BAB III

KEGIATAN SELAMA KERJA PRAKTEK

3.1 Jadwal Kerja Praktek

Kerja praktek yang dilaksanakan di Pusat Penelitian Informatika - LIPI berlangsung dari tanggal 7 Juli 2008 sampai dengan 6 Agustus 2008. Pada hari pertama pembimbing perusahaan memberikan penjelasan mengenai tugas yang harus dikerjakan selama kerja praktek. Penyelesaian tugas kerja praktek dilakukan di rumah. Hasil pekerjaan diserahkan dan disetujui kepada pembimbing perusahaan yang selanjutnya dilaksanakan pembuatan laporan.

3.2 Teknik Kerja Praktek

Adapun teknik kerja praktek yaitu mencangkup studi literatur dan wawancara.

3.2.1 Studi Literatur

Melakukan penelusuran pustaka dan sumber lainnya mengenai segala sesuatu yang dibutuhkan dalam merancang suatu aplikasi registrasi rumah sakit. Dalam mengerjakan laporan kerja praktek ini, penulis melakukan studi pustaka, untuk memperoleh informasi yang berhubungan dengan materi laporan kerja praktek melalui buku-buku, bahan kuliah dan bacaan lainnya yang memiliki relevansi dengan perancangan aplikasi registrasi pasien rumah sakit.

3.2.2 Wawancara

Melakukan kegiatan tanya jawab ke pebimbing lapangan kerja praktek. mengenai permintaan apa saja yang diinginkan dalam aplikasi yang akan dibangun sebagai inforamsi tambahan agar sesuai dan dapat digunakan dengan baik.

3.3 Data Kerja Praktek

Data kerja praktek yaitu mencangkup landasan teori, perancangan database, dan perancangan antarmuka.

3.3.1 Landasan Teori

Landasan teori merupakan teori-teori yang berhubungan dengan pembangunan aplikasi registrasi pasien pada rumah sakit.

3.3.1.1 Pengertian Sistem

Pengertian sistem menurut Jerry FithGerald yang menyatakan : **"***sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu".*

Sistem dapat didefinisikan juga sebagai suatu kesatuan yang terdiri dari dua atau lebih komponen atau sub-sistem yang berinteraksi untuk mencapai suatu tujuan .

Dari pengertian diatas dapat diambil kesimpulan bahwa didalam sistem terdiri dari sistem-sistem bagian (subsistem). Masing-masing subsistem dapat

terdiri dari subsistem yang lebih kecil lagi. Subsistem ini saling berinteraksi dan saling berhubungan membentuk satu kesatuan (terintegrasi) sehingga tujuan sistem tersebut dapat tercapai.

Suatu sistem mempunyai karakteristik yang tidak dapat dipisahkan antara satu sistem dengan sistem yang lainnya. Adapun karakteristik sistem tersebut adalah :

a. Komponen Sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen- komponen atau subsistem-subsistem.

b. Batas Sistem

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem ini dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

c. Lingkungan Luar Sistem

   Lingkungan luar dari suatu sistem adalah apapun diluar dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga dapat bersifat merugikan sistem.

d. Penghubung Sistem

Penghubung merupakan media penghubung antara satu subsistem dengan satu subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumberdaya mengalir dari satu subsistem ke subsistem lainnya.

e. Masukan Sistem

Masukan *(input)* adalah energi yang dimasukkan kedalam sistem. Masukan dapat berupa masukan perawatan *(maintenance input)* dan masukan sinyal *(signal input)*. *Maintenance input* adalah energi yang dimasukan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran.

f. Keluaran Sistem

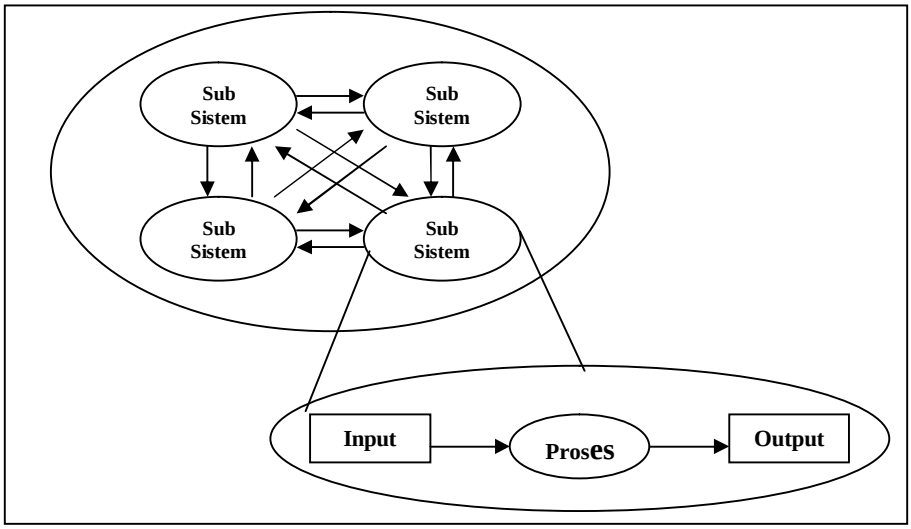
Keluaran *(input)* adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.

g. Pengolah Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran.

h. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan *(goal)* atau sasaran *(objective)*. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem da keluaran yang akan dihasilkan sistem.



Gambar 3.1 Karakteristik Sistem

Sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi, oleh karena itu sistem dapat diklasifikasikan kedalam beberapa sudut pandang, yaitu :

a. Sistem Abstrak *(abstract sistem)* adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya : Sistem *Teologia*, yaitu sistem yang berupa pemikiran-pemikiran hubungan manusia dengan Tuhan.

b. Sistem Fisik merupakan sistem yang ada secara fisik. Misalnya : Sistem Komputer,Sistem Akuntansi, dan Sistem produksi.

c. Sistem Alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Misalnya : Sistem perputaran bumi.

d. Sistem buatan manusia adalah sistem yang dirancang oleh manusia. Sistem yang melibatkan interaksi manusia dengan mesin disebut dengan *Human Machine Sistem*. Misalnya : Sistem Informasi.

e. Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti, sehingga keluaran dari sistem dapat diramalkan. Misalnya : Sistem Komputer.

f. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

g. Sistem tertutup adalah sitem yang tidak behubungan dan tidak terpengaruh dengan lingkungan luarnya. Siatem ini bekerja secara otomatis tanpa adanya turut campur dari pihak luarnya.

h. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Istem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya.

3.3.1.2 Pengertian Informasi

Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang berguna untuk pengambilan keputusan.

Sumber dari Informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan saat tertentu.

Informasi yang bersumber dari proses data harus merupakan informasi yang terstruktur. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

3.3.1.3 Pengertian Sistem Informasi

Informasi merupakan hal yang sangat penting bagi manajemen didalam pengambilan keputusan. Informasi dapat diperoleh dari sistem informasi *(Information Sistems)* atau disebut juga dengan *processing sistems.* Ssitem informasi didefinisikan sebagai berikut :*"Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan".*

Sistem informasi terdiri dari komponen-komponen yang saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarannya. Adapun komponen-komponen tersebut adalah :

1. Blok Masukan

   Input mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

   Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan suatu keluaran yang diinginkan.

3. Blok Keluaran

   Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

   Teknologi merupakan alat dari sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakse data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu : *Brainware, Software, dan Hardware*.

5. Blok Basis Data

   Basis data *(Database)* merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras computer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan didalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data didalam basis data perlu diorganisasikan sedemikin rupa, supaya informasi yang dihasilkan berkualitas. Basis data diakses atau

dimanipulasi dengan menggunkan perangkat lunak paket yang disebut dengan *DBMS (Database Management Sistems).*

6. Blok Kendali

   Banyak hal yang dapat merusak sistem informasi, seperti misalnya : bencana alam, api atau kegagalan sistem itu sendiri. Untuk itu beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah.

3.3.1.4 Perancangan Sistem

Perancangan sistem (*Sistem Design*) memiliki pengertian : "*Proses untuk mengkonversi/memindahkan sistem informasi dari spesifikasi logis atau konseptual kedalam sebuah bentuk yang dapat diimplementasikan pada sistem komputer*".

Tujuan dari *Sistem Design* yaitu :

a. Mengkonversi spesifikasi terstruktur kedalam sebuah desain yang handal dan dapat dipelihara.
b. Menyusun rencana yang memastikan bahwa sistem tersebut handal, lengkap dan akurat.
c. Menyusun manual sehingga pemakai dapat menggunakan sistem baru secara efektif dan efisien.

Tahap-tahap perancangan sistem :

a. *Output Design,* proses perancangan *layout* secara fisik untuk setiap hasil dari sistem dan mendefinisikan pembagian/penempatan dari *layout* tersebut.
b. *Input Design,* proses menentukan dan memperhitungkan bagaimana data dimasukan kedalam sistem.
c. *Database Systems Design,* proses perancangan database mulai dari pembuatan struktur sampai dengan bagaimana data tersebut dipergunakan sehingga mudah untuk digunakan dan dapat menghasilkan informasi yang dibutuhkan.
d. *Process Design,* proses perancangan kode-kode program dengan menggunakan bahasa pemrograman tertentu.
e. *Software Design,* proses perancangan aplikasi secara utuh.

3.3.1.5 Basis Data

Basis Data (*Database*) dapat dibayangkan sebagai sebuah lemari arsip. Basis Data terdiri dari atas dua kata, yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan dan sebagainya, yang direkam dalam bertuk angka, huruf, symbol, teks, gambar, bunyi, atau kombinasinya.

Basis Data dapat diartikan sebagai himpunan atau sekumpulan data bisa berupa tabel atau file yang saling berhubungan dan disimpan dalam media penyimpanan elektronis tanpa pengulangan (redudansi).

Setiap basis data umumnya dibuat untuk mewakili sebuah semesta data yang spesifik. Karena itu, operasi-operasi dasar yang dapat dilakukan berkenaan dengan basis data dapat meliputi :

a. pembuatan database baru (*create database*) yaitu membuat database baru.

b. Penghapusan database (*drop database*) yaitu penghapusan suatu database.

c. Pembuatan tabel baru ke dalam suatu databse yaitu penambahan file/tabel ke dalam database.

d. Penambahan/pengisian data baru ke sebuah file/tabel disebuah database (*insert*)

e. Pengambilan data dari sebuah file/tabel (*retrieve/search*) yaitu pencarian data pada suatu file.

f. Pengubahan data dari sebuah file/tabel (*update*) yaitu perbaikan data pada suatu file.

g. Penghapusan data dari sebuah file/tabel (*delete*).

Operasi yang berkenaan dengan pembuatan objek (basis data dan tabel) merupakan operasi awal yang hanya dilakukan sekali dan berlaku seterusnya.

Sedangkan operasi-operasi yang berkaitan dengan isi tabel (data) merupakan operasi rutin yang akan berlangsung berulang-ulang dank arena itu operasi-operasi inilah yang lebih tepat mewakili aktifitas pengelolaan (*management*) dan pengolahan (*processing)* data dalam basis data.

Tujuan awal dan utama dalam pengelolaan data dalam sebuah basis data agar memperoleh atau menemukan kembali data yang dicari dengan mudah dan cepat. Disamping itu, pemanfaatan basis data untuk pengelolaan data, juga memiliki tujuan lain seperti berikut :

1. Kecepatan dan Kemudahan (*Speed)*
2. Efisiensi Ruang Penyimpanan (*Space)*
3. Keakuratan (*Accuracy)*
4. Ketersediaan (*Availability)*
5. Kelengkapan (*Completeness)*
6. Keamanan (*Security)*
7. Kebersamaan Pemakaian (*Sharability*)

Basis Data hanya sebuah objek yang pasif/mati. Basis data tidak akan pernah berguna jika tidak ada penggeraknya. Yang menjadi pengelola atau penggeraknya secara langsung adalah program atau aplikasi (*software)*. Gabungan keduanya (basis data dan pengelolanya) menghasilkan sebuah sistem. Karena itu, secara umum sistem basis data merupakan sistem yang terdiri atas kumpulan file (tabel) yang saling berhubungan dan sekumpulan program (DBMS) yang memungkinkan beberapa pemakai (program lain) untuk mengakses dan

memanipulasi file-file (tabel-tabel) tersebut. Sistem basis data memiliki beberapa komponen pendukung diantaranya :

a. Perangkat Keras

   Perangkat keras yang biasanya terdapat dalam sebuah sistem basis data adalah

   1. Komputer (satu untuk sistem yang *stand-alone* atau lebih untuk sistem jaringan).
   2. Memori sekunder yang *on-line* (*Harddisk).*
   3. Meori sekunder yang *off-line* (*Tape* atau *Removeable Disk)* untuk keperluan backup data.
   4. Media/perangkat komunikasi (untuk sistem jaringan).

b. Sistem Operasi (*Operating Sistem)*

   Secara sederhana, Sistem Operasi merupakan program yang mengaktifkan/memfungsikan sistem computer, mengendalikan seluruh sumber daya (*resource)* dalam computer dan melakukan operasi-operasi dasar dalam computer (operasi I/O, pengelolaan file, dan lain-lain). Sejumlah Sistem Operasi yang banyak digunakan seperti : MS-DOS, MS-Windows 3.1, MS-Windows XP, MS-Windows Vista (2007), UNIX, dan lain-lain. Program pengelola basis data hanya dapat aktif jika sistem operasi yang dikehendaki (sesuai) telah aktif.

c. Basis Data (*Database)*

   Sebuah sistem basis data dapat memiliki beberapa basis data. Setiap basis data dapat berisi/memiliki sejumlah objek basis data (seperti file/tabel, indeks, dan lain-lain). Disamping berisi/menyimpan data, setiap basis data juga

mengandung definisi struktur (baik untuk bais data maupun objek-objeknya secara detail).

d. Sistem Pengelola Basis Data (*Database Management System/DBMS*)

Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak (sistem) yang khusus/spesefik. Perangkat lunak inilah yang akan menentukan bagaimana data organisasi, disimpan, diubah dan diambil kembali. DBMS juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan konsistensi data, dan sebagainya.

Perangkat lunak yang termasuk DBMS seperti dBase III+, dBaseIV, FoxBase, Rbase, MS-Access dan Borland-Paradox (untuk kelas sederhana) atau Borland-Interbase, MS-SQLServer, CA-Open Ingres, Oracle, Informix dan Sybase (Untuk kelas kompleks/berat).

e. Pemakai (*User)*

Ada beberapa jenis/tipe pemakai terhadap suatu sistem basis data yang dibedakan berdasarkan cara mereka berinteraksi terhadap sistem :

1. Programmer Aplikasi

   Pemakai yang berinteraksi dengan basis data melalui *Data Manipulation Language* (DML), yang disertakan dalam program yang ditulis dalam bahasa pemrograman induk (seperti C, Pascal, Cobol, dan lain-lain).

2. User Mahir (*Casual User*)

Pemakai yang berinteraksi dengan sistem tanpa menulis modul program. Mereka menyatakan *guery* (untuk akses data) dengan bahasa *query* yang telah disediakan oleh DBMS.

3. User Umum (*End User Naive User)*

   Pemakai yang berinteraksi dengan sistem basis data melalui pemanggilan satu program aplikasi permanen (*Executable program)* yang telah ditulis/disediakan sebelumnya.

4. User Khusus (*Specialized User)*

   Pemakai yang menulis aplikasi basis data non konvensional tetapi untuk keperluan-keperluan khusus, seperti untuk aplikasi AI, Sistem Pakar, Pengolahan Citra, dan lain-lain, yang bisa saja mengakses basis data dengan /tanpa DBMS yang bersangkutan.

   Untuk sebuah basis data yang *stand-alone,* maka pada suatu saat hanya ada satu pemakai yang dapat bekerja. Sedangkan untuk sistem basis data dalam jaringan, maka pada suatu saat ada banyak pemakai yang dapat berhubungan (menggunakan) basis data yang sama.

f. Aplikasi (Perangkat Lunak) Lain

   Aplikasi (Perangkat Lunak) lain ini bersifat opsional. Artinya ada tidaknya tergantung kebutuhan. DBMS yang digunakan lebih berperan dalam pengorganisasian data dalam basis data, sementara bagi pemakai basis data (khususnya yang menjadi *end-user/naïve-user)* dapat disediakan program khusus lain untuk melakukan pengisian, pengubahan dan pengambilan data. Program ini ada yang sudah disediakan bersama dengan DBMS-nya, ada juga

yang harus dibuat sendiri dengan menggunakan aplikasi lain yang khusus untuk itu (*development tools).*

3.3.1.6 *Structured Query Language (SQL)*

DBMS merupakan perantara bagi pemakai dengan basis data dalam *disk.* Cara berinteraksi/berkomunikasi antara pemakai dengan basis data tersebut diatur dalam suatu bahasa khusus yang ditetapkan oleh perusahaan pembuat DBMS. Bahasa itu dapat kita sebut sebagai bahasa basis data yang terdiri atas sejumlah perintah yang diformulasikan dan dapat diberikan *user* dan dikenali/diproses oleh DBMS untuk melakukan suatu aksi/pekerjaan tertentu.

Sebuah Bahasa Basis Data atau *Structured Query Language* dapat dibagi kedalam tiga bentuk yaitu :

1. *Data Definition Language* (DDL)

   DDL merupakan bagian dari sistem manajemen *database*, dipakai untuk mendefinisikan dan mengatur semua atribut dan properti dari sebuah *database.* Dengan DDL kita dapat menentukan tata letak baris, definisi kolom, kolom-kolom kunci, lokasi *file* dan strategi penyimpanan. Dengan DDL kita juga dapat mendefinisikan *database*, tabel dan *view.*

2. *Data Manipulation Language* (DML)

   Merupakan bentuk Bahasa Basis Data yang berguna untuk melakukan manipulasi dan pengambilan data pada suatu *database*. Manipulasi data dapat berupa :

a. Penyisipan / Penambahan data baru ke suatu basis data.
b. Penghapusan data dari suatu basis data.
c. Pengubahan data di suatu basis data

3. *Data Control Language* (DCL)

   Dipakai untuk mengontrol hak-hak pada objek-objek *database*. Seperti memberikan hak kepada *user* untuk mengakses sebuah *database*, membuang hak yang telah diberikan, membuat sebuah izin pada sebuah *account* melalui *group* atau keanggotaan *role*.

3.3.1.7 Relasi Tabel

Relasi tabel menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Dari adanya relasi tabel tersebut terdapat kardinalitas relasi yang menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Terdapat empat kemungkinan kombinasi kardinalitas yaitu : satu kesatu *(One To One)*, satu ke banyak *(One To Many)*, dan banyak ke banyak *Many To Many)*.

3.3.1.8 MYSQL

MySQL merupakan sebuah *software* yang berguna sebagai suatu *Database Server* yang cukup terkenal. Kepopulerannya seiring dengan penggunaan *script* PHP untuk *web programming*. *Database server* itu sendiri merupakan suatu *software* yang bertugas untuk melayani permintaan (*request*) query dari *client*.

MySQL sebagai suatu *database server* mempunyai beberapa kemampuan, salah satunya harus menyediakan suatu sistem manajemen *databas*e yang dapat mengatur bagaimana menyimpan, menambah, mengakses data dan transaksi-transaksi *database* lainnya. MySQL cepat sekali berkembang, karena MySQL merupakan suatu *software* yang *Open Source.*

Website MySQL adalah *www.mysql.com,* di *website* inilah ada banyak hal yang bisa ditemukan seperti MySQL versi terbaru atau dokumentasi-dokumentasi serta program-program utility MySQL.

Selain itu MySQL juga memiliki beberapa fitur, antara lain :

1. *Portability*

   MySQL dapat berjalan stabil pada berbagai sistem operasi seperti *Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga,* dan masih banyak lagi.

2. *Open Source*

   MySQL didistribusikan secara *open source* (gratis), dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.

3. *Multiuser*

   MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

4. *Performance tuning*

   MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. *Column types*

   MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.

6. *Security*

   MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta *password* terenkripsi.

7. *Scalability dan limits*

   MySQL mampu menangani database dalam skala besar, dengan jumlah *records* lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

8. *Connectivity*

   MySQL dapat melakukan koneksi dengan *client* menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).

9. *Localisation*

   MySQL dapat mendeteksi pesan kesalahan pada *client* dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk didalamnya.

10. *Interface*

    MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

11. *Clients* dan *tools*

    MySQL dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi database, dan pada setiap *tool* yang ada disertakan petunjuk online.

12. Struktur tabel

    MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan database lainnya semacam *PostgreSQL* ataupun *Oracle*

3.3.1.9 Java

Bahasa pemrograman Java adalah bahasa pemrograman berorientasi objek yang mirip dengan bahasa C++ dan *Smalltalk.* Java bersifat netral, tidak bergantung pada suatu platform, dan mengikuti prinsip WORA (*Write Once and Run Anywhere).* Tidak seperti C atau C++, Anda dapat menulis program dalam bahasa Java, cukup sekali mengkompilasi dan dapat dijalankan pada berbagai sistem operasi, seperti *Windows, Linux, Solaris, MacOS*. Sebaliknya bahasa C++ bergantung pada suatu platform, karena mengharuskan kita mengkompilasi program tersebut pada setiap sistem operasi yang dituju.

Adapun keuntungan dari java adalah sebagai berikut :

a. Mulai dengan cepat : Java merupakan bahasa pemrograman berorientasi objek, mudah dipelajari, terutama untuk programmer yang sudah menguasai C atau C++

b. Menghindari kebergantungan pada platform tertentu : dapat menjalankan program pada banyak platform dengan tidak menggunakan *library* yang ditulis spesifik untuk platform tertentu.
c. Tulis sekali, jalankan di mana saja : Karena aplikasi yang ditulis dalam bahasa Java dikompilasi ke dalam kode *byte* yang bebas *platform*, aplikasi yang ditulis dapat jalan secara konsisten pada *platform* apa saja.

3.3.1.10 Netbeans

NetBeans mengacu pada dua hal, yakni platform untuk pengembangan aplikasi desktop java, dan sebuah *Integrated Development Environment* (IDE) yang dibangun menggunakan platform NetBeans. Platform NetBeans memungkinkan aplikasi dibangun dari sekumpulan komponen perangkat lunak moduler yang disebut 'modul'. Sebuah modul adalah suatu arsip Java (*Java archive*) yang memuat kelas-kelas Java untuk berinetraksi dengan *NetBeans Open API* dan file manifestasi yang mengidentifikasinya sebagai modul. Aplikasi yang dibangun dengan modul-modul dapat dikembangkan dengan menambahkan modul-modul baru. Karena modul dapat dikembangkan secara independen, aplikasi berbasis *platform* NetBeans dapat dengan mudah dikembangkan oleh pihak ketiga secara mudah dan *powerful*.

a. Platform NetBeans

Platform NetBeans adalah *framework* yang dapat digunakan kembali (*reusable*) untuk menyederhanakan pengembangan aplikasi desktop. Ketika aplikasi berbasis *platform* NetBeans dijalankan, kelas Main dari platform dieksekusi. Modul-modul yang tersedia ditempatkan di sebuah *registry* di dalam memori, dan tugas *startup* modul dijalankan. Secara umum, kode modul dimuatkan ke dalam memori hanya ketika ia diperlukan.

Aplikasi dapat menginstal modul secara dinamis. Aplikasi dapat memasukkan modul *Update Center* untuk mengijinkan pengguna aplikasi men-*download digitally-signed upgrade* dan fitur-fitur baru secara langsung ke dalam aplikasi yang berjalan. Penginstalan kembali sebuah upgrade atau rilis baru tidak memaksa pengguna untuk men-*download* keseluruhan aplikasi lagi.

Platform NetBeans menawarkan layanan-layanan yang umum bagi aplikasi desktop, mengijinkan pengembang untuk fokus ke logika yang spesifik terhadap aplikasi. Fitur-fitur yang disediakan oleh platform NetBeans:

- Manajemen antarmuka (misal: menu & toolbar)
- Manajemen pengaturan pengguna
- Manajemen penyimpanan (menyimpan dan membuka berbagai macam data)
- Manajemen jendela
- *Wizard framework* (mendukung dialog langkah demi langkah).

b. NetBeans IDE

NetBeans IDE adalah IDE open source yang ditulis sepenuhnya dengan bahasa pemrograman Java menggunakan platform NetBeans. NetBeans IDE mendukung pengembangan semua tipe aplikasi Java (J2SE, web, EJB, dan aplikasi *mobile*). Fitur lainnya adalah sistem proyek berbasis Ant, kontrol versi, dan *refactoring*.

Versi terbaru saat ini adalah NetBeans IDE 5.5.1 yang dirilis Mei 2007 mengembangkan fitur-fitur Java EE yang sudah ada (termasuk *Java Persistence support*, EJB-3 dan JAX-WS). Sementara paket tambahannya, NetBeans Enterprise Pack mendukung pengembangan aplikasi perusahaan Java EE 5, meliputi alat desain visual SOA, skema XML, *web service* dan pemodelan UML. NetBeans C/C++ Pack mendukung proyek C/C++.

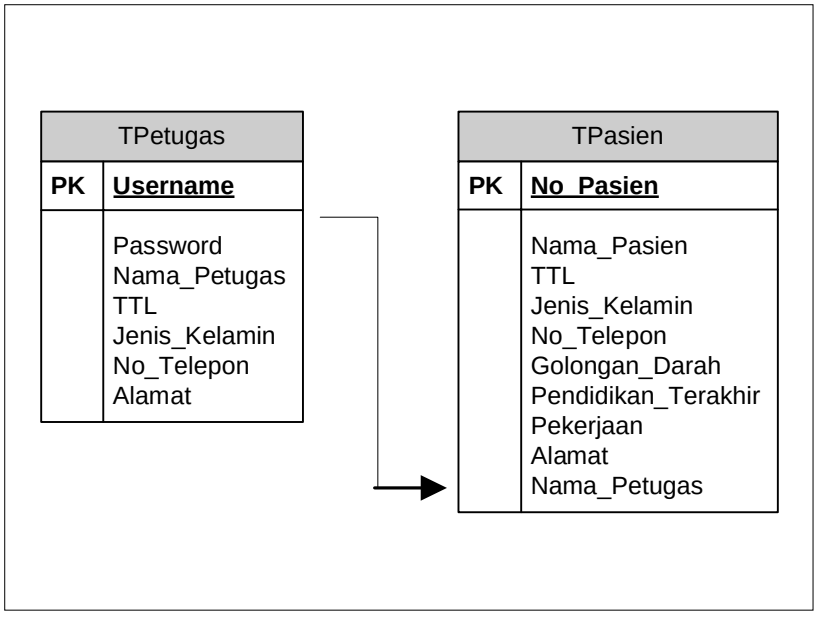
Modularitas: Semua fungsi IDE disediakan oleh modul-modul. Tiap modul menyediakan fungsi yang didefinisikan dengan baik, seperti dukungan untuk bahasa pemrograman Java, editing, atau dukungan bagi CVS. NetBeans memuat semua modul yang diperlukan dalam pengembangan Java dalam sekali *download*, memungkinkan pengguna untuk mulai bekerja sesegera mungkin. Modul-modul juga mengijinkan NetBeans untuk bisa dikembangkan. Fitur-fitur baru, seperti dukungan untuk bahasa pemrograman lain, dapat ditambahkan dengan menginstal modul tambahan. Sebagai contoh, Sun Studio, Sun Java Studio Enterprise, dan Sun Java Studio Creator dari Sun Microsystem semuanya berbasis NetBeans IDE.

3.3.2 Perancangan Database

Pemakaian *Database* dalam aplikasi registrasi pasien pada rumah sakit ini agar dalam pengoperasian dan implementasikannya dapat diperoleh suatu informasi yang lengkap dan mempermudah proses manipulasi data. Adapun perancangan basis data yang akan dibagun adalah sebagai berikut.

3.3.2.1 Skema Relasi

Skema relasi merupakan hubungan dua tabel atau lebih pada sistem. Tabel tersebut menampung atau berisi data-data yang ada pada system tersebut. Gambar dibawah ini menjelaskan skema relasi yang akan dibangun pada Aplikasi Registrasi Pasien pada Rumah Sakit.



Gambar 3.2 Skema Relasi Aplikasi Registrasi Pasien pada Rumah Sakit

3.3.2.2 Struktur File

Struktur file merupakan isi data yang berada dalam satu record pada satu tabel. Struktur file dalam Aplikasi Registrasi Pasien pada Rumah Sakit adalah :

Nama File : TPetugas

Media Penyimpanan : Harddisk

Keterangan : Digunakan untuk menyimpan data petugas

Tabel 3.1 Struktur File TPetugas

Nama Field	Type	Size	Keterangan
Username	Integer	6	Key
NIP	Varchar	12	-
Nama_pegawai	Varchar	30	-
TTL	Varchar	30	-
Jenis Kelamin	Varchar	6	-
No_Telepon	Integer	12	-
Alamat	Varchar	30	-
Password	Varchar	6	-

Nama File : TPasien

Media Penyimpanan : Harddisk

Keterangan : Digunakan untuk menyimpan data pasien

Tabel 3.2 Struktur File Tpasien

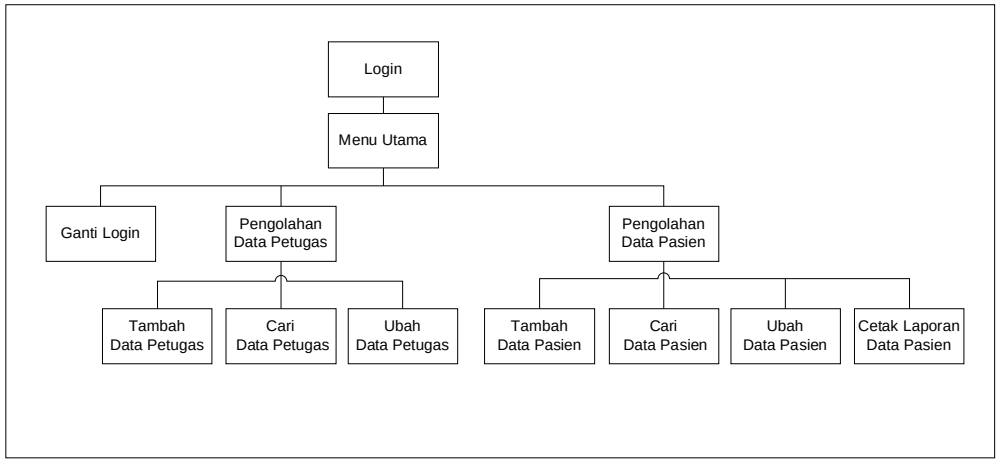
Nama Field	Type	Size	Keterangan
No_Pasien	Varchar	5	Key
Nama_Pasien	Char	30	-
TTL	Varchar	Shortdate	-
Jenis_Kelamin	Char	6	-
No_Telepon	Integer	12	-
Golongan_Darah	Char	2	-
Pendidikan_Terakhir	Varchar	20	-
Pekerjaan	Varchar	15	-
Alamat	Varchar	40	-

3.3.3 Perancangan Antarmuka

Perancangan antarmuka meliputi beberapa perancangan diantaranya meliputi perancangan struktur menu, spesifikasi antarmuka, perancangan pesan dan jaringan semantik.

3.3.3.1 Perancangan Struktur Menu

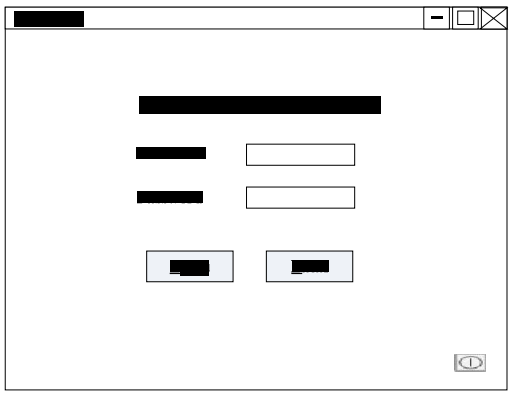
Dalam pembuatan suatu sistem informasi diperlukan suatu struktur menu yang berisikan mengenai menu dan submenu. Berikut ini struktur menu dari Aplikasi Registrasi Pasien pada Rumah Sakit.



Gambar 3.3 Struktur Menu Registrasi Pasien pada Rumah Sakit.

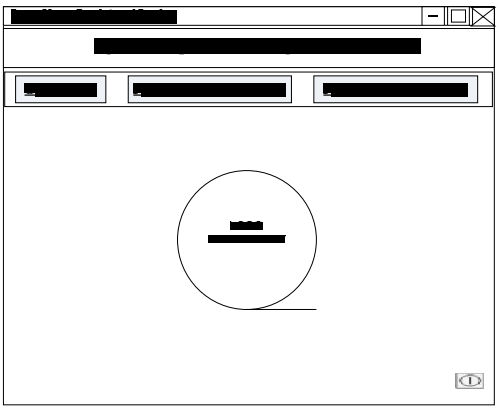
3.3.3.2 Spesifikasi Antarmuka

Spesifikasi antarmuka merupakan suatu bentuk tampilan dari program yang akan dibuat untuk kebutuhan *interface* dengan user**.** Berikut ini merupakan rancangan antarmuka dari Aplikasi Registrasi Pasien pada Rumah Sakit.:

T01	**Navigator**
	- Tampilan *login* ini akan meminta *Username* dan *Password* sebelum masuk ke Tampilan. - Klik Login (Alt+L), jika UserName ditemukan dan password cocok akan menuju T02 (Pengelolahan Data

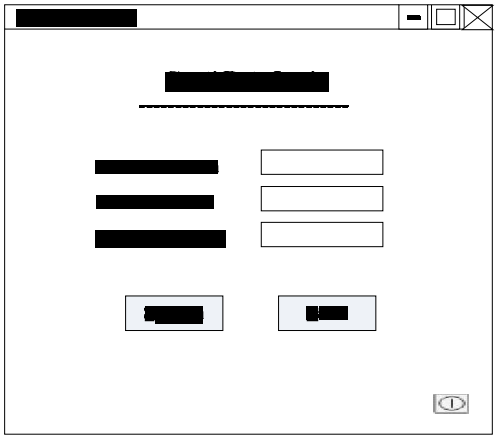
	Pasien) - Klik Batal (Alt+B), text kembali menjadi kosong. - Klik Gambar keluar dari tampilan login.
Keterangan : - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal. - Warna Backgraund Putih dengan gambar berwarna biru muda. - Ukuran Form 460x360 pixel.	

Gambar 3.4 Spesifikasi Antarmuka Login

T02	**Navigator**
	- Klik Ganti Login (Alt+G), menuju T03 - Klik Pengolahan Data Petugas (Alt+D) menuju ke T04 - Klik Pengolahan Data Pasien (Alt+P) menuju ke T05

	- Klik Gambar maka keluar dari aplikasi.
Keterangan - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal - Warna Backgraund Putih dengan gambar berwarna biru muda - Ukuran Form 1288x783 pixel.	

Gambar 3.5 Spesifikasi Antarmuka Menu Registrasi Pasien

T03	**Navigator**
	- Klik Simpan (Alt+S), untuk menyimpan data. - Klik Batal (Alt+B),text akan kembali kosong. - Klik Gambar kembali T02.
Keterangan: - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal - Warna Backgraund Putih dengan gambar berwarna biru muda	

- Ukuran Form 460x360 pixel.

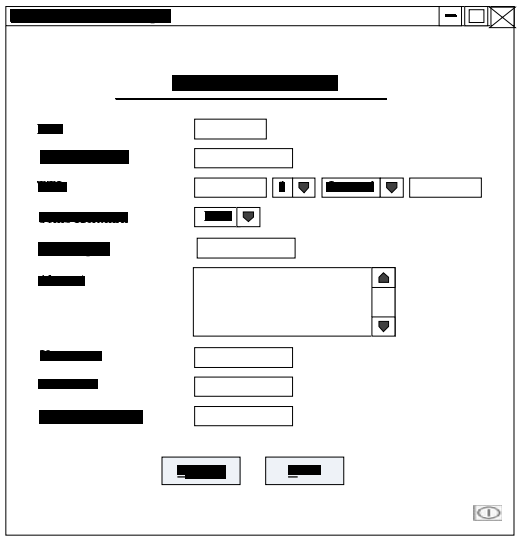
Gambar 3.6 Spesifikasi Antarmuka Ganti Login

T04	**Navigator**
	- Klik Cari Petugas maka akan muncul nama petugas yang di cari melalui table. - Klik Tambah Petugas (Alt+T) maka menuju T06 - Klik Ubah Petugas (Alt+U) maka menuju T07 - Klik Gambar kembali T02.
Keterangan - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal - Warna Backgraund Putih dengan gambar berwarna biru muda - Ukuran Form 1288x783 pixel.	

Gambar 3.7 Spesifikasi Antarmuka Pengolahan Data Petugas

T05	Navigator
	- Klik Cari Pasien maka akan muncul nama pasiens yang di cari melalui table. - Klik Tambah Pasien (Alt+T) menuju T08. - Klik Ubah Pasien (Alt+U) menuju T09. - Klik Cetak Laporan (Alt+C) menampilkan laporan data pasien. - Klik Gambar menuju T02
Keterangan - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal - Warna Backgraund Putih dengan gambar berwarna biru muda - Ukuran Form 1288x783 pixel.	

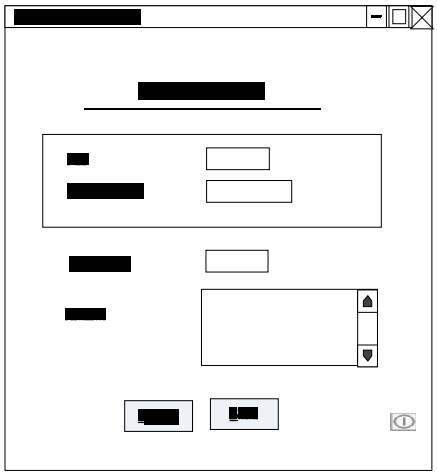
Gambar 3.8 Spesifikasi Antarmuka Pengolahan Data Pasien

T06	**Navigator**
	- Klik Simpan (Alt+S), untuk menyimpan data petugas. - Klik Batal (Alt+B), maka text akan kembali kosong. - Klik Gambar Kembali T02

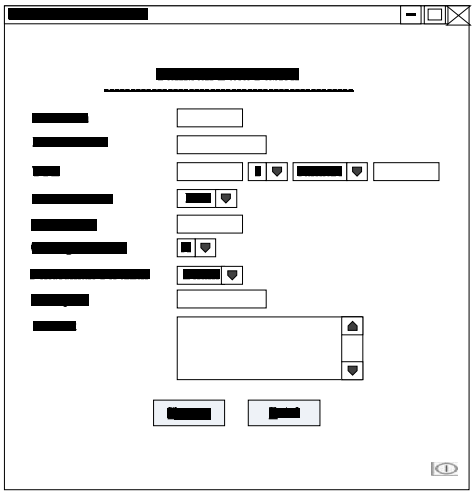
Keterangan

- Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal
- Warna Backgraund Putih dengan gambar berwarna biru muda
- Ukuran Form 395x405 pixel.

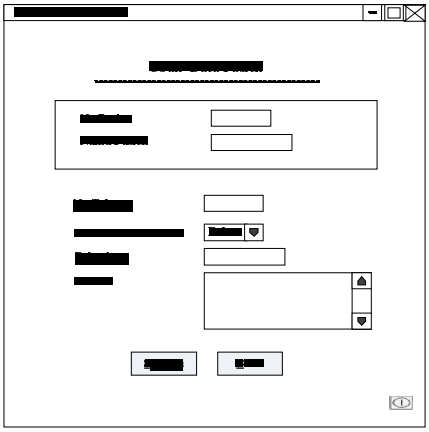
Gambar 3.9 Spesifikasi Antarmuka Tambah Data Petugas

T07	**Navigator**
	- Klik Simpan (Alt+S) untuk menyimpan data petugas - Klik Batal (Alt+B) text akan kembali kosong. - Klik Gambar Kembali T02
Keterangan - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal - Warna Backgraund Putih dengan gambar berwarna biru muda - Ukuran Form 395x405 pixel.	

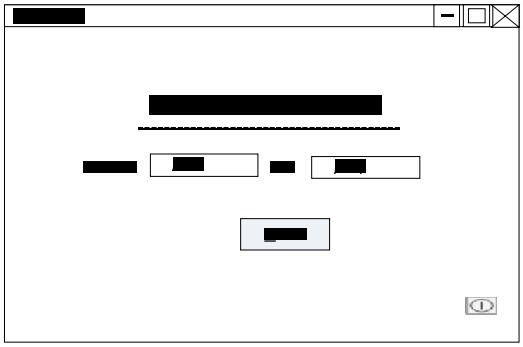
Gambar 3.10 Spesifikasi Antarmuka Ubah Data Petugas

T08	**Navigator**
	- Klik Simpan (Alt+S), untuk menyimpan data pasien. - Klik Batal (Alt+B), maka text akan kembali kosong. - Klik Gambar Kembali T02
Keterangan - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal - Warna Backgraund Putih dengan gambar berwarna biru muda - Ukuran Form 395x405 pixel.	

Gambar 3.11 Spesifikasi Antarmuka Tambah Data Pasien

T09	**Navigator**
	- Klik Simpan (Alt+S), untuk menyimpan data - Klik Batal (Alt+B),, text akan kembali kosong. - Klik Gambar Kembali T02
Keterangan - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal - Warna Backgraund Putih dengan gambar berwarna biru muda - Ukuran Form 395x405 pixel.	

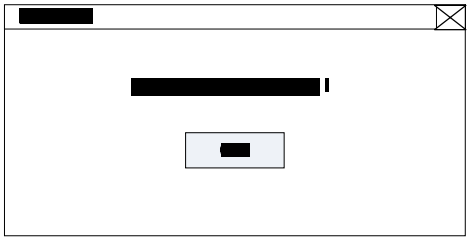
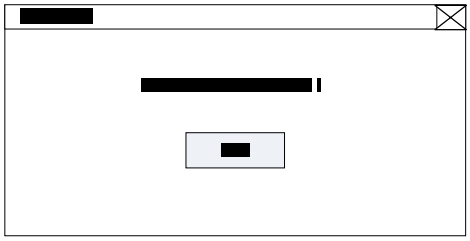
Gambar 3.12 Spesifikasi Antarmuka Ubah Data Pasien

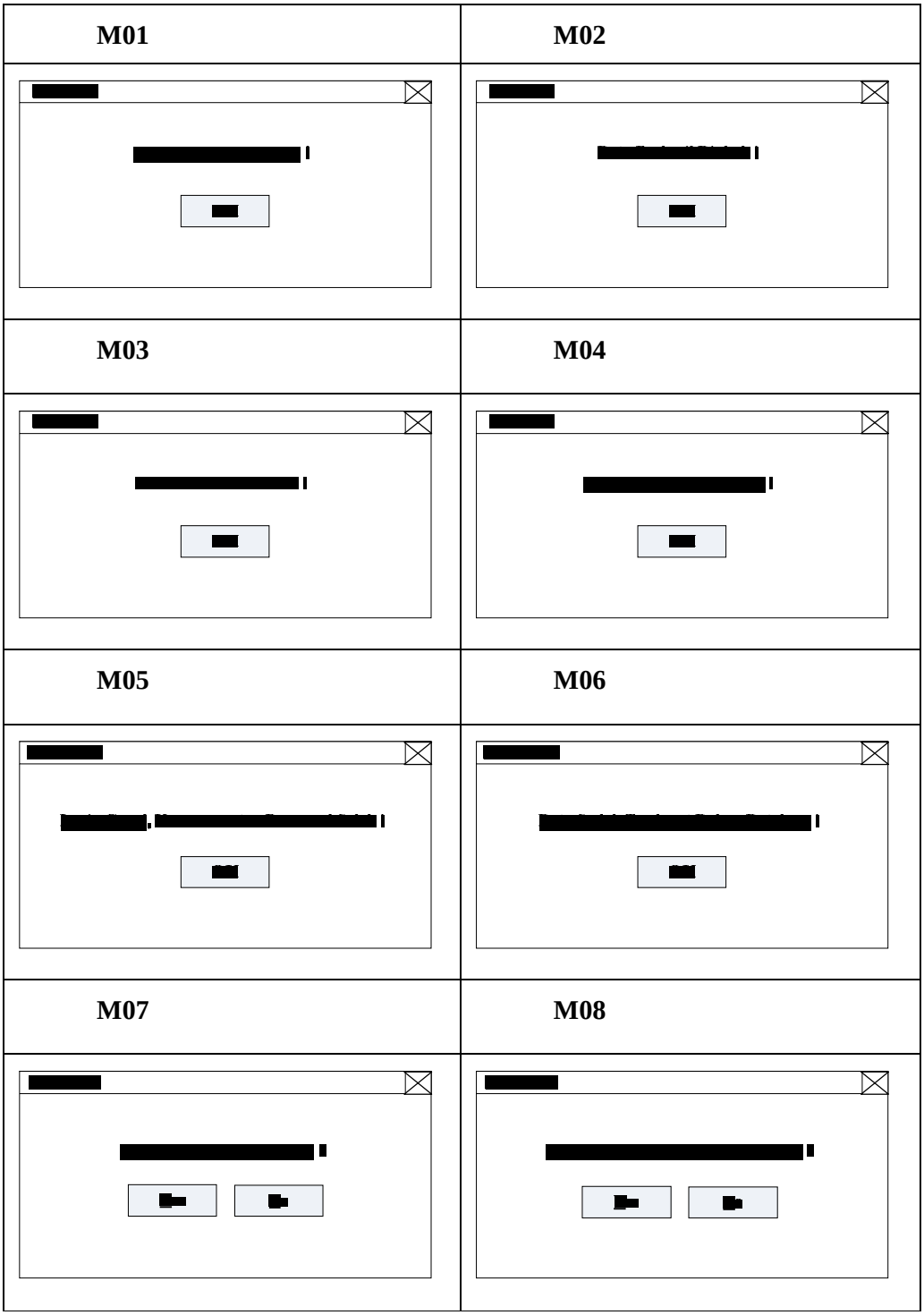
T10	Navigator
	- Klik Cetak (Alt+C), menampilkan laporan berdasarkan periode yang dipilih. - Klik Gambar menuju T02
Keterangan : - Font menggunakan Times New Roman dan Impact ukuran 10, 12, cetak tebal. - Warna Backgraund Putih dengan gambar berwarna biru muda. - Ukuran Form 460x360 pixel.	

Gambar 3.13 Spesifikasi Antarmuka Cetak Laporan

3.3.3.3 Perancangan Pesan

Perancangan pesan dari aplikasi registrasi pasien yang akan dibangun ini adalah sebagai berikut :

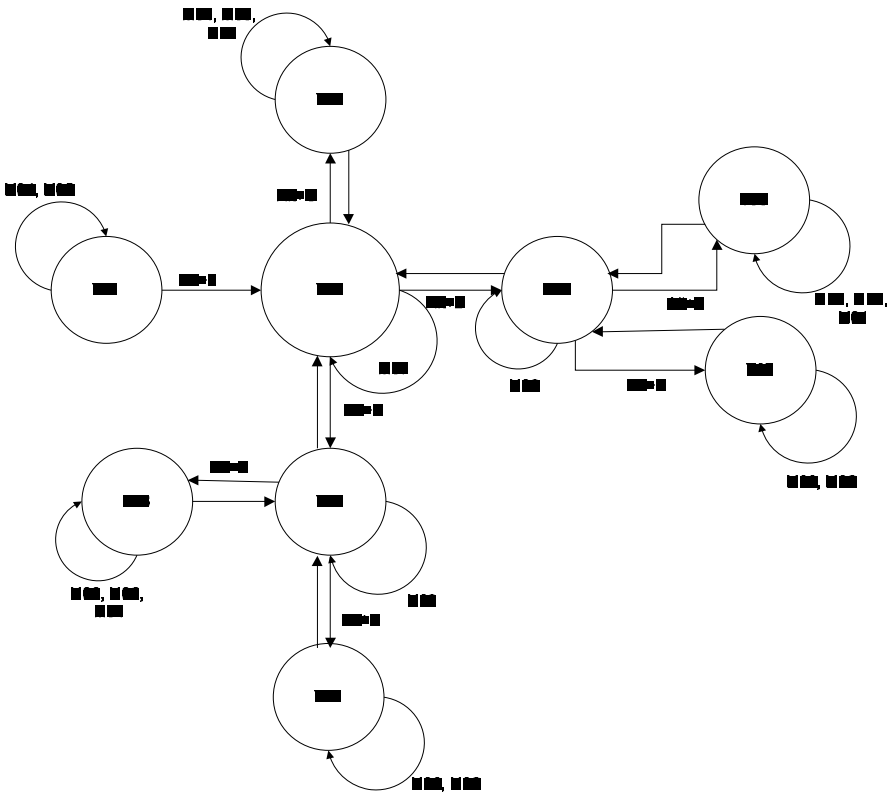
M01	M02
	



Gambar 3.14 Rancangan Pesan Aplikasi Registrasi Pasien

3.3.3.4 Jaringan Semantik

Jaringan semantik merupakan grafik yang terdiri dari simpul-simpul yang mempresentasikan objek pisik atau objek konsep dan busur-busur yang menunjukan relasi antar simpul-smpul tersebut. Digunakan untuk mempermudah bagi pemogram pada saat menulisakan program pada setiap lembar kerja. Pada jaringan simantik jaringan terdiri atas dua komponen : nomer tampilan (biasa diberi notasi dengan lingkaran), dan transaksi yang menyababkan perpindahan-perpindahan ke tampilan yang lain (biasa diberi notasi dengan anak panah). Jaringan semantic merupakan alat efektif untuk mempresentasikan pemetaan data yang bertujuan mencegah terjadinya duplikasi data.



Gambar 3.15 Jaringan Semantik Aplikasi Registrasi Pasien