BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Tahap implementasi sistem merupakan kegiatan yang dilakukan setelah tahap perancangan sistem selesai dilaksanakan. Tujuan yang ingin dicapai pada tahap ini adalah dapat dioperasikanya hasil perancangan sistem yang telah dibuat. Dengan demikian dapat diketahui apakah perangkat lunak ini dapat menghasilkan sistem informasi rekam medis pasien yang sesuai dengan tujuan yang diharapkan. Implementasi Sistem Informasi Rekam Data Medis yang dilakukan menggunakan bahasa pemrograman PHP , dan basis data yang digunakan ialah MYSQL.

5.1.1 Batasan Implementasi

Dalam mengimplementasikan perangkat lunak, batasan sebagai pengendalian aplikasi ini ada beberapa hal yang perlu dibatasi dalam implementasi ini yaitu :

1. Perangkat lunak yang digunakan dalam pengimplementasian basis data ini adalah MySQL yang telah terkompilasi kedalam sebuah perangkat lunak bebas yaitu xampp.
2. Hanya bagian terkait yang dapat menggunakan sistem informasi rekam medis pasien ini.

3. Tidak semua kebutuhan Sistem Informasi Rekam Data medis dapat terpenuhi, tetapi hanya pada subsistem yang meliputi pengelolaan data pasien, pendaftaran pasien, resep, data obat, data tindakan, cetak kwitansi pembayaran.
4. Sistem yang diterapkan masih mengacu kepada sistem terakhir yang di observasi.

5.1.2 Implementasi Perangkat Lunak

Perangkat lunak (*Software*) digunakan untuk mendukung sistem operasi (SO) dan aplikasi *Database*. Perangkat lunak yang dibutuhkan oleh Komputer *server* maupun Komputer *client*, adapun kebutuhan perangkat lunak yang digunakan sebagai berikut :

1. Komputer *Server*

   Komputer *server* adalah komputer yang mempunyai kemampuan yang lebih dari komputer *client*, dimana didalamnya tersimpan data-data yang akan didistribusikan. Dimana software yang dibutuhkan diantaranya :

   a. Sistem Operasi Windows XP SP 2
   b. XAMPP 1.8.1
   c. MySQL.
   d. Adobe Reader
   e. Browser (mozilla firefox, chrome atau yang lainnya)

2. Komputer *Client*

   Komputer *Client* adalah komputer yang didalamnya tersimpan data-data yang dapat digunakan secara perorangan, mengumpulkan dan menampilkan data, serta menyimpan data ke *server*. Dimana software yang dibutuhkan diantaranya :

   a. Sistem Operasi Windows XP SP 2
   b. XAMPP 1.8.1
   c. MySQL
   d. Adobe Reader
   e. Browser (mozilla firefox, chrome atau yang lainnya)

5.1.3 Implementasi Perangkat Keras

Untuk dapat menjalankan aplikasi yang dirancang maka dibutuhkan suatu perangkat keras sebagai penunjangnya. Adapun perangkat kebutuhan perangkat keras yang dibutuhkan tersebut adalah sebagai berikut :

1. *Server* :
   a. *Processor* Minimal *Pentium* 4
   b. Menggunakan minimal RAM 512 MB.
   c. *Harddisk* minimal 80 Gb.
   d. *Keyboard*, *Monitor* dan *Mouse.*
2. *Client*
   a. *Processor* Minimal *Pentium* 4
   b. Menggunakan minimal RAM 512 MB.

c. *Harddisk* minimal 40 Gb.

d. *Keyboard*, *Monitor* dan *Mouse*.

Selain itu, harus pula tersedia Connector (Hub) dan kabel jaringan sebagai penghubung antara client dengan server.

5.1.4 Implementasi Basis Data (Sintaks SQL)

Implementasi basis data dilakukan dengan menggunakan bahasa SQL, dimana aplikasi pemrograman yang digunakan adalah MySQL, implementasi basis datanya dalam bahasa SQL adalah sebagai berikut :

```
--
-- Database: `db_praktek`
--
-- --------------------------------------------------------
--
-- Table structure for table `ci_sessions`
CREATE TABLE IF NOT EXISTS `ci_sessions` (
  `session_id` varchar(40) CHARACTER SET latin1 NOT NULL DEFAULT '0',
  `session_start` int(10) unsigned NOT NULL DEFAULT '0',
  `session_last_activity` int(10) unsigned NOT NULL DEFAULT '0',
  `session_ip_address` varchar(16) CHARACTER SET latin1 NOT NULL
DEFAULT '0',
  `session_user_agent` varchar(50) CHARACTER SET latin1 NOT NULL,
  `session_data` text CHARACTER SET latin1 NOT NULL,
  PRIMARY KEY (`session_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
ROW_FORMAT=DYNAMIC
-- --------------------------------------------------------
--
Table structure for table `daftar_menu`
--
CREATE TABLE IF NOT EXISTS `daftar_menu` (
  `menu_id` int(11) NOT NULL AUTO_INCREMENT,
  `menu_parent` int(11) NOT NULL DEFAULT '0',
  `menu_name` varchar(255) DEFAULT NULL,
  `menu_path` mediumtext,
  `menu_icon` varchar(255) NOT NULL,
  `sorter` int(11) NOT NULL,
```

```
  `subsorter` int(11) NOT NULL,
  `modulstat` varchar(1) NOT NULL,
  PRIMARY KEY (`menu_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 PACK_KEYS=0
AUTO_INCREMENT= 0
-- --------------------------------------------------------
--
-- Table structure for table `data_berobat`
--

CREATE TABLE IF NOT EXISTS `data_berobat` (
  `brb_id` int(11) NOT NULL AUTO_INCREMENT,
  `psn_id` int(11) NOT NULL,
  `dokter_id` int(11) NOT NULL DEFAULT '0',
  `no_urut` varchar(25) NOT NULL,
  `date_register` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `date_konsultasi_ulang` date DEFAULT '0000-00-00',
  `status` char(1) NOT NULL DEFAULT '0' COMMENT 'status periksa 0=
register, 1=dipanggil lagi, 2=sudah diobati, 3=batal periksa',
  `status_periksa` char(1) NOT NULL DEFAULT '0' COMMENT '0= berobat , 1
= konsultasi',
  `user_add` int(11) DEFAULT NULL,
  `date_add` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `ip_add` varchar(12) DEFAULT NULL,
  `user_update` int(11) DEFAULT '0',
  `date_update` datetime DEFAULT '0000-00-00 00:00:00',
  `ip_update` varchar(23) DEFAULT NULL,
  PRIMARY KEY (`brb_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT= 0
-- --------------------------------------------------------
--
Table structure for table `data_jabatan`
--

CREATE TABLE IF NOT EXISTS `data_jabatan` (
  `jab_id` int(11) NOT NULL AUTO_INCREMENT,
  `jab_name` varchar(50) COLLATE latin1_general_ci NOT NULL,
  `rec_creator` int(4) NOT NULL,
  `rec_created` date NOT NULL,
  `rec_edit` int(1) NOT NULL DEFAULT '0',
  `rec_editor` int(4) NOT NULL,
  `rec_edited` date NOT NULL,
  PRIMARY KEY (`jab_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=0
-- --------------------------------------------------------
```

```
--
Table structure for table `data_obat`
--

CREATE TABLE IF NOT EXISTS `data_obat` (
  `id_obat` int(3) NOT NULL AUTO_INCREMENT,
  `kode_obat` varchar(5) NOT NULL,
  `nama_obat` varchar(50) NOT NULL,
  `harga_obat` int(50) NOT NULL,
  `stok_obat` int(30) NOT NULL,
  PRIMARY KEY (`id_obat`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=0
-- --------------------------------------------------------
--
Table structure for table `data_pasien`
--

CREATE TABLE IF NOT EXISTS `data_pasien` (
  `psn_id` int(11) NOT NULL AUTO_INCREMENT,
  `psn_medrec` varchar(20) NOT NULL DEFAULT '0',
  `psn_name` varchar(250) NOT NULL,
  `psn_tempat_lahir` varchar(100) DEFAULT NULL,
  `psn_tgl_lahir` date NOT NULL DEFAULT '0000-00-00',
  `umur` int(3) NOT NULL,
  `psn_jk` char(1) NOT NULL DEFAULT '0' COMMENT 'jenis kelamin 1 =
Laki-Laki, 2=Perempuan',
  `psn_alamat` text,
  `psn_telepon` varchar(25) DEFAULT NULL,
  `psn_foto` tinytext,
  `user_add` int(11) DEFAULT NULL,
  `date_add` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `ip_add` varchar(12) DEFAULT NULL,
  `user_update` int(11) DEFAULT '0',
  `date_update` datetime DEFAULT '0000-00-00 00:00:00',
  `ip_update` varchar(23) DEFAULT NULL,
  PRIMARY KEY (`psn_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=0
-- --------------------------------------------------------
--
Table structure for table `data_transaksi`
--

CREATE TABLE IF NOT EXISTS `data_transaksi` (
  `id_transaksi` int(3) NOT NULL AUTO_INCREMENT,
  `psn_id` int(11) NOT NULL,
  `no_transaksi` varchar(15) NOT NULL,
```

```
  `tgl_transaksi` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `diagnosis` varchar(100) NOT NULL,
  `sorter` int(20) NOT NULL,
  PRIMARY KEY (`id_transaksi`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=0
-- --------------------------------------------------------
--
Table structure for table `history_obat`
--

CREATE TABLE IF NOT EXISTS `history_obat` (
  `psn_id` int(11) NOT NULL,
  `obat` varchar(50) NOT NULL,
  `jml_obat` int(10) NOT NULL,
  `harga` int(50) NOT NULL,
  `sorter` int(20) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1
-- --------------------------------------------------------
--
Table structure for table `history_tindakan`

--

CREATE TABLE IF NOT EXISTS `history_tindakan` (

  `psn_id` int(11) NOT NULL,

  `tindakan` varchar(50) NOT NULL,

  `biaya` int(50) NOT NULL,

  `sorter` int(20) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- --------------------------------------------------------
--
Table structure for table `prc_sys_client`
--

CREATE TABLE IF NOT EXISTS `prc_sys_client` (
  `client_id` int(11) NOT NULL AUTO_INCREMENT,
  `client_name` varchar(300) NOT NULL,
  `client_legal` varchar(10) NOT NULL,
  `module_program` varchar(50) NOT NULL,
  `module_type` varchar(50) NOT NULL,
```

```
  `module_version` varchar(6) NOT NULL,
  `module_revision` int(11) NOT NULL,
  `module_package` varchar(50) NOT NULL DEFAULT 'Development',
  `image` varchar(50) NOT NULL,
  PRIMARY KEY (`client_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=0
-- --------------------------------------------------------
--

Table structure for table `prc_sys_counter`

--

CREATE TABLE IF NOT EXISTS `prc_sys_counter` (

  `thn` year(4) NOT NULL DEFAULT '0000',

  `bln` int(2) NOT NULL DEFAULT '1',

  `psn_no` int(11) NOT NULL DEFAULT '1',

  `mr_no` int(11) NOT NULL DEFAULT '1',

  `rfq_no` int(11) NOT NULL DEFAULT '1',

  `po_no` int(11) NOT NULL DEFAULT '1',

  `rec_no` int(11) NOT NULL DEFAULT '1',

  `ret_no` int(11) NOT NULL DEFAULT '1',

  `grl_no` int(11) NOT NULL DEFAULT '1',

  `sr_no` int(11) NOT NULL DEFAULT '1',

  `srfq_no` int(11) NOT NULL DEFAULT '1',

  `so_no` int(11) NOT NULL DEFAULT '1',

  `pcv_no` int(11) NOT NULL DEFAULT '1',

  `cash_no` int(11) NOT NULL DEFAULT '1',

  `con_no` int(11) NOT NULL DEFAULT '1',
```

```
  `adj_no` int(11) NOT NULL DEFAULT '1',
  `bkbk_no` int(11) NOT NULL DEFAULT '1',
  KEY `bkbk_no` (`bkbk_no`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1

-- --------------------------------------------------------
--
Table structure for table `tindakan`

--

CREATE TABLE IF NOT EXISTS `tindakan` (
  `id_tin` int(3) NOT NULL AUTO_INCREMENT,
  `kode_tindakan` varchar(20) NOT NULL,
  `n_tindakan` varchar(50) NOT NULL,
  `biaya` int(50) NOT NULL,
  PRIMARY KEY (`id_tin`),
  KEY `n_tindakan` (`n_tindakan`),
  KEY `n_tindakan_2` (`n_tindakan`),
  KEY `n_tindakan_3` (`n_tindakan`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=0

-- --------------------------------------------------------
--
Table structure for table `total_transaksi`

--

CREATE TABLE IF NOT EXISTS `total_transaksi` (
  `id_transaksi` int(3) NOT NULL,
  `psn_id` int(11) NOT NULL,
```

```
  `no_transaksi` varchar(15) NOT NULL,
  `tgl_transaksi` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `diagnosis` varchar(100) NOT NULL,
  `total_bayar` int(50) NOT NULL,
  `dibayar` int(50) NOT NULL,
  `sorter` int(20) NOT NULL,
  PRIMARY KEY (`id_transaksi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1

-- --------------------------------------------------------
--
Table structure for table `trans_obat`

--

CREATE TABLE IF NOT EXISTS `trans_obat` (
  `psn_id` int(11) NOT NULL,
  `obat` varchar(50) NOT NULL,
  `jml_obat` int(10) NOT NULL,
  `harga` int(50) NOT NULL,
  `sorter` int(20) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1

-- --------------------------------------------------------
--
Table structure for table `trans_tindakan`

--

CREATE TABLE IF NOT EXISTS `trans_tindakan` (
  `psn_id` int(11) NOT NULL,
```

```
  `tindakan` varchar(50) NOT NULL,

  `biaya` int(50) NOT NULL,

  `sorter` int(20) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=latin1

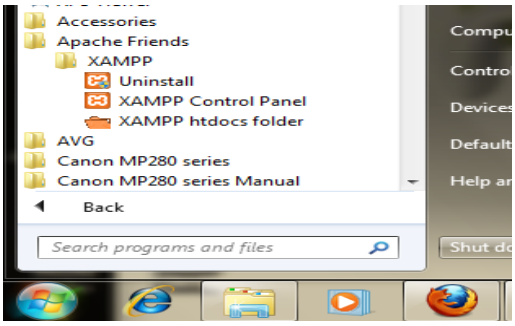
-- --------------------------------------------------------
--
Table structure for table `user_menu`
--

CREATE TABLE IF NOT EXISTS `user_menu` (
  `usr_id` int(4) NOT NULL DEFAULT '0',
  `menu_id` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`usr_id`,`menu_id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
```

5.1.5 Implementasi Instalasi Program

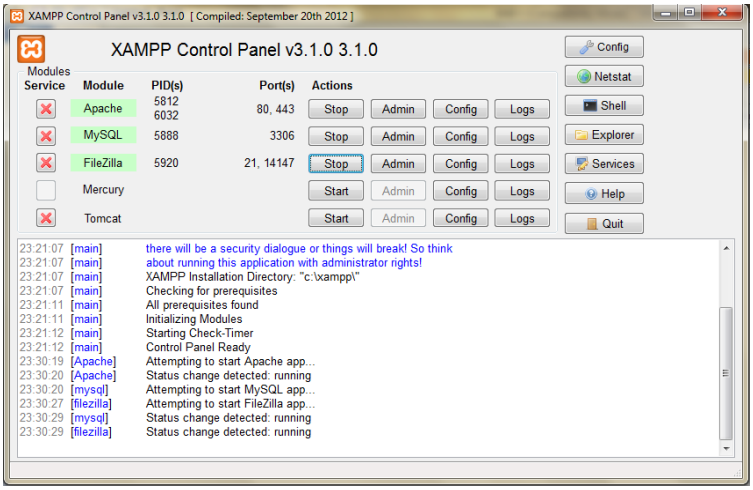
Berikut merupakan implementasi instalasi program bagaimana cara menggunakan program Aplikasi Sistem Informasi Rekam Medis Pasien. Langkah-langkahnya adalah sebagai berikut :

1. Klik menu start kemudian pilih XAMPP Control Panel



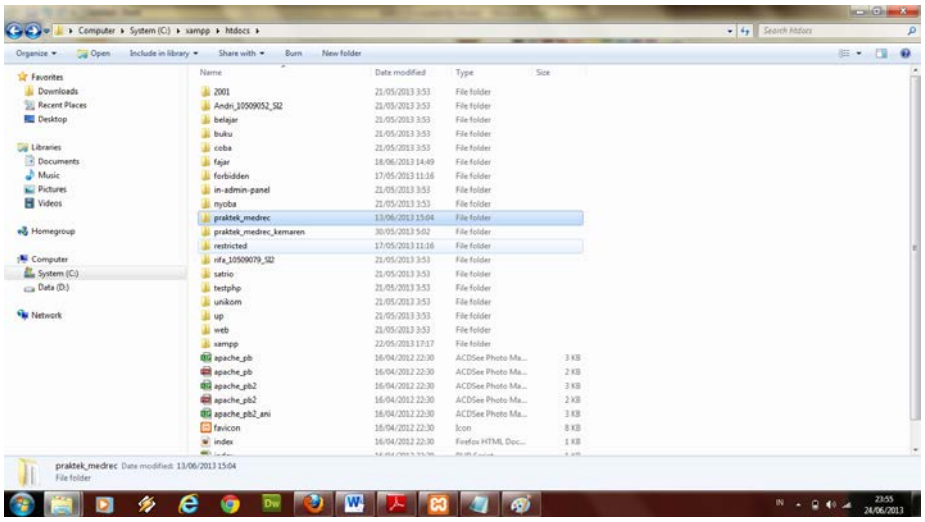
Gambar 5.1 Menjalankan XAMPP

2. Setelah di klik XAMPP, aktifkan Apache, MySQL dan FileZilla.



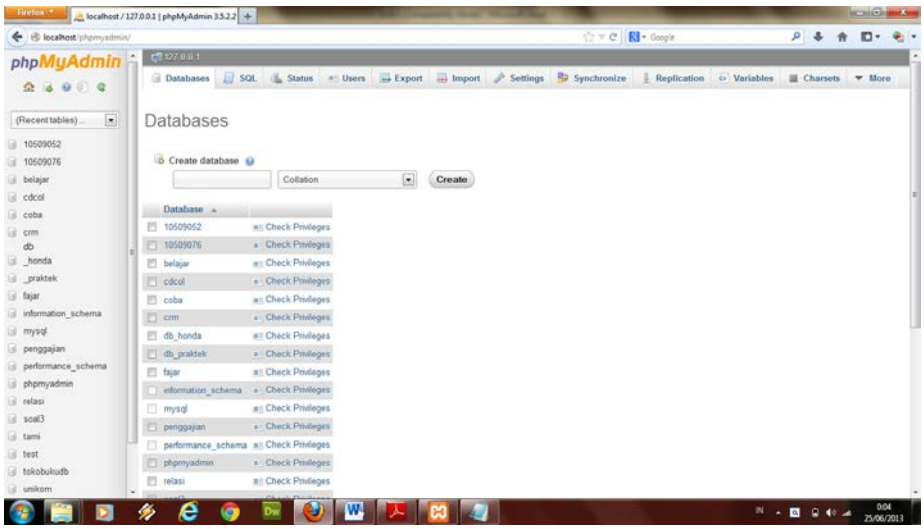
Gambar 5.2 XAMPP Aktif

3. Setelah XAMPP di aktifkan, langkah selanjutnya adalah *copy* folder "praktek_medrec" ke folder htdoc, langkah yang dilakukan : Buka windows explorer – Xampp – htdoc

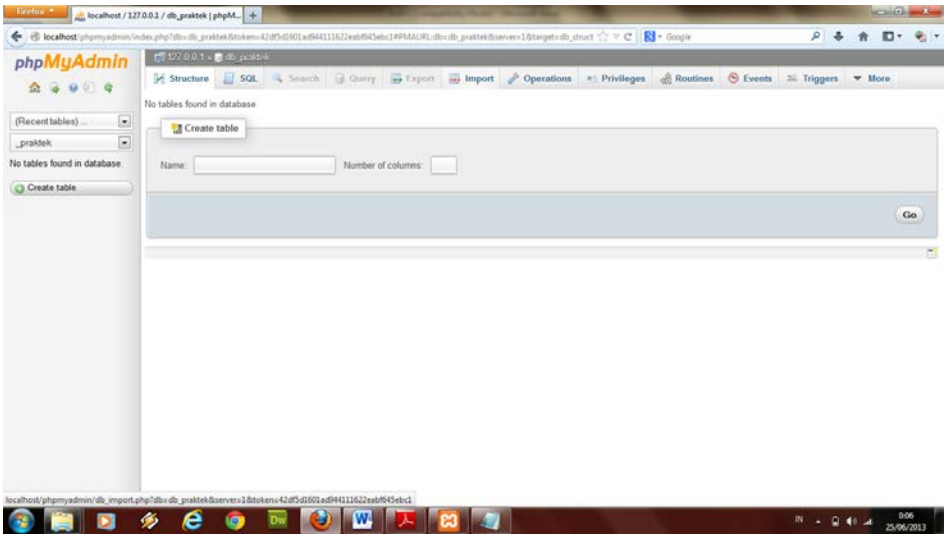


Gambar 5.3 Aplikasi Rekam Medis Pasien dalam folder htcoc

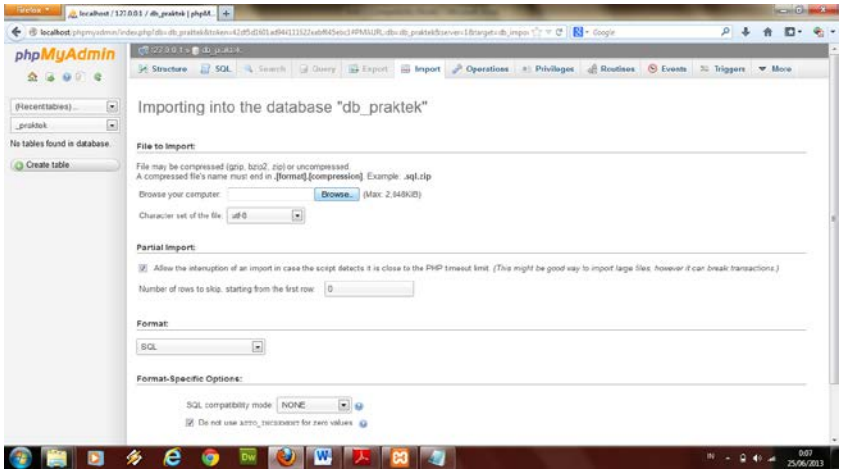
4. Import database “db_praktek.sql” dalam folder db di dalam praktek_medrec kedalam phpmyadmin, langkah yang dilakukan : Buka browser (mozilla firefox, chrome, opera atau yang lainnya) – ketikan localhost/phpmyadmin – create database dengan nama “db_praktek” – Create – Klik Import – Browse – Pilih “db_praktek.sql” dalam folder db di dalam praktek_medrec – Open - Go



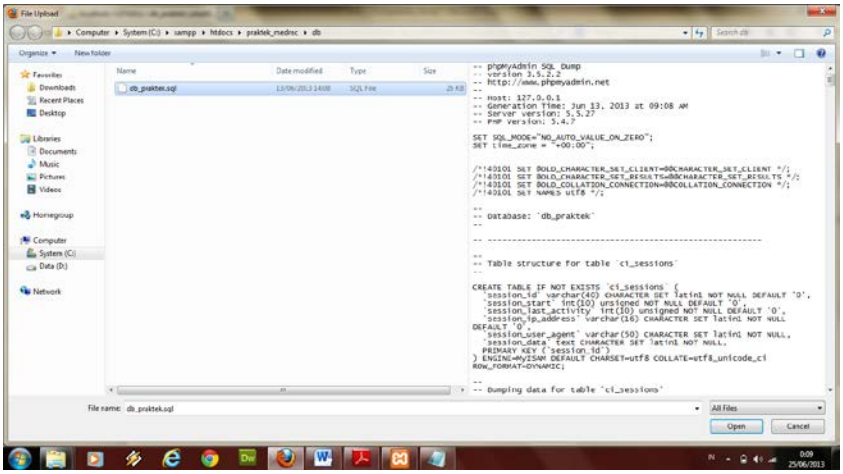
Gambar 5.4 Tampilan phpMyAdmin



Gambar 5.5 Tampilan Create Database



Gambar 5.6 Tampilan Import database



Gambar 5.7 Tampilan pilih database



Gambar 5.8 Tampilan Import database berhasil

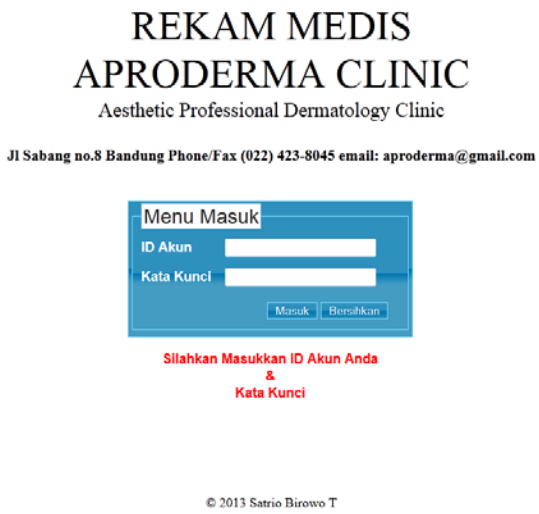
5.1.6 Penggunaan Program

Penggunaan program bertujuan untuk memberikan informasi tentang bagaimana tata cara dalam menjalankan dan menggunakan program Sistem Informasi Rekam Medis Pasien ini.

Berikut adalah petunjuk penggunaan program aplikasi Sistem Informasi Rekam Medis Pasien di APRODERMA CLINIC :

1. Halaman Login

   Menu *login* digunakan untuk masuk ke aplikasi, dimana setiap *user* yang akan masuk ke aplikasi adalah *user* yang sudah terdaftar. *ID Akun* dan *Kata Kunci* diisi sesuai dengan nama yang ada dalam database. Status menentukan hak akses dari sistem yang dibuat.

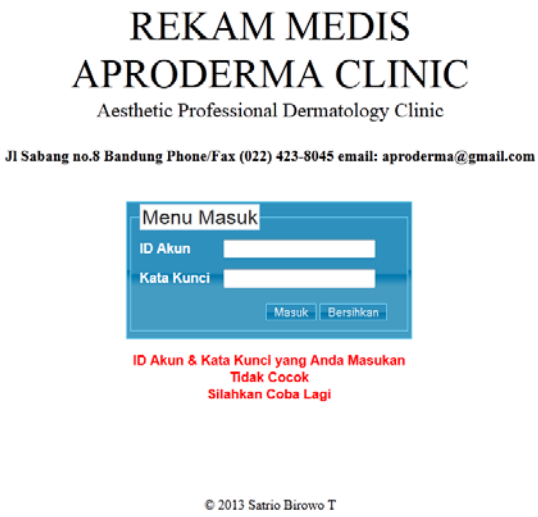


Gambar 5.9 Halaman Login

Langkah- langkah menjalankan form *login* :

1. *User* memasukan *ID Akun*, dan *Kata kunci*. Jika data yang di masukan salah maka akan memunculkan pesan berwarna merah ID Akun & Kata

Kunci yang Anda Masukan Tidak Cocok Silahkan Coba Lagi, dan kembali ke form login.



Gambar 5.10 Pesan *login* salah

2. Jika user memasukkan *ID Akun*, dan *Kata kunci* sesuai, maka akan masuk ke halaman utama.

2. Halaman Utama

Setelah *user* melakukan proses login, maka sistem akan menampilkan halaman utama sesuai dengan hak akses nya masing-masing.



Gambar 5.11 Halaman Utama

3. Form Pendaftaran Pasien

   Form ini digunakan untuk menambah pasien kedalam daftar tunggu.



Gambar 5.12 Form Pendaftaran Pasien

4. Form Form Tambah Pasien

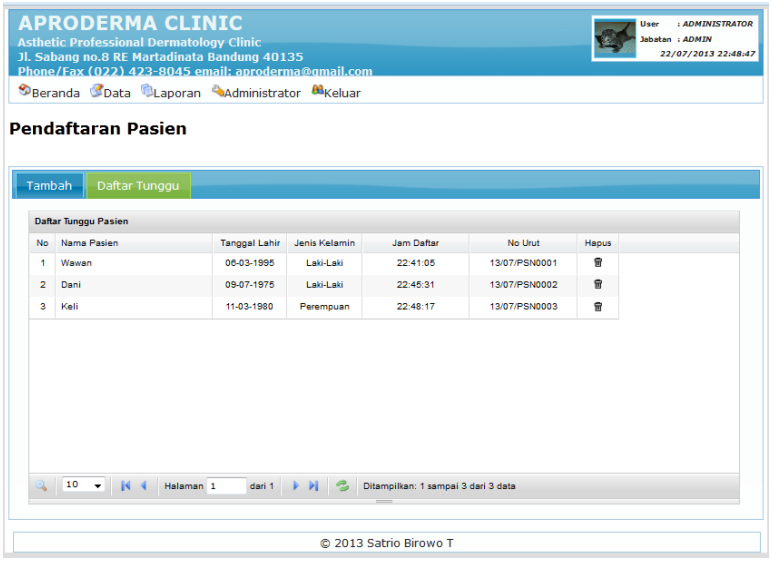
   Jika belum terdaftar dalam database, maka tambah pasien baru dan mengisi form tambah baru pasien.



Gambar 5.13 Form Tambah Pasien

5. Form Daftar Tunggu

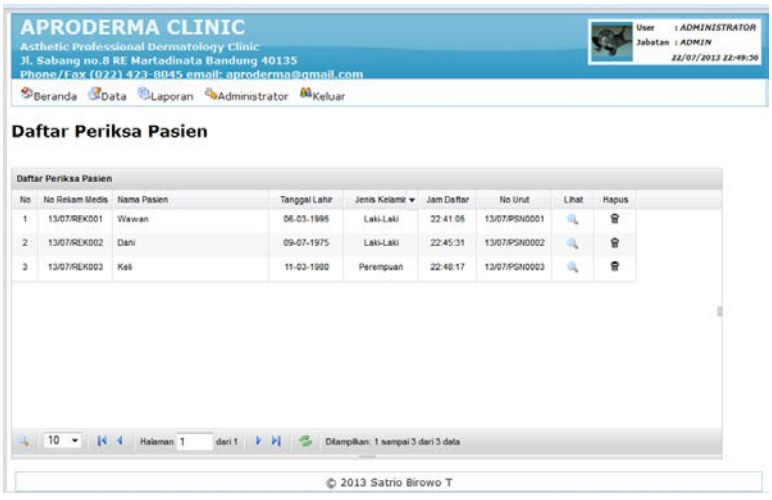
Form ini digunakan untuk memberikan informasi daftar tunggu pasien sesuai dengan no urut terendah.



Gambar 5.14 Form Daftar Tunggu Pasien

6. Form Daftar Periksa Pasien

Form ini digunakan untuk memberikan informasi daftar pasien yang akan diperiksa sesuai dengan daftar tunggu pasien.



. **Gambar 5.15 Form Daftar Periksa Pasien**

7. Form Periksa Pasien

Form ini digunakan untuk menginputkan hasil pemeriksaan dokter terhadap pasien yang nantinya akan menjadi riwayat rekam medis pasien.



Gambar 5.16 Gambar Form Periksa Pasien

8. Form Resep

Form ini digunakan untuk mencetak resep pasien yang diberikan setelah proses pemeriksaan selesai.



Gambar 5.17 Form Resep

9. Form Obat Pasien

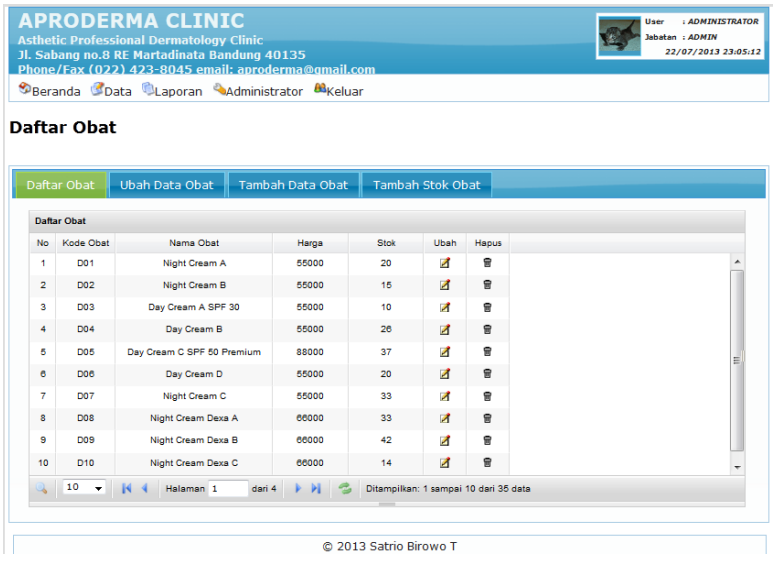
Form ini digunakan untuk memberikan informasi dan mencocokan apakah obat sesuai dengan resep yang diberikan oleh dokter.



Gambar 5.18 Form Obat Pasien

10. Form Persediaan Obat

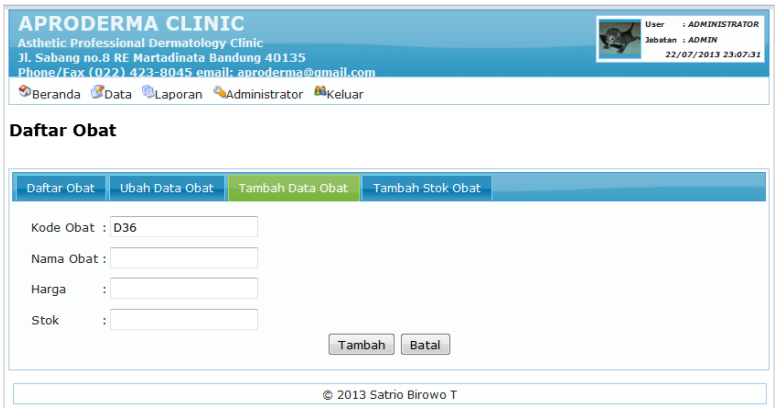
Form ini digunakan untuk melihat daftar obat, menambah stok obat, menambah, mengubah, menghapus data obat.



Gambar 5.19 Form Daftar Obat



Gambar 5.20 Form Ubah Data Obat



Gambar 5.21 Form Tambah Data Obat



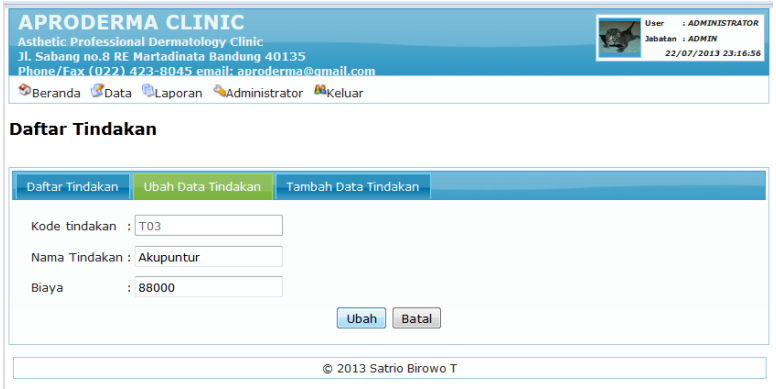
Gambar 5.22 Form Tambah Stok Obat

11. Form Tindakan

Form ini digunakan untuk melihat daftar tindakan, menambah, mengubah, menghapus data tindakan.



Gambar 5.23 Form Daftar Tindakan



Gambar 5.24 Form Ubah Data Tindakan



Gambar 5.25 Form Tambah Data Tindakan

12. Form Transaksi

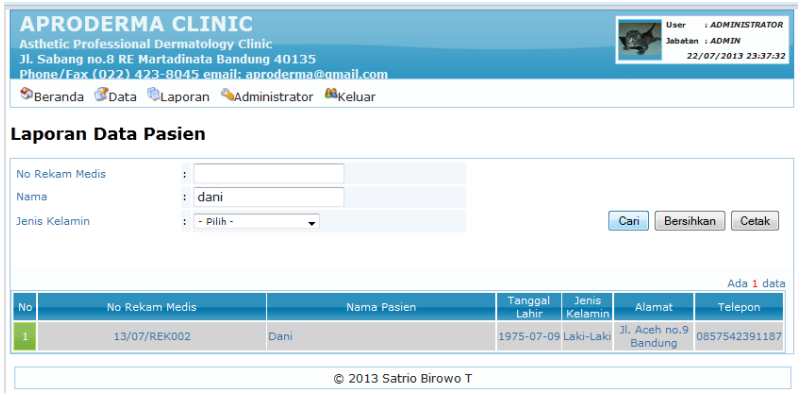
Form ini digunakan untuk memunculkan segala biaya baik tindakan maupun obat yang harus dibayar oleh pasien.



Gambar 5.26 Form Transaksi

13. Form Laporan Data Pasien

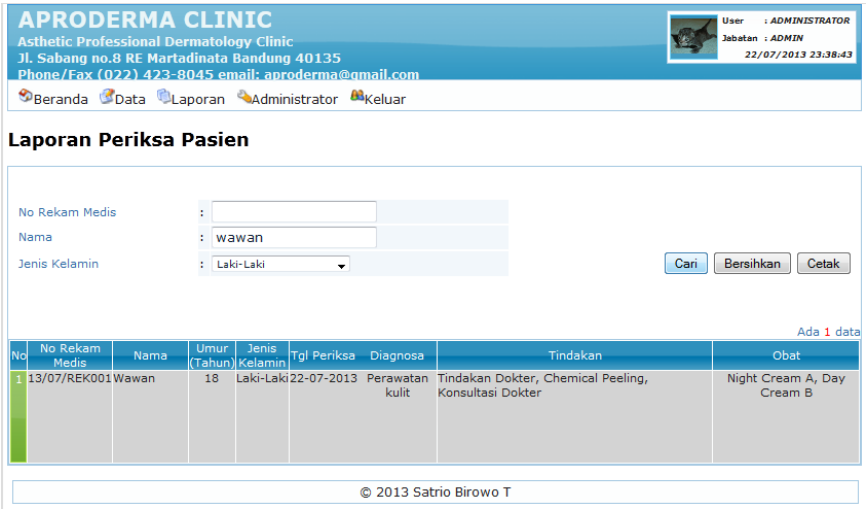
Form ini digunakan untuk melihat pasien yang sudah terdaftar dalam database.



Gambar 5.27 Form Laporan Data Pasien

14. Form Laporan Periksa Pasien

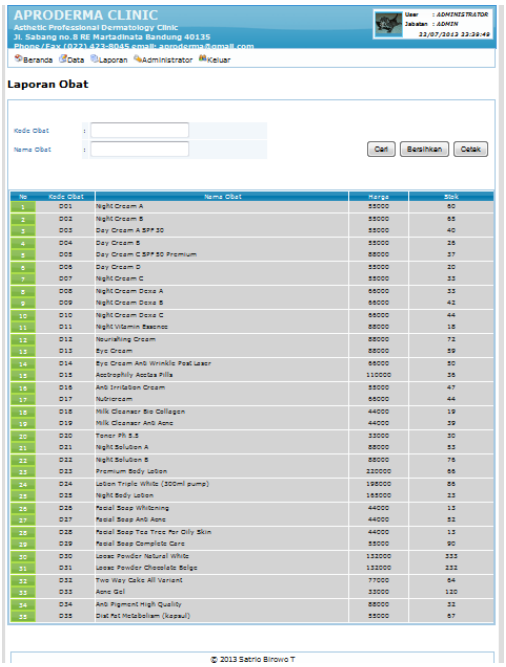
Form ini digunakan untuk melihat riwayat rekam medis pasien dari awal berobat yang sudah terdaftar dalam database.



Gambar 5.28 Form Laporan Periksa Pasien

15. Form Laporan Obat

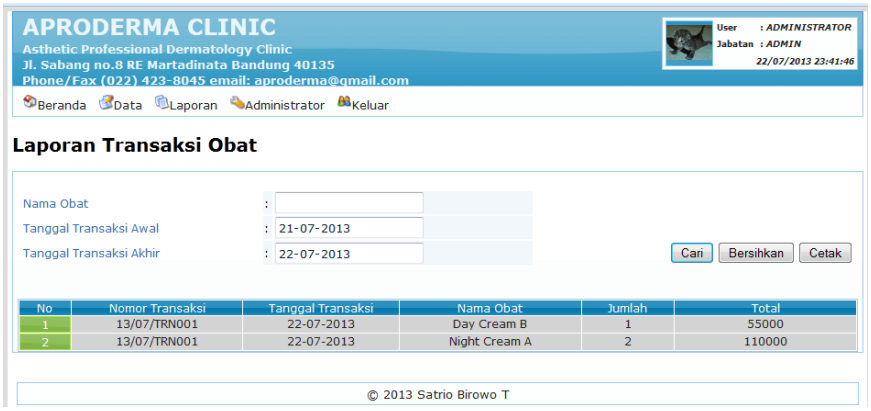
Form ini digunakan untuk melihat riwayat rekam medis pasien dari awal berobat yang sudah terdaftar dalam database.



Gambar 5.29 Form Laporan Obat

16. Form Laporan Transaksi Obat

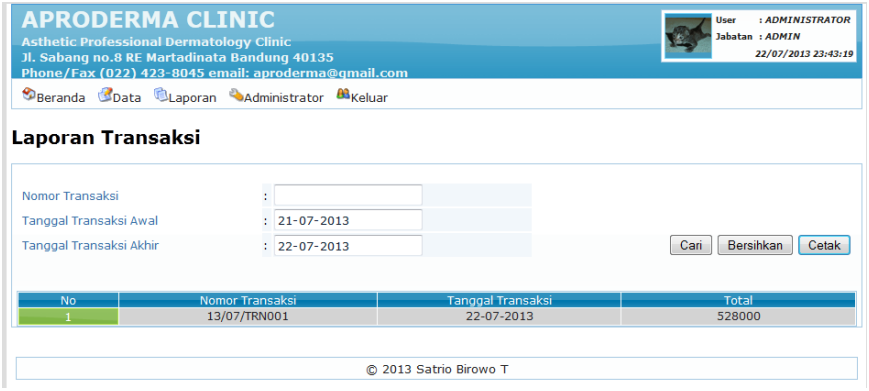
Form ini digunakan untuk melihat segala jenis trasnsaksi obat yang tercatat di setiap pembayaran.



Gambar 5.30 Form Laporan Transaksi Obat

17. Form Laporan Transaksi

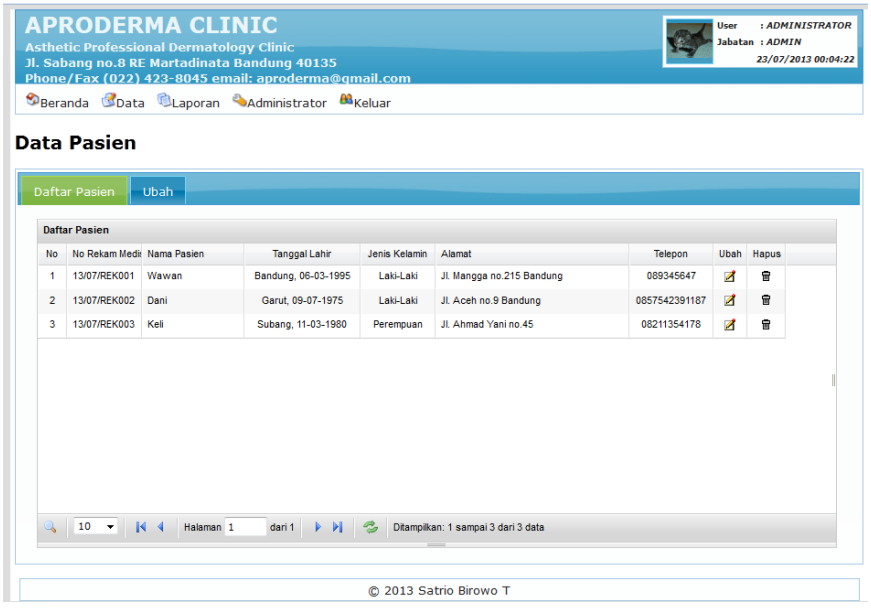
Form ini digunakan untuk melihat segala jenis trasnsaksi yang tercatat di setiap pembayaran.



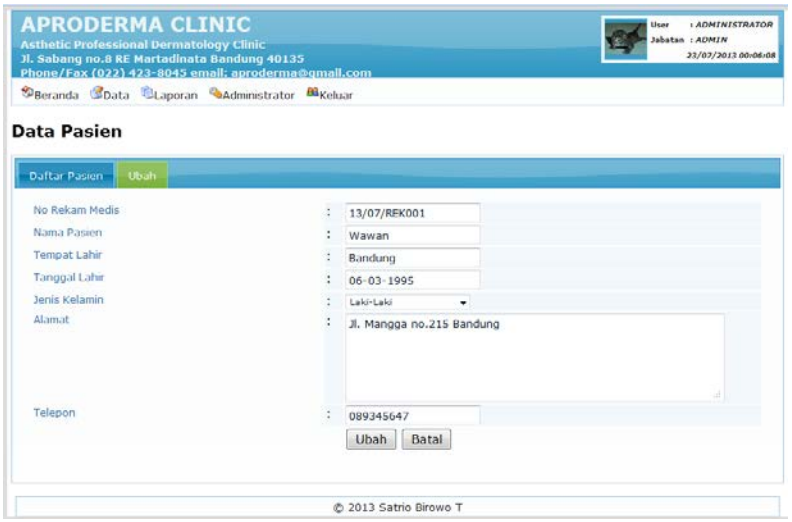
Gambar 5.31 Form Laporan Transaksi

18. Form Data Pasien

Form ini digunakan untuk melihat biodata pasien dan dapat melakukan ubah dan hapus data pasien.



Gambar 5.32 Form Data Pasien



Gambar 5.33 Form Ubah Data Pasien

19. Form Daftar Jabatan

Form ini digunakan sebagai identitas dan untuk mengisi jabatan yang berada di user panel.



Gambar 5.34 Form Daftar Jabatan



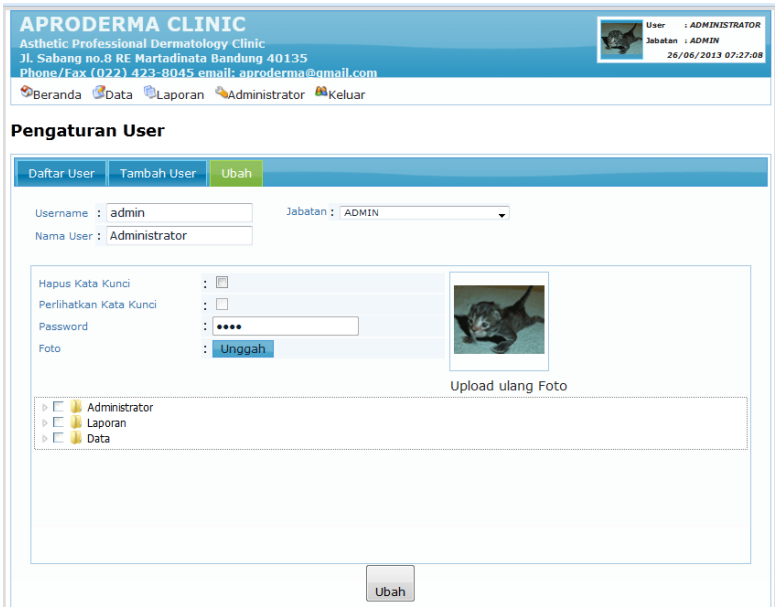
Gambar 5.35 Form Tambah Jabatan

20. Form Daftar User

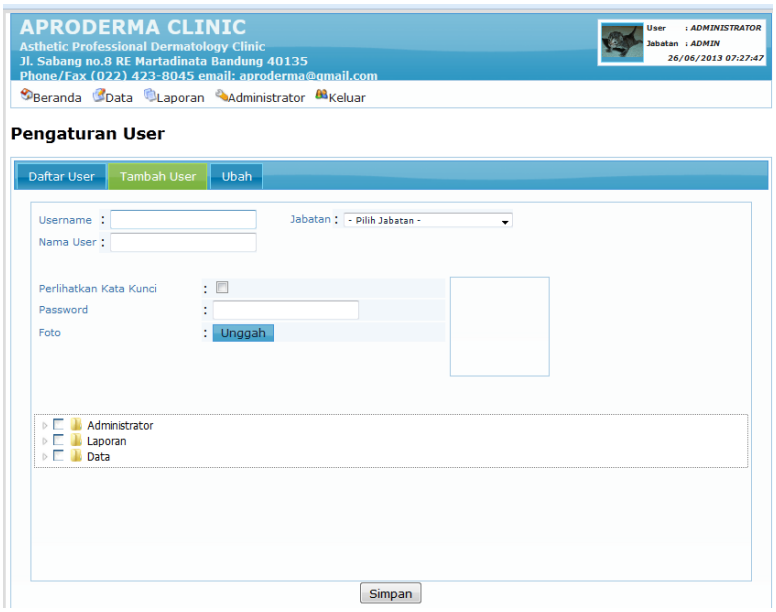
Form ini berfungsi untuk mengetahui user atau pengguna (hak akses) sistem informasi rekam medis pasien dan dapat melakukan tambah, ubah dan hapus user.



Gambar 5.36 Form Daftar User



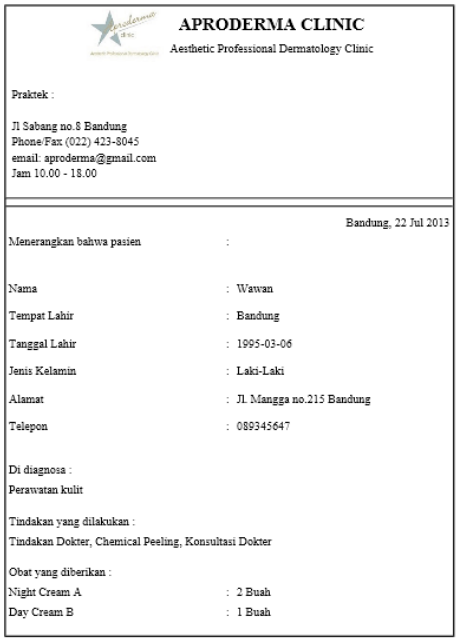
Gambar 5.37 Form Ubah User



Gambar 5.38 Form Tambah User

21. Cetak Resep

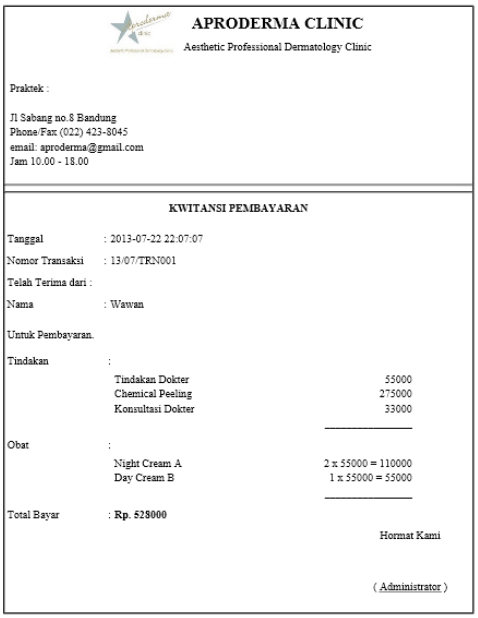
Resep ini dapat dicetak setelah melakukan proses pemeriksaan.



Gambar 5.39 Cetak Resep

22. Cetak Kwitansi Pembayaran

Kwitansi pembayaran dapat dicetak sesuai dengan segala jenis biaya baik tindakan maupun obat yang harus dibayar oleh pasien dan sebagai bukti transaksi pembayaran.



APRODERMA CLINIC
Aesthetic Professional Dermatology Clinic

Praktek :

Jl Sabang no.8 Bandung
Phone/Fax (022) 423-8045
email: aproderma@gmail.com
Jam 10.00 - 18.00

KWITANSI PEMBAYARAN

Tanggal : 2013-07-22 22:07:07
Nomor Transaksi : 13/07/TRN001
Telah Terima dari :
Nama : Wawan

Untuk Pembayaran.

Tindakan	:	
	Tindakan Dokter	55000
	Chemical Peeling	275000
	Konsultasi Dokter	33000
Obat	:	
	Night Cream A	2 x 55000 = 110000
	Day Cream B	1 x 55000 = 55000
Total Bayar	: **Rp. 528000**	

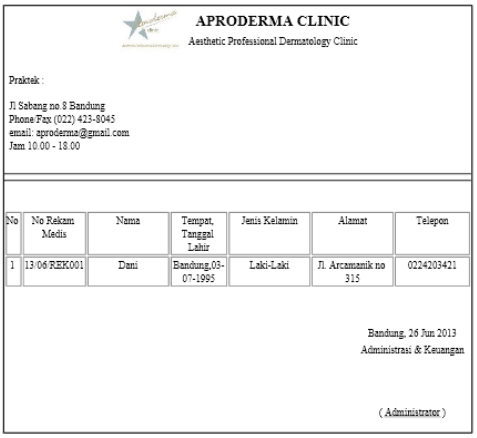
Hormat Kami

(Administrator)

Gambar 5.40 Cetak Kwitansi Pembayaran

23. Cetak Laporan Data Pasien

Laporan Data Pasien dapat dicetak sesuai dengan data pasien yang dimaksud.



APRODERMA CLINIC
Aesthetic Professional Dermatology Clinic

Praktek :

Jl Sabang no 8 Bandung
Phone/Fax (022) 423-8045
email: aproderma@gmail.com
Jam 10.00 - 18.00

No	No Rekam Medis	Nama	Tempat, Tanggal Lahir	Jenis Kelamin	Alamat	Telepon
1	13/06/REK001	Dani	Bandung,03-07-1995	Laki-Laki	Jl. Arcamanik no 315	0224203421

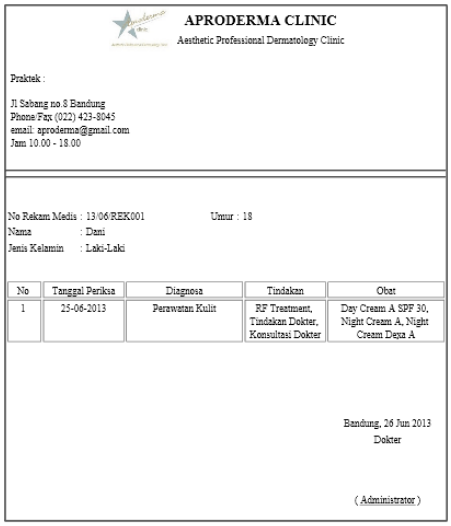
Bandung, 26 Jun 2013
Administrasi & Keuangan

(Administrator)

Gambar 5.41 Cetak Laporan Data Pasien

24. Cetak Laporan Periksa Pasien

Laporan Periksa Pasien dapat dicetak sesuai dengan data pasien yang dimaksud sebagai bukti rekam medis pasien.



APRODERMA CLINIC
Aesthetic Professional Dermatology Clinic

Praktek :

Jl Sabang no.8 Bandung
Phone/Fax (022) 423-8045
email: aproderma@gmail.com
Jam 10.00 - 18.00

No Rekam Medis : 13/06/REK001 Umur : 18
Nama : Dani
Jenis Kelamin : Laki-Laki

No	Tanggal Periksa	Diagnosa	Tindakan	Obat
1	25-06-2013	Perawatan Kulit	RF Treatment, Tindakan Dokter, Konsultasi Dokter	Day Cream A SPF 30, Night Cream A, Night Cream Dexa A

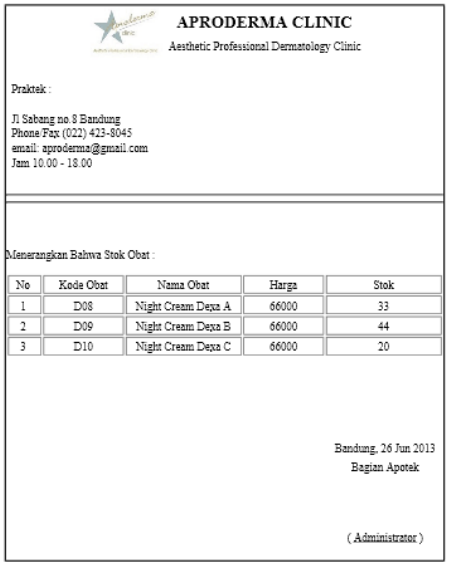
Bandung, 26 Jun 2013
Dokter

(Administrator)

Gambar 5.42 Cetak Laporan Periksa Pasien

25. Cetak Laporan Obat

Laporan Obat dapat dicetak sesuai dengan data obat yang dimaksud sebagai bukti stok obat terakhir.



APRODERMA CLINIC
Aesthetic Professional Dermatology Clinic

Praktek :

Jl Sabang no.8 Bandung
Phone/Fax (022) 423-8045
email: aproderma@gmail.com
Jam 10.00 - 18.00

Menerangkan Bahwa Stok Obat :

No	Kode Obat	Nama Obat	Harga	Stok
1	D08	Night Cream Dexa A	66000	33
2	D09	Night Cream Dexa B	66000	44
3	D10	Night Cream Dexa C	66000	20

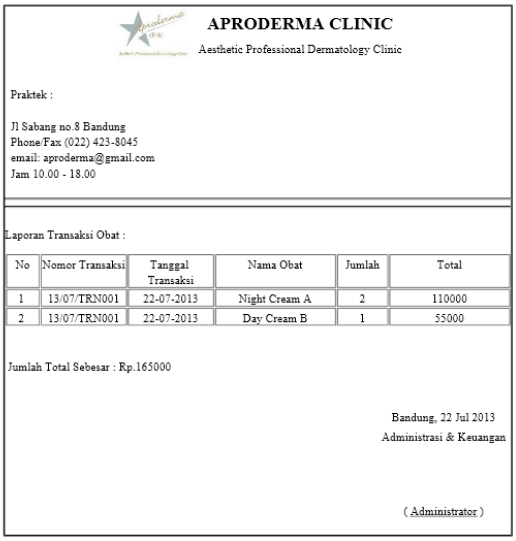
Bandung, 26 Jun 2013
Bagian Apotek

(Administrator)

Gambar 5.43 Cetak Laporan Obat

26. Cetak Laporan Transaksi Obat

Laporan Transaksi Obat dapat dicetak sesuai dengan nama obat dan tanggal transaksi yang dimaksud sebagai bukti obat keluar.



APRODERMA CLINIC

Aesthetic Professional Dermatology Clinic

Praktek :

Jl Sabang no.8 Bandung
Phone/Fax (022) 423-8045
email: aproderma@gmail.com
Jam 10.00 - 18.00

Laporan Transaksi Obat :

No	Nomor Transaksi	Tanggal Transaksi	Nama Obat	Jumlah	Total
1	13/07/TRN001	22-07-2013	Night Cream A	2	110000
2	13/07/TRN001	22-07-2013	Day Cream B	1	55000

Jumlah Total Sebesar : Rp.165000

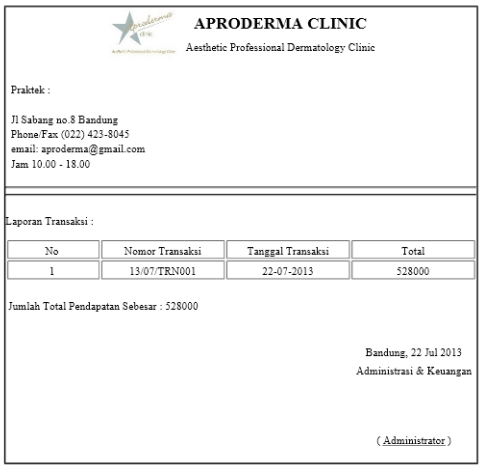
Bandung, 22 Jul 2013
Administrasi & Keuangan

(Administrator)

Gambar 5.44 Cetak Laporan Transaksi Obat

27. Cetak Laporan Transaksi

Laporan Transaksi dapat dicetak sesuai dengan nomor transaksi dan tanggal transaksi yang dimaksud.



APRODERMA CLINIC

Aesthetic Professional Dermatology Clinic

Praktek :

Jl Sabang no.8 Bandung
Phone/Fax (022) 423-8045
email: aproderma@gmail.com
Jam 10.00 - 18.00

Laporan Transaksi :

No	Nomor Transaksi	Tanggal Transaksi	Total
1	13/07/TRN001	22-07-2013	528000

Jumlah Total Pendapatan Sebesar : 528000

Bandung, 22 Jul 2013
Administrasi & Keuangan

(Administrator)

Gambar 5.45 Cetak Laporan Transaksi

5.2 Pengujian

Pengujian merupakan suatu set aktivitas yang direncanakan dan merupakan bagian yang penting dalam siklus pengembangan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Pengujian terhadap program itu sendiri yang bertujuan agar program dapat berjalan dengan baik tanpa mengalami gangguan atau *error*, dan memungkinkan untuk dilakukannya pengembangan sistem lebih lanjut. Pengujian diperlukan tidak hanya untuk meminimalisasi kesalahan secara teknis tetapi juga kesalahan nonteknis

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. Berikut ini adalah rencana pengujian input dan output program Sistem Informasi Rekam Medis Pasien.

5.2.1 Rencana Pengujian

Pengujian perangkat lunak ini menggunakan metode pengujian *Black Box*. Pengujian *Black Box* berfokus pada persyaratan fungsional perangkat lunak yang dibuat. Rencana pengujian selengkapnya terlihat pada tabel dibawah ini :

Tabel 5.1 Rencana Pengujian

Requirement yang diuji	Butir Uji
Login	Melakukan Login
Pendaftaran Pasien	Mencari nama pasien (bagi pasien lama) dan pengisian data pasien

	(bagi pasien baru) serta menambahkan ke daftar tunggu pasien
Pemeriksaan Pasien	Pemeriksaan pasien sesuai dengan daftar periksa pasien, pengisian dan menyimpan hasil pemeriksaan pasien.
Resep	Pengisian data pasien dan cetak resep
Obat Pasien	Pengisian data obat pasien dan menampilkan obat pasien
Persediaan Obat	Menampilkan data obat, melakukan tambah stok obat, tambah, ubah dan hapus data obat.
Tindakan	Menampilkan data tindakan, melakukan tambah , ubah dan hapus data tindakan.
Transaksi	Menampilkan record transaksi beserta seluruh biaya yang akan dibayar dan cetak kwitansi pembayaran
Laporan Data Pasien	Pengisian data pasien dan cetak laporan
Laporan Periksa Pasien	Pengisian data pasien dan cetak laporan
Laporan Obat	Pengisian data obat dan cetak laporan
Laporan Transaksi Obat	Pengisian data transaksi dan cetak laporan
Laporan Transaksi	Pengisian data transaksi dan cetak laporan

Data Pasien	Menampilkan data pasien, melakukan ubah dan hapus data pasien.
Jabatan	Menampilkan daftar jabatan, melakukan hapus dan tambah jabatan.
Daftar User	Menampilkan daftar user, melakukan ubah, hapus dan tambah user.

5.2.2 Kasus dan Hasil Pengujian

Berikut ini uraian hasil pengujian dengan teknik pengujian black boxberdasarkan *requirement* pada rencana pengujian :

Tabel 5.2 Kasus Dan Hasil Pengujian

Requirement	**Skenario Uji**		**Hasil yang diharapkan**	**Hasil Pengujian**
Login	1	Input data login (benar)	Masuk ke menu utama sesuai hak akses	[√] Diterima [ ] Ditolak
	2	Input data login ID Akun salah	Kembali ke form login dan menampilkan pesan ID Akun & Kata Kunci yang anda masukan tidak cocok silahkan coba lagi	[√] Diterima [ ] Ditolak
	3	Input data login Kata Kunci salah		[√] Diterima [ ] Ditolak
Pendaftaran Pasien	1	Input nama pasien dan klik tombol cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak

	2	Klik pilih pada tampilan hasil pencarian data	Pasien secara otomatis terdaftar di daftar tunggu pasien	[√] Diterima [ ] Ditolak
	3	Klik tambah baru	Menampilkan form tambah pasien	[√] Diterima [ ] Ditolak
	4	Klik Batal	Masuk ke daftar tunggu pasien namun tidak menginputkan data pasien	[√] Diterima [ ] Ditolak
	5	Klik Simpan pada form tambah pasien	Menyimpan data pasien dan masuk ke daftar tunggu pasien	[√] Diterima [ ] Ditolak
	6	Klik Batal	Masuk ke daftar tunggu pasien namun tidak menginputkan data pasien	[√] Diterima [ ] Ditolak
	7	Klik Hapus pada daftar tunggu pasien	Data pasien terhapus dari daftar tunggu pasien	[√] Diterima [ ] Ditolak
Pemeriksaan Pasien	1	Klik lihat pada daftar periksa pasien	Masuk ke form input periksa	[√] Diterima [ ] Ditolak
	2	Klik Hapus pada daftar periksa pasien	Data pasien terhapus dari daftar periksa pasien	[√] Diterima [ ] Ditolak
	3	Klik simpan pada input periksa pasien	Menyimpan data periksa pasien dan menampilkan daftar periksa pasien	[√] Diterima [ ] Ditolak
	4	Klik Batal pada	Menampilkan daftar	[√] Diterima [ ] Ditolak

		input periksa pasien	periksa pasien	
Resep	1	Input data pasien, klik cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak
	2	Input data pasien, klik bersihkan	Menampilkan form resep	[√] Diterima [ ] Ditolak
	3	Klik cetak resep	Menampilkan cetak resep	[√] Diterima [ ] Ditolak
Obat Pasien	1	Klik cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak
	2	Klik bersihkan	Menampilkan form Obat pasien	[√] Diterima [ ] Ditolak
Persediaan Obat	1	Klik ubah	Menampilkan form ubah	[√] Diterima [ ] Ditolak
	2	Klik hapus	Menghapus data obat	[√] Diterima [ ] Ditolak
	3	Klik tambah stok obat	Menampilkan form tambah stok obat	[√] Diterima [ ] Ditolak
	4	Klik tambah data obat	Menampilkan form tambah data obat	[√] Diterima [ ] Ditolak
	5	Klik simpan pada tambah stok obat	Menyimpan data stok obat dan menampilkan data obat	[√] Diterima [ ] Ditolak
	6	Klik batal pada tambah stok obat	Menampilkan data obat	[√] Diterima [ ] Ditolak
	7	Klik simpan pada tambah data obat	Menyimpan data obat dan menampilkan data obat	[√] Diterima [ ] Ditolak
	8	Klik batal pada tambah data obat	Menampilkan data obat	[√] Diterima [ ] Ditolak

Tindakan	1	Klik ubah	Menampilkan form ubah	[√] Diterima [ ] Ditolak
	2	Klik hapus	Menghapus data tindakan	[√] Diterima [ ] Ditolak
	3	Klik tambah data tindakan	Menampilkan form tambah data tindakan	[√] Diterima [ ] Ditolak
	4	Klik simpan pada tambah data tindakan	Menyimpan data tindakan dan menampilkan data tindakan	[√] Diterima [ ] Ditolak
	5	Klik batal pada tambah data tindakan	Menampilkan data tindakan	[√] Diterima [ ] Ditolak
Transaksi	1	Klik Combo box rekam medis	Memunculkan data transaksi	[√] Diterima [ ] Ditolak
	2	Klik Simpan	Menyimpan transaksi kedalam database	[√] Diterima [ ] Ditolak
	3	Klik Batal	Menampilkan form transaksi	[√] Diterima [ ] Ditolak
	4	Klik Cetak	Mencetak kwitansi pembayaran	[√] Diterima [ ] Ditolak
Laporan Data Pasien	1	Klik Cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak
	2	Klik Bersihkan	Menampilkan form laporan data pasien	[√] Diterima [ ] Ditolak
	3	Klik Cetak	Mencetak laporan data pasien	[√] Diterima [ ] Ditolak
Laporan Periksa Pasien	1	Klik Cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak
	2	Klik Bersihkan	Menampilkan form	[√] Diterima

			laporan periksa pasien	[ ] Ditolak
	3	Klik Cetak	Mencetak laporan periksa pasien	[√] Diterima [ ] Ditolak
Laporan Obat	1	Klik Cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak
	2	Klik Bersihkan	Menampilkan form laporan obat	[√] Diterima [ ] Ditolak
	3	Klik Cetak	Mencetak laporan obat	[√] Diterima [ ] Ditolak
Laporan Transaksi Obat	1	Klik Cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak
	2	Klik Bersihkan	Menampilkan form laporan transaksi obat	[√] Diterima [ ] Ditolak
	3	Klik Cetak	Mencetak laporan transaksi obat	[√] Diterima [ ] Ditolak
Laporan Transaksi	1	Klik Cari	Menampilkan hasil pencarian	[√] Diterima [ ] Ditolak
	2	Klik Bersihkan	Menampilkan form laporan transaksi	[√] Diterima [ ] Ditolak
	3	Klik Cetak	Mencetak laporan transaksi	[√] Diterima [ ] Ditolak
Data Pasien	1	Klik Ubah	Masuk ke form ubah	[√] Diterima [ ] Ditolak
	2	Klik Hapus	Menghapus data pasien	[√] Diterima [ ] Ditolak
Jabatan	1	Klik Hapus	Menghapus data pasien	[√] Diterima [ ] Ditolak
	2	Klik form tambah	Masuk ke form tambah jabatan	[√] Diterima [ ] Ditolak

	3	Klik simpan pada form tambah jabatan	Menyimpan jabatan dan menampilkan form jabatan	[√] Diterima [ ] Ditolak
Daftar User	1	Klik ubah	Masuk ke form ubah	[√] Diterima [ ] Ditolak
	2	Klik Hapus	Menghapus data user	[√] Diterima [ ] Ditolak
	3	Klik tambah user	Masuk ke form tambah user	[√] Diterima [ ] Ditolak
	4	Klik Simpan pada form tambah user	Menyimpan user dan menampilkan form daftar user	[√] Diterima [ ] Ditolak

5.2.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus *sample* uji yang telah dilakukan memberikan kesimpulan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

Namun pengujian tersebut dapat dikatakan belum sempurna, karena hanya dilakukan pada satu sisi pengujian. Dari semua yang telah dilakukan dalam pengujian ini diharapkan dapat mewakili pengujian fungsi yang lain dalam program Sistem Informasi Rekam Medis Pasien ini.