BAB IV

HASIL DAN PEMBAHASAN

4.1 Perancangan Sistem

Perancangan sistem merupakan tahap selanjutnya dari fase pembuatan sistem informasi yang diusulkan setelah mengetahui bagaimana kondisi sistem yang sedang berjalan untuk memperbaiki kelemahan yang ada didalam sistem tersebut.

4.1.1. Tujuan Perancangan Sistem

Tujuan perancangan suatu sistem secara global adalah membentuk kerangka sistem pengolahan data dengan bantuan komputer agar sistem yang ada menjadi lebih terkomputerisasi.

Sedangkan tujuan utama dari perancangan sistem secara umum adalah untuk memberikan gambaran secara umum kepada pemakai (*user*) mengenai sistem informasi yang baru, perancangan sistem secara umum juga sudah dapat mengenai komponen sistem informasi yang akan di desain.

Penentuan persyaratan sistem dilakukan agar arah perancangan sistem dapat benar - benar terarah pada sasaran, oleh sebab itu sistem yang dirancang harus memenuhi batasan sistem dimana perancangan sistem ini merupakan kebutuhan fungsional dan persiapan untuk rancang bangun implementasi menggambarkan bagaimana suatu sistem di bentuk. Pada tahap perancangan sistem informasi di rancang dengan tujuan komunikasi kepada pemakai bukan untuk pembuat program.

4.1.2. Gambaran Umum Sistem yang Diusulkan

Dimana pada prosedur yang dibuat tidak mengalami banyak perubahan dari sistem yang sedang berjalan, hanya berbeda dalam penggunaan sistem komputerisasi yang dapat membantu proses penginputan, pengeditan, dan penghapusan data serta dapat mempermudah dalam hal pencarian data.

4.1.3. Perancangan Prosedur yang Diusulkan

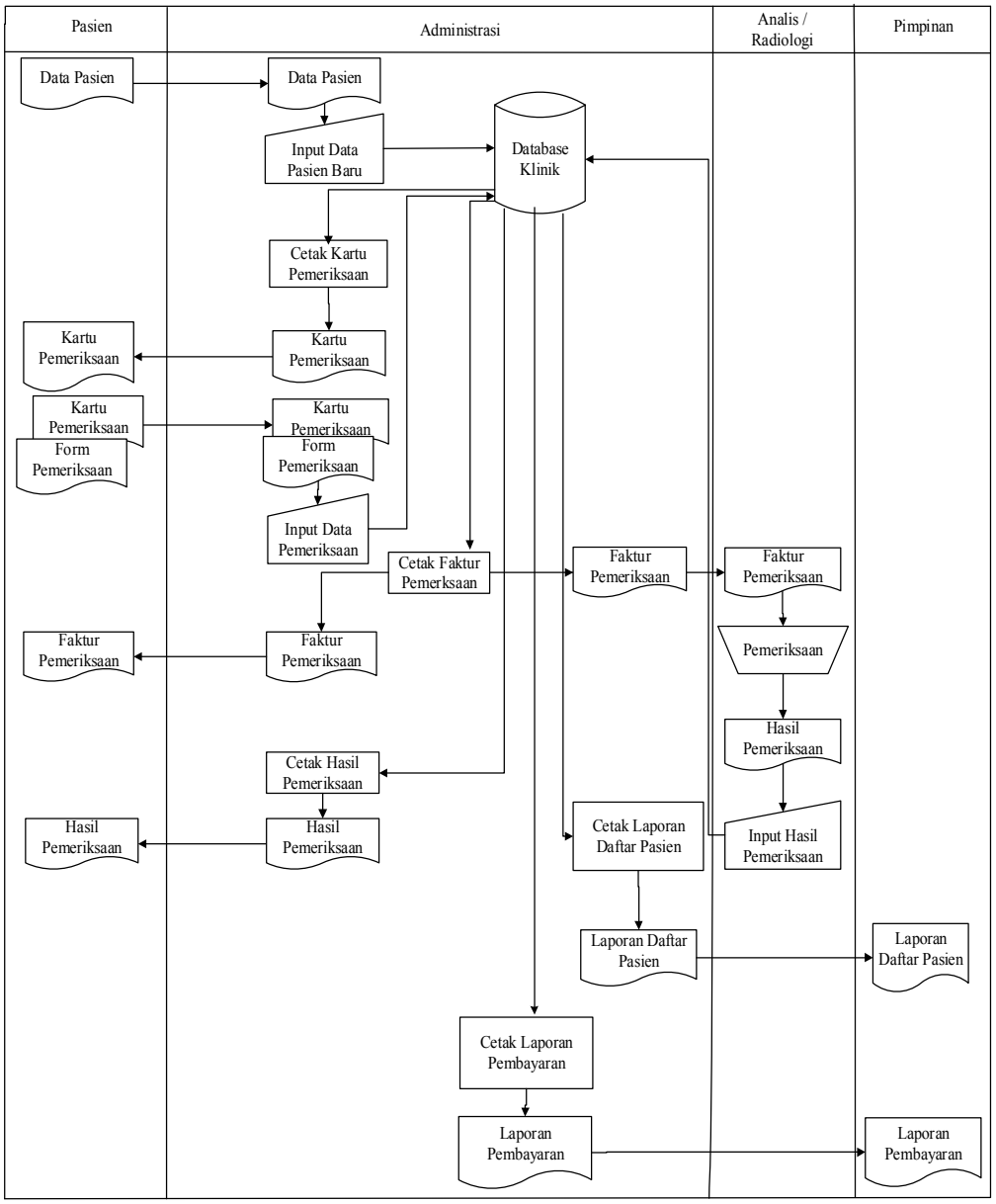
Perancangan prosedur yang diusulkan adalah jika yang lama menggunakan cara manual dan dokumen atau data - data berbentuk buku sedangkan sistem informasi yang baru akan memaksimalkan alat bantu komputer yang ada dan dilakukan secara otomatis.

Proses atau prosedur pelayanan kesehatan yang diusulkan adalah :

1. Pasien datang ke bagian administrasi untuk melakukan pendaftaran, kemudian bagian administrasi akan menanyakan apakah sudah pernah melakukan pemeriksaan sebelumnya atau belum. Apabila belum pernah maka akan dibuatkan kartu pemeriksaan untuk pasien.
2. Selanjutnya bagian administrasi akan menginputkan pemeriksaan apa yang akan dilakukan oleh pasien berdasarkan dari form pemeriksaan ataupun permintaan pasien ke dalam database.
3. Selanjutnya bagian administrasi membuat faktur pembayaran lalu pasien melakukan pembayaran secara tunai.
4. Selanjutnya pasien diperiksa oleh pihak medis dan hasil pemeriksaan akan diinputkan ke database.
5. Hasil pemeriksaan di cetak kemudian diberikan kepada pasien.

4.1.3.1. Flow Map

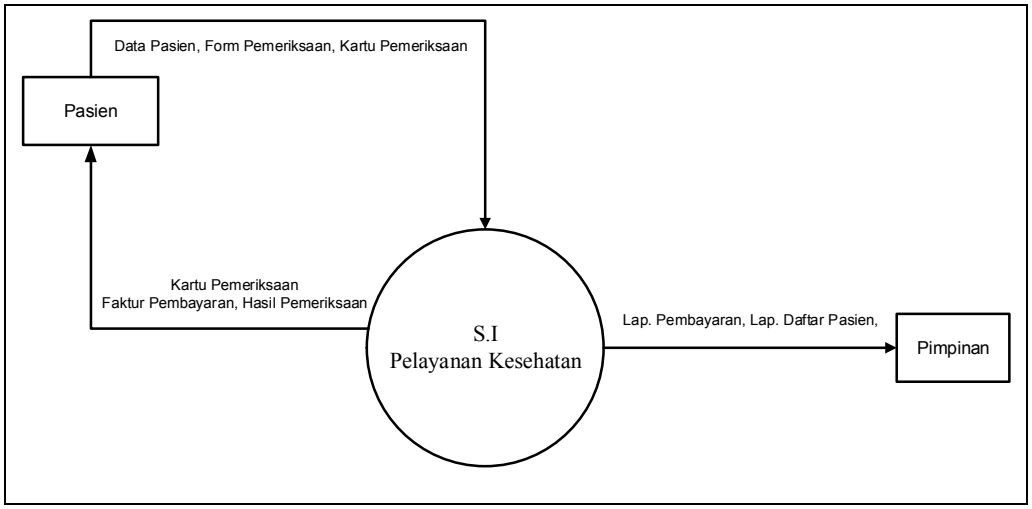
Flow map menggambarkan aliran dan informasi antar area di dalam sebuah organisasi. Bagan alir ini menelusuri sebuah dokumen dari asalnya sampai tujuannya. Secara rinci bagan alur ini menunjukkan dari mana dokumen tersebut berasal, distribusinya dan tujuan digunakannya dokumen tersebut. Berikut ini gambar flowmap prosedur pelayanan kesehatan yang diusulkan.



Gambar 4.1 Flow Map yang diusulkan

4.1.3.2 Diagram Konteks

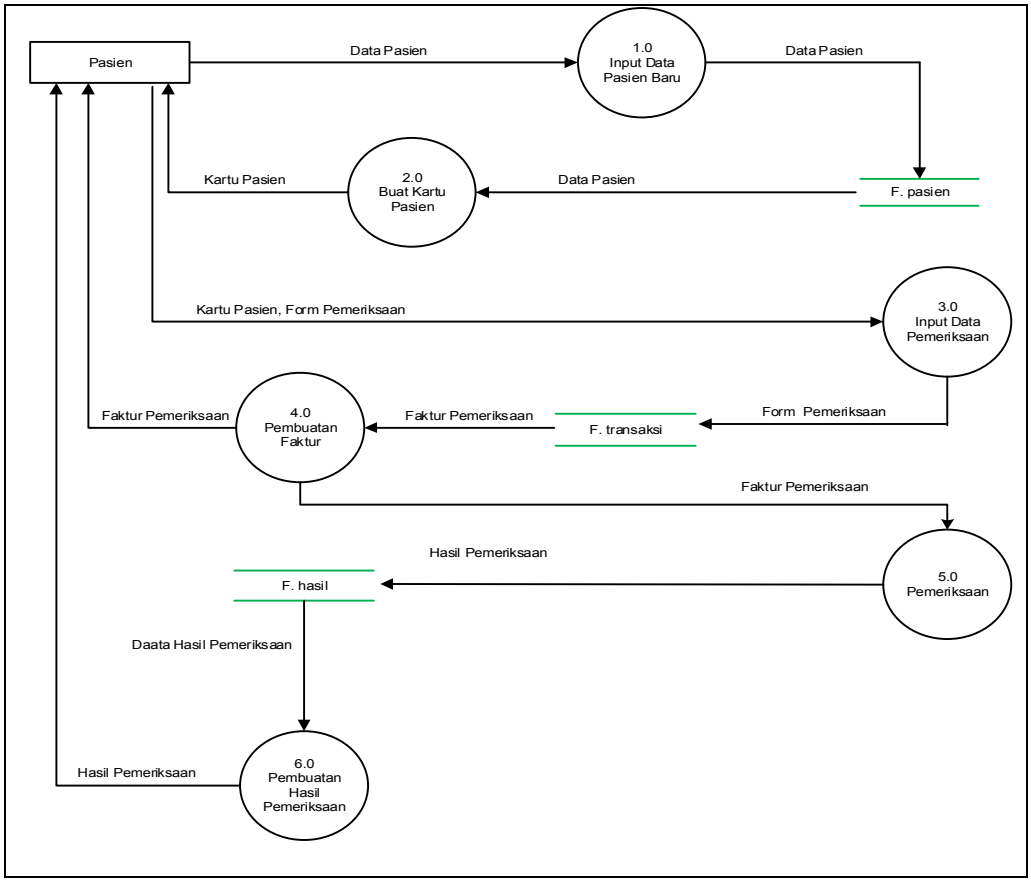
Diagram konteks merupakan alat untuk struktur analisis yang menggambarkan sistem secara umum. Sistem informasi yang dibuat akan menghasilkan sumber informasi yang dibutuhkan dan tujuan informasi yang dihasilkan.



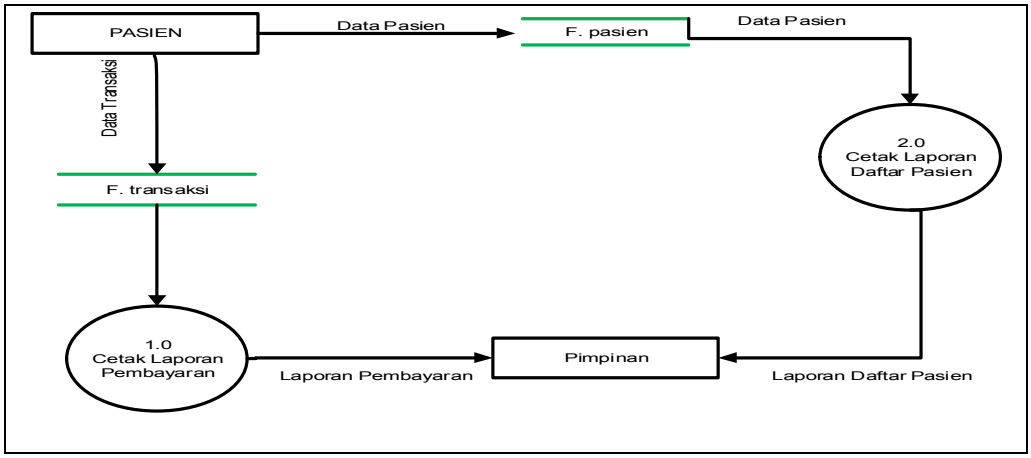
Gambar 4.2 Diagram Konteks yang Diusulkan

4.1.3.3 Data Flow Diagram (DFD)

DFD (*Data Flow Diagram*) merupakan alat pada metodologi pengembangan sistem secara terstruktur, yang menggambarkan alur data dari suatu sistem.



Gambar 4.3 DFD level 1 yang diusulkan



Gambar 4.4 DFD level 2 yang diusulkan Laporan Pembayaran dan Laporan Daftar Pasien

4.1.3.4 Kamus Data

Kamus Data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi yang lengkap. Keterangan lebih lanjut tentang struktur data suatu arus data di DFD secara lebih terperinci dapat dilihat di kamus data.

1. Nama dokumen : Data Pasien
   Alias : -
   Arus Data : Pasien – Proses1, Proses 1– F.Pasien,
   F.Pasien – Proses2
   Atribut : Nama_Pasien, Tempat_lahir, Tanggal_lahir,
   Jenis_Kelamin, Alamat, Telp.

2. Nama dokumen : Kartu Pemeriksaan
   Alias : Kartu Pasien
   Arus Data : FPasien – Proses2, Proses2 – Pasien
   Atribut : No_Pasien, Nama_Pasien, Tempat_Lahir,
   Tanggal_lahir, Jenis_Kelamin, Alamat, Telp
   Tgl_Daftar.

3. Nama dokumen : Form Pemeriksaan
   Alias : -
   Arus Data : Pasien – Proses3, Proses3 – F.Transaksi,
   F.Transaksi – Proses4

Atribut : No_Pasien, Nama_Pasien, Jenis_Kelamin,
Alamat_Pasien, Telp, Dokter,
Tgl Periksa, Kode_Pemeriksaan, Pemeriksaan.

4. Nama dokumen : Hasil Pemeriksaan Pasien
   Alias : -
   Arus Data : Proses5 – F.Hasil, F.Hasil – Proses6,
   Proses6 – Pasien
   Atribut : Register, Tgl_Periksa, No_Pasien, Nama_Pasien,
   Jenis_Kelamin, Alamat, Telp, Dokter,
   Kode_Pemeriksaan, Pemeriksaan,
   Hasil_Pemeriksaan, Satuan, Metoda,
   Nilai_Normal, Keterangan.

5. Nama dokumen : Faktur Pembayaran
   Alias : -
   Arus Data : F.Transaksi – Proses4, Proses4 – Pasien
   Atribut : Register, Tgl_Periksa, No_Pasien, Nama_Pasien,
   Jenis_Kelamin, Alamat, Tlp, Dokter,
   Kode_Pemeriksaan, Pemeriksaan,
   Harga, Diskon, Total, Total_Biaya, Terbilang.

6. Nama dokumen : Laporan Data Pembayaran

   Alias : -

   Arus Data : Pasien – FTransaksi, FTransaksi – Proses2.1, Proses2.1 – Pimpinan

   Atribut : Register, No_Pasien, Nama_Pasien, Dokter, Tanggal_Periksa, Kode_Pemeriksaan, Pemeriksaan, Harga, Diskon, Total, Total_Biaya.

7. Nama dokumen : Laporan Daftar Pasien

   Alias : -

   Arus Data : Pasien – FPasien, FPasien – Proses2.2, Proses.2.2 – Pimpinan

   Atribut : Tgl_Daftar, No_Pasien, Nama_Pasien, Tgl_Lahir, Jenis_Kelamin, Alamat, Telp.

4.1.4 Perancangan Basis Data

Setelah merancang prosedur sistem yang baru, penulis dapat merancang basis data untuk sistem ini, yang dimana akan memudahkan dalam membuat database dan program yang akan dirancang.

4.1.4.1 Normalisasi

Normalisasi memberikan panduan yang sangat membantu bagi pengembang untuk mencegah penciptaan struktur tabel yang kurang fleksibel atau mengurangi ketidak efisienan.

1. **Bentuk Unnormal**

   {Nama_Pasien, Tempat_lahir, Tanggal_lahir, Jenis_Kelamin, Alamat, Telp, Kontak_Person, Bagian, No_Pasien, Nama_Pasien, Tempat_Lahir, Tanggal_lahir, Jenis_Kelamin, Alamat, Telp, Tgl_Daftar, No_Pasien, Nama_Pasien, Jenis_Kelamin, Alamat_Pasien, Telp, Dokter, Tgl_Periksa, Kode_Pemeriksaan, Pemeriksaan, Register, Tgl_Periksa, No_Pasien, Nama_Pasien, Jenis_Kelamin, Alamat, Tlp, Dokter, Kode_Pemeriksaan, Pemeriksaan, Hasil_Pemeriksaan, Nama_Kontraktor, Satuan, Metoda, Nilai_Normal, Keterangan, Register, Tgl_Periksa, No_Pasien, Nama_Pasien, Jenis_Kelamin, Alamat, Telp, Dokter, Kode_Pemeriksaan, Pemeriksaan, Harga, Diskon, Total, Total_Biaya, Terbilang, Kontak_Person, Tanggal, Jam, Kontak_Person, Tanggal, Jam, Register, No_Pasien, Nama_Pasien, Dokter, Tgl_Periksa, Kode_Pemeriksaan, Pemeriksaan, Harga, Diskon, Total, Total_Biaya, Tgl_Daftar, No_Pasien, Nama_Pasien, Tgl_Lahir, Jenis_Kelamin, Alamat, Telp}

2. **Bentuk Normal 1**

   Bentuk normal pertama dapat terpenuhi apabila tidak memiliki atribut bernilai banyak atau lebih dari satu atribut dengan domain bernilai sama.

   {No_Pasien, Nama_Pasien, Tempat_Lahir, Tanggal_lahir, Jenis_Kelamin, Alamat, Telp, Tgl_Daftar, Dokter, Tgl_Periksa, Kode_Pemeriksaan, Pemeriksaan, Register, Hasil_Pemeriksaan, Satuan, Metoda, Nilai_Normal, Keterangan, Harga, Diskon, Total, Total_Biaya, Terbilang, Kontak_Person, Tgl, Jam}

3. Bentuk Normal 2

Bentuk normal kedua terpenuhi jika pada sebuah tabel semua atribut yang tidak termasuk primary key memiliki ketergantungan fungsional pada primary key secara utuh. Sebuah tabel dikatakan tidak memenuhi 2nf, jika ketergantungannya hanya bersifat parsial (hanya tergantung pada sebagian dari primary key).

a. Pasien : {*No_Pasien, Tgl_Daftar, Nama_Pasien, Jenis_Kelamin, Tgl_Lahir, Alamat, Telp}

b. Harga : {*Kode_Pemeriksaan, Pemeriksaan, Harga}

c. Transaksi : {*Register, Tgl_Periksa, **No_Pasien, Dokter Nama_Kontraktor, **Kode_Pemeriksaan, Diskon, Total, Total_Biaya, Terbilang}

d. Hasil : {*Register, **No_Pasien, Dokter, Nama_Kontraktor, Tgl_Periksa, **Kode_Pemeriksaan, Satuan, Hasil_Pemeriksaan, Metoda, Nilai_Normal, Keterangan}

e. Nilai : {*Kode_Pemeriksaan, Nama_Pemeriksaan, Satuan, Nilai_Normal}

4. Bentuk Normal3

Bentuk normal ketiga terpenuhi jika pada sebuah tabel semua atribut yang tidak termasuk primary key memiliki ketergantungan fungsional pada primary key secara utuh. Sebuah tabel dikatakan tidak memenuhi 2nf, jika

ketergantungannya hanya bersifat parsial (hanya tergantung pada sebagian dari primary key)

a. Pasien : {*No_Pasien, **No_Kontraktor, Tgl_Daftar, Nama_Pasien, Jenis_Kelamin, Tgl_Lahir, Alamat, Telp}

b. Harga : {*Kode_Pemeriksaan, Nama_Pemeriksaan, Harga, Kode_Kategori, Kode_Subkategori}

c. Transaksi : {*No_Transaksi, **No_Pasien, Tgl_Periksa, Dokter, No_Kontraktor**, Total_Biaya, Terbilang}

d. Detail_Transaksi : {*No_Transaksi, **Kode_Pemeriksaan, Diskon, Total}

e. Hasil : {*No_Transaksi, **No_Pasien, **No_Dokter, **No_Kontraktor, Tgl_Periksa, **Kode_Pemeriksaan, Hasil_Pemeriksaan , Keterangan, Status}

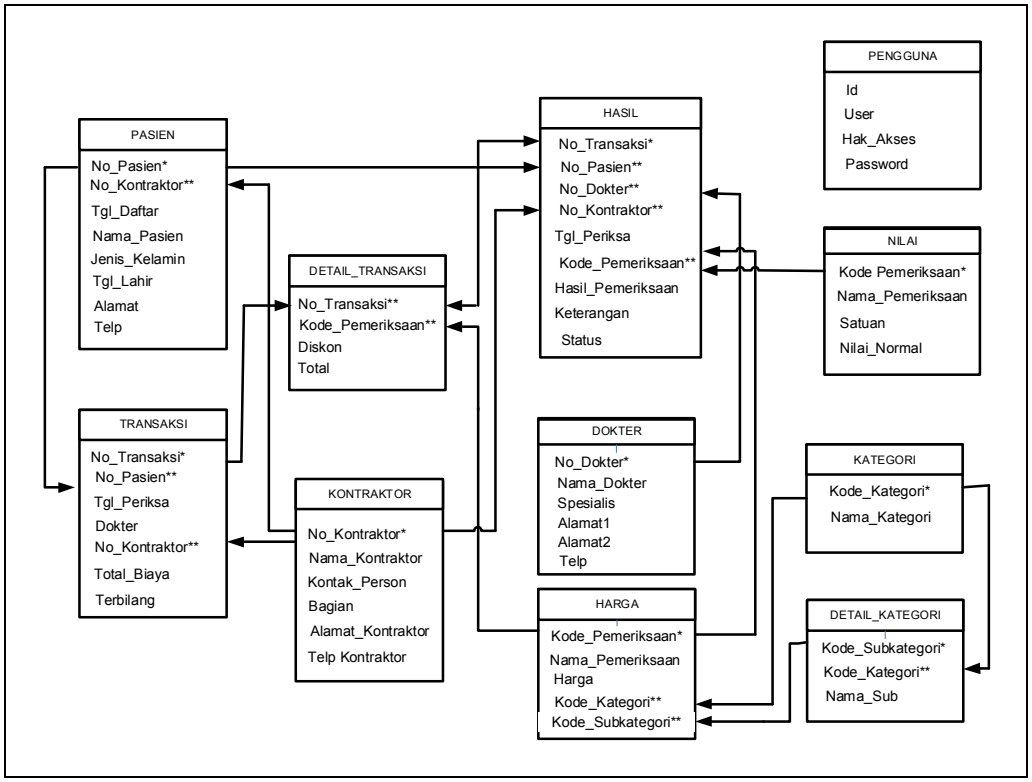
f. Nilai : {*Kode_Pemeriksaan, Nama_Pemeriksaan, Satuan, Nilai_Normal}

g. Dokter : { *No_Dokter, Nama_Dokter, Spesialis, Alamat1, Alamat2, Telp}

h. Kontraktor : { *No_Kontraktor, Nama_Kontraktor, Kontak_Person, Bagian, Alamat_Kontraktor, Telp_Kontraktor)

4.1.4.2. Tabel Relasi

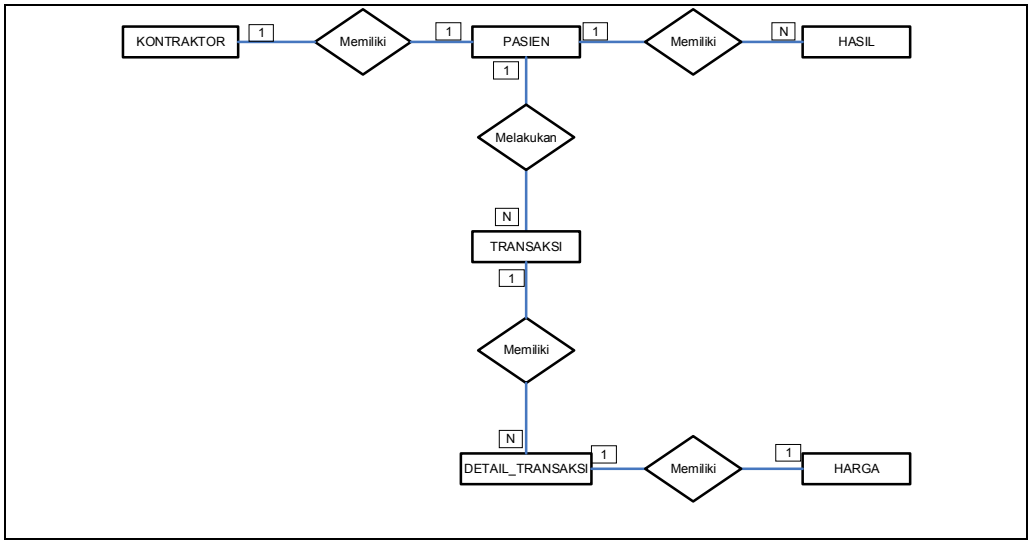
Tabel Relasi menggambarkan hubungan antara tabel - tabel yang terdapat pada *database* klinik, yang dimana didalam tabel tersebut terdapat *field* kunci (*Primary Key*) dan terdapat kunci tamu (*Foreign Key*). Kedua kunci (*Key*) ini digunakan untuk menghubungkan antara tabel.



Gambar 4.5 Relasi Tabel

4.1.4.3. Entity Relationship Diagram (ERD)

Diagram hubungan entitas atau dikenal dengan diagram ER adalah rotasi grafik dari sebuah model data atau sebuah model jaringan yang menjelaskan tentang data yang tersimpan dari sebuah sistem.



Gambar 4.6 *Entity Relationship Diagram*

4.1.4.4. Struktur File

Struktur *file* yang akan digunakan didalam perancangan sistem ini akan menentukan struktur *file database* yang menunjukan struktur dari elemen - elemen yang menyatakan panjang tipe datanya. Pengembangan struktur *file* yang akan diuraikan sebagai berikut :

1. File Pasien

   a. Nama File : Pasien

   b. Primary Key : no_pasien

   c. Jumlah Field : 8

Tabel 4.1 File Pasien

No.	Field Name	Type	Size	Description
1	No_Pasien	Varchar	20	No Pasien
2	No_Kontraktor	Varchar	20	No Kontraktor
3	Tgl_Daftar	Varchar	20	Tanggal Daftar

4	Nama_Pasien	Varchar	50	Nama Pasien
5	Jenis_Kelamin	Varchar	20	Jenis Kelamin
6	Tgl_Lahir	Varchar	10	Tanggal Lahir
7	Alamat	Varchar	50	Alamat
8	Telp	Varchar	12	Telp

2. File Harga

   a. Nama File : Harga

   b. Primary Key : kode_pemeriksaan

   c. Jumlah Field : 5

Tabel 4.2 File Harga

No.	Field Name	Type	Size	Description
1	Kode_pemeriksaan	Varchar	20	Kode Pemeriksaan
2	Nama_Pemeriksaan	Varchar	50	Nama Pemeriksaan
3	Harga	Double		Harga Pemeriksaan
4	Kode_Kategori	Varchar	20	Kode Pemeriksaan
5	Kode_Subkategori	Varchar	20	Kode Pemeriksaan

3. File Hasil

   a. Nama File : Hasil

   b. Primary Key : no_transaksi

   c. Jumlah Field : 9

Tabel 4.3 File Hasil

No.	Field Name	Type	Size	Description
1	no_transaksi	Varchar	20	No Transaksi
2	no_pasien	Varchar	20	No Pasien
3	no_dokter	Varchar	50	No Dokter
4	no_kontraktor	Varchar	100	No Kontraktor
5	tgl_periksa	Varchar	20	Tanggal Pemeriksaan
6	Kode_pemeriksaan	Varchar	50	Kode Pemeriksaan
7	Hasil_Pemeriksaan	Varchar	50	Hasil Pemeriksaan
8	Keterangan	Varchar	20	Keterangan
9	Status	Varchar	20	Status

4. File Detail_Transaksi

   a. Nama File : Detail_Transaksi

   b. Primary Key : no_transaksi

   c. Jumlah Field : 4

Tabel 4.4 File Detail Transaksi

No.	Field Name	Type	Size	Description
1	No_Transaksi	Varchar	20	No Transaksi
2	Kode_Pemeriksaan	Varchar	20	Kode Pemeriksaan
3	Diskon	Double		Diskon Pemeriksaan
4	Total	Double		Total Pemeriksaan

5. File Transaksi

a. Nama File : Transaksi

b. Primary Key : no_transaksi

c. Jumlah Field : 7

Tabel 4.5 File Transaksi

No.	Field Name	Type	Size	Description
1	No_Transaksi	Varchar	20	No Transaksi
2	No_Pasien	Varchar	20	No Pasien
3	Tgl_Pemeriksaan	Varchar	20	Tanggal Pemeriksaan
4	Dokter	Varchar	50	Nama Dokter
5	No_Kontraktor	Varchar	50	No Kontraktor
6	Total_Biaya	Double		Total Biaya
7	Terbilang	Varchar	150	Terbilang

6. File Nilai

a. Nama File : Nilai

b. Primary Key : Kode_Pemeriksaan

c. Jumlah Field : 4

Tabel 4.6 File Nilai

No.	Field Name	Type	Size	Description
1	Kode_Pemeriksaan	Varchar	20	Kode Pemeriksaan
2	Nama_Pemeriksaan	Varchar	20	Nama Pemeriksaan
3	Satuan	Varchar	20	Satuan

4	Nilai_Normal	Double		Nilai Normal

7. File Dokter

a. Nama File : Dokter

b. Primary Key : No_Dokter

c. Jumlah Field : 6

Tabel 4.7 File Dokter

No.	Field Name	Type	Size	Description
1	No_Dokter	Varchar	20	No Dokter
2	Nama_Dokter	Varchar	40	Nama Dokter
3	Spesialis	Varchar	20	Spesialis
4	Alamat1	Varchar	150	Alamat Rumah
5	Alamat1	Varchar	150	Alamat Prakter
6	Telp	Double		Telp Dokter

8. File Kontraktor

a. Nama File : Kontraktor

b. Primary Key : No_Kontraktor

c. Jumlah Field : 6

Tabel 4.8 File Kontraktor

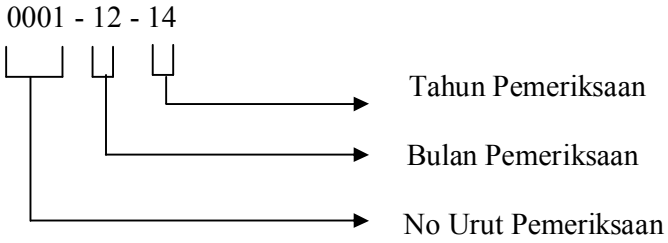
No.	Field Name	Type	Size	Description
1	No_Kontraktor	Varchar	20	No Kontraktor
2	Nama_Kontraktor	Varchar	40	Nama Kontraktor

3	Kontak Person	Varchar	20	Kontak Person
4	Bagian	Varchar	150	Bagian
5	Alamat_Kontraktor	Varchar	150	Alamat Kontraktor
6	Telp_ Kontraktor	Double		Telp Kontraktor

4.1.4.5. Kodifikasi

Pengkodean ini dibuat untuk mendefinisikan suatu objek secara singkat. Dengan adanya sistem ini pengkodean ini diharapkan dapat mengklasifikasikan data, dan memasukkan data ke dalam komputer dan untuk mengambil informasi yang terkait. Kode dapat dibentuk dari kumpulan huruf, angka dan karakter khusus. Adapun pengkodean pada sistem informasi pelayanan kesehatan di Laboratorium Klinik PARANIDA yaitu :

1. Register atau No Transaksi

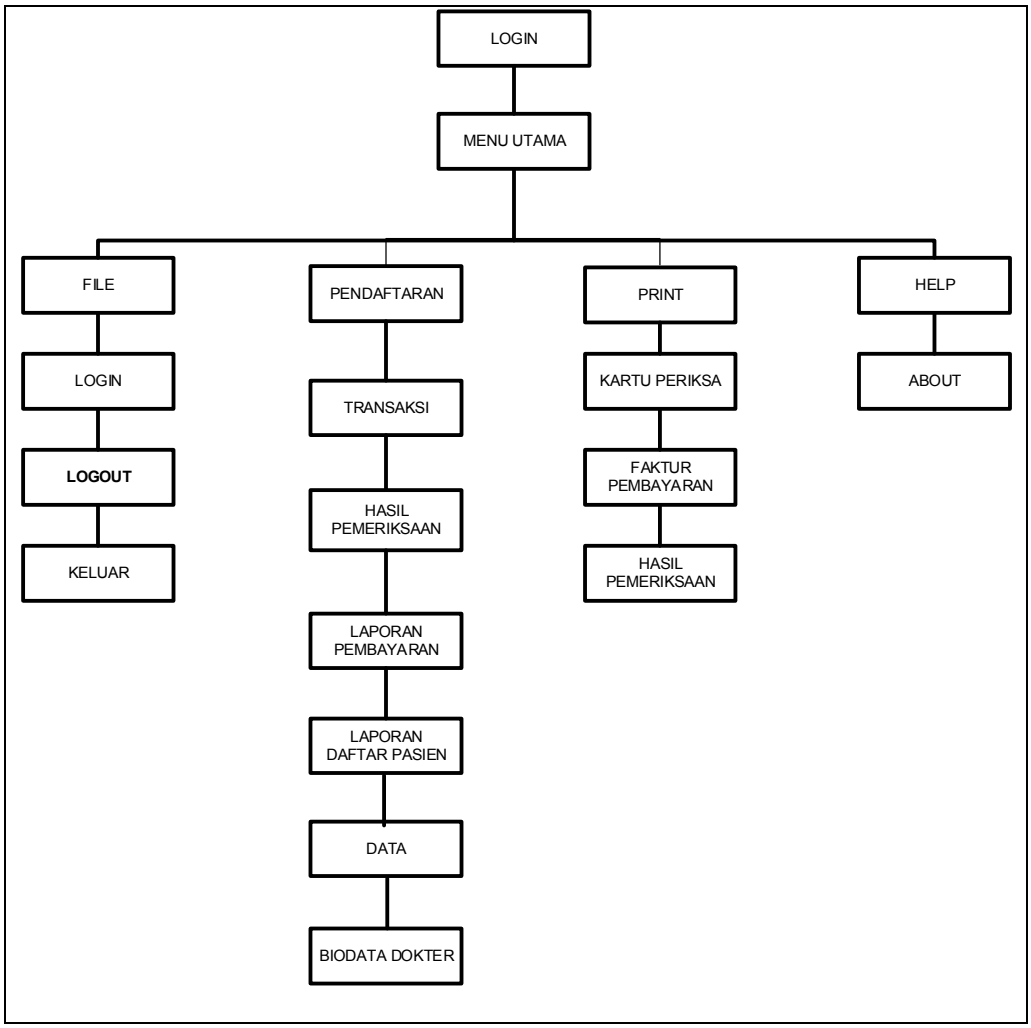


4.2. Perancangan Antar Muka

Perancangan antar muka merupakan suatu langkah dalam membuat sebuah program aplikasi. Program dirancang sesuai dengan kebutuhannya. Perancangan program dibuat meliputi beberapa perancangan diantaranya perancangan struktur menu, perancangan *input* dan perancangan *output*.

4.2.1. Struktur Menu

Perancangan struktur menu merupakan suatu bentuk yang terdapat pada form utama yang dibuat untuk memudahkan user untuk memilih menu yang dikehendaki. Berikut ini gambar perancangan struktur menu.



Gambar 4.7 Struktur Menu

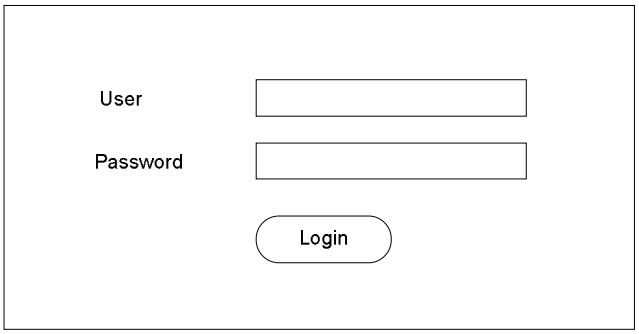
4.2.2. Perancangan Input

Perancangan *input* merupakan suatu bentuk tampilan dari program yang akan dibuat untuk kebutuhan *interface* dengan *user*.

Berikut ini gambar dari perancangan *input* :

1. Tampilan Login

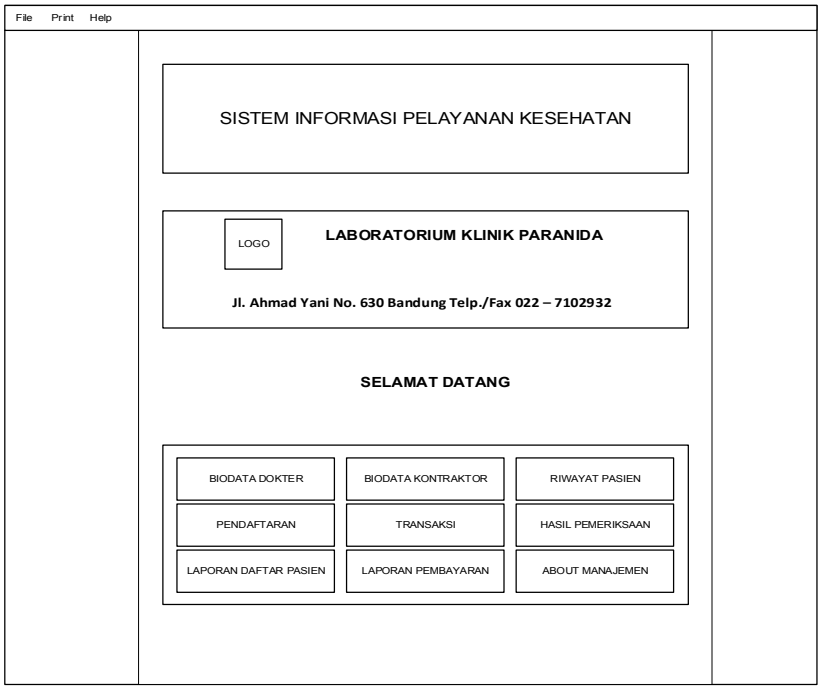
Perancangan login merupakan tampilan pertama kali muncul ketika program aplikasi penggajian pertama kali dijalankan. Aplikasi Login ini dirancang untuk pengamanan setiap aplikasi. Berikut tampilan perancangan Login.



Gambar 4.8 Rancangan Menu Login

2. Tampilan Menu Utama

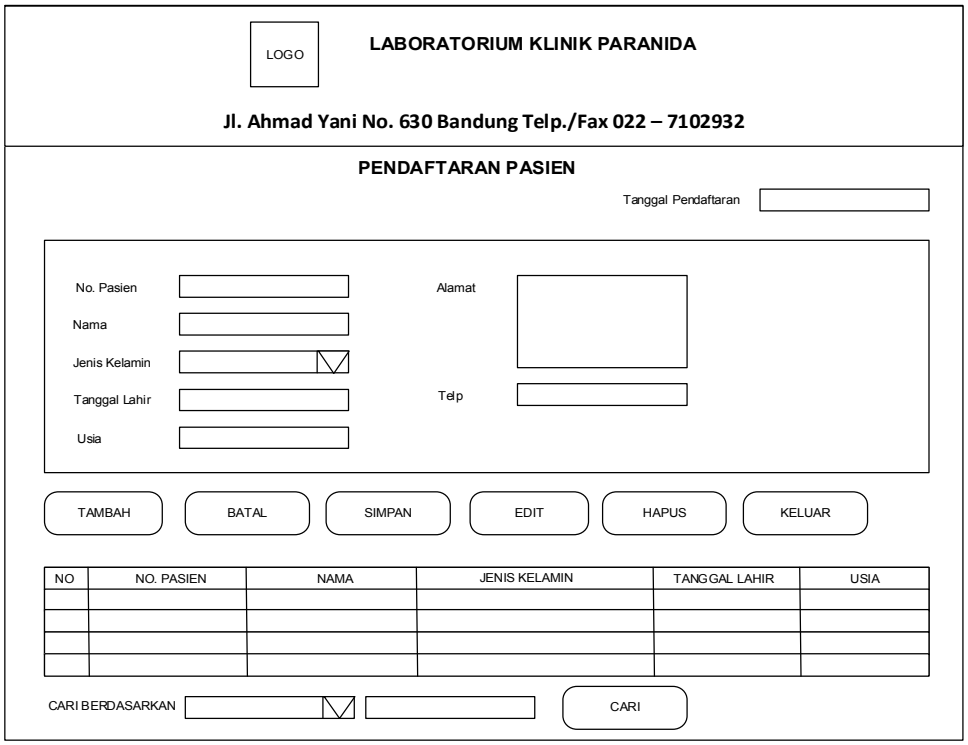
Tampilan menu utama yang akan penulis buat terdapat 8 tombol yaitu Biodata Dokter, Biodata Kontraktor, Pendaftaran, Transaksi, Hasil Pemeriksaan, Laporan Daftar Pembayaran, Laporan Daftar Pasien, Keluar. Form ini akan tampil jika *user* berhasil masuk ke form login.



Gambar 4.9 Rancangan Menu Utama

3. Tampilan Form Pendaftaran Pasien

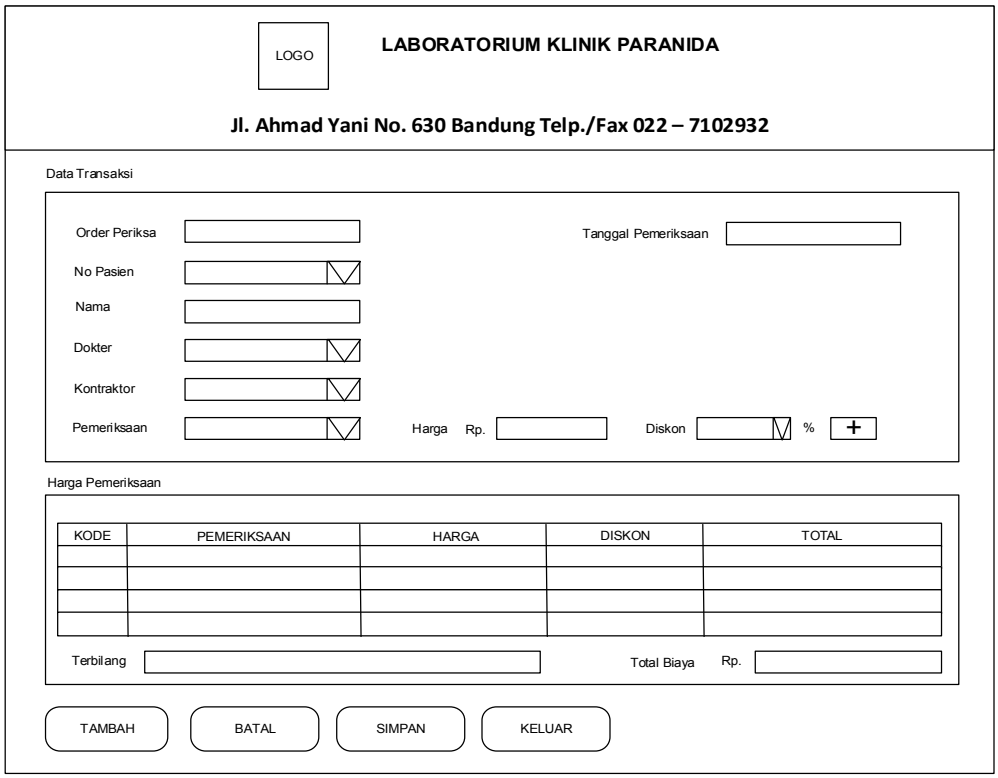
Perancangan form pendaftaran merupakan rancangan yang dibuat untuk melakukan proses penginputan pendaftaran pasien. Berikut adalah rancangan form pendaftaran data pasien.



Gambar 4.10 Rancangan Pendaftaran Pasien

4. Tampilan Form Transaksi Pembayaran

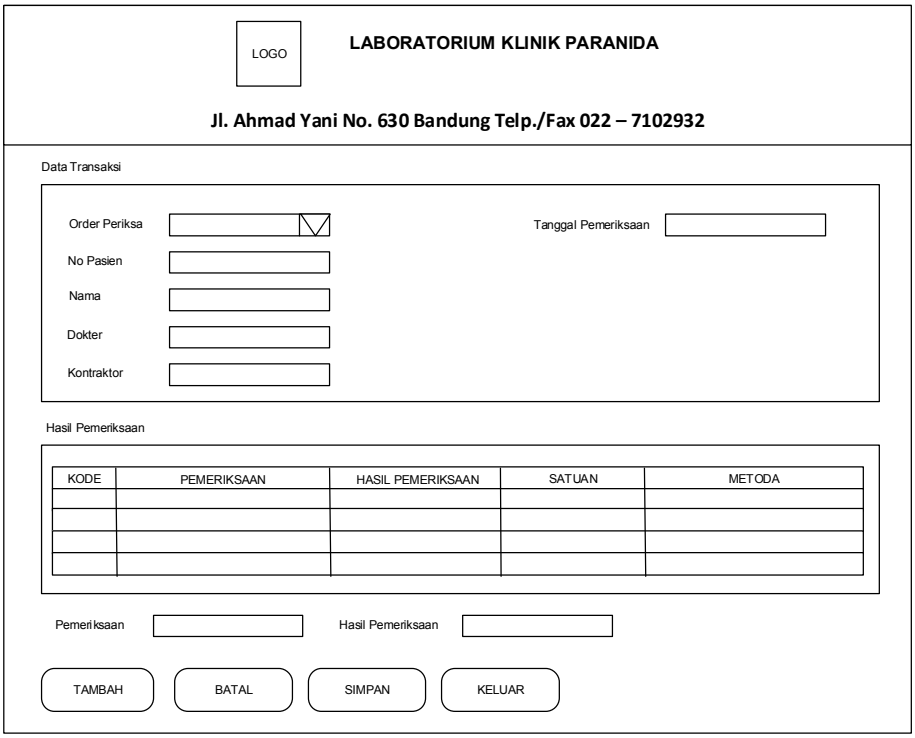
Perancangan form transaksi merupakan rancangan yang dibuat untuk melakukan proses penginputan data transaksi pembayaran. Berikut adalah rancangan form transaksi pembayaran.



Gambar 4.11 Rancangan Transaksi Pembayaran

5. Tampilan Form Hasil Pemeriksaan

Perancangan form hasil pemeriksaan merupakan rancangan yang dibuat untuk melakukan proses hasil pemeriksaan. Berikut adalah rancangan form hasil pemeriksaan.



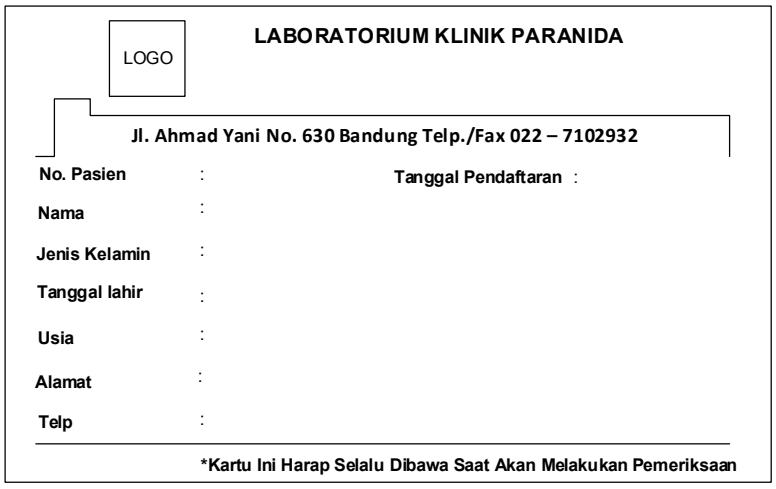
Gambar 4.12 Rancangan Hasil Pemeriksaan

4.2.3. Perancangan Output

Perancangan *output* dapat digunakan untuk menghasilkan suatu informasi. Perancangan *output* ini akan menampilkan data keluaran atau laporan yang diinginkan untuk siap melakukan percetakan dengan sumber data. Adapun bentuk tampilan keluaran berupa laporan-laporan seperti yang dijelaskan dibawah ini :

a. Kartu Pasien

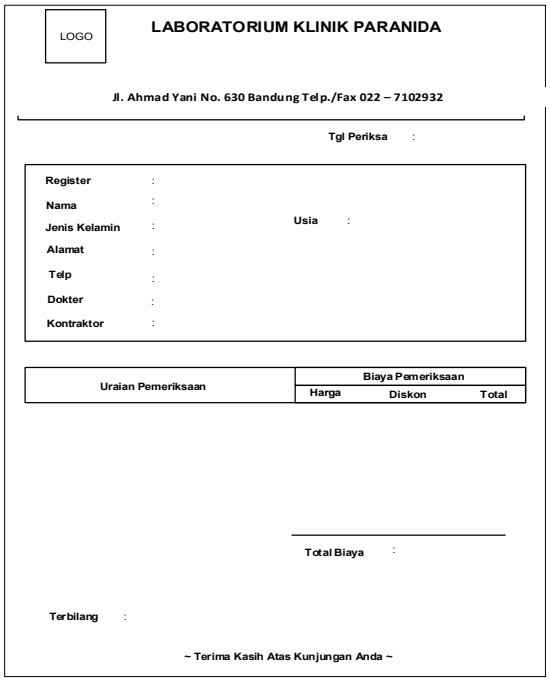
   Kartu pemeriksaan berisi tentang data pasien yang harus dibawa setiap akan melakukan pemeriksaan. Berikut gambar dari perancangan kartu pasien :



Gambar 4.13 Rancangan Kartu Pasien

b. Faktur Pembayaran

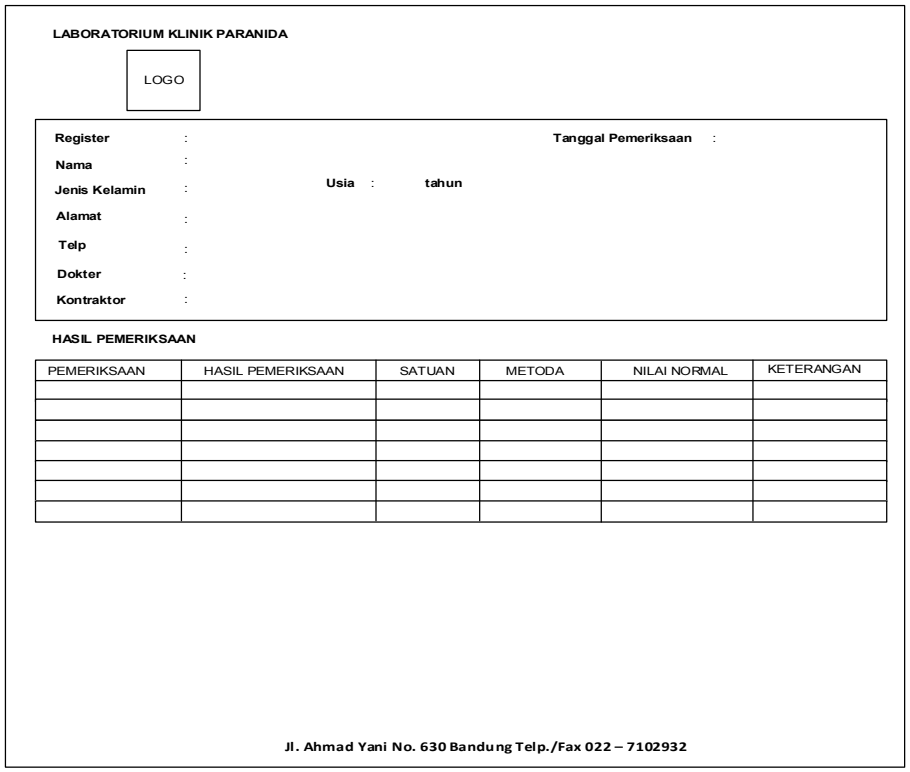
Faktur pembayaran berisi tentang rincian biaya pemeriksaan yang harus dibayar oleh pasien. Berikut gambar dari perancangan faktur pembayaran :



Gambar 4.14 Rancangan Faktur Pembayaran

c. Hasil Pemeriksaan

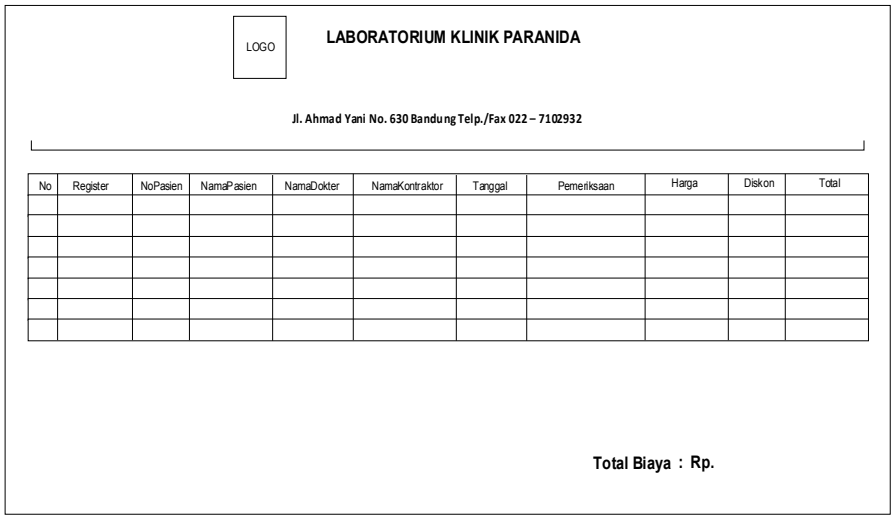
Hasil pemeriksaan berisi tentang seluruh hasil pemeriksan medis yang telah dilakukan dan akan diterima oleh pasien. Berikut gambar dari perancangan hasil pemeriksaan :



Gambar 4.15 Rancangan Hasil Pemeriksaan

d. Laporan Pembayaran

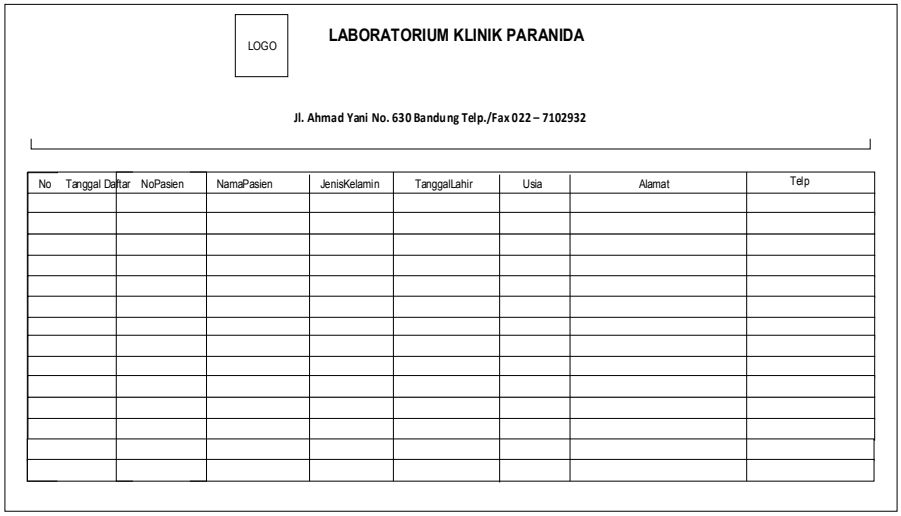
Laporan ini berisi tentang transaksi pembayaran pemeriksaan untuk diserahkan ke pimpinan dari bagian administrasi. Berikut gambar dari perancangan laporan pembayaran :



Gambar 4.16 Rancangan Laporan Pembayaran

e. Laporan Daftar Pasien

Laporan ini berisi tentang seluruh data pasien yang terdaftar untuk diserahkan ke pimpinan dari bagian administrasi. Berikut gambar dari perancangan laporan daftar pasien :

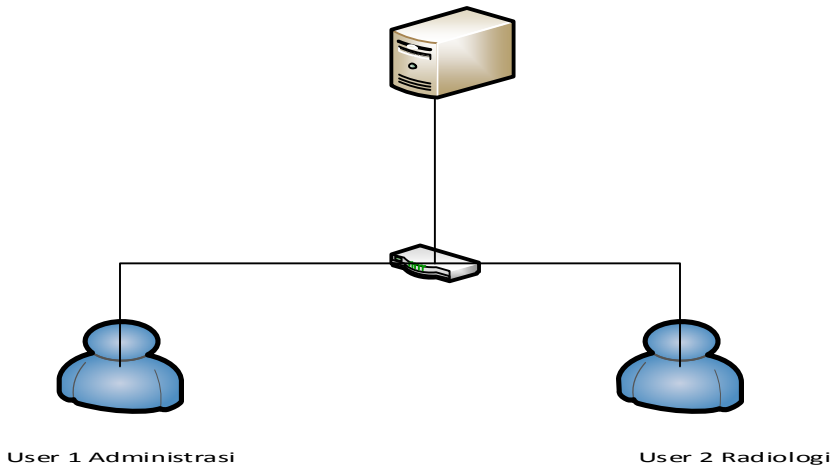


Gambar 4.17 Rancangan Laporan Daftar Pasien

4.3. Perancangan Arsitektur jaringan

Arsitektur Jaringan terdiri dari perkabelan, topologi, media metoda akses dan format paket. Arsitektur yang umum digunakan dalam jaringan adalah berbasis kabel elektrik, melalui perkembangan teknologi optik kini banyak digunakan juga serat kabel optik sebagai media alternatif beserta kelebihan dan kekurangannya.Tujuan dari suatu jaringan adalah menghubungkan jaringan-jaringan yang telah ada dalam jaringan tersebut sehingga informasi dapat di transfer dari satu lokasi ke lokasi yang lain.

Berikut adalah gambar perancangan arsitektur yang penulis rancang :



Gambar 4.18 Perancangan Arsitektur Jaringan.

4.4. Implementasi

Tahap implementasi merupakan tahap kelanjutan setelah melakukan perancangan sistem. Setelah melakukan perancangan sistem ini maka sistem tersebut diimplementasikan. Hal ini merupakan suatu usaha untuk mewujudkan suatu sistem yang dirancang agar sesuai dengan kebutuhan. Implementasi

perancangan Sistem Informasi Pelayanan Kesehatan pada LABORATORIUM KLINIK PARANIDA ini dilakukan dengan menggunakan bahasa pemrograman Java Netbeans IDE 6.9.1 dengan basis data yang digunakan adalah MySQL. Program tersebut dijalankan pada sistem operasi dan perangkat keras, tetapi implementasi dan pengujian sepenuhnya hanya dilakukan di PC (*Personal Computer*) dengan sistem operasi Microsoft Windows XP.

4.4.1. Batasan Implementasi (Optional)

Mengimplementasikan *system* aplikasi ini ada beberapa hal yang yang dijadikan batasan-batasan implementasi yaitu sebagai berikut :

1. Implementasi perangkat lunak dilakukan pada PC (Personal Computer) dengan *system* operasi Microsoft windows 7 secara *client server.*
2. Basis data yang digunakan dalam pengimplementasian ini adalah MySQL (phpMyAdmin) sebagai *database.*
3. Pembuatan aplikasi *system* Informasi Klinik Laboratorium Paranida ini menggunakan Bahasa Pemrograman NetBeans 6.9.1.

4.4.2. Implementasi Perangkat Lunak

Perangkat lunak merupakan alat pendukung sistem yang terdiri dari sistem operasi dan aplikasi database. Perangkat lunak yang di gunakan penulis dalam pembuatan aplikasi adalah sebagai berikut:

1. Sistem Operasi Windows 7
2. Bahasa Pemrograman (JAVA), NetBeans 6.9.1

3. MySQL (phpMyAdmin) sebagai *database*
4. iReports *version* 4.1.3

4.4.3. Implementasi Perangkat Keras

Selain perangkat lunak, kita juga membutuhkan perangkat keras (*Hardware*) untuk dapat menjalankan beberapa perangkat lunak. *Hardware* merupakan perangkat keras komputer, seperti *monitor*, *motherboard*, *printer*, *keyboard*, dan sebagainya. Minimum spesifikasi hardware untuk dapat membuat dan menjalankan aplikasi ini adalah sebagai berikut :

1. Menggunakan *processor* Intel Pentium atau yang sekelasnya.
2. Menggunakan RAM 512 MB.
3. Kapasitas harddisk atau media penyimpanan minima 40 GB.
4. *Mouse*, *keyboard*, dan *monitor* sebagai peralatan antar muka.

4.4.4. Implementasi Basis Data (Syntak SQL)

-- phpMyAdmin SQL Dump

-- version 2.11.9.2

-- http://www.phpmyadmin.net

--

-- Host: localhost

-- Generation Time: Nov 18, 2012 at 12:26 AM

-- Server version: 5.0.67

-- PHP Version: 5.2.6

```
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
--
-- Database: `klinik`
--
-- --------------------------------------------------------
--
-- Table structure for table `detail_transaksi`
--
CREATE TABLE IF NOT EXISTS `detail_transaksi` (
  `no_transaksi` varchar(20) NOT NULL,
  `kode_pemeriksaan` varchar(20) NOT NULL,
  `nama_pemeriksaan` varchar(50) NOT NULL,
  `harga` double NOT NULL,
  `diskon` double NOT NULL,
  `total` double NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Dumping data for table `detail_transaksi`
-- --------------------------------------------------------
-- Table structure for table `dokter`
--
CREATE TABLE IF NOT EXISTS `dokter` (
  `no_dokter` varchar(20) NOT NULL,
```

```
  `nama_dokter` varchar(50) NOT NULL,
  `kode_spesialis` varchar(50) NOT NULL,
  `alamat1` varchar(50) NOT NULL,
  `alamat2` varchar(50) NOT NULL,
  `telp` varchar(12) NOT NULL,
  PRIMARY KEY  (`no_dokter`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Dumping data for table `dokter`
-- --------------------------------------------------------
--
-- Table structure for table `harga`
--
CREATE TABLE IF NOT EXISTS `harga` (
  `kode_pemeriksaan` varchar(20) NOT NULL,
  `pemeriksaan` varchar(50) NOT NULL,
  `harga` double NOT NULL,
  PRIMARY KEY  (`kode_pemeriksaan`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Dumping data for table `harga`
-- --------------------------------------------------------
--
```

```
-- Table structure for table `hasil`
--
CREATE TABLE IF NOT EXISTS `hasil` (
  `no_transaksi` varchar(20) NOT NULL,
  `no_pasien` varchar(20) NOT NULL,
  `nama_pasien` varchar(50) NOT NULL,
  `nama_dokter` varchar(50) NOT NULL,
  `nama_kontraktor` varchar(50) NOT NULL,
  `tanggal_pemeriksaan` varchar(20) NOT NULL,
  `kode_pemeriksaan` varchar(20) NOT NULL,
  `nama_pemeriksaan` varchar(200) NOT NULL,
  `hasil_pemeriksaan` varchar(100) default NULL,
  `satuan` varchar(50) default NULL,
  `metoda` varchar(50) default NULL,
  `nilai_normal` varchar(50) default NULL,
  `keterangan` varchar(100) default NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Dumping data for table `hasil`
--
-- --------------------------------------------------------
--
-- Table structure for table `kontraktor`
```

```
--
CREATE TABLE IF NOT EXISTS `kontraktor` (
  `no_kontraktor` varchar(20) NOT NULL,
  `nama_kontraktor` varchar(50) NOT NULL,
  `kontak` varchar(25) NOT NULL,
  `bagian` varchar(50) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  `telp` varchar(12) NOT NULL,
  PRIMARY KEY  (`no_kontraktor`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Dumping data for table `kontraktor`
--
-- --------------------------------------------------------
--
-- Table structure for table `nilai`
--

CREATE TABLE IF NOT EXISTS `nilai` (
  `kode_pemeriksaan` varchar(20) NOT NULL,
  `nama_pemeriksaan` varchar(100) NOT NULL,
  `hasil_pemeriksaan` varchar(100) default NULL,
  `satuan` varchar(100) default NULL,
```

```
  `metoda` varchar(100) default NULL,
  `nilai_normal` varchar(100) default NULL,
  `keterangan` varchar(100) default NULL,
  PRIMARY KEY  (`kode_pemeriksaan`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Dumping data for table `nilai`
--
-- --------------------------------------------------------
--
-- Table structure for table `pasien`
--
CREATE TABLE IF NOT EXISTS `pasien` (
  `no_pasien` varchar(20) NOT NULL,
  `tanggal_daftar` varchar(25) NOT NULL,
  `nama_pasien` varchar(50) NOT NULL,
  `jenis_kelamin` varchar(20) NOT NULL,
  `tanggal_lahir` varchar(10) NOT NULL,
  `usia` varchar(3) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  `telp` varchar(12) NOT NULL,
  UNIQUE KEY `no_pasien` (`no_pasien`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
--
-- Dumping data for table `pasien`
-- --------------------------------------------------------
-- Table structure for table `pengguna`
--
CREATE TABLE IF NOT EXISTS `pengguna` (
  `id` int(4) NOT NULL auto_increment,
  `user` varchar(25) NOT NULL,
  `hakakses` varchar(25) NOT NULL,
  `password` varchar(11) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
--
-- Dumping data for table `pengguna`
-- --------------------------------------------------------
--
-- Table structure for table `transaksi`
--
CREATE TABLE IF NOT EXISTS `transaksi` (
  `no_transaksi` varchar(20) NOT NULL,
  `tanggal_pemeriksaan` varchar(20) NOT NULL,
  `no_pasien` varchar(20) NOT NULL,
  `nama_pasien` varchar(50) NOT NULL,
```

```
  `nama_dokter` varchar(100) NOT NULL,
  `nama_kontraktor` varchar(100) NOT NULL,
  `total_biaya` double NOT NULL,
  `terbilang` varchar(100) NOT NULL,
  PRIMARY KEY  (`no_transaksi`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Dumping data for table `transaksi`
--
```

4.4.5. Implementasi Antar Muka

Dalam pemrograman Java NetBeans implementasi antar muka dibuat dalam sebuah frame yang berektensi frame. Setiap halaman dalam perangkat lunak dibuat file program yang ditulis dengan berekstensi frame. Berikut adalah implementasi antar muka sistem informasi pelayanan kesehatan adalah sebagai berikut :



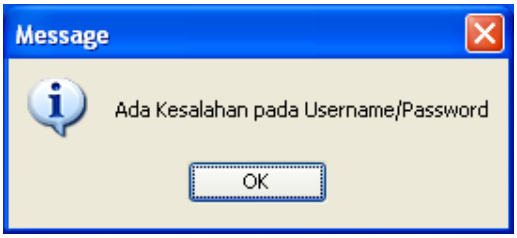
Gambar 4.19 *Splash Screen*

Di dalam tampilan awal program, yang pertama kali muncul adalah frame *splash screen*, setelah itu masuk ke frame login dimana user harus memasukkan *username* dan *password* yang benar.



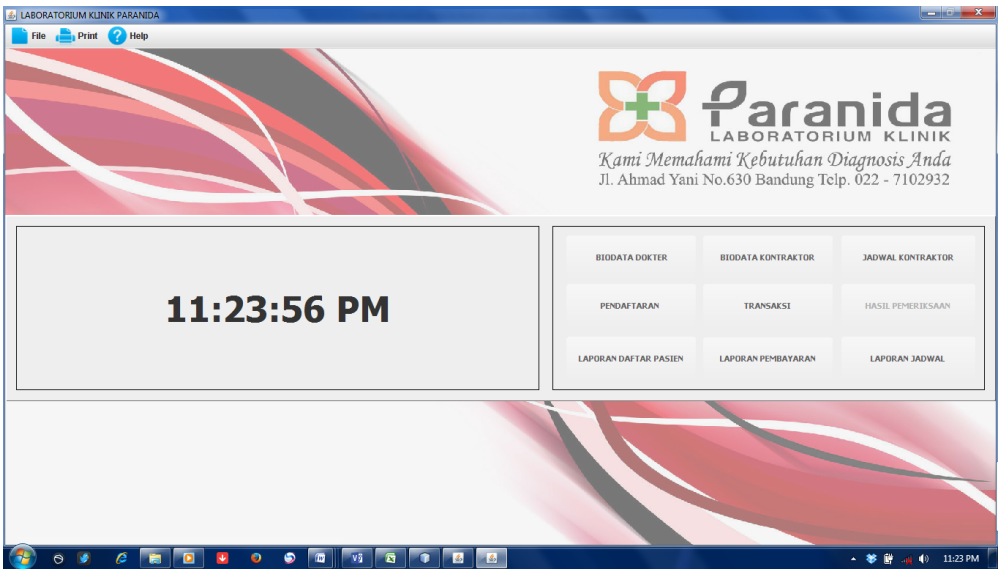
Gambar 4.20 Frame Login

Jika ada kesalahan saat memasukkan *username* dan *password* maka pada saat tombol "Login" diklik, akan muncul peringatan seperti dibawah ini :



Gambar 4.21 Message Salah Username/Password

Setelah Login berhasil, dengan memasukan *username* dan *password* yang benar, maka akan muncul frame utama, yang dimana terdapat menu - menu pilihan yang dapat dipilih sesuai kebutuhan, sebagai berikut :

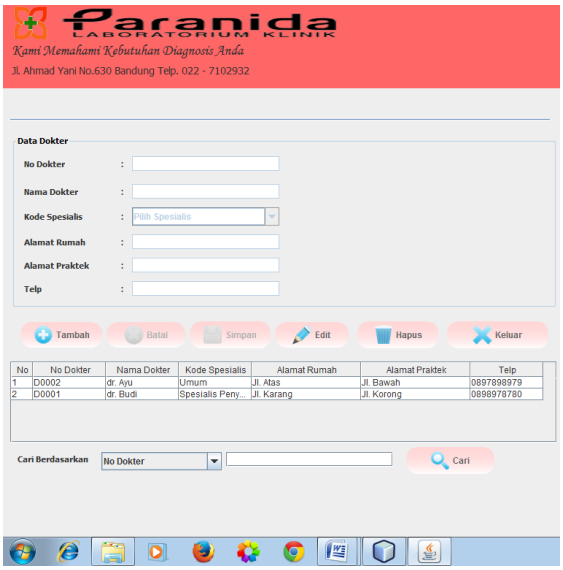


Gambar 4.22 Frame Utama

Keterangan :

1. Di dalam menu file ada sub menu login, logout, keluar
2. Di dalam menu print ada sub menu kartuperiksa, faktur pembayaran, dan hasil pemeriksaan.
3. Di dalam menu help ada sub menu about.
4. Di dalam menu utama terdapat menu biodata dokter, pendaftaran, transaksi, hasil pemeriksaan, laporan pembayaran, laporan daftar pasien, dan riwayat pasien.

Jika dalam menu utama dan dipilih biodata dokter maka akan muncul frame data dokter sebagai berikut :



Gambar 4.23 Menu Data Dokter

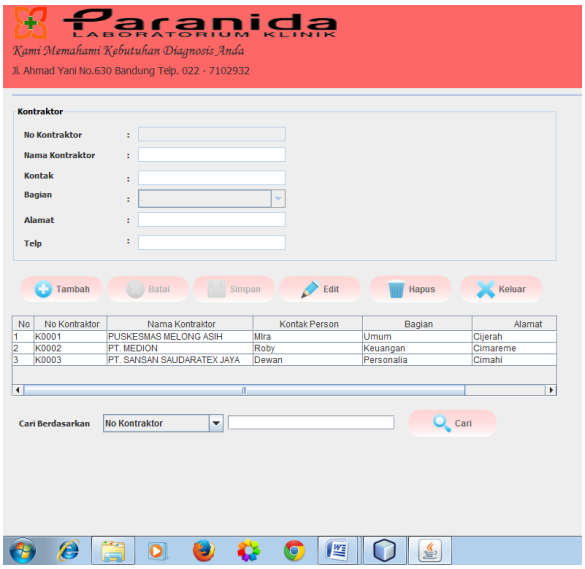
Frame ini digunakan untuk menginput data dokter dan juga dapat mengedit data dokter yang sudah ada. Jika di klik tombol “Tambah”, maka text box no kontraktor akan secara otomatis terisi dan text box yang lain akan diaktifkan dan dikosongkan.

Jika selesai menginput, maka klik tombol “Simpan”, dan semua textbox akan nonaktif. Dan jika tombol “Batal” diklik maka data tidak akan bertambah ke database dan jika tombol “Edit” diklik, maka semua textbox juga akan aktif. Dan

jika tombol “hapus” diklik maka akan keluar konfirmasi dan data pun terhapus dari table.

Jika ingin mencari data dokter bedasarkan no dokter atau nama dokter, maka pilih combobox cari berdasarkan no dokter atau nama dokter, lalu inputkan ke dalam text dan klik tombol cari. Untuk keluar dari frame data dokter, klik tombol “keluar” maka akan kembali ke frame menu utama

Jika dalam menu utama dan dipilih biodata kontraktor maka akan muncul frame data kontraktor sebagai berikut :



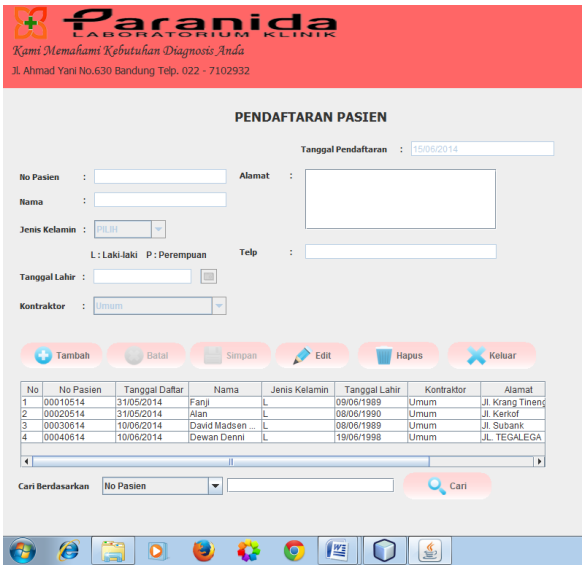
Gambar 4.24 Menu Data Kontraktor

Frame ini digunakan untuk menginput data kontraktor dan juga dapat mengedit data kontraktor yang sudah ada. Jika di klik tombol “Tambah”, maka text box no dokter akan secara otomatis terisi dan text box yang lain akan diaktifkan dan dikosongkan.

Jika selesai menginput, makal klik tombol “Simpan”, dan semua textbox akan nonaktif. Dan jika tombol “Batal” diklik maka data tidak akan bertambah ke database dan jika tombol “Edit” diklik, maka semua textbox juga akan aktif. Dan jika tombol “hapus” diklik maka akan keluar konfirmasi dan data pun terhapus dari table.

Jika ingin mencari data kontraktor bedasarkan no kontraktor atau nama kontraktor, maka pilih combobox cari berdasarkan no kontraktor atau nama kontraktor, lalu inputkan ke dalam text dan klik tombol cari. Untuk keluar dari frame data kontraktor, klik tombol “keluar” maka akan kembali ke frame menu utama.

Jika dalam menu utama dan dipilih pendaftaran maka akan muncul frame data pendaftaran pasien sebagai berikut :



Gambar 4.25 Menu Pendaftaran Pasien

Frame ini digunakan untuk menginput pendaftaran pasien dan juga dapat mengedit data pasien yang sudah ada. Jika di klik tombol “Tambah”, maka text

box no pasien dan text box tanggal pendaftaran akan secara otomatis terisi dan text box yang lain akan diaktifkan dan dikosongkan.

Jika selesai menginput, maka klik tombol “Simpan” dan akan langsung mencetak kartu pemeriksaan, dan semua textbox akan nonaktif.

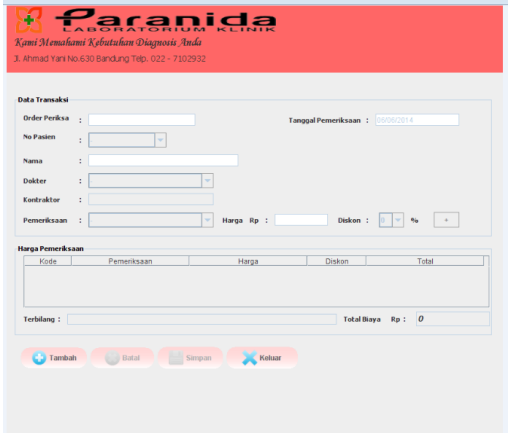


Gambar 4.26 Kartu Pasien

Dan jika tombol “Batal” diklik maka data tidak akan bertambah ke database dan jika tombol “Edit” diklik, maka semua textbox juga akan aktif. Dan jika tombol “hapus” diklik maka akan keluar konfirmasi dan data pun terhapus dari table.

Jika ingin mencari data pasien bedasarkan no pasien atau nama kontraktor, maka pilih combobox cari berdasarkan no pasien atau nama pasien, lalu inputkan ke dalam text dan klik tombol cari. Untuk keluar dari frame data pasien, klik tombol “keluar” maka akan kembali ke frame menu utama.

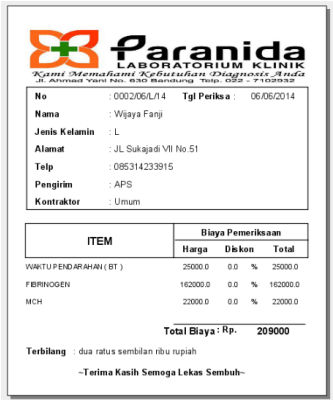
Jika dalam menu utama dan dipilih transaksi maka akan muncul frame transaksi sebagai berikut :



Gambar 4.27 Menu Data Transaksi

Frame ini digunakan untuk menginput transaksi pembayaran dan pembuatan faktur pembayaran. Jika di klik tombol “Tambah”, maka text box order periksa dan text box tanggal pemeriksaan akan secara otomatis terisi dan text box yang lain akan diaktifkan dan dikosongkan.

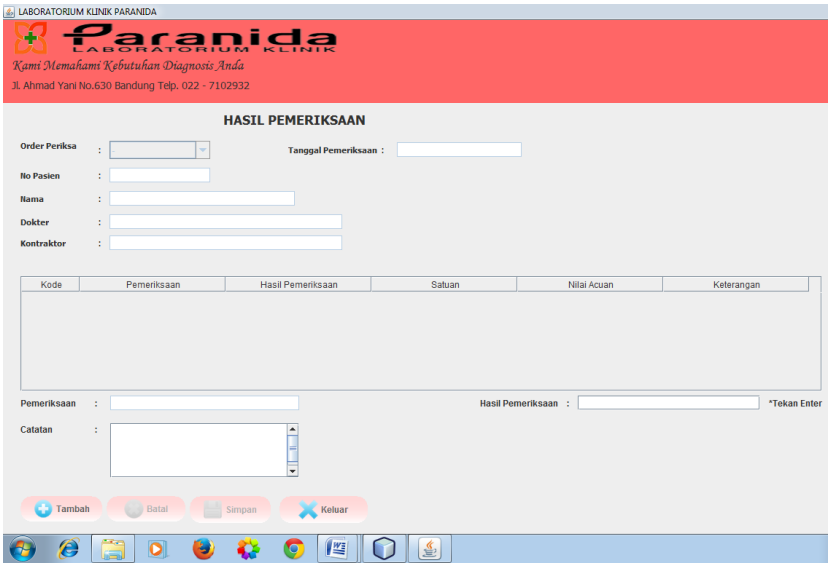
Jika selesai menginput, maka klik tombol “Simpan” dan akan langsung mencetak faktur pembayaran, dan semua textbox akan nonaktif.



Gambar 4.28 Faktur Pembayaran

Dan jika tombol “Batal” diklik maka data tidak akan bertambah ke database. Untuk keluar dari frame data pasien, klik tombol “keluar” maka akan kembali ke frame menu utama.

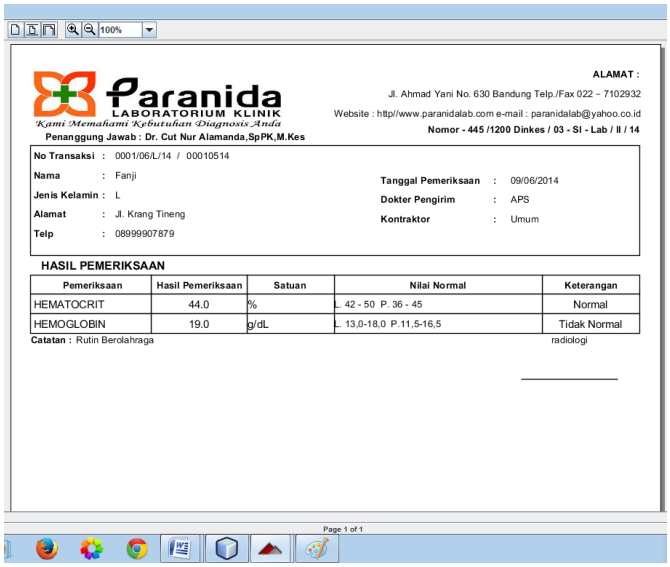
Jika dalam menu utama dan dipilih hasil pemeriksaan maka akan muncul frame hasil pemeriksaan sebagai berikut :



Gambar 4.29 Menu Hasil Pemeriksaan

Frame ini digunakan untuk menginput hasil pemeriksaan. Jika di klik tombol “Tambah”, text box tanggal pemeriksaan akan secara otomatis terisi dan text box yang lain akan diaktifkan dan dikosongkan.

Jika selesai menginput, maka klik tombol “Simpan” dan akan langsung mencetak hasil pemeriksaaan , dan semua textbox akan nonaktif.



Paranida
LABORATORIUM KLINIK
Kami Memahami Kebutuhan Diagnosa Anda

ALAMAT :
Jl. Ahmad Yani No. 630 Bandung Telp./Fax 022 - 7102932
Website : http//www.paranidalab.com e-mail : paranidalab@yahoo.co.id
Nomor - 445 /1200 Dinkes / 03 - SI - Lab / II / 14

Penanggung Jawab : Dr. Cut Nur Alamanda,SpPK,M.Kes

No Transaksi : 0001/06/L/14 / 00010514
Nama : Fanji
Jenis Kelamin : L
Alamat : Jl. Krang Tineng
Telp : 0899907879

Tanggal Pemeriksaan : 09/06/2014
Dokter Pengirim : APS
Kontraktor : Umum

HASIL PEMERIKSAAN

Pemeriksaan	Hasil Pemeriksaan	Satuan	Nilai Normal	Keterangan
HEMATOCRIT	44.0	%	L. 42 - 50 P. 36 - 45	Normal
HEMOGLOBIN	19.0	g/dL	L. 13,0-18,0 P.11,5-16,5	Tidak Normal

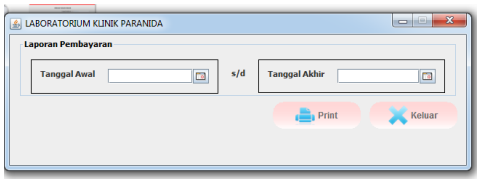
Catatan : Rutin Berolahraga

radiologi

Page 1 of 1

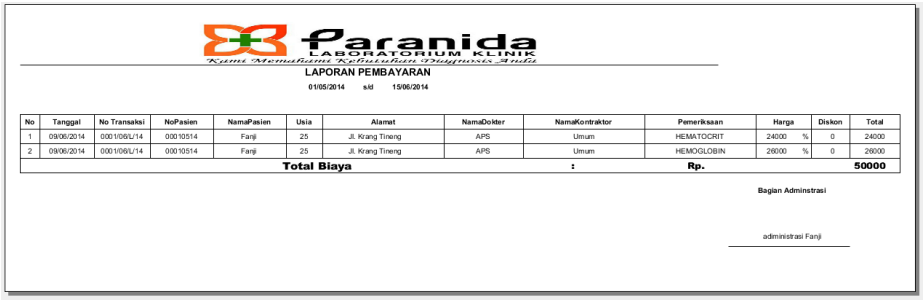
Gambar 4.30 Laporan Hasil Pemeriksaan

Jika dalam menu utama dan dipilih laporan pembayaran maka akan muncul frame laporan pembayaran sebagai berikut :



Gambar 4.31 Menu Laporan Pembayaran

Pilih tanggal awal dan tanggal akhir kemudian klik “Print” kemudian akan muncul tampilan print preview seperti berikut ini :


Paranida
LABORATORIUM KLINIK
Kami Memahami Kebutuhan Diagnosis Anda

LAPORAN PEMBAYARAN

01/05/2014 s/d 15/06/2014

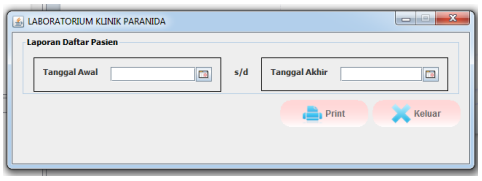
No	Tanggal	No Transaksi	NoPasien	NamaPasien	Usia	Alamat	NamaDokter	NamaKontraktor	Pemeriksaan	Harga	Diskon	Total
1	09/06/2014	0001/06/L/14	0001054	Fanji	25	Jl. Krang Tineng	APS	Umum	HEMATOCRIT	24000 %	0	24000
2	09/06/2014	0001/06/L/14	0001054	Fanji	25	Jl. Krang Tineng	APS	Umum	HEMOGLOBIN	26000 %	0	26000
Total Biaya								:	Rp.			50000

Bagian Adminstrasi

administrasi Fanji

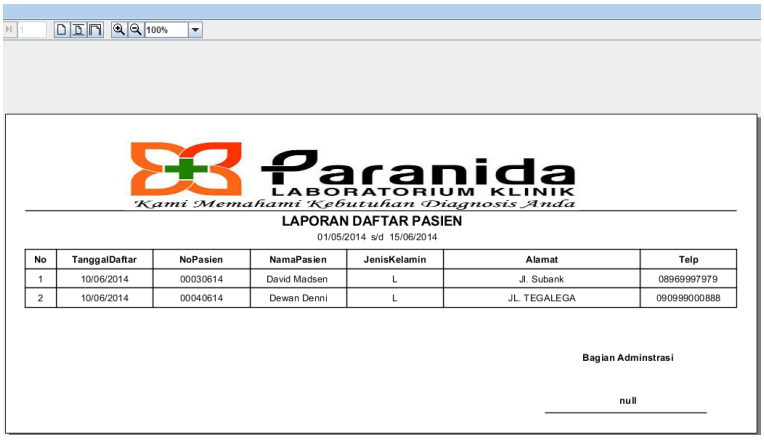
Gambar 4.32 Laporan Pembayaran

Jika dalam menu utama dan dipilih laporan daftar pasien maka akan muncul frame laporan daftar pasien sebagai berikut :



Gambar 4.33 Menu Laporan Daftar Pasien

Pilih tanggal awal dan tanggal akhir kemudian klik “Print” kemudian akan muncul tampilan print preview seperti berikut ini :


100%

Paranida
LABORATORIUM KLINIK
Kami Memahami Kebutuhan Diagnosis Anda

LAPORAN DAFTAR PASIEN

01/05/2014 s/d 15/06/2014

No	TanggalDaftar	NoPasien	NamaPasien	JenisKelamin	Alamat	Telp
1	10/06/2014	00030614	David Madsen	L	Jl. Subank	08969997979
2	10/06/2014	00040614	Dewan Denni	L	JL. TEGALEGA	090999000888

Bagian Adminstrasi

null

Gambar 4.34 Laporan Daftar Pasien

4.4.6. Implementasi Instalasi Program

Pertama yang harus dilakukan adalah membuka *file* instalasi program Klinik Laboratorium Panadina Bandung.exe, *double click* untuk menginstalnya.



Gambar 4.35 Icon Instalasi Program Klinik Laboratorium Paranida

Maka akan terlihat tampilan awal proses instalasi seperti pada gambar dibawah ini, lalu klik tombol **Next**.



Gambar 4.36 Instalasi Program

Kemudian akan muncul informasi "*I agree with the above terms and conditions*" lalu klik **Next** sampai muncul tampilan sebagai berikut :



Gambar 4.37 Lisensi Instalasi Program

Kemudian akan muncul sebuah tampilan untuk menentukan dimana lokasi program akan diinstal dan klik **Next,** untuk memulai proses instalasi program.



Gambar 4.38 Lokasi Instalasi Program

Setelah proses instalasi selesai maka di desktop anda akan terdapat shortcut program si pelayanan kesehatan yang siap untuk digunakan.



Gambar 4.39 Tampilan Shortcut Program

4.4.7 Penggunaan Program

Penjelasan penggunaan program dilakukan dengan tujuan agar pengguna perangkat lunak dapat mengerti mengenai program aplikasi yang dibuat dan cara kerjanya. Tahapannya adalah :

a. Login

Pertama, pengguna diharuskan untuk mengisikan *username* serta *password* yang benar untuk masuk kedalam program aplikasi.

b. Pendaftaran

Untuk melakukan pendaftaran, klik menu pendaftaran pada main menu, kemudian akan muncul form pendaftaran dimana pengguna dapat mengisikan data - data pasien sesuai dengan *field* yang disediakan pada form tersebut. Jika sudah, klik tombol simpan dan akan langsung menyimpan data pasien, yang selanjutnya akan mencetak kartu pemeriksaan pasien.

c. Transaksi

Setelah pasien terdaftar maka selanjutnya adalah melakukan transaksi pembayaran pemeriksaan. Untuk melakukan transaksi, klik menu transaksi pada main menu kemudian akan muncul form transaksi dimana pengguna akan

mengisikan data jenis pemeriksaan apa yang akan dilakukan oleh pasien, maka akan muncul biaya, total biaya dan terbilang secara otomatis.

d. Hasil Pemeriksaan

Jika telah diperoleh data hasil laboratorium pasien, maka petugas medis cukup untuk menyimpan hasil tersebut pada form pemeriksaan ini. Berisi data - data jenis pemeriksaan beserta nilai normalnya sehingga memudahkan pengguna dan pasien dalam proses pemeriksaan laboratorium. Jika data sudah diisi, maka klik tombol simpan untuk menyimpan data, yang selanjutnya akan mencetak kartu pemeriksaan pasien.

4.5. Pengujian Program

Pada tahapan ini dilakukan pengujian pada program aplikasi yang telah dibuat, apakah program aplikasi telah berjalan sesuai dengan yang diinginkan oleh pengguna. Adapun pengujian menggunakan metode *Black Box* yang terfokus pada persyaratan fungsional perangkat lunak yang dibuat.

4.5.1. Rencana Pengujian

Pengujian sistem informasi pelayanan kesehatan di Laboratorium Klinik Paranida menggunakan data uji berupa masukan dari pengguna. Tahapan pengujian yang dilakukan dapat dilihat pada tabel pengujian dibawah ini.

Tabel 4.9 Rencana Pengujian SI Pelayanan Kesehatan

Kelas Uji	**Butir Uji**	**Tingkat Pengujian**	**Jenis Pengujian**
Login	Pengecekan Login User	Sistem	Black Box
Data Pasien	Pengisian Data Pasien	Sistem	Black Box
	Pengecekan Data Pasien Yang Telah Ada	Sistem	Black Box
Data Transaksi	Pengisian Data Transaksi	Sistem	Black Box
	Pengecekan Data Transaksi Yang Telah Ada	Sistem	Black Box
Data Hasil Pemeriksaan	Pengisian Data Pemeriksaan	Sistem	Black Box
	Pengecekan Data Pemeriksaan Yang Telah Ada	Sistem	Black Box
Data Kontraktor	Pengisian Data Kontraktor	Sistem	Black Box
	Pengecekan Data Kontraktor	Sistem	Black Box

4.5.2 Kasus dan Hasil Pengujian

Setelah melaksanakan rencana pengujian maka dilanjutkan dengan kasus dan hasil pengujian. Berikut kasus dan hasil pengujian:

1. Pengecekan Login :

   Berikut ini kasus dan hasil pengujian login user

Tabel 4.10 Pengecekan Login

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username : Pendaftaran Bagian : administrasi Password : Admin	Tercantum pada textbox username, label bagian, dan textbox password	Dapat mengisi login sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Klik tombol LOGIN	Login berhasil dengan hak aksesnya dan menampilkan pesan "Login Berhasil "	Tombol dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Username : tidak sesuai Bagian : - Password : Tidak sesuai	Tercantum pada textbox username, label bagian, dan textbox password	User tidak dapat login	[X] Diterima [ ] Ditolak
Klik tombol LOGIN	Tidak dapat masuk ke form home dan menampilkan pesan "Ada kesalahan pada username dan password !"	Tombol dapat berfungsi sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak

2. Pendaftaran Pasien :

Berikut ini kasus dan hasil pengujian pendaftaran pasien.

Tabel 4.11 Pendaftaran Pasien

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol TAMBAH	Textbox no_pasien otomatis terisi, dan menampilkan seluruh inputan data pasien	Tombol berfungsi dengan baik dan semua inputan menjadi aktif	[X] Diterima [ ] Ditolak
Data pasien diisi secara lengkap.	Penyimpanan data kedalam file pasien	Dapat melakukan penyimpanan data dengan benar.	[X] Diterima [ ] Ditolak

Klik tombol SIMPAN	Seluruh data yang diinputkan akan tersimpan ke database dan langsung dicetak	Dapat melakukan penyimpanan data dengan benar, muncul pesan "Input Berhasil" dan langsung mencetak kartu pemeriksaan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data pasien diisi secara tidak lengkap dan ada yang kosong	Penyimpanan tidak dapat dilakukan	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak
Klik tombol SIMPAN	Penyimpanan gagal dan tampil pesan "Pengisian Data Tidak Boleh Kosong"	Tombol dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak

3. Transaksi Pembayaran :

Berikut ini kasus dan hasil pengujian transaksi pembayaran.

Tabel 4.12 Transaksi Pembayaran

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol TAMBAH	Textbox no_transaksi otomatis terisi, dan menampilkan seluruh inputan data transaksi	Tombol berfungsi dengan baik dan semua inputan menjadi aktif	[X] Diterima [ ] Ditolak
Klik Combobox no_pasien	Nama pasien akan muncul secara otomatis di textbox nama pasien setelah kita memilih combobox no_pasien	Combobox berfungsi Sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak
Data transaksi diisi secara lengkap.	Penyimpanan data kedalam file transaksi dan detail_transaksi.	Dapat melakukan penyimpanan data dengan benar.	[X] Diterima [ ] Ditolak
Klik tombol SIMPAN	Seluruh data yang diinputkan akan tersimpan ke database dan langsung dicetak	Dapat melakukan penyimpanan data dengan benar, muncul pesan "Input Berhasil" dan	[X] Diterima [ ] Ditolak

		langsung mencetak faktur pemeriksaan	
Kasus dan Hasil Uji (Data Salah)			
Data transaksi diisi secara tidak lengkap dan ada yang kosong	Penyimpanan tidak dapat dilakukan	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak
Klik tombol SIMPAN	Penyimpanan gagal dan tampil pesan "Pengisian Data Tidak Boleh Kosong"	Tombol dapat berfungsi sesuai dengan yang diharapkan	[X] Diterima [ ] Ditolak

4. Hasil Pemeriksaan :

Berikut ini kasus dan hasil pengujian hasil pemeriksaan.

Tabel 4.13 Hasil Pemeriksaan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik tombol TAMBAH	Mengaktifkan seluruh inputan data hasil transaksi yang dibutuhkan	Tombol berfungsi dengan baik dan semua inputan menjadi aktif	[X] Diterima [ ] Ditolak
Data hasil pemeriksaan diisi secara lengkap.	Penyimpanan data kedalam file hasil.	Dapat melakukan penyimpanan data dengan benar.	[X] Diterima [ ] Ditolak
Klik tombol SIMPAN	Seluruh data yang diinputkan akan tersimpan ke database dan langsung dicetak	Dapat melakukan penyimpanan data dengan benar, muncul pesan "Input Berhasil" dan langsung mencetak hasil pemeriksaan	[X] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data hasil pemeriksaan diisi secara tidak lengkap dan ada yang kosong	Penyimpanan tidak dapat dilakukan	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak
Klik tombol	Penyimpanan gagal dan	Tombol dapat	[X] Diterima

SIMPAN	tampil pesan "Pengisian Data Tidak Boleh Kosong"	berfungsi sesuai yang diharapkan	[ ] Ditolak

5. Data Kontraktor :

Berikut ini kasus dan hasil pengujian Data Kontraktor.

Tabel 4.14 Data Kontraktor

Kasus dan Hasil Uji (Data Salah)			
Data kontraktor diisi secara tidak lengkap dan ada yang kosong	Penyimpanan tidak dapat dilakukan	Tidak dapat melakukan penyimpanan data	[X] Diterima [ ] Ditolak
Memasukan Data kontraktor yang sudah ada	Data tidak dapat tersimpan	Muncul pesan bahwa Data yang diinputkan sudah Ada	[X] Diterima [ ] Ditolak
Klik tombol SIMPAN	Penyimpanan gagal dan tampil pesan "Pengisian Data Tidak Boleh Kosong !"	Tombol dapat berfungsi sesuai yang diharapkan	[X] Diterima [ ] Ditolak

4.5.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dari kasus uji sample di atas, dapat ditarik kesimpulan bahwa perangkat lunak bekerja sesuai dengan harapan dan berjalan dengan sebagaimana mestinya.