tombol tambah resep yang berfugsi untuk menambahkan obat-obat apa saja yang ada didalam kartu rekam medis, setelah data sudah lengkap maka klik tombol simpan. Maka data pasien akan masuk kedalam data rekam medis.

3. Menu utama pada bagian apoteker



Gambar 5.16 menu utama pada bagian apoteker

Pada menu utama bagian apoteker ini user dapat mengakses data-data obat untuk membuat faktur resep, melihat status faktur, mencetak laporan faktur resep obat.

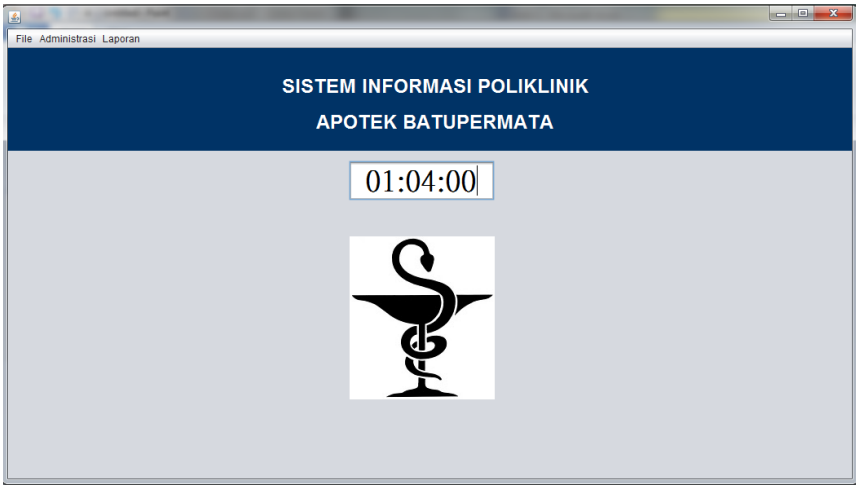
a. Form pembuatan faktur



Gambar 5.17 Form pembuatan faktur

Form ini dapat menampilkan daftar yang sudah di anjurkan oleh dokter berdasarkan resep dokter, terdapat tombol cetak , yaitu untuk mencetak faktur yang isinya tentang rincian biaya tindakan dokter dan rincian biaya obat.

4. Menu utama bagian administrasi



Gambar 5.18 menu utama bagian administrasi

Pada menu utama bagian administrasi ini user dapat mengakses data-data faktur yang telah dibuat dibagian apoteker, membuat surat keterangan sakit, membuat surat rujukan.

a. Form pembuatan surat keterangan sakit dan rujukan



Gambar 5.19 form pembuatan surat sakit dan rujukan

Form ini berfungsi untuk mencetak surat keterangan sakit dan surat rujukan berdasarkan keterangan dari dokter. Terdapat tombol surat rujukan yaitu untuk mencetak surat rujukan dan tombol surat sakit untuk mencetak surat keterangan sakit.

5.2. Pengujian

Pengujian sistem dilakukan untuk memeriksa sistem yang akan diterapkan mempunyai fungsi yang sesuai dengan kebutuhan yang diharapkan. Tujuan utama dari sistem ini untuk memastikan bahwa semua komponen-komponen yang ada didalam sistem berjalan dengan fungsinya, sehingga dapat menghasilkan sebuah informasi yang diharapkan.

5.2.1. Rencana Pengujian

Rencana pengujian yang dilakukan adalah menggunakan pengujian *black box* . metode pengujian dengan *black box* ini fokus terhadapa persyaratan fungsional perangkat lunak yang dibuat. Pengujian aplikasi sistem informasi pelayanan kesehatan pada poliklinik apotek batupermata ini terdiri dari :

Tabel 5.7. Rencana Pengujian

Requirement yang di uji	Butir uji
Login User	Melakukan login
Pendaftaran	menampilkan form pendaftaran
Input data rekam medis	menampilkan form daftar pasien yang sudah melakukan registrasi sesuai poliklinik yang dipilih
Data dokter	Menampilkan form tabel dokter
Data poli	Menampilkan form data poli
Data penyakit	Menampilkan form data penyakit
Pembuatan faktur	Menampilkan form pembuatan faktur
Lihat status faktur	Menampilkan form status faktur
Data obat	Menampilkan form data obat
Input pembayaran	Menampilkan form pembayaran

Surat rujukan	Menampilkan form pembuatan surat rujukan
Surat sakit	Menampilkan form pembuatan surat sakit
Laporan data pasien	Menampilkan form pembuatan laporan data pasien
Laporan data kunjungan pasien	Menampilkan form pembuatan laporan data kunjungan pasien
Laporan data faktur	Menampilkan form pembuatan laporan data faktur
Laporan data obat	Menampilkan form pembuatan laporan data obat
Laporan data penyakit	Menampilkan form pembuatan laporan data penyakit

5.2.2. Kasus dan Hasil Pengujian

Berikut ini merupakan uraian pengujian dengan menggunakan pengujian *Black Box* berdasarakan Requirement rencana pengujian.

Tabel 5.8. Kasus dan Hasil Pengujian

Requirement	**Sekenario uji**		**Hasil yang diharapkan**	**Hasil pengujian**
Login	1	Input data login (benar)	Masuk ke menu sesuai hak akses yang telah ditentukan	[ ] Diterima [ ] Ditolak
	2	Input data login (username atau password salah)	Menampilkan “username atau password tidak sesuai”	[ ] Diterima [ ] Ditolak
Pendaftaran	1	Klik tombol Pasien baru	Menampilkan form untuk menambahkan pasien baru	[ ] Diterima [ ] Ditolak
	2	Klik tombol Print	Menampilkan pilihan untuk menctekan kartu pasien	[ ] Diterima [ ] Ditolak

			dan kartu rekam medis	
	3	Klik tombol Hapus	Menghapus data pasien	[ ] Diterima [ ] Ditolak
	4	Klik tombol Daftar	Menampilkan form unuk memilih poli setelah pasien melakukan pendaftaran	[ ] Diterima [ ] Ditolak
Input data rekam medis	1	Pilih isi yang ada di dalam tabel dan klik tombol diagnosa	Menampilkan form pengisian hasil rekam medis berdasarkan kartu rekam medis dokter	[ ] Diterima [ ] Ditolak
Data dokter	1	Klik tombol dokter baru	Menampilkan form untuk menambah dokter baru	[ ] Diterima [ ] Ditolak
	2	Pilih isi yang ada didalam tabel dan	Menampilkan form untuk edit	[ ] Diterima [ ] Ditolak

		klik tombol edit	data dokter	
	3	Pilih isi yang ada didalam tabel dan klik tombol hapus	Menghapus data dokter	[ ] Diterima [ ] Ditolak
Data poli	1	Klik tombol simpan	Menyiman data poli berdasarkan data-data yang sudah terisi	[ ] Diterima [ ] Ditolak
	2	Pilih isi tabel poli dan klik tombol edit	Mengedit data poli	[ ] Diterima [ ] Ditolak
	3	Klik tombol update	Memperbaharui data poli	[ ] Diterima [ ] Ditolak
	4	Klik tombol hapus	Menghapus data poli	[ ] Diterima [ ] Ditolak
Input pembayaran	1	Klik tombol simpan	Menyimpan perubahan status data faktur	[ ] Diterima [ ] Ditolak
	2	Klik tombol batal	Keluar dari sistem	[ ] Diterima [ ] Ditolak
Surat rujukan	1	Klik tombol cetak	Mencetak surat	[ ] Diterima

			rujukan	[ ] Ditolak
	2	Klik tombol batal	Keluar dari sistem	[ ] Diterima [ ] Ditolak
Surat sakit	1	Klik tombol cetak	Mencetak surat keterangan sakit	[ ] Diterima [ ] Ditolak
	2	Klik tombol batal	Keluar dari sistem	[ ] Diterima [ ] Ditolak
Laporan data pasien	1	Klik tombol cetak	Mencetak laporan data pasien berdasarkan tanggal dan tahun yang sudah di pilih	[ ] Diterima [ ] Ditolak
Laporan data kunjungan pasien	1	Klik tombol cetak	Mencetak laporan data kunjungan pasien berdasarkan tanggal dan tahun yang sudah di pilih	[ ] Diterima [ ] Ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji *sample* di atas dapat ditarik kesimpulan bahwa perangkat lunak Sistem Informasi pelayanan kesehatan pada poliklinik apotek batupermata secara fungsional mengeluarkan hasil output sesuai dengan yang diharapkan.