BAB IV

ANALISIS DAN DESAIN
PROGRAM MEDIFIRST 2000 EDISI LABORATORIUM

4.1 Kebutuhan Perangkat Lunak

4.1.1 Deskripsi Umum Sistem

Sistem Informasi Manajemen Laboratorium Klinik adalah sistem informasi yang menangani pengelolaan data pasien yang meliputi pendaftaran pasien dan pemeriksaan laboratorium. Dimulai dari pasien melakukan pendaftaran hingga pasien mendapatkan kartu berobat. Selain itu juga menangani pengelolaan pemeriksaan pasien, pengelolaan data dokter, pengelolaan penyakit, pengelolaan hasil pemeriksaan laboratorium, pengelolaan hasil rontgen, pengelolaan transaksi pembayaran, menangani pembuatan laporan data pasien, laporan transaksi pembayaran, serta pencetakan kartu pasien

4.1.2 Fungsi Utama Perangkat Lunak

Produk perangkat lunak ini berfungsi untuk memudahkan dalam melayani pendaftaran pasien, pemeriksaan pasien, pengelolaan hasil pemeriksaan pasien, transaksi pembayaran, dan cetak kartu pasien.

Adapun fungsi-fungsi utama dari *Sistem Informasi Manajemen Laboratorium Klinik* adalah sebagai berikut :

- Mengelola data pasien yang meliputi proses pendaftaran, pemeriksaan laboratorium, history pasien dan cetak kartu berobat.
- Mengelola data user
- Mengolah data dokter
- Mengelola data penyakit
- Mengelola data hasil pemeriksaan laboratorium
- Mengelola data transaksi pembayaran
- Membuat laporan data pasien
- Membuat laporan transaksi pembayaran

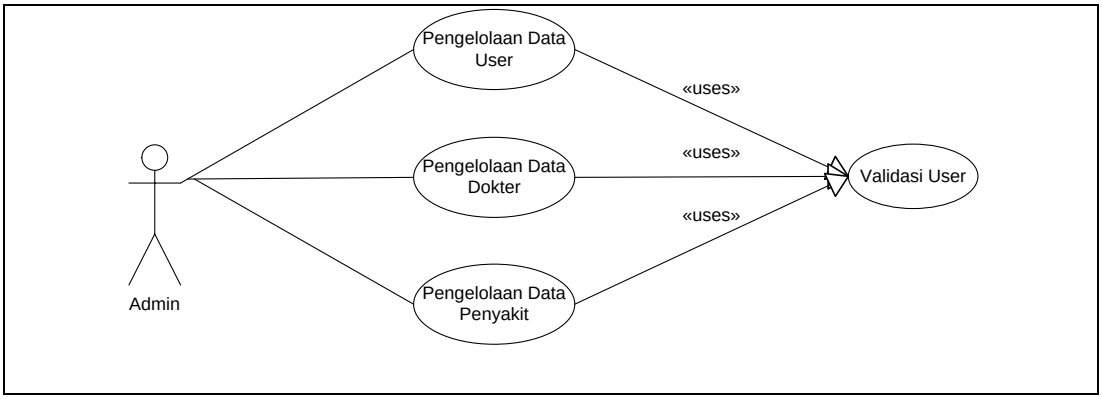
4.1.3 Model Use case

Pada bagian ini akan dijelaskan tentang definisi *Actor*, definisi *Use case* dan *Use case Diagram*.

4.1.3.1 Diagram Use case

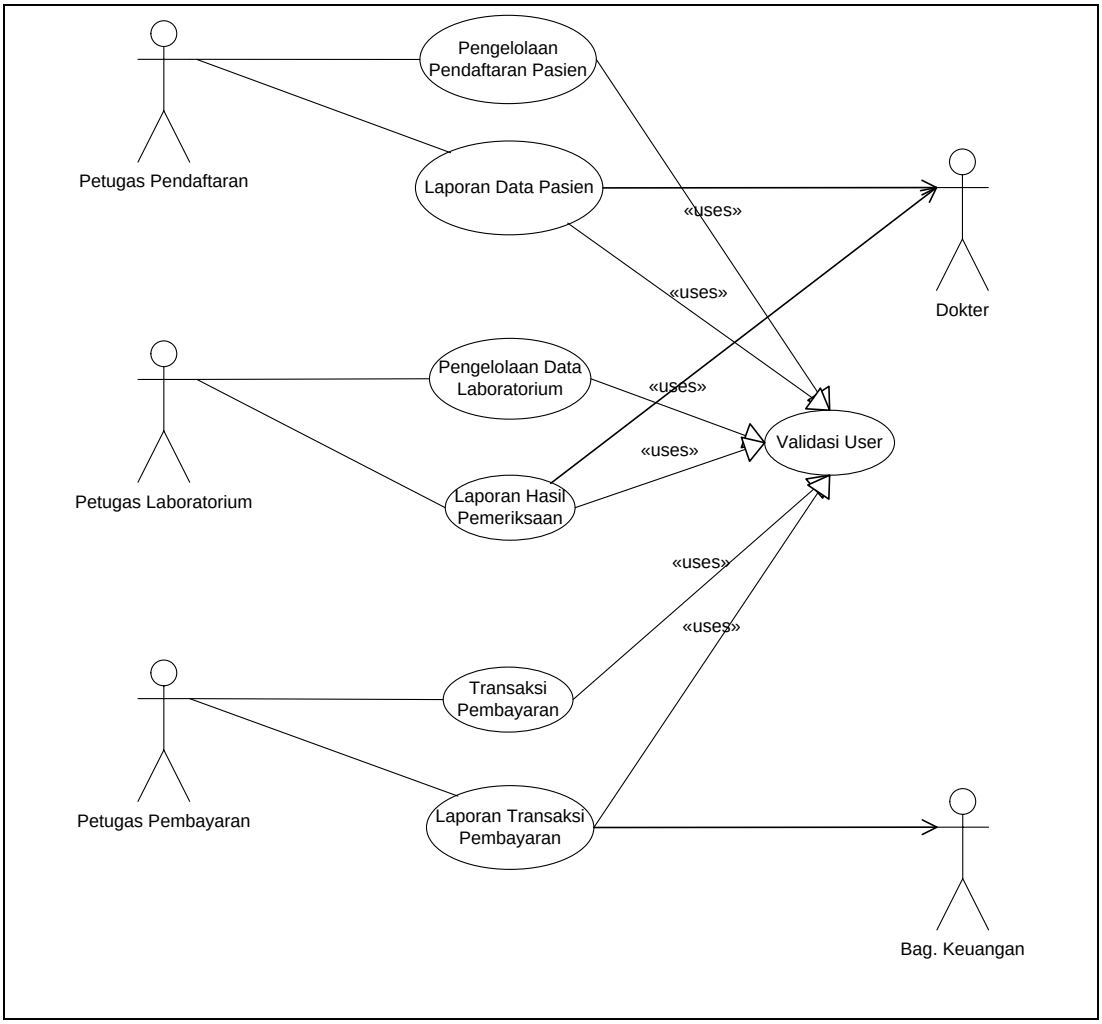
Use case Diagram menampilkan *Actor*, *Use case* dan hubungan antara keduanya. *Actor* mana yang menggunakan *Use case* mana dan *Use case* mana yang memasukkan *Use case* lain. *Use case Diagram* yang menggambarkan *sistem* dibagi ke dalam 2(dua) paket, diantaranya paket Sistem untuk Admin dan paket Sistem untuk Petugas.

a. Paket Sistem untuk Admin



Gambar 4.1 Paket Sistem untuk Admin

b. Paket Sistem untuk Petugas



Gambar 4.2 Paket Sistem untuk Petugas

4.1.3.1 Definisi Actor

Dalam *Use case Diagram*, ada beberapa actor yang terlibat dalam sistem. Diantaranya adalah Admin, Petugas Pendaftaran, Petugas Laboratorium, Petugas Pembayaran, Dokter dan Bag. Keuangan.

Tabel 4.1 Definisi Actor

No	Actor	Deskripsi
1.	Admin	Merupakan primary actor yang mempunyai hak untuk

		mengelola data user, data dokter, dan data penyakit
2.	Petugas Pendaftaran	Merupakan primary actor yang mempunyai hak untuk melayani pendaftaran pasien dan membuat laporan data pasien
3.	Petugas Laboratorium	Merupakan primary actor yang mempunyai hak untuk melayani pemeriksaan laboratorium, dan mencatat hasil pemeriksaannya
4.	Petugas Pembayaran	Merupakan primary actor yang mempunyai hak untuk melayani transaksi pembayaran dan membuat laporan transaksi pembayaran
5.	Dokter	Merupakan secondary actor yang menerima laporan data pasien dan laporan hasil pemeriksaan laboratorium
6.	Bag. Keuangan	Merupakan secondary actor yang menerima laporan transaksi pembayaran dari petugas pembayaran

4.1.3.2 Definisi *Use case*

Use case yang ada dalam diagram didefinisikan pada tabel dibawah ini **:**

Tabel 4.2 Definisi *Use case*

No	***Use case***	**Deskripsi**
1.	Validasi User	Proses validasi user dan pengaturan hak akses
2.	Pengelolaan Data User	Proses dimana admin menginputkan data user. Jika terjadi perubahan data user maka admin akan melakukan edit data
3.	Pengelolaan Data Dokter	Proses dimana admin menginputkan data dokter. Jika terjadi perubahan data dokter maka admin akan melakukan edit data
4.	Pengelolaan Data Penyakit	Proses dimana admin menginputkan data penyakit. Jika terjadi perubahan data penyakit maka admin akan melakukan edit data
5.	Pengelolaan Pendaftaran Pasien	Proses dimana petugas pendaftaran melayani pendaftaran pasien, mencetak kartu pasien, dan mengelola data pasien. Jika terjadi perubahan data

No	*Use case*	Deskripsi
		pasien maka petugas akan melakukan edit data
6.	Pengelolaan Data Laboratorium	Proses dimana petugas laboratorium mengelola data hasil laboratorium pasien
7.	Transaksi Pembayaran	Proses dimana petugas pembayaran membuat dan mencetak laporan berdasarkan data transaksi pembayaran
8.	Laporan Data Pasien	Proses dimana petugas pendaftaran membuat dan mencetak laporan berdasarkan data pasien
9.	Laporan Hasil Pemeriksaan	Proses dimana petugas laboratorium membuat dan mencetak laporan hasil pemeriksaan berdasarkan data pemeriksaan dari dokter
10.	Laporan Transaksi Pembayaran	Proses dimana petugas pembayaran membuat dan mencetak laporan berdasrkan data transaksi pembayaran

4.1.3.3 Skenario Use case

Menjelaskan skenario dari masing-masing *use case*.

Tabel 4.4 *Use case* Validasi User

No	:	SUC-01
Name	:	Validasi User
Actor	:	Admin, Petugas Pendaftaran, Dokter, Petugas Laboratorium, Petugas Pembayaran, dan Bag. Keuangan
Precondition	:	User belum melakukan login, user telah memiliki user_id dan password
Postcondition	:	User melakukan login dengan benar & Hak akses masing-masing user dapat ditampilkan
Main Flow of Event		

Actor Actions	Respon System
1. User menginputkan user_id dan password	2. Sistem mengecek validitas user_id dan password yang diinputkan
	3. Tampil info validitas user
4. User dapat menggunakan system sesuai	

hak aksesnya	
Exceptional Flow of Event	
1a. Jika username dan password salah maka sistem akan menampilkan pesan kesalahan 1b. Jika yang diinputkan username nya saja, maka sistem akan menampilkan pesan kesalahan 1c. Jika yang diinputkan password nya saja, maka sistem akan menampilkan pesan kesalahan 1d. Jika username dan password tidak diisi, maka sistem akan menampilkan pesan kesalahan	

Tabel 4.5 *Use case* Pengelolaan Data User

No	:	SUC-02
Name	:	Pengelolaan Data User
Actor	:	Admin
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Data user telah disimpan dan dapat ditampilkan
Main Flow of Event		
Actor Actions		**Respon System**
1. Admin menerima data user 2. Admin menginputkan data user 4. Admin mencari data user berdasarka Id user 6. Admin mengedit data user 8. Admin melakukan penghapusan data user		3. Sistem menampilkan info data user telah ditambahkan 5. Sistem akan menampilkan info pencarian data user 7. Sistem akan menampilkan info data user telah diedit 9. Sistem akan menampilkan info data user telah dihapus
Exceptional Flow of Event		
2a. Jika data user tidak lengkap, maka sistem akan menampilkan pesan ”data user tidak lengkap” 4a. Jika data user yang dicari tidak terdapat di database, maka system akan menampilkan pesan “data user dengan ID tersebut tidak ada di database”		

Tabel 4.6 *Use case* Pengelolaan Data Dokter

No	:	SUC-03
Name	:	Pengelolaan Data Dokter
Actor	:	Admin
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Data dokter telah disimpan dan dapat ditampilkan

Main Flow of Event

Actor Actions	**Respon System**
1. Admin mengumpulkan semua data mengenai dokter 2. Admin menginputkan data dokter 4. Admin mencari data dokter berdasarka Id dokter 6. Admin melakukan pengeditan data dokter 8. Admin melakukan penghapusan data dokter	3. Sistem menampilkan info data dokter telah ditambahkan 5. Sistem akan menampilkan info pencarian data dokter 7. Sistem akan menampilkan info data dokter telah diedit 9. Sistem akan menampilkan info data dokter telah dihapus

Exceptional Flow of Event

2a. Jika data dokter tidak lengkap, maka sistem akan menampilkan pesan ”data dokter tidak lengkap”

4a. Jika data dokter yang dicari tidak terdapat di database, maka system akan menampilkan pesan “data dokter dengan ID tersebut tidak ada di database”

Tabel 4.7 *Use case* Pengelolaan Data Penyakit

No	:	SUC-04
Name	:	Pengelolaan Data Penyakit

Actor	:	Admin
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Data penyakit telah disimpan dan dapat ditampilkan

Main Flow of Event

Actor Actions	**Respon System**
1. Admin mengumpulkan semua data mengenai penyakit 2. Admin menginputkan data penyakit 4. Admin mencari data dokter berdasarka Id penyakit 6. Admin melakukan pengeditan data penyakit 8. Admin melakukan penghapusan data penyakit	3. Sistem menampilkan info data penyakit telah ditambahkan 5. Sistem akan menampilkan info pencarian data penyakit 7. Sistem akan menampilkan info data penyakit telah diedit 9. Sistem akan menampilkan info data penyakit telah dihapus

Exceptional Flow of Event

2a. Jika data penyakit tidak lengkap, maka sistem akan menampilkan pesan ”data penyakit tidak lengkap”

4a. Jika data penyakit yang dicari tidak terdapat di database, maka system akan menampilkan pesan “data penyakit dengan ID tersebut tidak ada di database”

Tabel 4.8 *Use case* Pengelolaan Pendaftaran Pasien

No	:	SUC-05
Name	:	Pengelolaan Pendaftaran Pasien
Actor	:	Petugas Pendaftaran
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Data pasien disimpan dan dapat ditampilkan

Main Flow of Event

Actor Actions	**Respon System**
1. Petugas menerima data pasien baru 2. Petugas menginputkan data pasien baru 5. Petugas memberikan kartu pasien	3. Sistem menampilkan info data pasien telah ditambahkan 4. Cetak kartu pasien

Exceptional Flow of Event
2a. Jika data pasien tidak lengkap, maka sistem akan menampilkan pesan “data pasien tidak lengkap

Tabel 4.9 *Use case* Pengelolaan Data Laboratorium

No	:	SUC-06
Name	:	Pengelolaan Data Laboratorium
Actor	:	Petugas Laboratorium
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Data laboratorium telah disimpan dan dapat ditampilkan

Main Flow of Event	
Actor Actions	**Respon System**
1. Petugas lababoratorium menerima data hasil pemeriksaan pasien 2. Petugas menginputkan data hasil laboratorium pasien	3. Sistem menampilkan info data laboratorium telah disimpan 4. Cetak hasil laboratorium

Exceptional Flow of Event
2a. Jika data laboratorium tidak lengkap, maka sistem akan menampilkan pesan “data laboratorium tidak lengkap

Tabel 4.10 *Use case* Transaksi Pembayaran

No	:	SUC-07
Name	:	Transaksi Pembayaran
Actor	:	Petugas Pembayaran
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Data transaksi dicatat dan disimpan

Main Flow of Event	
Actor Actions	**Respon System**

1. Petugas pembayaran menerima data laboratorium yang harus dibayar pasien 2. Petugas pembayaran menginputkan data pembayaran laboratorium yang harus dibayar pasien 3. Petugas pembayaran menjumlahkan total biaya 5. Petugas pembayaran mencatat data pembayaran laboratorium	4. Sistem menampilkan seluruh info total biaya yang harus dibayar pasien 6. Sistem akan mencetak bukti pembayaran laboratorium
Exceptional Flow of Event	
2a. Jika data yang harus dibayar pasien tidak lengkap, maka sistem akan menampilkan pesan "data laboratorium yang harus dibayar pasien tidak lengkap"	

Tabel 4.11 *Use case* Laporan Data Pasien

No	:	SUC-08
Name	:	Laporan Data Pasien
Actor	:	Petugas Pendaftaran
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Laporan data pasien disimpan dan dicetak

Main Flow of Event	
Actor Actions	**Respon System**
1. Petugas pendaftaran mengambil data pasien 2. Petugas pendaftaran merekapitulasi data pasien sehingga menjadi laporan data pasien 4. Petugas pendaftaran menyimpan data laporan 6. Petugas pendaftaran menginputkan kategori	3. Sistem akan merekapitulasi data pasien 5. Sistem akan menyimpan data laporan 7. Sistem akan mencetak laporan data pasien sesuai dengan kategori
Exceptional Flow of Event	
6a. Jika data laporan yang diminta belum ada, maka sistem akan menampilkan pesan "Data yang dicari belum ada"	

Tabel 4.12 *Use case* Laporan Hasil Pemeriksaan

No	:	SUC-09
Name	:	Laporan Hasil Pemeriksaan
Actor	:	Petugas Laboratorium
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Laporan hasil pemeriksaan disimpan dan dicetak

Main Flow of Event

Actor Actions	**Respon System**
1. Petugas laboratorium mengambil data hasil pemeriksaan pasien 2. Petugas laboratorium merekapitulasi data hasil pemeriksaan menjadi laporan hasil pemeriksaan pasien 4. Petugas laboratorium menyimpan data hasil pemeriksaan 6. Petugas laboratorium menginputkan kategori	3. Sistem akan merekapitulasi data hasil pemeriksaan 5. Sistem akan menyimpan data hasil pemeriksaan pasien 7. Sistem akan mencetak laporan hasil pemeriksaan sesuai dengan kategori

Exceptional Flow of Event

6a. Jika data laporan yang diminta belum ada, maka sistem akan menampilkan pesan

"Data yang dicari belum ada"

Tabel 4.13 *Use case* Laporan Transaksi Pembayaran

No	:	SUC-10
Name	:	Laporan Transaksi Pembayaran
Actor	:	Petugas Pembayaran
Precondition	:	User_id dan password sudah teridentifikasi
Postcondition	:	Laporan transaksi pembayaran disimpan dan dicetak

Main Flow of Event

Actor Actions	**Respon System**
1. Petugas pembayaran mengambil data bukti pembayaran 2. Petugas pembayaran merekapitulasi bukti pembayaran sehingga menjadi data	3. Sistem akan merekapitulasi data transaksi pembayaran 5. Sistem akan menyimpan data laporan transaksi pembayaran

laporan transaksi pembayaran 4. Petugas pembayaran menyimpan data laporan transaksi pembayaran 6. Petugas pembayaran menginputkan kategori	7. Sistem akan mencetak laporan transaksi pembayaran berdasarkan kategori
Exceptional Flow of Event	
6a. Jika data laporan yang diminta belum ada, maka sistem akan menampilkan pesan "Data yang dicari belum ada"	

4.1.4 Spesifikasi Tambahan

Adapun kebutuhan non fungsional diantaranya :

Tabel 4.14 Spesifikasi Tambahan

No	Keterangan	Kapasitas
1	Mampu menangani *request*	
2	Mampu menangani penyimpanan data	
3	Aspek *Security*	
4	Mudah digunakan	
5	*User Friendly*	
6	*Distributed System*	

4.1.5 Glossary

Daftar istilah yang digunakan adalah sebagai berikut:

Tabel 4.15 Glossary

Istilah	Definisi
Actor	Elemen luar sistem yang berkepentingan secara langsung dengan sistem.
Constraint	Kondisi sematik atau batasan elemen.
Class	Deskripsi dari kumpulan objek yang memiliki atribut, methode, operasi, relasi dan sematik yang sama.

Extend	Menerangkan bagaimana perilaku dari extention usecase dapat disisipkan kedalam perilaku base usecase.
Generalisasi	Hubungan antar elemen yang lebih umum dengan elemen yang lebih khusus.
Postcondition	Batasan / Kondisi yang harus dilakukan sebelum melakukan proses.
Precondition	Batasan / Kondisi yang harus dilakukan setelah melakukan proses.
System Sequence Diagram	Penggambaran perilaku pada sebuah skenario
UML	Merupakan bahasa pemodelan secara grafik untuk mendokumentasikan seluruh artifak sistem perangkat lunak.
Usecase	Deskripsi fungsi dari sebuah system dari persfektif pengguna

4.2 Model Analisis

4.2.1 Realisasi *Use Case* Tahap Analisis

Realisasi use case tahap analisis terdiri dari Domain Model dan Sistem Sequence Diagram yang menggambarkan interaksi setiap objek dari kelas analisis yang terlibat di dalam use case tersebut.

4.2.1.1 System Sequence Diagram (SSD)

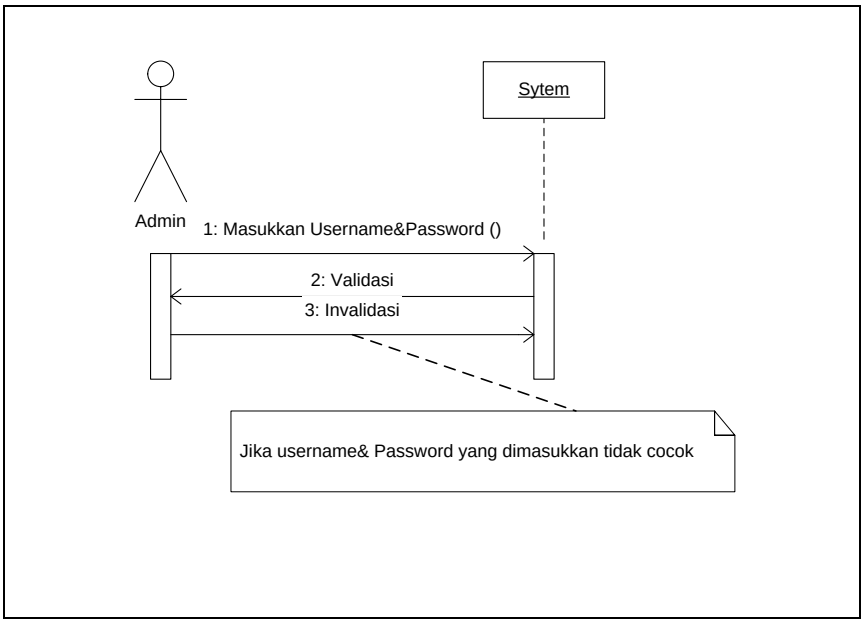
System Sequence Diagram terbagi ke dalam 2 paket, yaitu paket sistem untuk admin, dan paket sistem untuk petugas. Terdapat pula sequence diagram yang berada di setiap paket, yaitu *System Sequence Diagram* validasi user.

- Validasi user

a. User meng-entry username dan password

b. Jika username dan password yang dientrykan benar maka masuk ke aplikasi

c. Jika username dan password yang dientrykan salah maka kembali ke proses validasi



Gambar 4.3 SSD validasi user

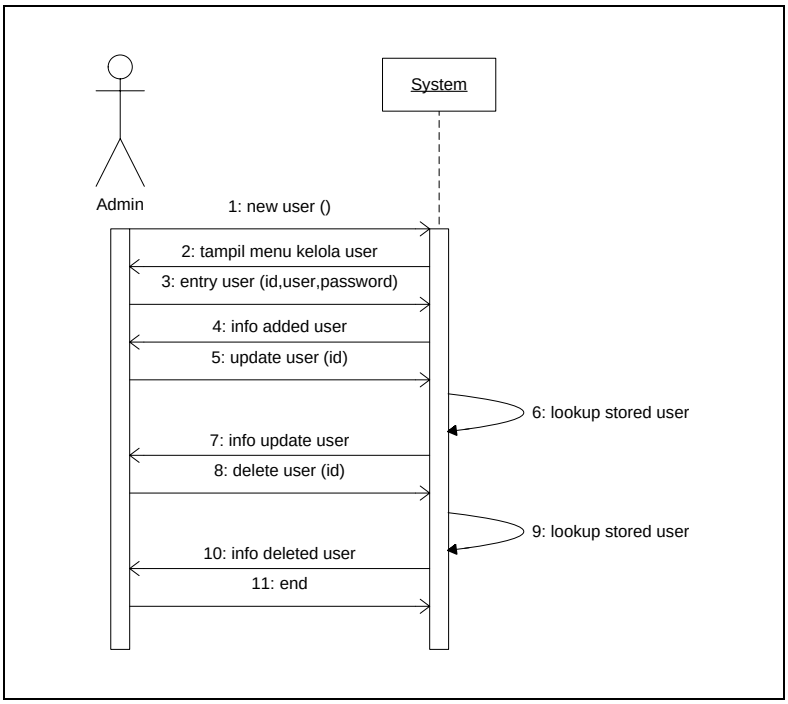
- ***System Sequence Diagram* Paket Sistem untuk Admin**

System Sequence Diagram yang terdapat dalam paket sistem untuk admin terdiri dari 3 macam, yaitu SSD Pengelolaan Data User, Pengelolaan Data Dokter, dan Pengelolaan Data Penyakit.

Event yang dilakukan setiap *actor* eksternal kepada sistem dalam setiap *use case* digambarkan dengan *System Sequence Diagram* dibawah ini:

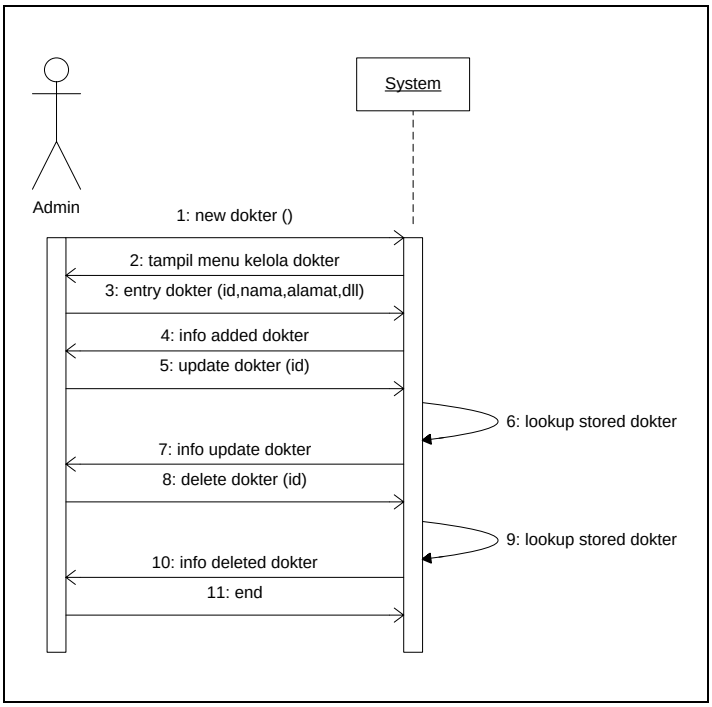
***System Sequence Diagram* Pengelolaan Data User**

(1) Pengelolaan data user



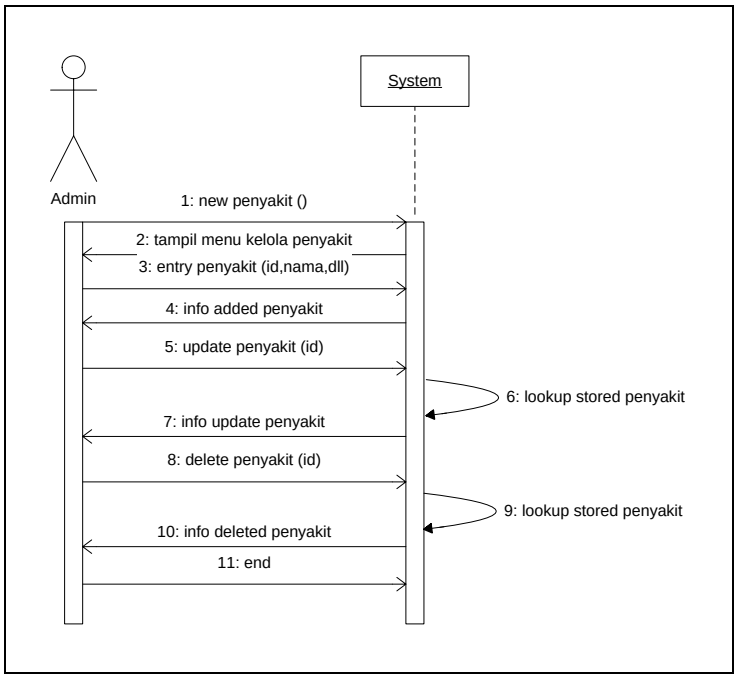
Gambar 4.4 SSD Pengelolaan Data User

(2) Pengelolaan data dokter



Gambar 4.5 SSD Pengelolaan Data Dokter

(3) Pengelolaan data penyakit



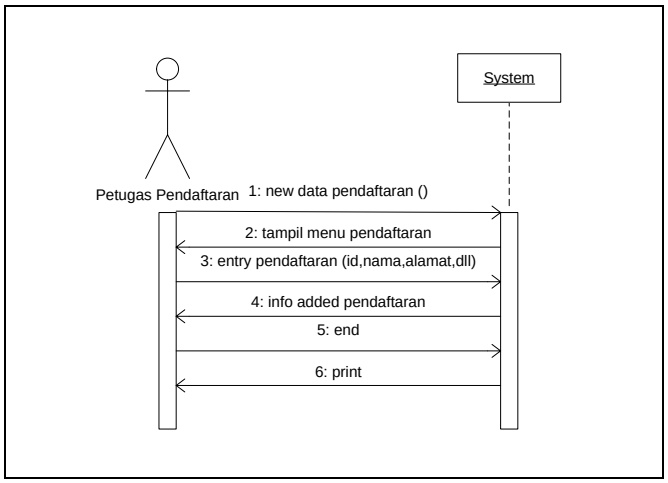
Gambar 4.6 SSD Pengelolaan Data Penyakit

- ***System Sequence Diagram* Paket Sistem untuk Petugas**

System Sequence Diagram yang terdapat dalam paket kajur terdiri dari 3 macam, yaitu system sequence diagram Pengelolaan Pendaftaran Pasien, Pengelolaan Data Laboratorium, Transaksi Pembayaran, Laporan Data Pasien, Laporan Hasil Pemeriksaan, dan Laporan Transaksi Pembayaran.

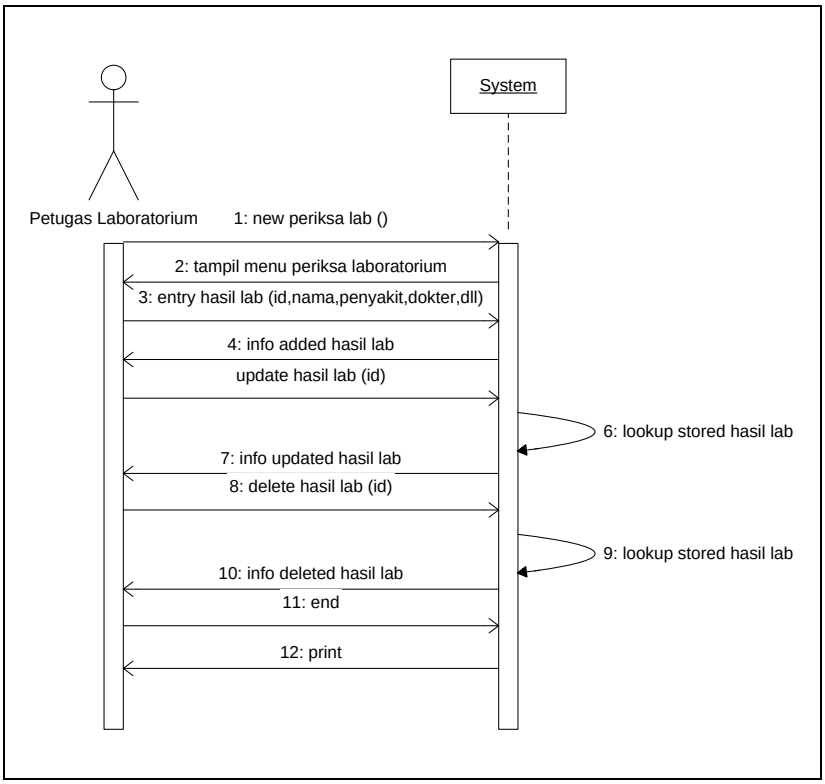
***System Sequence Diagram* Pengelolaan Pendaftaran Pasien**

(1) Pengelolaan Pendaftaran Pasien



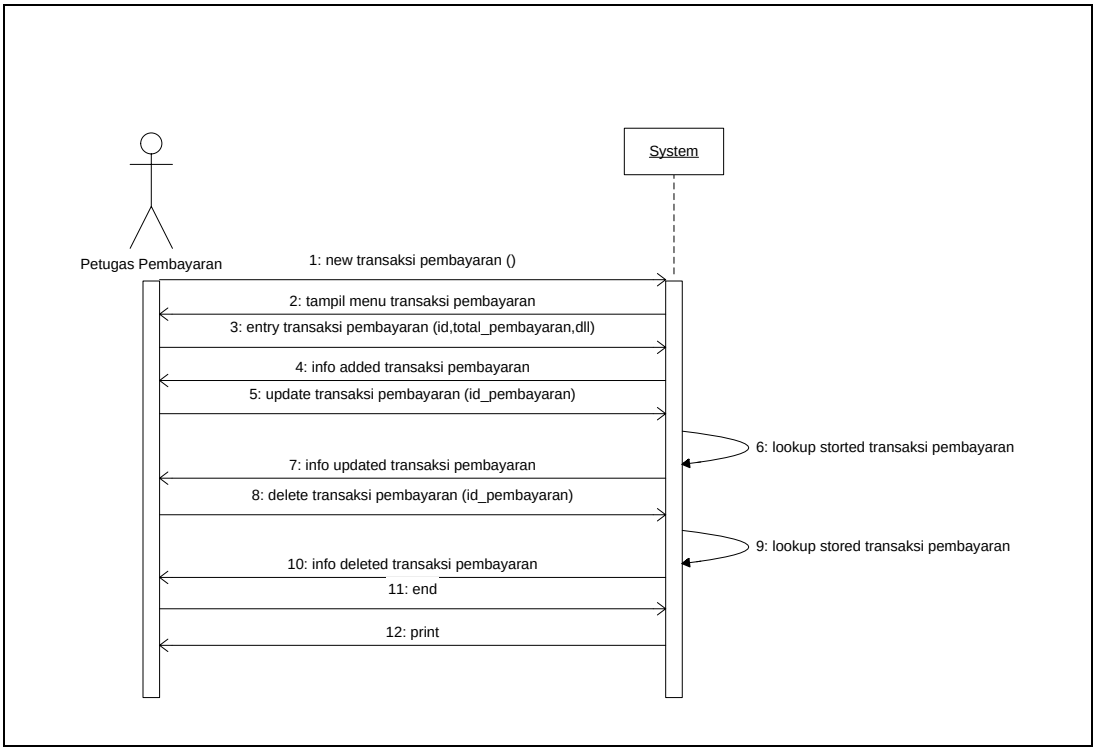
Gambar 4.7 SSD Pengelolaan Pendaftaran Pasien

(2) Pengelolaan Data Laboratorium



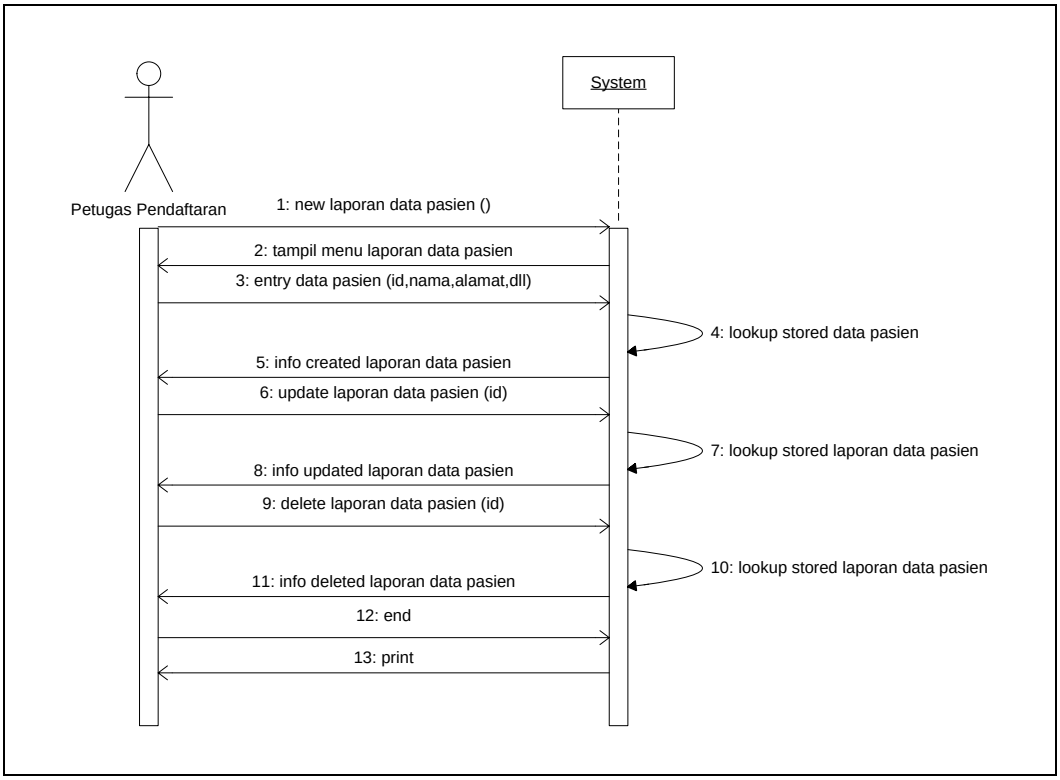
Gambar 4.8 SSD Pengelolaan Data Laboratorium

(3) Transaksi Pembayaran



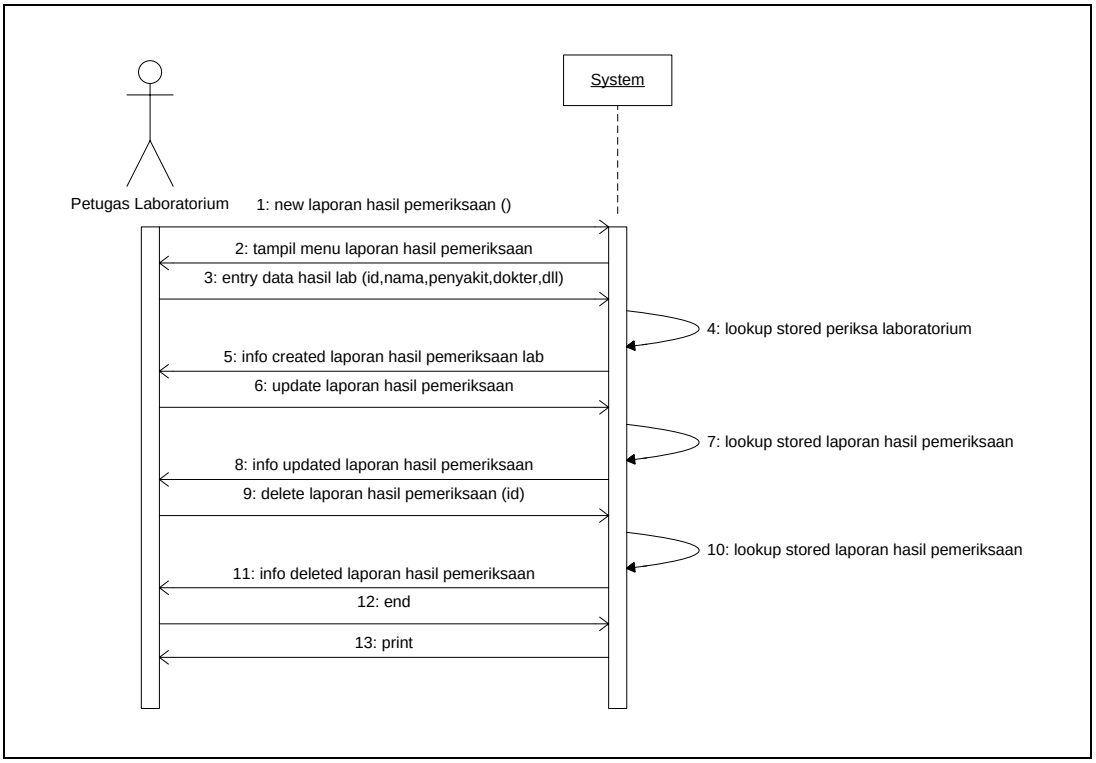
Gambar 4.9 SSD Transaksi Pembayaran

(4) Laporan Data Pasien



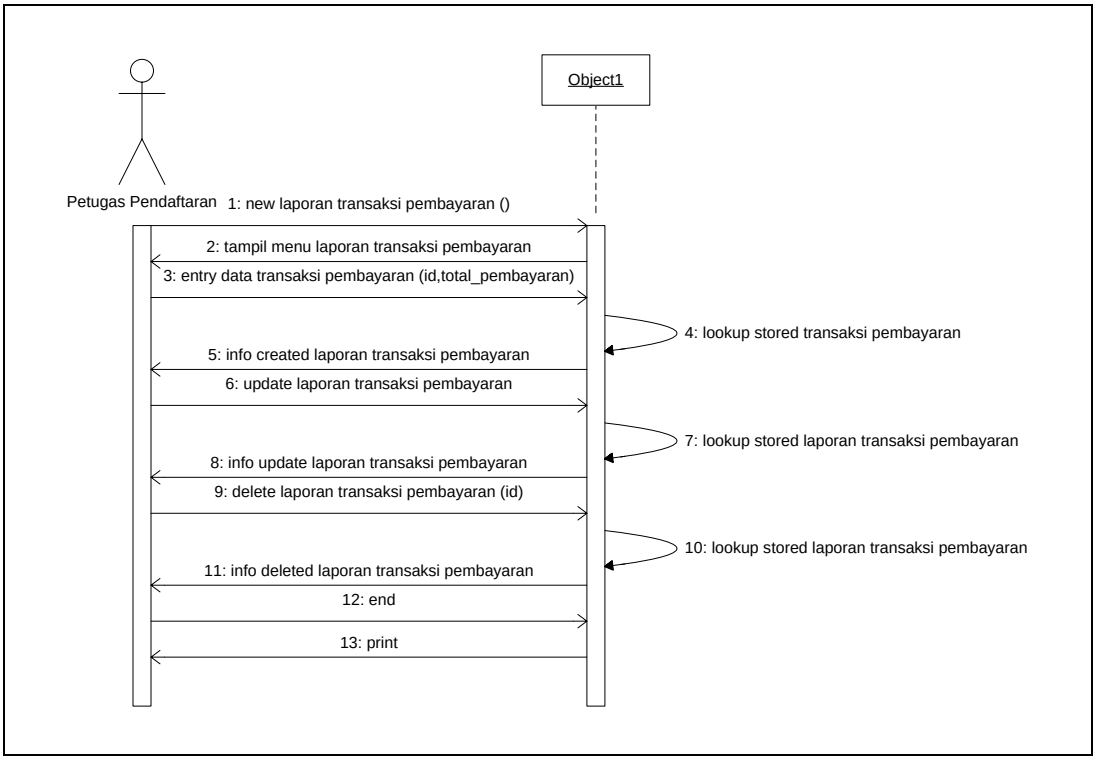
Gambar 4.10 SSD Laporan Data Pasien

(5) Laporan Hasil Pemeriksaan



Gambar 4.11 SSD Laporan Hasil Pemeriksaan

(6) Laporan Transaksi Pembayaran

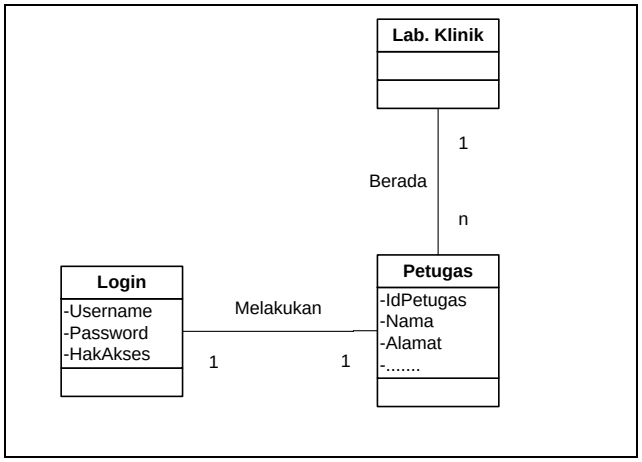


Gambar 4.12 SSD Laporan Transaksi Pembayaran

4.2.1.2 Domain Model

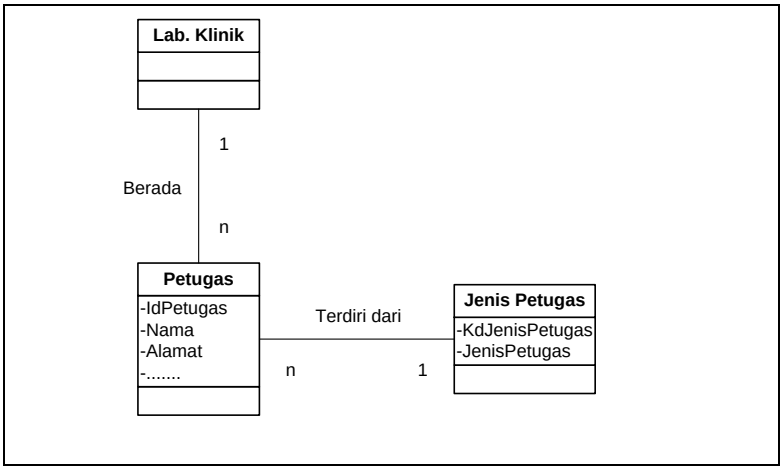
Domain model merupakan penggambaran keterkaitan objek-objek yang ada dalam lingkungan system informasi yang dikembangkan. Berikut adalah domain model dari Sistem Informasi Manajemen Laboratorium Klinik.

Domain Model Validasi User



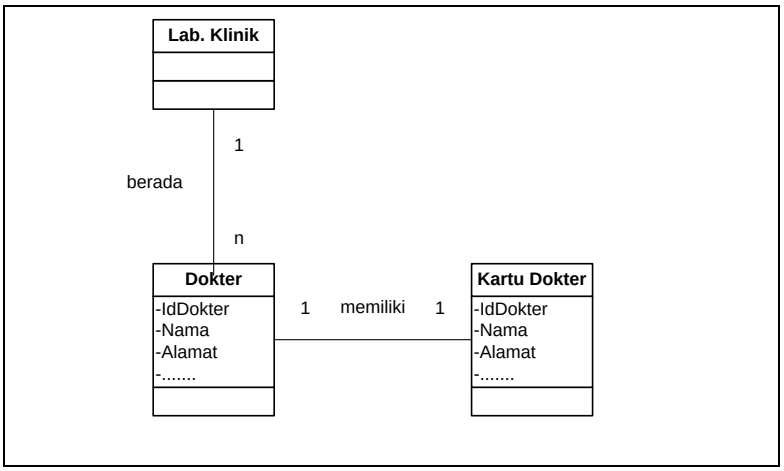
Gambar 4.13Domain Model Validasi User

Domain Model Pengelolaan Data User



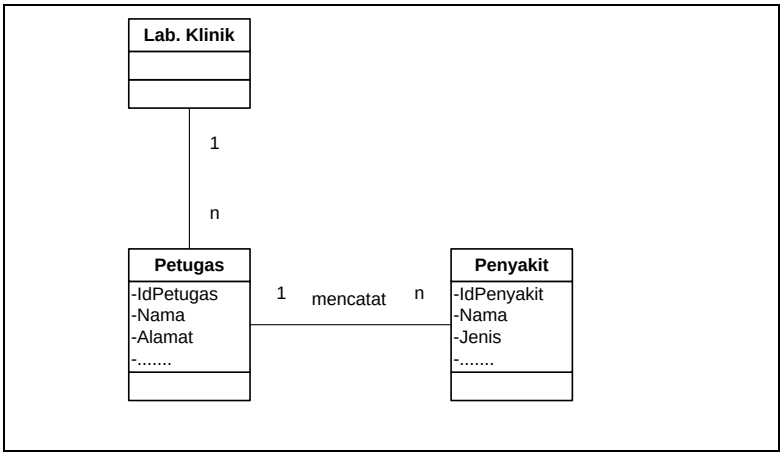
Gambar 4.14 Domain Model Pengelolaan Data User

Domain Model Pengelolaan Data Dokter



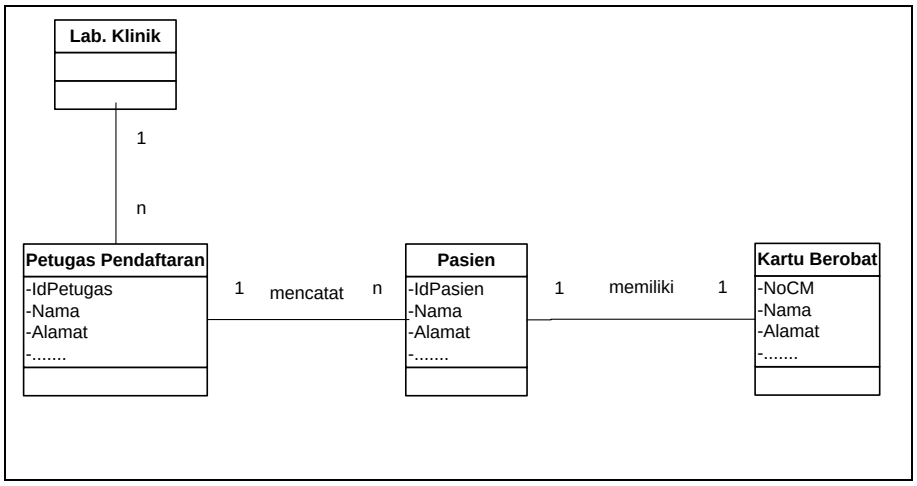
Gambar 4.15 Domain Model Pengelolaan Data Dokter

Domain Model Pengelolaan Data Penyakit



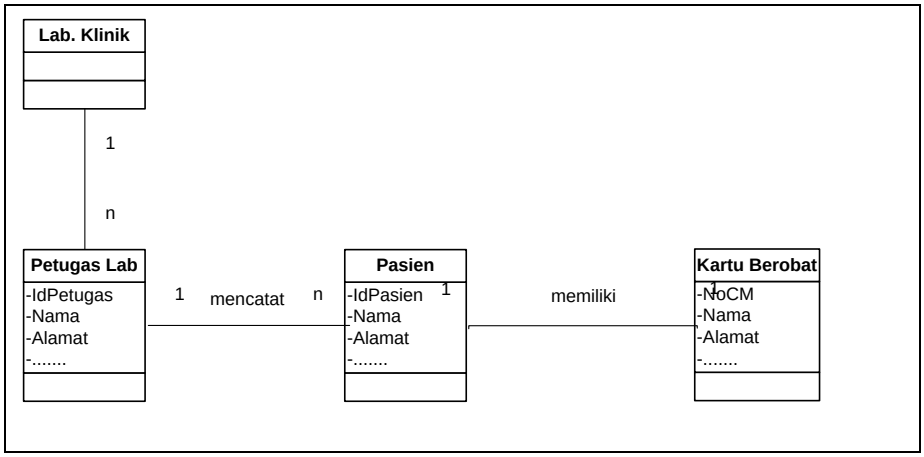
Gambar 4.16 Domain Model Pengelolaan Data Penyakit

Domain Model Pengelolaan Pendaftaran Pasien



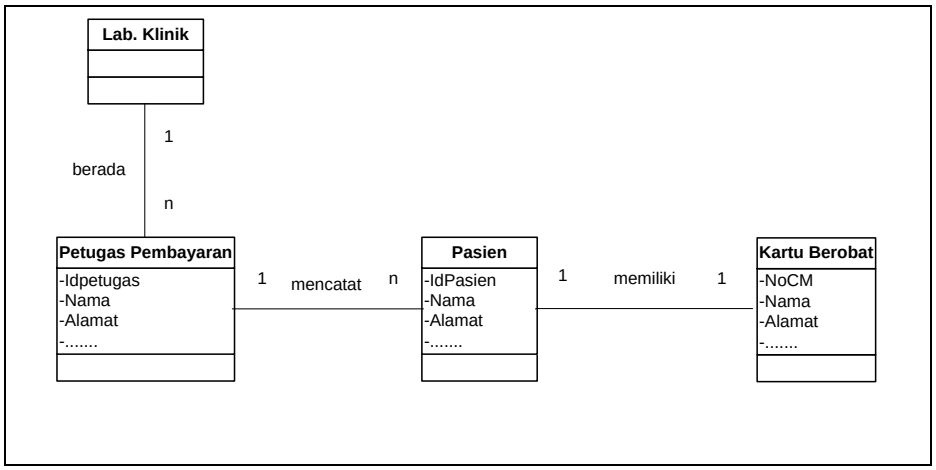
Gambar 4.17 Domain Model Pengelolaan Pendaftaran Pasien

Domain Model Pengelolaan Data Laboratorium



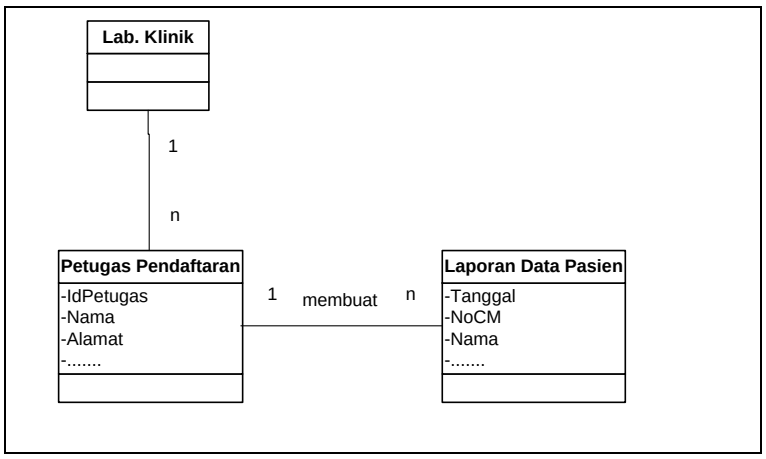
Gambar 4.18 Domain Model Pengelolaan Data Laboratorium

Domain Model Transaksi Pembayaran



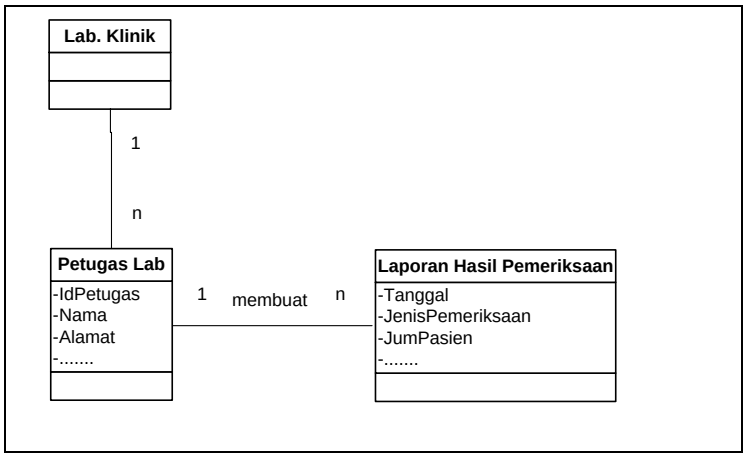
Gambar 4.19 Model Transaksi Pembayaran

Domain Model Lapora Data Pasien



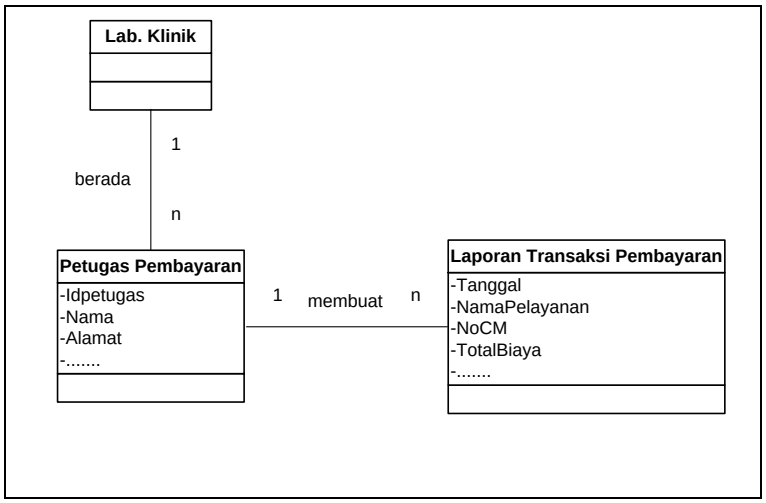
Gambar 4.20 Domain Model Laporan Data Pasien

Domain Model Laporan Hasil Pemeriksaan



Gambar 4.21 Domain Model Laporan Hasil Pemeriksaan

Domain Model Laporan Transaksi Pembayaran



Gambar 4.22 Domain Model Laporan Transaksi Pembayaran

4.2.2 Kelas Analisis

Kelas – kelas yang terbentuk dari tahap analisis, dan akan digunakan pada tahap perancangan, adalah sebagai berikut :

Tabel 4.16 Kelas Analisis

No	Nama Kelas	Jenis
1.	User	Control
2.	Pasien	Entity
3.	Dokter	Entity
4.	HasilLab	Entity
5.	Transaksi	Entity

4.2.2.1 Tanggung Jawab dan Atribut

Tabel 4.17 Tanggung Jawab dan Atribut

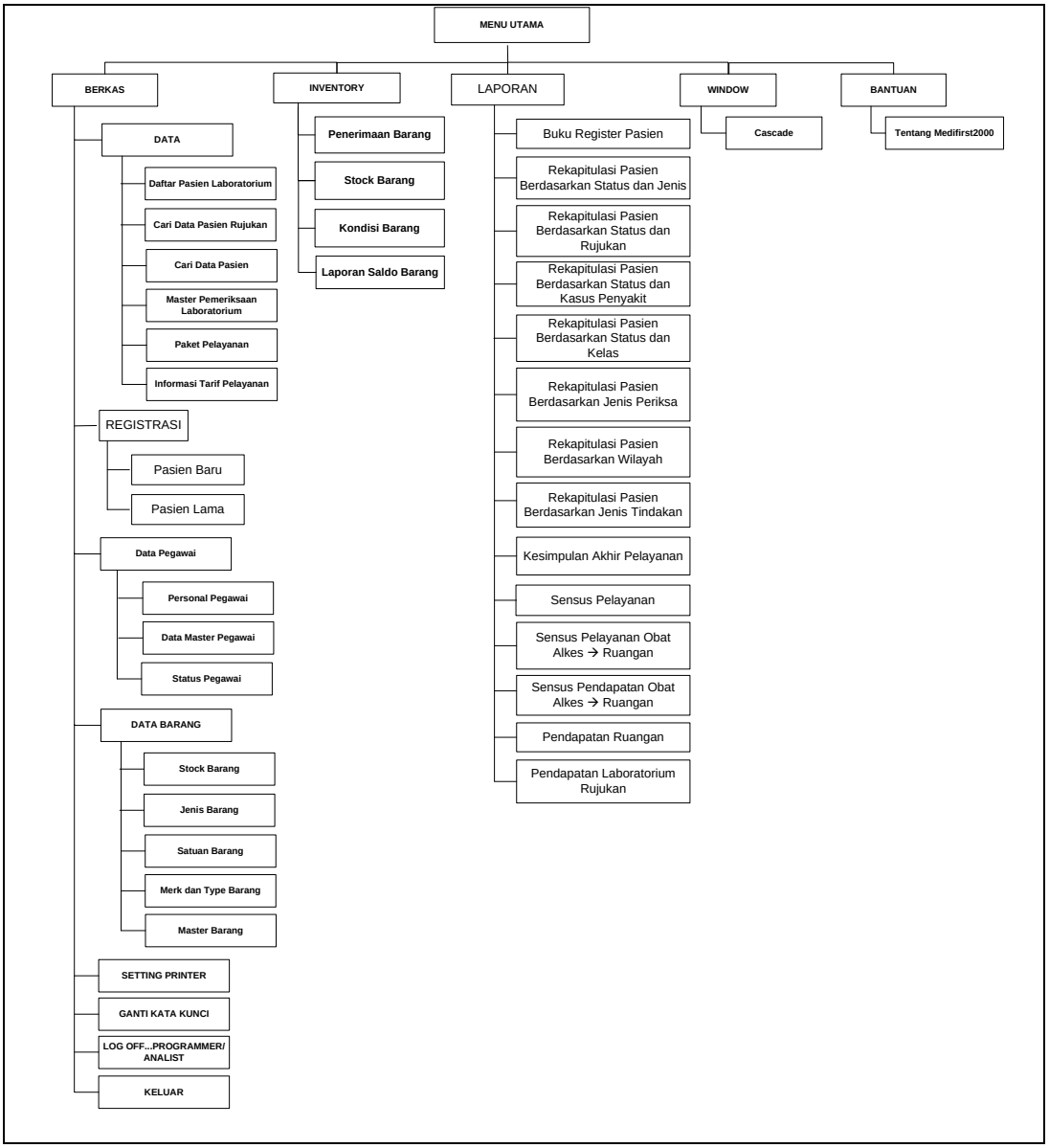
Nama Kelas	Daftar Tanggung-Jawab	Daftar Atribut
User	1. addUser 2. searchUser 3. updateUser 4. deleteUser 5. getUserNamePassword 6. validateUser	IdUser Username Password HakAkses
Pasien	1. addPasien 2. searchPasien 3. getIdPasien 4. getNamaPasien	IdPasien NamaLengkap TempatLahir TglLahir JenisKelamin Alamat Telepon Propinsi Kota Kecamatan

		Kelurahan RTRW KodePos
Dokter	1. addDokter 2. searchDokter 3. updateDokter 4. deleteDokter 5. getNamaDokter	IdPegawai NamaLengkap TempatLahir TglLahir JenisKelamin TglMasuk Alamat Telepon Propinsi Kota Kecamatan Kelurahan RTRW KodePos
HasilLab	1. addHasilLab 2. getHasilLab 3. getDiagnosa	NoCM IdPasien IdDokter JenisPemeriksaan NamaPemeriksaan DetailPemeriksaan HasilPemeriksaan NilaiNormal SatuanHasil
Transaksi	1. addTransaksi 2. caclTotalBiaya	NoTransaksi IdPasien IdDokter

		IdLab NamaPelayanan TotalBiayaPelayanan

Prototipe Antarmuka

Struktur Menu Aplikasi Laboratorium Klinik seperti di bawah ini :



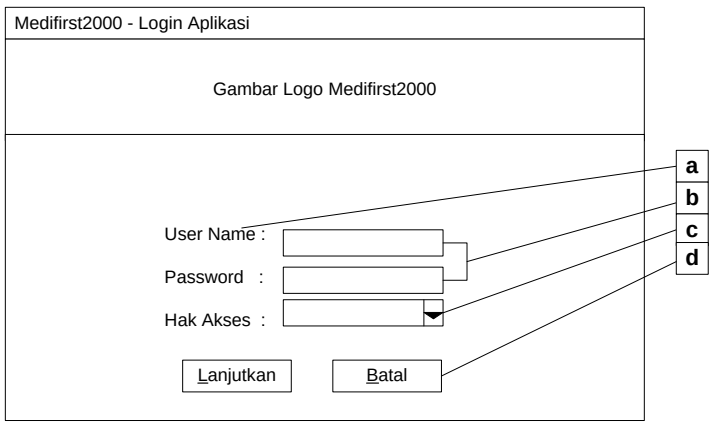
Tampilan antarmuka sistem informasi yang dibangun digambarkan seperti di bawah ini :

✓ **Login**

No Layar : PA-01

Nama Layar : Login

Deskripsi : Tampilan antarmuka yang digunakan untuk proses validasi berdasarkan hak akses



Gambar 4.23 Antarmuka Login

Keterangan :

Form untuk mengisi data User yang dapat mengakses aplikasi.

a. *Label*, digunakan untuk menuliskan nama User Name, Password, dan Hak Akses

b. *TextBox*, tempat untuk memasukkan User Name atau Password yang dapat mengakses aplikasi

c. *DataCombo*, jika di klik akan menampilkan data-data nama Hak Akses yang sudah ada di dalam *database*

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Jika mengklik tombol Lanjutkan maka akan masuk ke form Menu Utama (**Gambar 2**) sesuai dengan hak akses masing-masing user. Tetapi jika mengklik tombol Batal maka proses login akan dibatalkan

```
DataPegawai,Login

Algoritma :

Nama Operasi        : Login

If UserName = True Then
  If Password = True Then
    Koneksi ke database
    If koneksi gagal Then
      Kembali ke layar login
    Else
      Select user from Login where UserName=UserName and  Password=Password
      If tidak ada kesalahan Then
        Masuk ke menu (sesuai dengan hak akses)
      Else
        "Login Salah"
      Endif
```

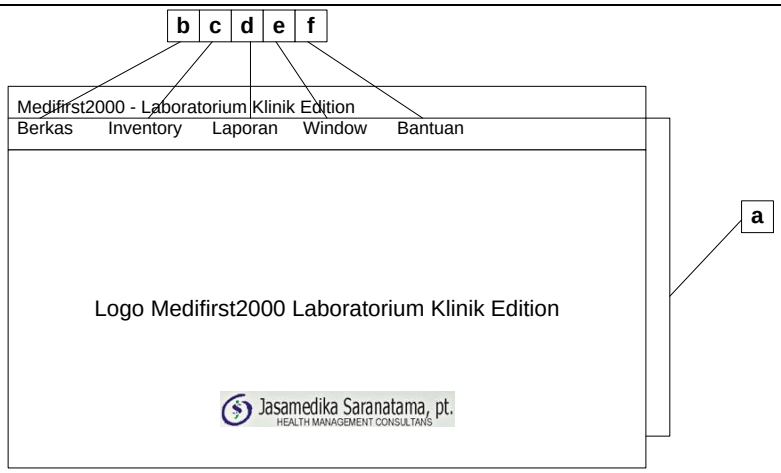
```
    Endif
  Endif
Endif
```

✓ **Menu Utama**

No Layar : PA-02

Nama Layar : Menu Utama

Deskripsi : Tampilan antarmuka yang pertama kali muncul setelah melakukan proses login



Gambar 4.24 Antarmuka Menu Utama

Keterangan :

Form yang pertama ditampilkan ketika user berhasil login ke aplikasi

a. *MDIUtama*, form yang digunakan sebagai tampilan utama dari aplikasi yang terdiri dari Menu Berkas, Inventory, Laporan, Window, dan Bantuan

b. *Menu Berkas*, jika di klik akan muncul sub menu dari menu berkas, yaitu :

- Sub Menu Data. Pada Sub Menu Data terdiri dari beberapa sub-sub menu lainnya, yaitu :
  1) Daftar Pasien Laboratorium
  2) Cari Data Pasien Rujukan
  3) Cari Data Pasien
  4) Master Pemeriksaan Laboratorium
  5) Paket Pelayanan
  6) Informasi Tarif Pelayanan
- Sub Menu Registrasi. Pada Sub Menu Registrasi terdiri dari beberapa sub-sub menu lainnya, yaitu :
  1) Pasien Baru
  2) Pasien Lama
- Sub Menu Data Pegawai. Pada Sub Menu Data Pegawai terdiri dari beberapa sub-sub menu lainnya, yaitu :
  1) Personal Pegawai
  2) Data Master Pegawai
  3) Status Pegawai
- Sub Menu Data Barang. Pada Sub Menu Data Barang terdiri dari beberapa sub-sub menu lainnya, yaitu :
  1) Stock Barang
  2) Jenis Barang
  3) Satuan Barang

4) Merk dan Type Barang

5) Master Barang

- Sub Menu Setting Printer
- Sub Menu Ganti Kata Kunci
- Sub Menu Log Off
- Sub Menu Keluar

c. *Menu Inventory*, jika di klik akan muncul sub menu dari menu informasi, yaitu :

- Penerimaan Barang
- Stock Barang
- Kondisi Barang
- Laporan dan Saldo Barang

d. *Menu Laporan*, jika di klik akan muncul sub menu dari menu laporan, yaitu :

- Buku Register Pasien
- Laporan Kunjungan Pasien
- Rekapitulasi Pasien Berdasarkan Status dan Jenis
- Rekapitulasi Pasien Berdasarkan Status dan Rujukan
- Rekapitulasi Pasien Berdasarkan Status dan Kasus Penyakit
- Rekapitulasi Pasien Berdasarkan Status dan Kelas
- Rekapitulasi Pasien Berdasarkan Jenis Periksa
- Rekapitulasi Pasien Berdasarkan Wilayah
- Rekapitulasi Pasien Berdasarkan Jenis Tindakan

e. *Menu Window*, jika di klik akan muncul sub menu dari menu window, yaitu :

- Cascade

f. *Menu Bantuan*, jika di klik akan muncul sub menu dari menu bantuan, yaitu :

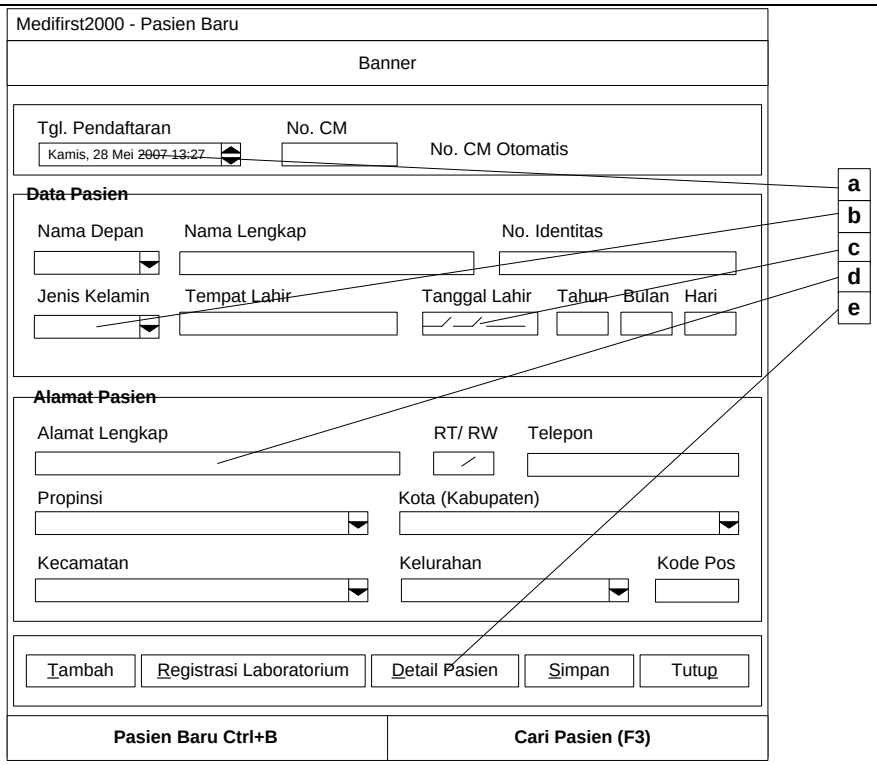
- Tentang Medifirst2000

✓ **Form Pasien Baru**

No Layar : PA-03

Nama Layar : Form Pasien Baru

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pasien



Gambar 4.25 Antarmuka Form Pasien Baru

Pasien,Propinsi,KotaKabupaten,Kecamatan,Kelurahan

Algoritma :

Nama Operasi : PasienBaru

Declare

NoCM NoIdentitas

TglDaftarMembership Title

NamaLengkap TempatLahir

TglLahir JenisKelamin

Alamat Telepon

```
Propinsi            Kota/Kabupaten
Kecamatan           Kelurahan
RT/RW               KodePos


If DataPasien terisi semua Then
  Insert INTO [Pasien]
  No. CM terisi otomatis
  Tombol Registrasi Pasien & Detail Pasien ➔ Aktif
    Else {DataPasien tidak lengkap}
     Akan tampil pesan kesalahan
    Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menginputkan data pasien baru yang akan melakukan pemeriksaan di laboratorium. Pada saat memilih form pasien baru, tampilan awal tombol Tambah, Simpan, dan Tutup dalam keadaan aktif sedangkan tombol Registrasi Laboratorium dan Detail Pasien dalam keadaan tidak aktif. Setelah data pasien baru diisi dan menekan tombol Simpan maka No. CM akan terisi otomatis, tombol Registrasi Laboratorium dan Detail Pasien akan menjadi aktif.

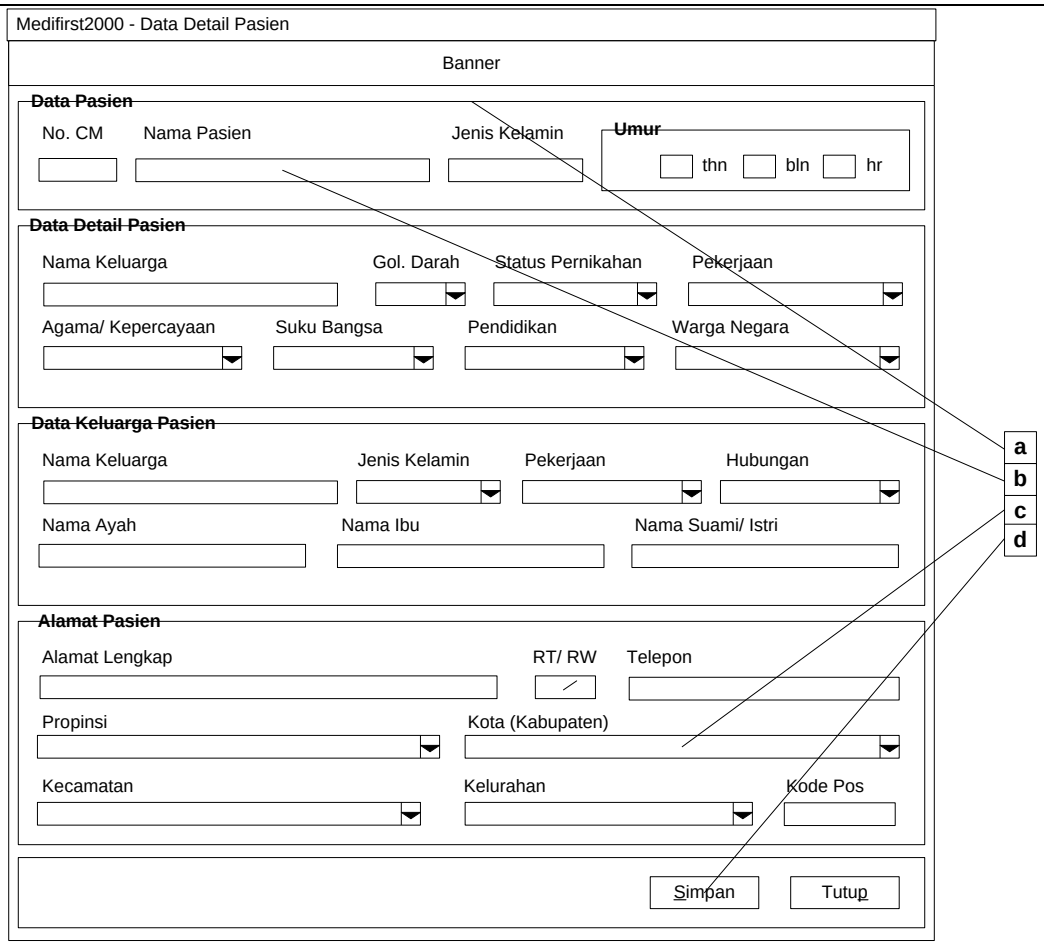
a. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data

b. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

c. *TextBox*, tempat untuk menginputkan data-data yang akan disimpan

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database. Tombol Tambah, jika diklik form akan kembali kosong. Tombol Detail Pasien, jika diklik akan menampilkan form seperti di bawah ini :

No Layar	:	PA-04
Nama Layar	:	Form Detail Pasien
Deskripsi	:	Tampilan antarmuka yang digunakan untuk menginputkan data detail pasien



Gambar 4.26 Antarmuka Form Data Detail Pasien

Pasien,DetailPasien,KeluargaPasien,Pekerjaan,Pendidikan,

Propinsi,KotaKabupaten,Kecamatan,Kelurahan

Algoritma :

Nama Operasi : DetailPasien

Declare

NamaKeluarga GolDarah

StatusPernikahan Pekerjaan

```
Agama/Kepercayaan      SukuBangsa
Pendidikan             WargaNegara
NamaKeluarga           JenisKelaminK
PekerjaanK             Hubungan
NamaAyah               NamaIbu
NamaSuami/Istri        AlamatLengkap
RT/RW                  Telepon
Propinsi               Kota/Kabupaten
Kecamatan              Kelurahan
KodePos


If DataDetailPasien terisi semua Then
  Data-data akan disimpan ke database
  Insert INTO [DetailPasien] & [KeluargaPasien]
    Else {DataDetailPasien tidak lengkap}
      Akan tampil pesan kesalahan
    Endif
Endif
```

Keterangan :

Form yang digunakan untuk menginputkan data detail pasien. No. CM, Nama Pasien, Jenis Kelamin dan Umur akan terisi sesuai dengan data pasien yang sudah dipilih.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

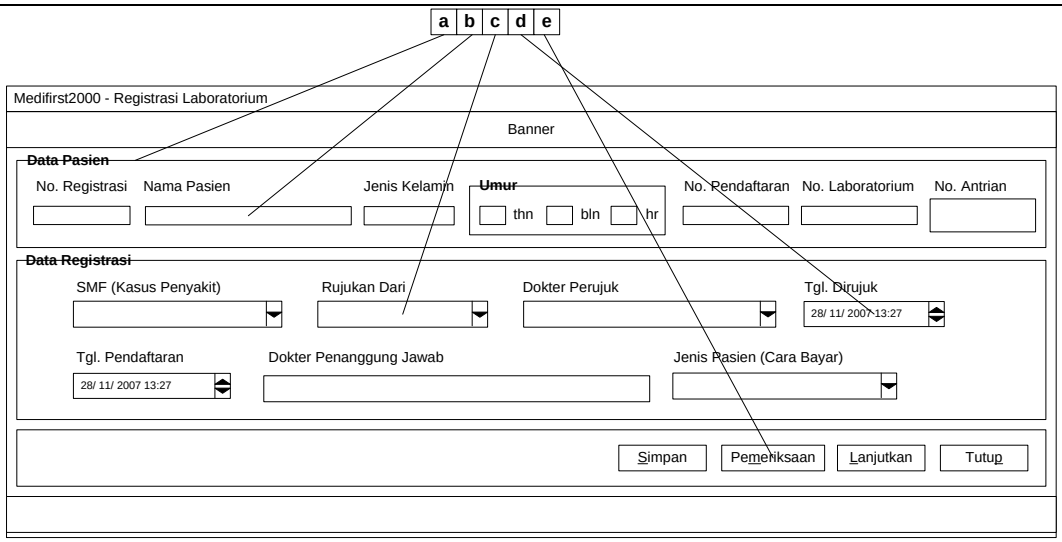
b. *TextBox*, tempat untuk menginputkan data-data yang akan dimasukkan

c. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database. Tombol Tutup, jika diklik akan kembali ke form Pasien Baru

Tombol Registrasi Laboratorium, jika diklik akan menampilkan form seperti di bawah ini :

No Layar	:	PA-05
Nama Layar	:	Form Registrasi Laboratorium
Deskripsi	:	Tampilan antarmuka yang digunakan untuk menginputkan data pada saat pasien melakukan registrasi



Gambar 4.27 Antarmuka Form Registrasi Laboratorium

Pasien,PasienDaftar,RegistrasiLaboratorium,Rujukan,

RujukanAsal,KelompokPasien,

Algoritma :

Nama Operasi : RegistrasiLaboratorium

Declare

NoCM NamaPasien

JenisKelamin NoPendaftaran

NoLaboratorium NoAntrian

RujukanAsal NamaPerujuk

TglDirujuk TglPendaftaran

NamaDokter JenisPasien

```
If DataRegistrasiLab terisi semua Then
  Data-data akan disimpan ke database
  Insert INTO [PasienDaftar],[RegistrasiLaboratorium],
        [Rujukan],[RujukanAsal],[KelompokPasien]
   Else {DataRegistrasiLab tidak lengkap}
    Akan tampil pesan kesalahan
   Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menginputkan data registrasi pasien. Data pasien akan terisi sesuai dengan data pasien yg diinputkan. No. Pendaftaran, No. Laboratorium, dan No. Antrian akan terisi otomatis setelah menekan tombol Simpan. Setelah menekan tombol Simpan, maka semua field dan tombol Simpan menjadi tidak Aktif.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *TextBox*, tempat untuk menginputkan data-data yang akan dimasukkan

c. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

d. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DD/ MM/ YYYY HH:SS

e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di

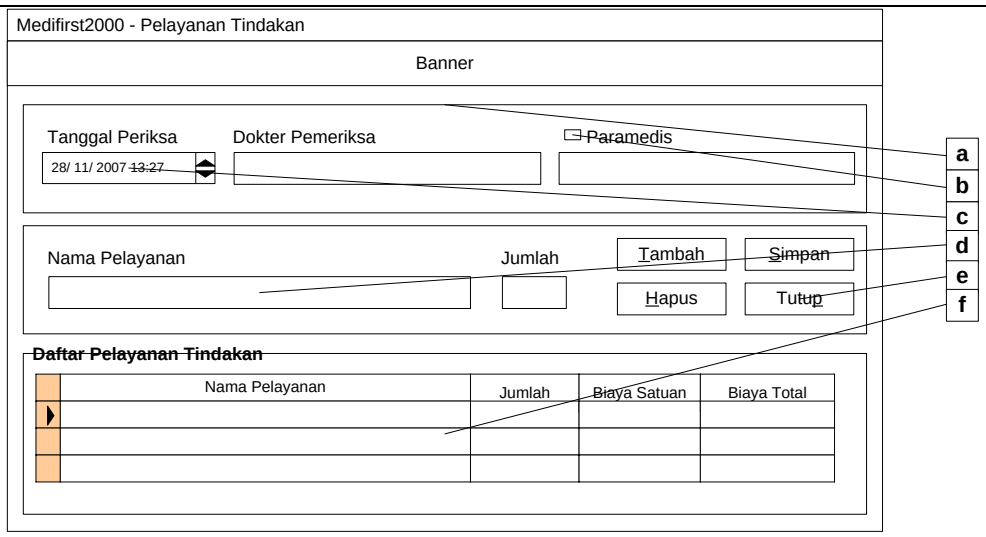
simpan ke database. Tombol Tutup, jika diklik akan kembali ke form Pasien Baru

Tombol Pemeriksaan, jika diklik akan menampilkan form seperti di bawah ini :

No Layar : PA-06

Nama Layar : Form Pelayanan Tindakan

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pelayanan tindakan yang diterima oleh pasien



Gambar 4.28 Antarmuka Form Pelayanan Tindakan

PasienDaftar,RegistrasiLaboratorium,DetailPelayananLab,

ListPelayananRS,JenisPelayanan,BiayaPelayanan

Algoritma :

Nama Operasi : PelayananTindakan

Declare

TglPelayanan NamaDokter

Paramedis NamaPelayanan

JmlPelayanan

If PelayananTindakan terisi semua Then

Insert INTO [DetailPelayananLab],[ListPelayanan],&

[BiayaPelayanan]

Data-data akan disimpan ke database

Else {PelayananTindakan tidak lengkap}

Akan tampil pesan kesalahan

Endif

Endif

Keterangan :

Form di atas digunakan untuk menginputkan data pelayanan tindakan yang dilakukan oleh pasien. Pada pengisian Nama Pelayanan, jika kita memasukkan huruf atau nama yang dimaksud maka akan ditampilkan Data Grid yang berisi Jenis Pelayanan, Nama Pelayanan, Kelas, Jenis Tarif, dan Tarif. Jika menekan tombol Tambah maka Nama Pelayanan yang sudah

diisi akan ditampilkan di Data Grid yang berada pada kolom Daftar Pelayanan Tindakan, kita dapat menginputkan data pelayanan lebih dari satu. Tekan tombol Simpan untuk menyimpan data pelayanan tindakan ke database.

a. *Frame,* digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *CheckBox*, digunakan untuk memilih paramedis

c. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data

d. *TextBox,* tempat untuk memasukkan data-data yang akan diinputkan

e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Tambah digunakan untuk menambahkan data yang akan disimpan ke DataGrid. Tombol Simpan digunakan untuk menyimpan data ke database. Tombol Hapus digunakan untuk menghapus data yang sudah ditambahkan ke DataGrid

f. *DataGrid*, tempat untuk menampilkan data pelayanan tindakan yang ditambahkan dan yang akan di simpan ke database

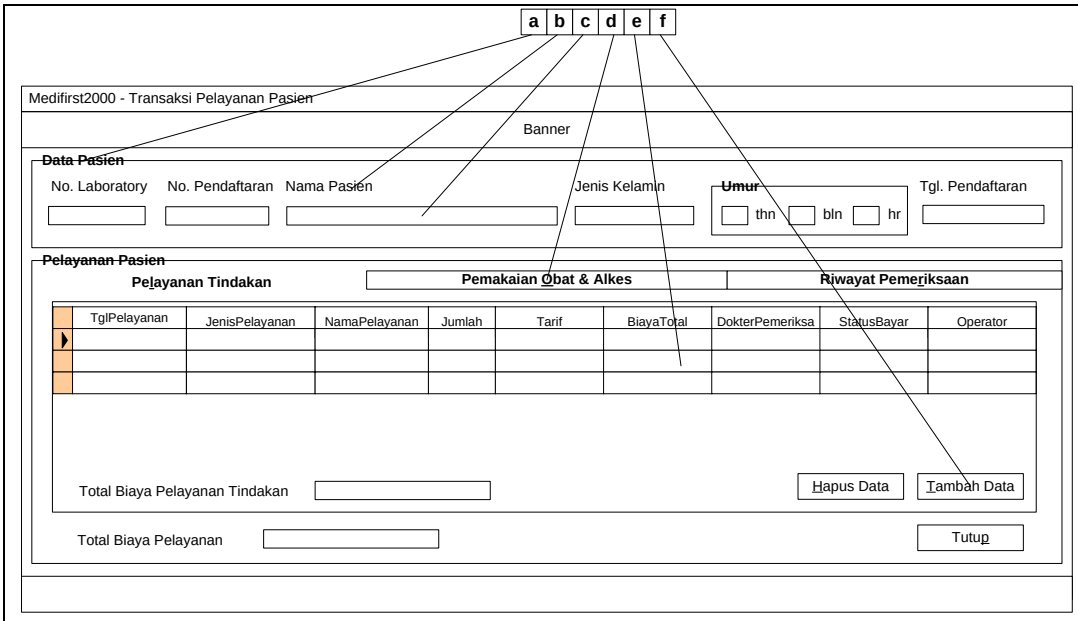
Tombol Lanjutkan, jika diklik akan menampilkan form seperti di bawah ini :

- Tab Pelayanan Tindakan

No Layar : PA-07

Nama Layar : Form Transaksi Pelayanan Pasien

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pelayanan pasien selama melakukan pemeriksaan



Gambar 4.29 Antarmuka Form Transaksi Pelayanan Pasien

Pasien,PasienDaftar,RegistrasiLaboratorium,DetailPelayananLab,

ListPelayananRS,JenisPelayanan,BiayaPelayanan

Algoritma :

Nama Operasi : PelayananTindakan

Declare

```
NoLaboratorium          NoPendaftaran
NamaPasien              JenisKelamin
TglPendaftaran          TglPelayanan
JenisPelayanan          NamaPelayanan
JmlPelayanan            Tarif
NamaDokter              Operator

If PelayananTindakan sudah ada di dalam database Then
  Menampilkan semua data yang ada di dalam database
  Select * from [Pasien],[PasienDaftar],
           [RegistrasiLaboratorium],[ListPelayanan],&
           [BiayaPelayanan]
  Else {PelayananTindakan blm ada}
    Tidak akan ada data yang ditampilkan
   Endif
Endif
```

Keterangan :

Form diatas digunakan untuk menampilkan pelayanan yang sudah dilakukan oleh pasien. Untuk No. Laboratory, No. Pendaftaran, Nama Pasien, Jenis Kelamin, Umur dan Tgl. Pendaftaran akan terisi sesuai dengan data pasien. Total Biaya Pelayanan Tindakan akan terisi otomatis

sesuai dengan total biaya atas pelayanan tindakan yang dilakukan oleh pasien. Total Biaya Pelayanan akan terisi otomatis sesuai dengan total biaya keseluruhan yang harus dibayar pasien.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

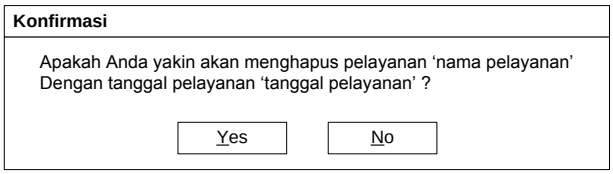
c. *TextBox*, tempat untuk memasukkan data-data yang akan diinputkan

d. *SsTab*, digunakan untuk membuat 3 form pengisian data dalam satu tampilan form

e. *DataGrid*, tempat untuk menampilkan data pelayanan tindakan yang sudah dilakukan oleh pasien

f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Tambah Data digunakan untuk menambahkan data pelayanan tindakan, maka akan tampil form **Pelayanan Tindakan** seperti yang sudah dibahas di atas. Tombol Hapus digunakan untuk menghapus data yang sudah ada di DataGrid.

Tombol Hapus Data, jika diklik akan menampilkan Konfirmasi ”Apakah Anda yakin akan menghapus pelayanan ’nama pelayanan’ Dengan tanggal pelayanan ‘tanggal pelayanan’?” seperti gambar di bawah ini :



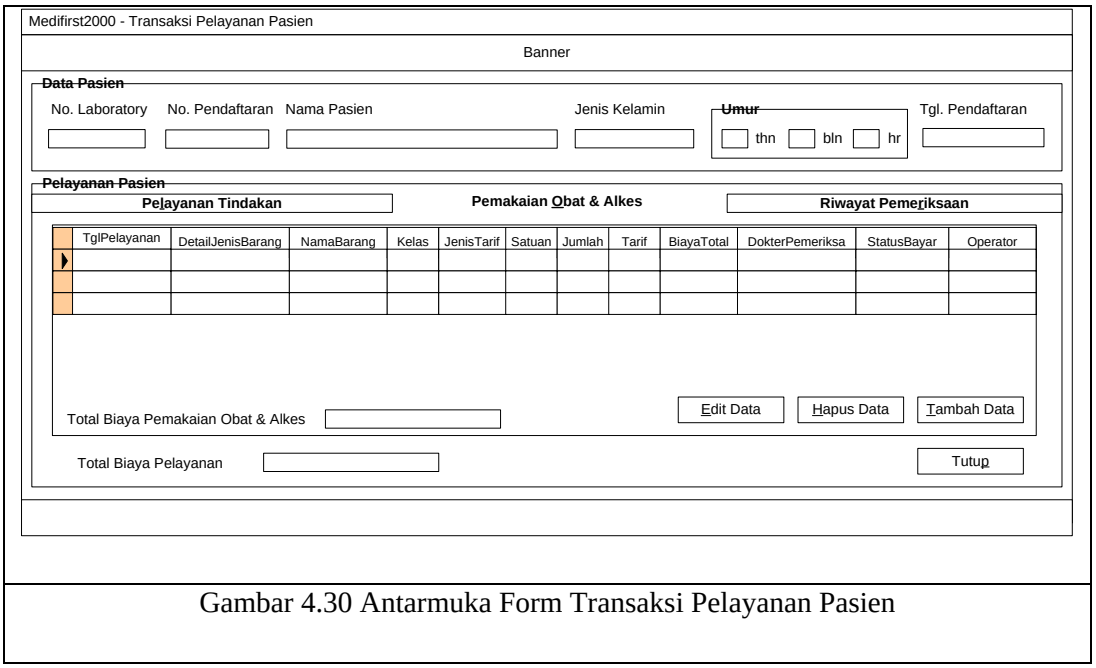
Tombol Tambah Data, jika diklik akan tampil form Pelayanan Tindakan seperti yang sudah dibahas di atas.

- Tab Pemakaian Obat & Alkes

No Layar : PA-08

Nama Layar : Form Transaksi Pelayanan Pasien

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pemakaian obat & alkes pasien selama melakukan pemeriksaan



Gambar 4.30 Antarmuka Form Transaksi Pelayanan Pasien

Pasien,PasienDaftar,RegistrasiLaboratorium,PemakaianAlkes,

DetailPemakaianAlkes,MasterBarang,DetailJenisBarang,

JenisBarang,Satuan,KelasPelayanan,BiayaPelayanan

Algoritma :

Nama Operasi : PemakaianObatAlkes

Declare

NoLaboratorium NoPendaftaran

NamaPasien JenisKelamin

TglPendaftaran TglPelayanan

DetailJenisBarang NamaBarang

Kelas Satuan

Jumlah Tarif

BiayaTotal NamaDokter

If PemakaianObatAlkes sudah ada di dalam database Then

Menampilkan semua data yang ada di dalam database

Select * from [Pasien],[PasienDaftar],

[RegistrasiLaboratorium],[PemakaianAlkes],

[MasterBarang],[Satuan]&[KelasPelayanan]

Else {PemakaianObatAlkes blm ada}

Tidak akan ada data yang ditampilkan

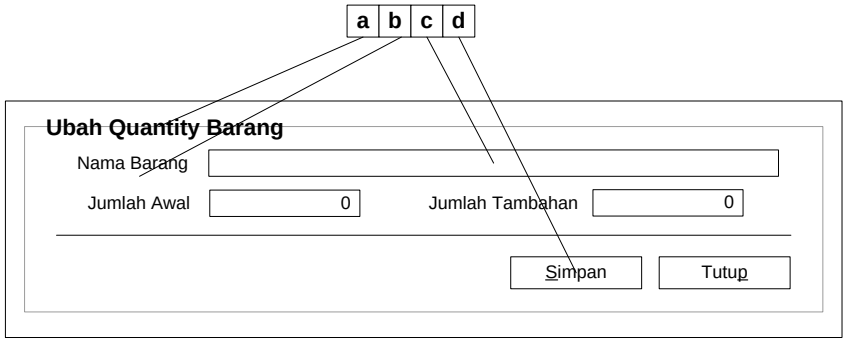
Endif

Endif

Keterangan :

Form diatas digunakan untuk menampilkan data pemakaian obat & alkes yang sudah diterima oleh pasien. Untuk No. Laboratory, No. Pendaftaran, Nama Pasien, Jenis Kelamin, Umur dan Tgl. Pendaftaran akan terisi sesuai dengan data pasien. Total Biaya Pemakaian Obat & Alkes akan terisi otomatis sesuai dengan total biaya atas pemakaian obat & alkes yang diterima oleh pasien. Total Biaya Pelayanan akan terisi otomatis sesuai dengan total biaya keseluruhan yang harus dibayar pasien.

Tombol Edit Data, jika diklik maka akan tampil form seperti di bawah ini :



MasterBarang,PemakaianAlkes

Algoritma :

Nama Operasi : EditPemakaianObat&Alkes

Declare

NamaBarang JmlAwal

```
    JmlTambahan


If PemakaianObatAlkes sudah dipilih Then
  Tekan tombol Edit
  Masukkan jumlah tambahannya
  Update [PemakaianAlkes] set jumlah=”jmltambahan”
    Where NamaBarang=”yangDipilih”
  Else {PemakaianObatAlkes blm dipilih}
    Tidak akan ada data yang bisa dirubah
   Endif
Endif
```

Keterangan :

Form di atas digunakan untuk melakukan edit pada data pemakaian obat & alkes yang sudah ada. Nama Barang dan Jumlah Barang dalam keadaan tidak aktif. Jumlah Tambahan untuk menginputkan jumlah baru.

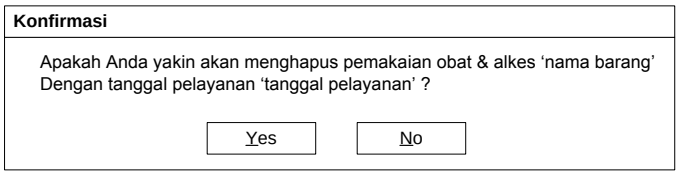
a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

c. *TextBox*, tempat untuk menampilkan data yang sudah ada dan tempat untuk memasukkan data-data yang akan diinputkan

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan digunakan untuk menyimpan data yang sudah diEdit. Tombol Tutup, jika diklik akan kembali ke form semula.

Tombol Hapus Data, jika diklik akan menampilkan Konfirmasi ”Apakah Anda yakin akan menghapus pemakaian obat & alkes ’nama barang’ Dengan tanggal pelayanan ‘tanggal pelayanan’?” seperti gambar di bawah ini :



Pasien,PasienDaftar,RegistrasiLaboratorium,PemakaianAlkes,

DetailPemakaianAlkes,MasterBarang,DetailJenisBarang,

JenisBarang,Satuan,KelasPelayanan,BiayaPelayanan

Algoritma :

Nama Operasi : HapusPemakaianObatAlkes

Declare

NoLaboratorium NoPendaftaran

NamaPasien JenisKelamin

TglPendaftaran TglPelayanan

DetailJenisBarang NamaBarang

```
    Kelas                Satuan
    Jumlah               Tarif
    BiayaTotal           NamaDokter

If PemakaianObatAlkes sudah dipilih Then
  Tekan tombol Hapus
  Akan tampil Konfirmasi penghapusan,jika tekan Yes
  Delete from [Pasien],[PasienDaftar],
        [RegistrasiLaboratorium],[PemakaianAlkes],
        [MasterBarang],[Satuan]&[KelasPelayanan]
   Where NamaBarang=”yangDipilih”
  Else {PemakaianObatAlkes blm dipilih}
     Tidak akan ada data yang bisa dihapus
   Endif
Endif
```

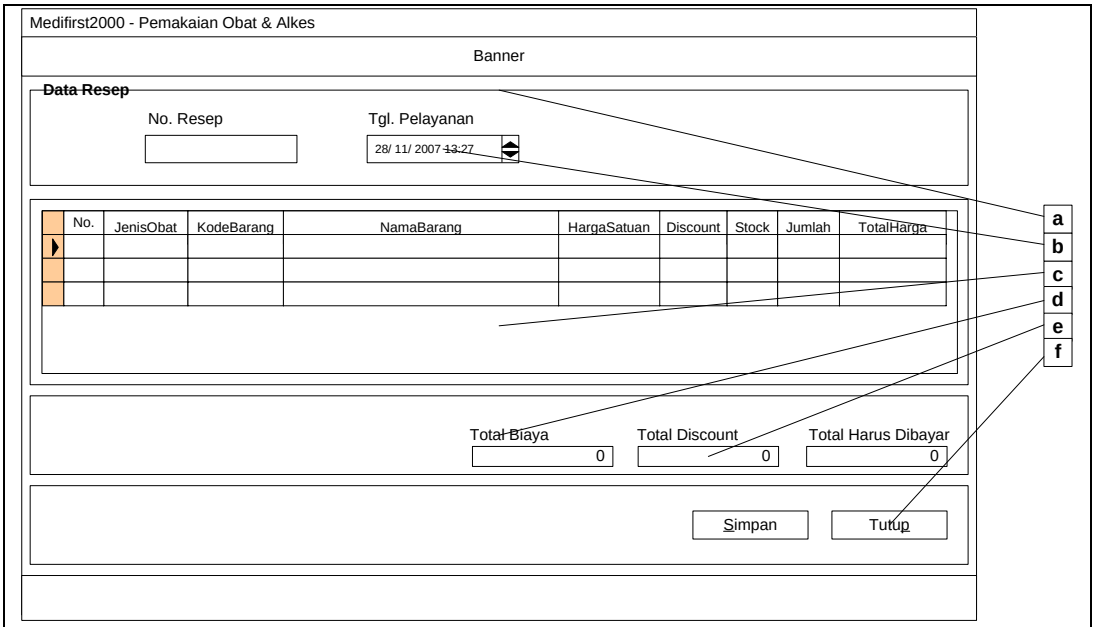
Tombol Tambah Data, jika diklik akan tampil form Pemakaian Obat & Alkes seperti di bawah ini :

No Layar : PA-09

Nama Layar : Form Pemakaian Obat & Alkes

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pemakaian obat & alkes pasien

selama melakukan pemeriksaan



Gambar 4.31 Antarmuka Form Pemakaian Obat & Alkes

PemakaianAlkes,MasterBarang,JenisObat,

StockBarang,HargaBarang

Algoritma :

Nama Operasi : TambahPemakaianObatAlkes

Declare

NoStruk TglPelayanan

JenisObat KodeBarang

NamaBarang HargaSatuan

Discount Stock

Jumlah TotalHarga

```
If NamaBarang yang dipilih ada Then
  Data-data yang lain akan terisi
  Insert INTO [JenisObat],[MasterBarang],[StockBarang],
          [PemakaianAlkes],&[HargaBarang]
  Else {NamaBarang yang dicari tidak ada}
     Lihat kembali tabel MasterBarang
    Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menginputkan data pemakaian obat & alkes. Data pemakaian obat & alkes diinputkan melalui DataGrid. Pada kolom Jenis Obat, tekan sembarang tombol maka akan ditampilkan nama-nama Jenis Obat yang sudah ada seperti tampilan DataCombo kemudian pilih salah satu lalu tekan Enter. Untuk mengisikan Nama Barang nya, pada kolom NamaBarang ketikkan nama yg mau dicari nanti akan ditampilkan data-datanya yang terdiri dari JenisBarang, NamaBarang, Satuan, HargaBarang, dan JmlStok setelah nama barangnya dipilih tekan Enter, maka KodeBarang, HargaSatuan, Discount, dan Stock akan otomatis terisi sesuai dengan NamaBarang yang dipilih. Pada kolom Jumlah ketikkan banyaknya barang yang dipakai, maka TotalHarga akan otomatis terisi

sesuai dengan banyaknya barang dikali harga barang (TotalHarga = Jumlah*HargaSatuan).

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data

c. *DataGrid*, tempat untuk menginputkan data pemakaian obat & alkes yang dipakai oleh pasien

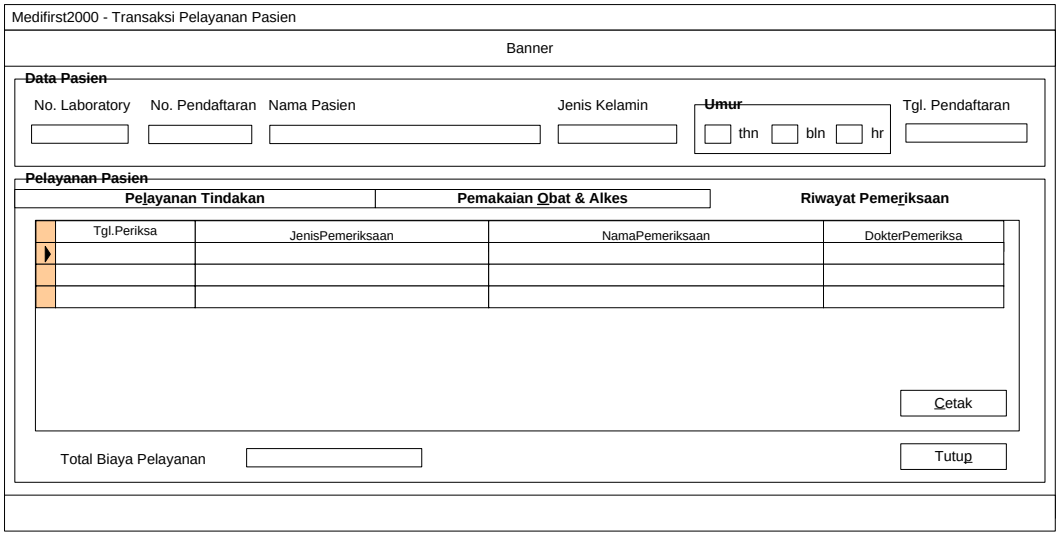
d. *Label*, digunakan untuk memberikan keterangan kepada item-item

e. *TextBox*, tempat untuk menampilkan data-data yang akan diinputkan

f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan digunakan untuk menyimpan data pemakaian obat & alkes. Tombol Tutup, jika diklik akan kembali ke form semula.

- Riwayat Pemeriksaan

No Layar	:	PA-10
Nama Layar	:	Form Transaksi Pelayanan Tindakan
Deskripsi	:	Tampilan antarmuka yang digunakan untuk menampilkan riwayat pemeriksaan yang pernah dilakukan oleh pasien



Gambar 4.32 Antarmuka Form Transaksi Pelayanan Tindakan

Pasien,PasienDaftar,RegistrasiLaboratorium,DetailPelayananLab,

ListPelayananRS,JenisPelayanan

Algoritma :

Nama Operasi : CetakPelayananTindakan

Declare

NoLaboratorium NoPendaftaran

NamaPasien JenisKelamin

TglPendaftaran TglPelayanan

JenisPelayanan NamaPelayanan

NamaDokter

```
If PelayananTindakan sudah ada di dalam database Then
  Menampilkan semua data yang ada di dalam database
  Select * from [Pasien],[PasienDaftar],
          [RegistrasiLaboratorium],&[ListPelayanan]
  Data dapat dicetak
  Else {PelayananTindakan blm ada}
    Tidak ada yang bisa dicetak
   Endif
Endif
```

Keterangan :

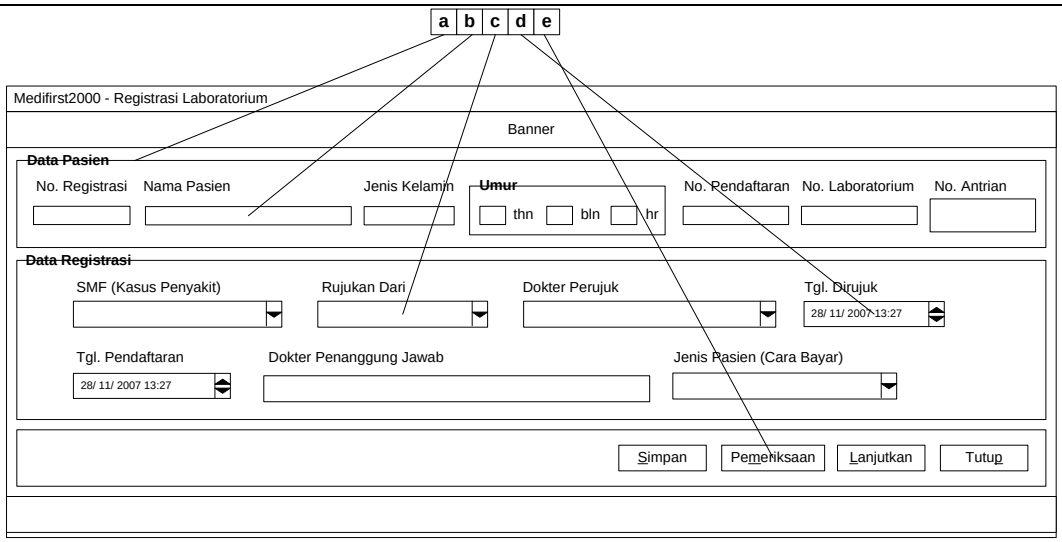
Form di atas digunakan untuk menampilkan seluruh Riwayat Pemeriksaan yang sudah dilakukan oleh pasien yang bersangkutan. Tombol Cetak digunakan untuk mencetak Riwayat Pemeriksaan pasien.

✓ **Form Pasien Lama**

No Layar : PA-11

Nama Layar : Form Registrasi Laboratorium

Deskripsi : Tampilan antarmuka yang digunakan untuk registrasi pasien lama pada saat akan melakukan pemeriksaan kembali



Gambar 4.33 Antarmuka Form Registrasi Laboratorium

Keterangan :

Form di atas digunakan untuk menginputkan data registrasi pasien lama. Masukkan No. Registrasi pasien kemudian tekan Enter, maka Nama Pasien, Jenis Kelamin, dan Umur akan tampil. No. Pendaftaran, No. Laboratorium, dan No. Antrian akan terisi otomatis setelah menekan tombol Simpan. Setelah menekan tombol Simpan, maka semua field dan tombol Simpan menjadi tidak Aktif.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *TextBox*, tempat untuk menginputkan data-data yang akan dimasukkan

c. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

d. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DD/ MM/ YYYY HH:SS

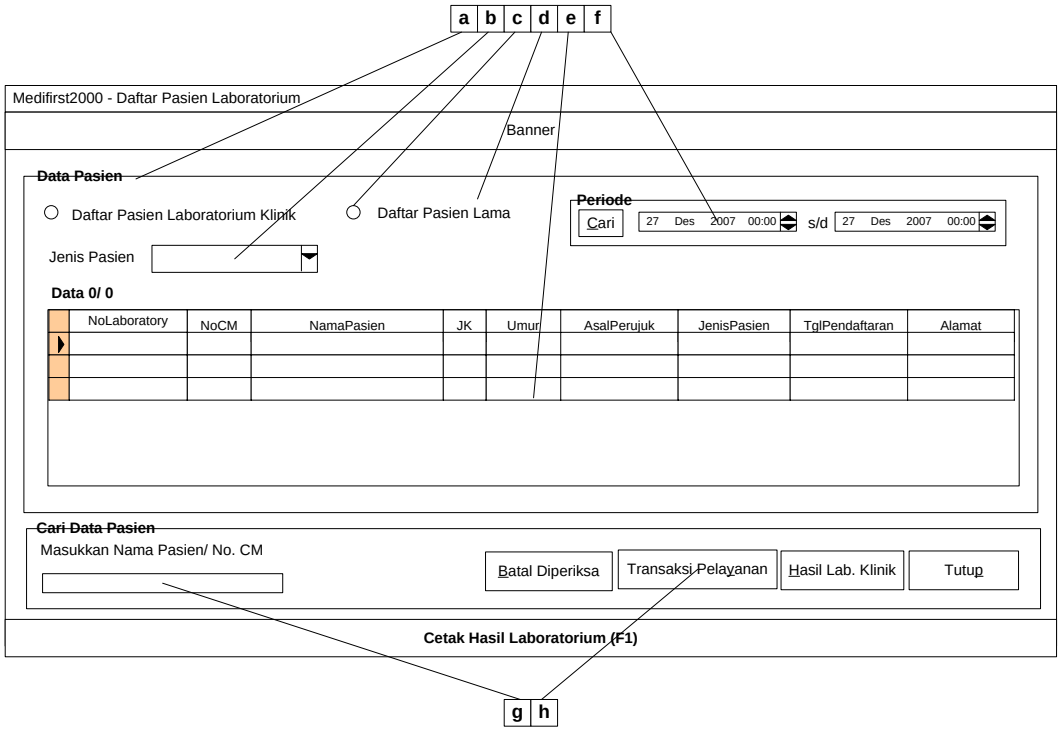
e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database. Tombol Pemeriksaan, jika diklik akan tampil form **Pelayanan Tindakan** seperti yang sudah dibahas di atas. Tombol Lanjutkan, jika diklik akan tampil form **Transaksi Pelayanan Pasien** seperti yang sudah dibahas di atas. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

✓ **Form Daftar Pasien Laboratorium**

No Layar : PA-12

Nama Layar : Form Daftar Pasien Laboratorium

Deskripsi : Tampilan antarmuka yang digunakan untuk menampilkan dan mencari data pasien laboratorium, dan juga dapat menambahkan pelayanan tindakan, melihat hasil pemeriksaan laboratorium, dan mencetak hasil laboratorium



Gambar 4.34 Antarmuka Form Daftar Pasien Laboratorium

Pasien,PasienDaftar,RegistrasiLaboratorium,

RujukanAsal,KelompokPasien,

Algoritma :

Nama Operasi : DaftarPasienLaboratorium

Declare

NoCM NamaPasien

JenisKelamin NoLaboratorium

RujukanAsal Alamat

JenisPasien

```
If DaftarPasienLaboratorium sudah ada dlm database Then
  Data-data akan ditampilkan
  Select * From [PasienDaftar],[RegistrasiLaboratorium],
          [RujukanAsal],[KelompokPasien]
    Else { DaftarPasienLaboratorium blm ada}
     Tidak akan ada yang ditampilkan
    Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menampilkan Daftar Pasien Laboratorium Klinik berdasarkan Jenis Pasien dalam Periode waktu tertentu, mencari data pasien Laboratorium Klinik dengan menginputkan Nama Pasien atau No. CM pasien yang akan dicari, membatalkan pemeriksaan, menambahkan Transaksi Pelayanan, melihat dan mencetak Hasil Laboratorium.

a. *Frame,* digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

c. *RadioButton*, digunakan untuk memilih salah satu pilihan yang sudah ada

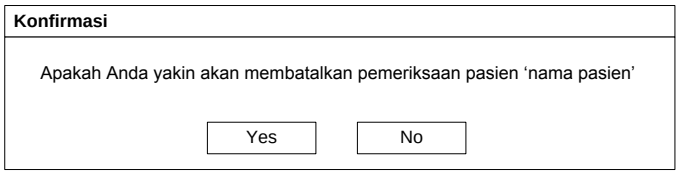
d. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

e. *DataGrid*, tempat untuk menampilkan data daftar pasien laboratorium klinik

f. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DD/ MM/ YYYY HH:SS

g. *TextBox*, tempat untuk menginputkan data-data yang akan dimasukkan

h. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Batal Diperiksa, jika diklik akan menampilkan Konfirmasi ”Apakah Anda yakin akan membatalkan pemeriksaan pasien ’nama pasien’?” seperti gambar di bawah ini :



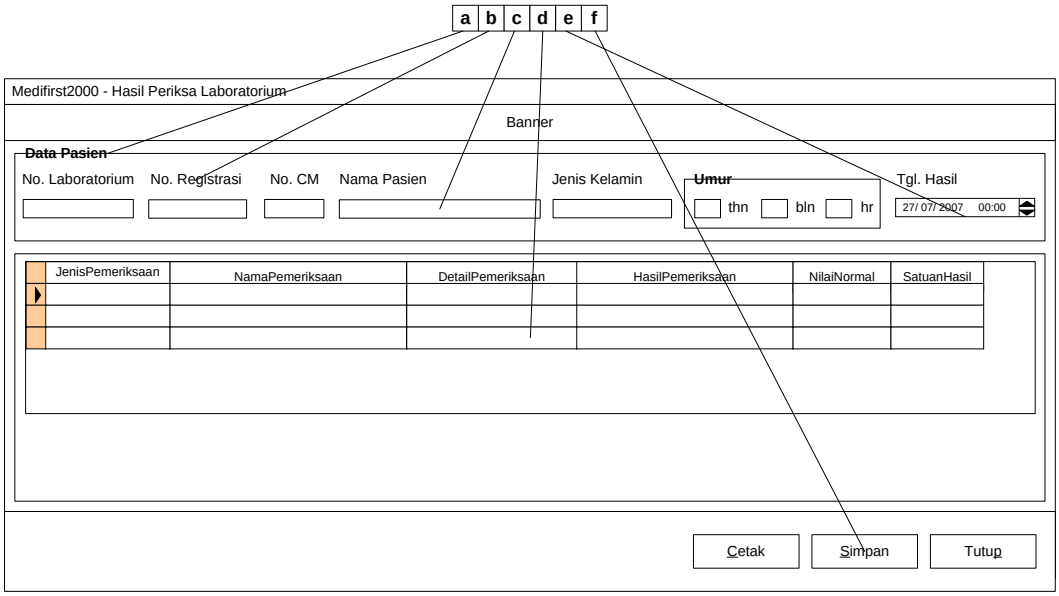
Tombol Transaksi Pelayanan, jika diklik akan tampil form **Transaksi Pelayanan Pasien** seperti yang sudah dibahas di atas.

Tombol Hasil Lab. Klinik, jika diklik akan tampil form Hasil Periksa Laboratorium seperti di bawah ini :

No Layar : PA-13

Nama Layar : Form Hasil Periksa Laboratorium

Deskripsi : Tampilan antarmuka yang digunakan untuk menampilkan data hasil pemeriksaan laboratorium pasien yang bersangkutan



Gambar 4.35 Antarmuka Form Daftar Pasien Laboratorium

Pasien,PasienDaftar,RegistrasiLaboratorium,

JenisPemeriksaan,ListPelayananRS,DetailPemeriksaan,

ConvertJenisPemeriksaanToPelayanan,KadarNormal,SatuanHasil,

HasilPemeriksaan

Algoritma :

Nama Operasi : HasilPemeriksaanLaboratorium

Declare

NoLaboratorium NoRegistrasi

NoCM NamaPasien

JenisKelamin TglHasil

```
      JenisPemeriksaan          NamaPemeriksaan
      DetailPemeriksaan         HasilPemeriksaan
      NilaiNormal               SatuanHasil


 If HasilLaboratorium sudah ada dlm database Then
   Data-data akan ditampilkan
   Select * From [PasienDaftar],[RegistrasiLaboratorium],
             [Pasien],[JenisPemeriksaan],
             [DetailPemeriksaan],[HasilPemeriksaan],
             [SatuanHasil],&[KadarNormal]
     Else { HasilLaboratorium blm ada}
      Tidak akan ada yang ditampilkan
     Endif
 Endif
```

Keterangan :

Form di atas digunakan untuk menampilkan dan menginputkan data hasil pemeriksaan laboratorium dari pasien yang bersangkutan. No. Laboratorium, No. Registrasi, No. CM, Nama Pasien, Jenis Kelamin, dan Umur dalam keadaan tidak aktif dan terisi dengan data-data pasien yang kita pilih. Pada DataGrid akan ditampilkan semua pemeriksaan yang sudah dilakukan oleh pasien tersebut, pada kolom HasilPemeriksaan tekan Enter terlebih dahulu untuk mengetikkan hasil pemeriksaan nya.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

c. *TextBox*, tempat untuk menampilkan data-data yang ada

d. *DataGrid*, tempat untuk menampilkan data pemeriksaan yang sudah dilakukan oleh pasien dan menginputkan data hasil pemeriksaan laboratorium

e. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DD/ MM/ YYYY HH:SS

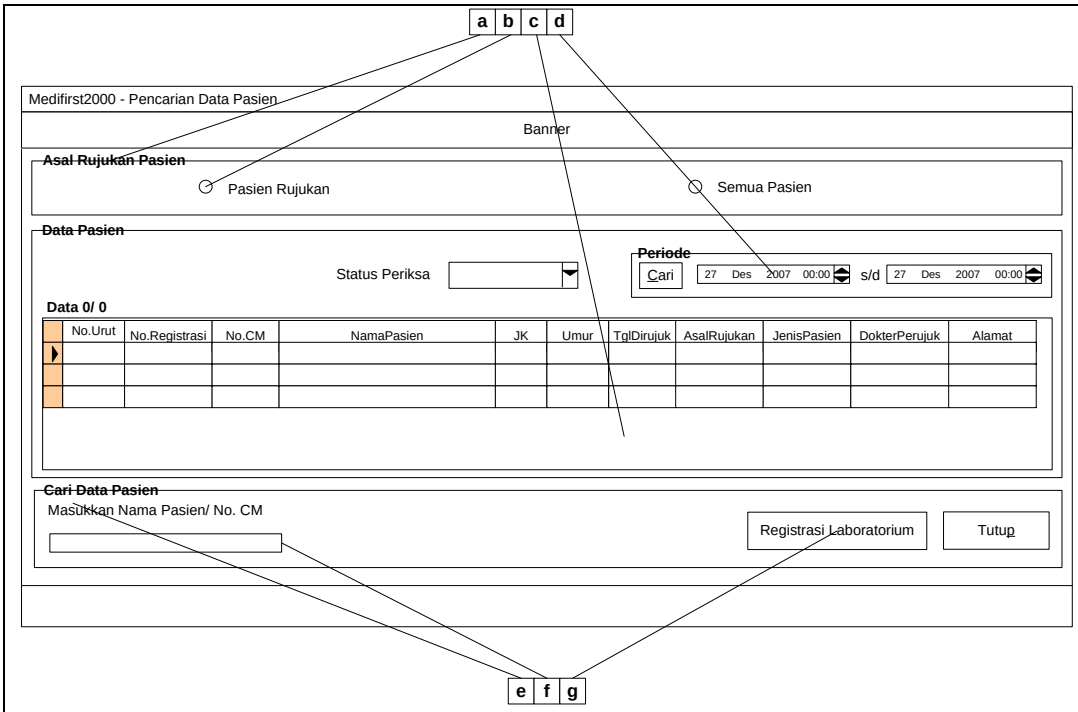
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Cetak untuk mencetak data hasil pemeriksaan laboratorium. Tombol Simpan untuk menyimpan data hasil pemeriksaan laboratorium ke dalam database. Tombol Tutup akan kembali ke form Daftar Pasien Laboratorium.

✓ **Form Cari Data Pasien Rujukan**

No Layar : PA-14

Nama Layar : Form Pencarian Data Pasien

Deskripsi : Tampilan antarmuka yang digunakan untuk menampilkan data pasien yang dirujuk dari rumah sakit lain



Gambar 4.36 Antarmuka Form Pencarian Data Pasien

Pasien,PasienDaftar,PasienMasukRumahSakit,

KelompokPasien,DataPegawai

Algoritma :

Nama Operasi : CariDataPasienRujukan

Declare

NoUrut NoCM

NamaPasien JK

Umur Kelas

JenisPasien DokterPemeriksa

Alamat

```
If DataPasien sudah ada dlm database Then
  Data-data akan ditampilkan
  Select * From [PasienDaftar],[PasienMasukRumahSakit],
       [Pasien],[KelasPelayanan],[KelompokPasien],&
       [DataPegawai]
    Else {DataPasien blm ada}
     Tidak akan ada yang ditampilkan
    Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menampilkan data pasien rujukan yang dicari. Jika Status Periksanya Sudah maka jika menekan tombol Registrasi Laboratorium akan tampil Informasi ”Pasien tersebut sudah diperiksa”.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *RadioButton*, digunakan untuk memilih salah satu pilihan yang sudah ada

c. *DataGrid*, tempat untuk menampilkan data pemeriksaan yang sudah dilakukan oleh pasien dan menginputkan data hasil pemeriksaan laboratorium

d. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DD/ MM/ YYYY HH:SS

e. *Label*, digunakan untuk memberikan keterangan kepada item-item

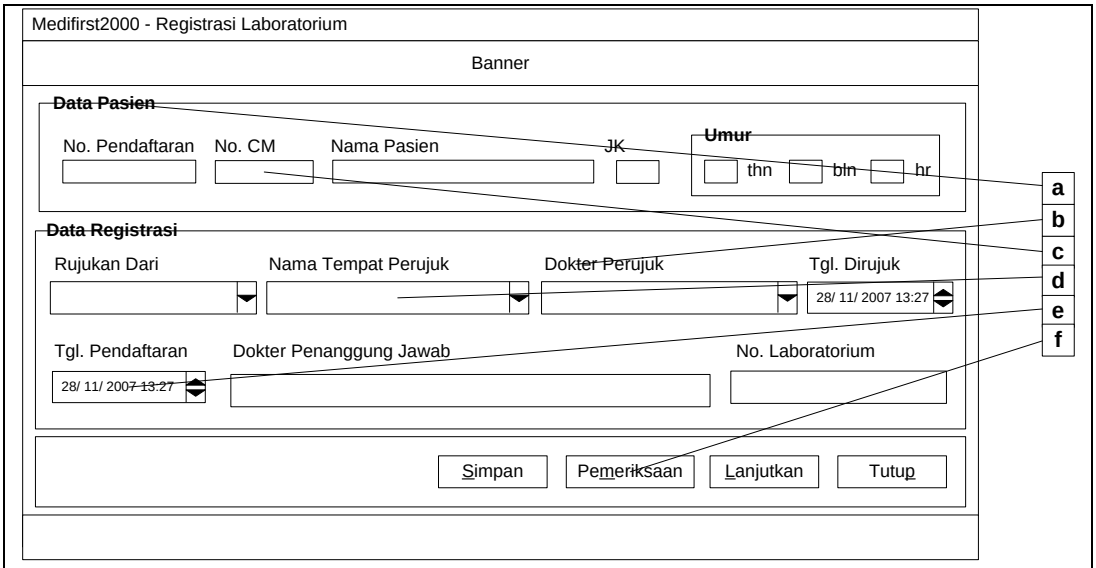
f. *TextBox*, tempat untuk menginputkan data pasien yang akan dicari

g. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Registrasi Laboratorium, jika diklik maka akan tampil form Registrasi Laboratorium seperti di bawah ini :

No Layar : PA-15

Nama Layar : Form Registrasi Laboratorium

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data registrasi pasien yang dicari dan statusnya belum diperiksa



Gambar 4.37 Antarmuka Form Registrasi Laboratorium

Pasien,PasienDaftar,RegistrasiLaboratorium,Rujukan,

RujukanAsal

Algoritma :

Nama Operasi : RegistrasiLaboratorium

```
Declare
    NoPendaftaran        NoCM
    NamaPasien           JK
    RujukanDari          NamaTempatPerujuk
    DokterPerujuk        TglDirujuk
    TglPendaftaran       DokterPenanggungJawab
    NoLaboratorium

If DataRegistrasiLab terisi semua Then
  Data-data akan disimpan ke database
  Insert INTO [PasienDaftar],[RegistrasiLaboratorium],
          [Rujukan],[RujukanAsal]
   Else {DataRegistrasiLab tidak lengkap}
    Akan tampil pesan kesalahan
   Endif
Endif
```

Keterangan :

Form yang digunakan untuk melakukan registrasi pasien yang dicari dan statusnya belum diperiksa. No. Pendaftaran, No. CM, Nama Pasien, JK, dan Umur sudah terisi data yang sesuai dengan data pasien yang dicari dan dalam keadaan tidak aktif. Untuk menginputkan nama Dokter Penanggung Jawab, ketikkan nama atau salah satu huruf maka akan ditampilkan data-data dokter

seperti KodeDokter, NamaDokter, JK, dan Jabatan kemudian pilih nama dokter yang dimaksud. No. Laboratorium akan terisi otomatis setelah menekan tombol Simpan.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

c. *TextBox*, tempat untuk menampilkan data-data pasien dan untuk menginputkan data-data yang akan dimasukkan

d. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

e. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DD/ MM/ YYYY HH:SS

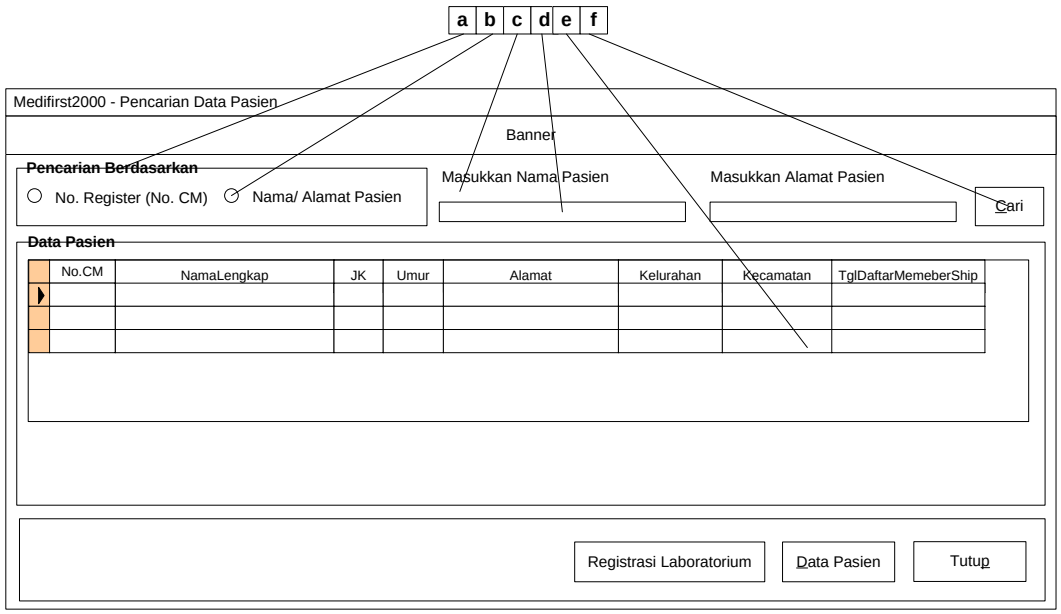
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database. Tombol Pemeriksaan, jika diklik akan tampil form **Pelayanan Tindakan** seperti yang sudah dibahas di atas. Tombol Lanjutkan, jika diklik akan tampil form **Transaksi Pelayanan Pasien** seperti yang sudah dibahas di atas. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

✓ **Form Cari Data Pasien**

No Layar : PA-16

Nama Layar : Form Pencarian Data Pasien

Deskripsi : Tampilan antarmuka yang digunakan untuk menampilkan data pasien yang dicari



Gambar 4.38 Antarmuka Form Pencarian Data Pasien

Pasien

Algoritma :

Nama Operasi : CariDataPasien

Declare

```
    NoCM          NamaPasien
    JK            Umur
    Alamat        Kelurahan
    Kecamatan     TglDaftarMemberShip

If DataPasien sudah ada dlm database Then
  Data-data akan ditampilkan
  Select * From [Pasien]
    Else {DataPasien blm ada}
      Tidak akan ada yang ditampilkan
    Endif
Endif
```

Keterangan :

Form di atas digunakan untuk mencari data pasien yang akan melakukan registrasi. Jika pencarian berdasarkan No. Registrasi (No. CM) Anda akan diminta untuk menginputkan No. Registrasi atau No. CM dari pasein yang akan dicari, tetapi jika pencarian berdasarkan Nama/ Alamat Pasien Anda akan diminta untuk menginputkan Nama Pasien dan atau Alamat Pasien yang dicari seperti gambar di atas.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *RadioButton*, digunakan untuk memilih salah satu pilihan yang sudah ada

c. *Label*, digunakan untuk memberikan keterangan kepada item-item

d. *TextBox*, tempat untuk menginputkan data pasien yang akan dicari

e. *DataGrid*, tempat untuk menampilkan data pasien yang dicari

f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Cari, jika diklik akan menampilkan data pasein yang dicari. Tombol Registrasi Laboratorium, jika diklik maka akan tampil form **Registrasi Laboratorium** seperti **gambar PA-** yang sudah dibahas di atas. Tombol Data pasien, digunakan untuk memperbaiki data pasien jika diklik akan tampil **Form Pasien Baru** seperti yang sudah dibahas di atas.

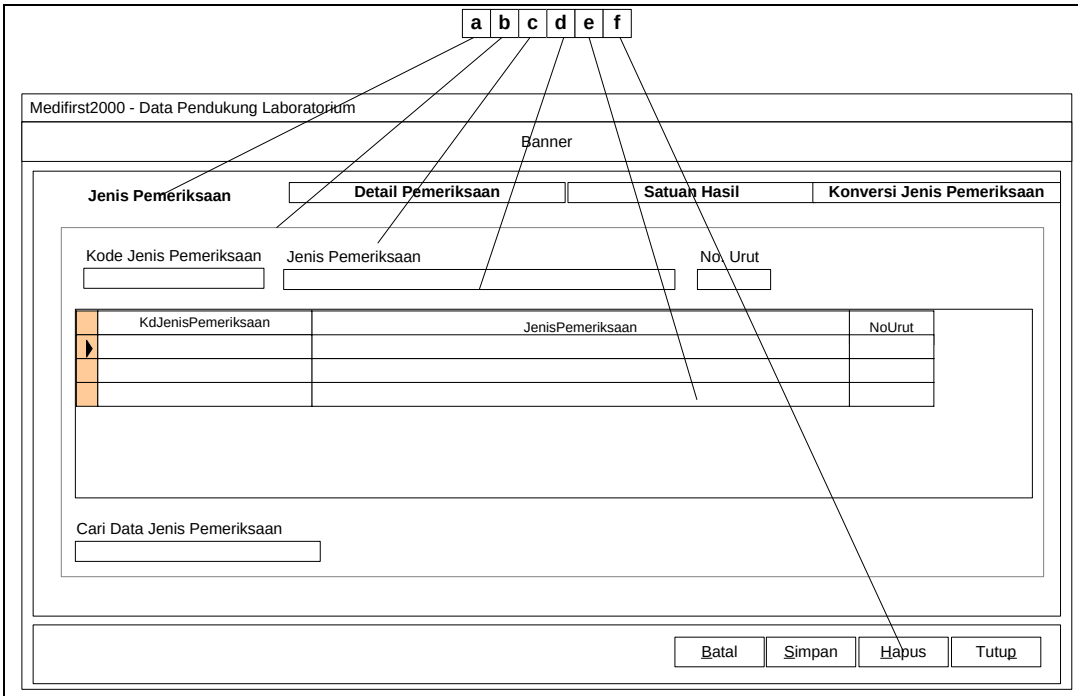
✓ **Form Master Pemeriksaan Laboratorium**

- Tab Jenis Pemeriksaan

No Layar : PA-17

Nama Layar : Form Data Pendukung Laboratorium

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pandukung pemeriksaan laboratorium



Gambar 4.39 Antarmuka Form Data Pendukung Laboratorium

JenisPemeriksaan

Algoritma :

Nama Operasi : SimpanJenisPemeriksaan

Declare

KdJenisPeriksa JenisPeriksa

NoUrut

If JenisPeriksa sudah ada Then

Akan tampil pesan Data Sudah Ada

Else {JenisPeriksa blm ada}

Data yang dimasukkan akan disimpan

```
        Insert INTO [JenisPemeriksaan]
      Endif
Endif


   Nama Operasi         : HapusJenisPemeriksaan
   Declare
      KdJenisPeriksa        JenisPeriksa
      NoUrut


If JenisPeriksa sudah dipilih Then
   Data yang dipilih akan dihapus
   Delete From [JenisPemeriksaan]
   Else {JenisPeriksa blm dipilih}
      Akan tampil pesan Data yang akan dihapus Belum Dipilih
     Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menginputkan data jenis pemeriksaan yang ada dilaboratorium. Kode Jenis Pemeriksaan dalam keadaan tidak aktif dan akan terisi setelah menekan tombol Simpan.

a. *SsTab,* digunakan untuk membuat 4 form pengisian data dalam satu tampilan form

b. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

c. *Label*, digunakan untuk memberikan keterangan kepada item-item

d. *TextBox*, tempat untuk memasukkan data-data yang akan diinputkan

e. *DataGrid*, tempat untuk menampilkan data jenis Pemeriksaan yang sudah disimpan

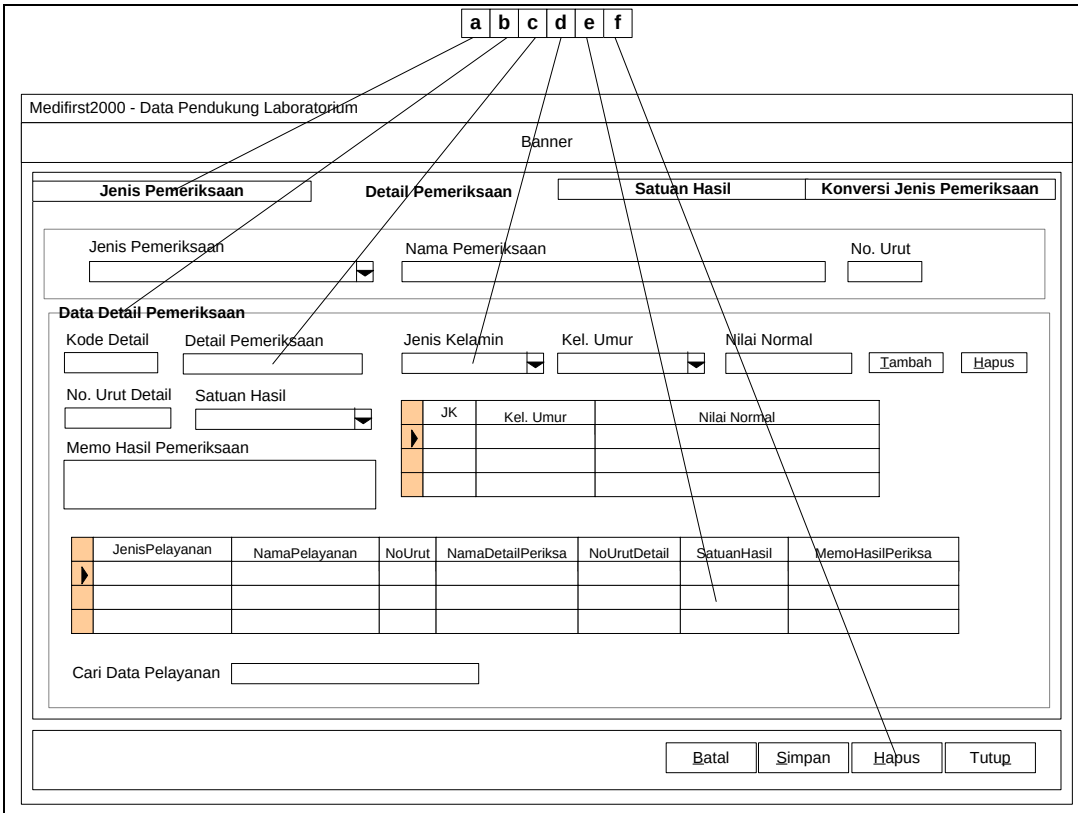
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Batal, jika diklik filed-field akan kembali kosong. Tombol Simpan, jika diklik data yang diinputkan akan disimpan ke database dan ditampilkan di DataGrid. Tombol Hapus, jika diklik data yang sudah dipilih akan dihapus. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

- Tab Detail Pemeriksaan

  No Layar : PA-18

  Nama Layar : Form Data Pendukung Laboratorium

  Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data detail pemeriksaan



Gambar 4.40 Antarmuka Form Data Pendukung Laboratorium

DetailPemeriksaan,ListPelayananRS,ListPemeriksaan,

SatuanHasil,KadarNormalDetailPemeriksaanPK,KelompokUmur

Algoritma :

Nama Operasi : SimpanDetailPemeriksaan

Declare

JenisPemeriksaan NamaPemeriksaan

NoUrut KodeDetail

DetailPemeriksaan NoUrutDetail

SatuanHasil MemoHasilPemeriksaan

JenisKelamin KelUmur

NilaiNormal

If JenisPemeriksaan sdh diConvert ke JenisPelayanan Then

Masukkan data-data DetailPemeriksaan nya

Insert INTO [DetailPemeriksaan],[ListPelayananRS],

[ListPemeriksaan],[SatuanHasil],

[KadarNormalDetailPemeriksaanPK,KelompokUmur]

Else {blm di convert}

Tidak bisa menginputkan data sesuai dengan jenis pemeriksaan nya

Endif

Endif

Nama Operasi : HapusJenisPemeriksaan

Declare

JenisPemeriksaan NamaPemeriksaan

NoUrut KodeDetail

DetailPemeriksaan NoUrutDetail

SatuanHasil MemoHasilPemeriksaan

JenisKelamin KelUmur

NilaiNormal

If DataDetailPemeriksaan sudah dipilih Then

```
Data yang dipilih akan dihapus
Delete From [DetailPemeriksaan],[ListPelayananRS],
    [ListPemeriksaan],[SatuanHasil],
    [KadarNormalDetailPemeriksaanPK,KelompokUmur]
Else {DataDetailPemeriksaan blm dipilih}
  Akan tampil pesan Data yang akan dihapus Belum Dipilih
  Endif
Endif
```

Keterangan :

Form di atas digunakan utnuk menginputkan data detail pemeriksaan. Kode Detail dalam keadaan tidak aktif dan akan otomatis terisi setelah menekan tombol Simpan.

a. *SsTab*, digunakan untuk membuat 4 form pengisian data dalam satu tampilan form

b. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

c. *TextBox*, tempat untuk memasukkan data-data yang akan diinputkan

d. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

e. *DataGrid*, tempat untuk menampilkan data Detail Pemeriksaan yang sudah disimpan

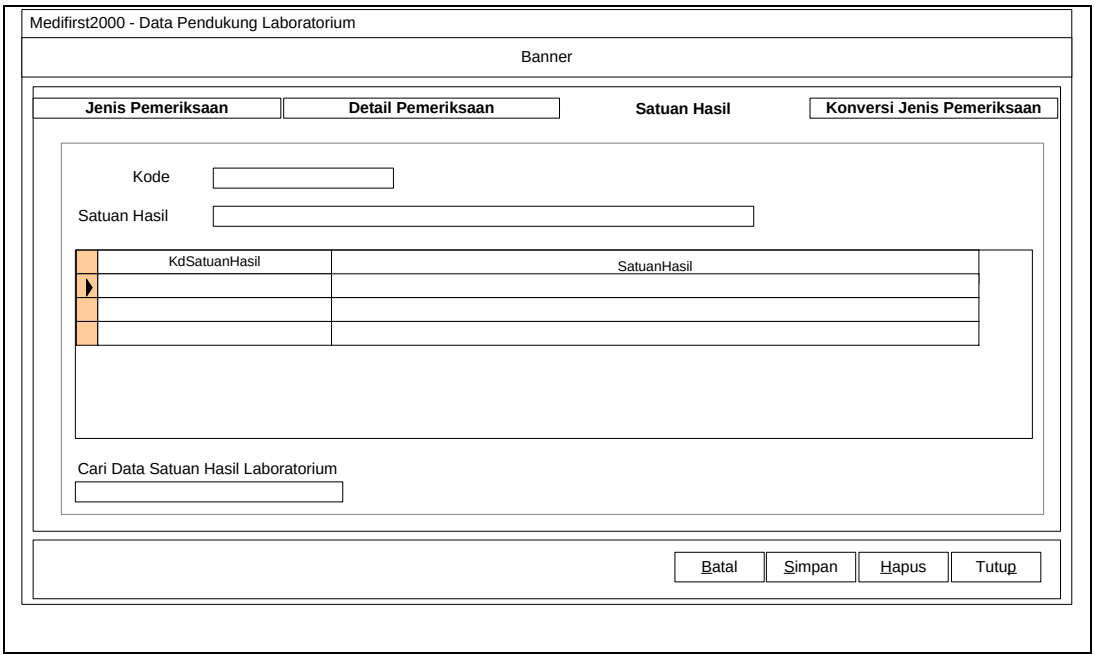
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Batal, jika diklik filed-field akan kembali kosong. Tombol Simpan, jika diklik data yang diinputkan akan disimpan ke database dan ditampilkan di DataGrid. Tombol Hapus, jika diklik data yang sudah dipilih akan dihapus. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

- Tab Satuan Hasil

  No Layar : PA-19

  Nama Layar : Form Data Pendukung Laboratorium

  Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data satuan hasil



Gambar 4.41 Antarmuka Form Data Pendukung Laboratorium

SatauanHasil

Algoritma :

Nama Operasi : SimpanSatuanHasil

Declare

KdSatuanHasil SatuanHasil

If SatuanHasil tidak di isi Then

Akan tampil pesan Nama Satuan Kosong

Else {SatuanHasil di isi}

Data yang dimasukkan akan disimpan

Insert INTO [SatuanHasil]

Endif

Endif

Nama Operasi : HapusSatuanHasil

Declare

KdSatuanHasil SatuanHasil

If SatuanHasil sudah dipilih Then

Data yang dipilih akan dihapus

Delete From [SatuanHasil]

Else {SatuanHasil blm dipilih}

Akan tampil pesan Data yang akan dihapus Belum Dipilih Endif Endif

Keterangan :

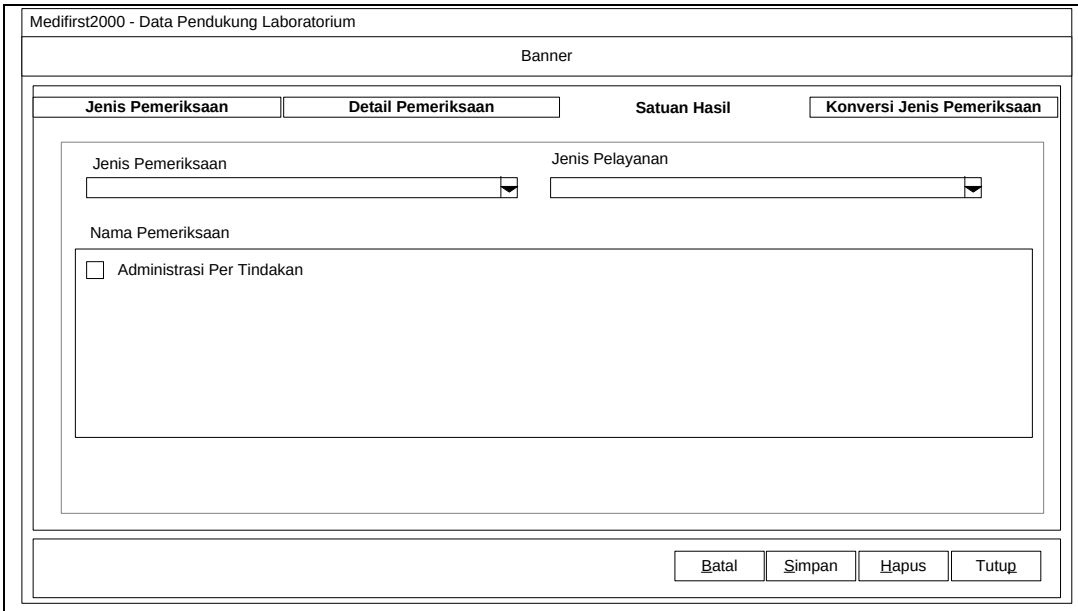
Form di atas digunakan untuk menginputkan data Satuan Hasil. Kode dalam keadaan tidak aktif dan akan otomatis terisi setelah menekan tombol Simpan.

- Tab Konversi Jenis Pemeriksaan

  No Layar : PA-20

  Nama Layar : Form Data Pendukung Laboratorium

  Deskripsi : Tampilan antarmuka yang digunakan untuk menyimpan data konveri jenis pemeriksaan ke pelayanan



Gambar 4.42 Antarmuka Form Data Pendukung Laboratorium

ConvertJenisPemeriksaanToPelayanan,JenisPemeriksaa,

ListPelayananRS

Algoritma :

Nama Operasi : SimpanConvertJenisPemeriksaanToPelayanan

Declare

JenisPeriksa NamaPelayanan

If JenisPeriksa atau NamaPelayanan tidak di isi Then

Akan tampil pesan JenisPemeriksaan atau NamaPelayanan

Kosong

Else { JenisPeriksa atau NamaPelayanan di isi}

Data yang dimasukkan akan disimpan

```
        Insert INTO [ConvertJenisPemeriksaanToPelayanan]
      Endif
Endif
```

Keterangan :

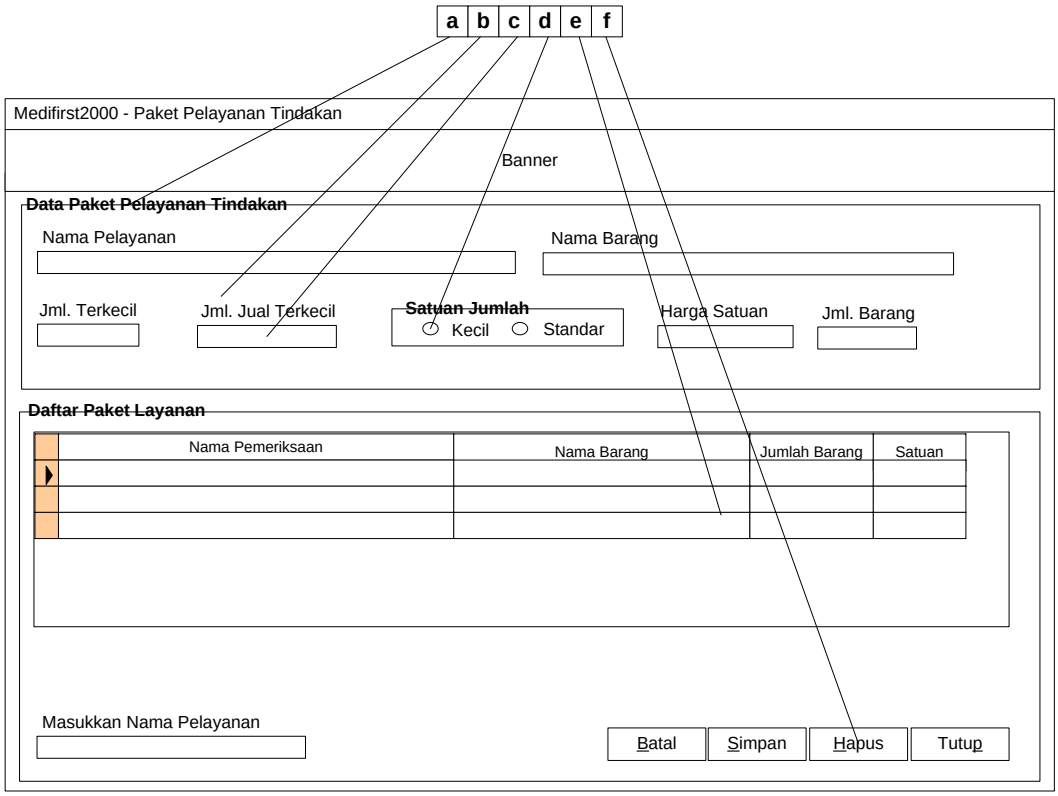
Form di atas digunakan untuk menyimpan data konversi jenis pemeriksaan ke jenis pelayanan.

✓ **Form Paket Pelayanan Tindakan**

No Layar : PA-21

Nama Layar : Form Paket Pelayanan Tindakan

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan daftar paket pelayanan tindakan



Gambar 4.43 Antarmuka Form Paket Pelayanan Tindakan

PaketLayanan,ListPelayananRS,MasterBarang,HargaBarang,

Satuan

Algoritma :

Nama Operasi : SimpanPaketPelayanan

Declare

NamaPelayanan NamaBarang

JmlTerkecil JmlJualTerkecil

Satuan HargaSatuan

```
      JmlBarang


If NamaPelayanan atau NamaBarang tidak di isi Then
  Akan tampil pesan NamaPelayanan atau NamaBarang
  Harus di Isi
  Else { NamaPelayanan atau NamaBarang di isi}
      Data yang dimasukkan akan disimpan
      Insert INTO [PaketLayanan]
    Endif
Endif


  Nama Operasi        : HapusPaketPelayanan
  Declare
    NamaPelayanan      NamaBarang
    JmlTerkecil        JmlJualTerkecil
    Satuan             HargaSatuan
    JmlBarang


If DataPaketLayanan sudah dipilih Then
  Data yang dipilih akan dihapus
  Delete From [PaketLayanan]
  Else { DataPaketLayanan blm dipilih}
     Akan tampil pesan Data yang akan dihapus Belum Dipilih
    Endif
```

```
Endif
```

Keterangan :

Form di atas digunakan untuk menginputkan data paket pelayanan tindakan dan dapat melakukan pencarian dengan memasukkan Nama Pelayanan. Jml. Terkecil, Jml. Jual Terkecil, Satuan Jumlah, dan Harga Satuan dalam keadaan tidak aktif dan akan terisi otomatis sesuai dengan Nama Barang yang sudah dipilih.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

c. *TextBox*, tempat untuk menginputkan data dan menampilkan data yang sesuai dengan data yang sudah dipilih

d. *RadioButton*, digunakan untuk menunjukkan salah satu pilihan

e. *DataGrid*, tempat untuk menampilkan data paket pelayanan yang diinputkan dan menampilkan data paket pelayanan yang dicari

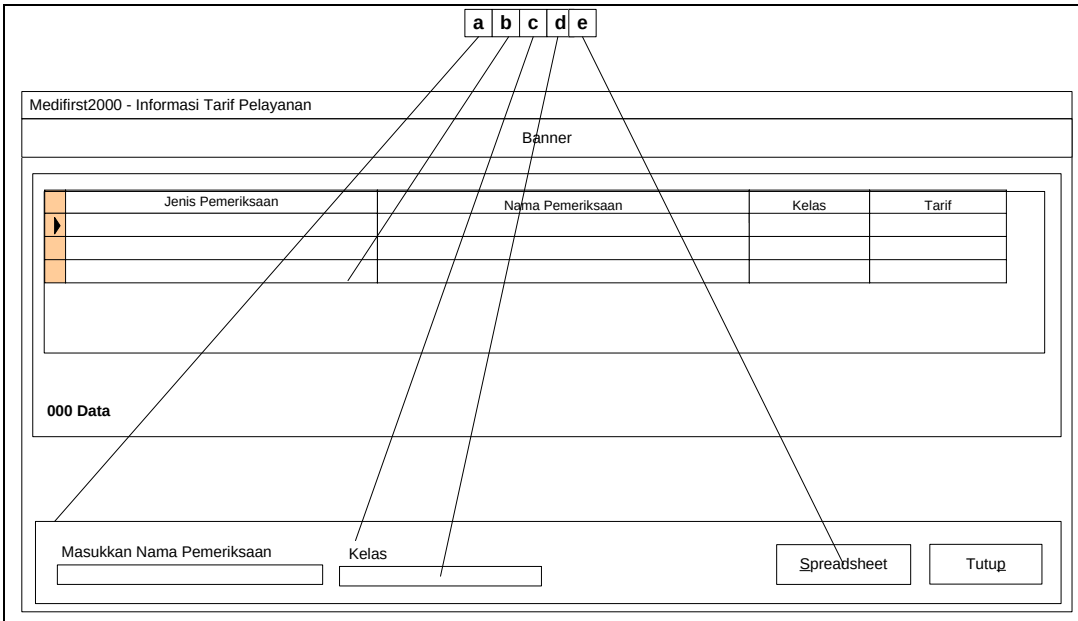
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan disimpan ke database. Tombol Baru, jika diklik filed-field yang sudah terisi akan menjadi kosong. Tombol Hapus, jika diklik akan menghapus data yang sudah dipilih. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

✓ **Form Informasi Tarif Pelayanan**

No Layar : PA-22

Nama Layar : Form Informasi Tarif Pelayanan

Deskripsi : Tampilan antarmuka yang digunakan untuk pencarian data informasi tarif pelayanan



Gambar 4.44 Antarmuka Informasi Tarif Pelayanan

TarifPelayanan,JenisPelayanan,ListPelayananRS,

JenisTarif,KelasPelayanan

Algoritma :

Nama Operasi : TampilTarifPelayanan

Declare

```
    JenisPelayanan        NamaPelayanan

    Kelas                 Tarif


If DataTarifPelayanan ada Then
  Data yang sudah ada dapat dicetak
  Select * From [TarifPelayanan],[JenisPelayanan],
        [ListPelayananRS],[jenisTarif],&
        [KelasPelayanan]
  Else { DataTarifPelayanan tidak ada }
     Data tidak akan ada yang dicetak
    Endif
Endif
```

Keterangan :

Form diatas digunakan untuk menampilkan data informasi tarif pelayanan berdasarkan Nama Pemeriksaan dan Kelas. Spreadsheet akan digunakan untuk mencetak Informasi Tarif Pelayanan berdasarkan Jenis Pemeriksaan dan Nama Pemeriksaan.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *DataGrid*, tempat untuk menampilkan data informasi tarif pelayanan

c. *Label*, digunakan untuk memberikan keterangan kepada item-item

d. *TextBox*, tempat untuk menginputkan data yang dicari

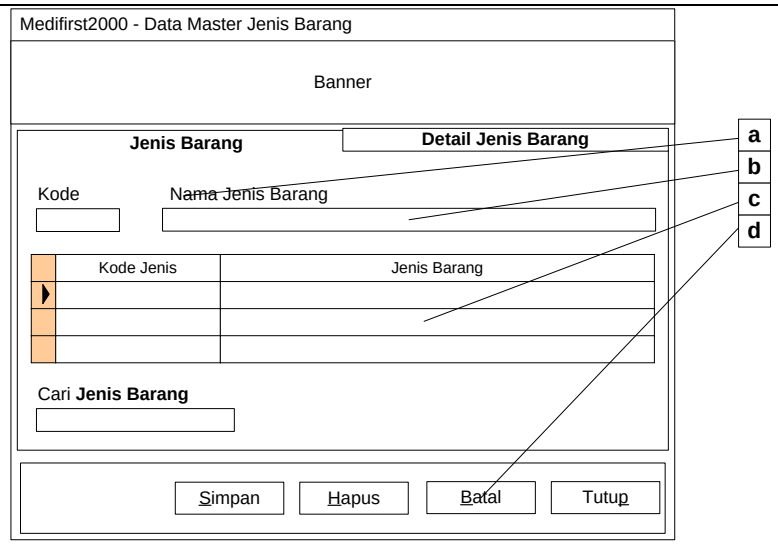
e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Spreadsheet, jika diklik data yang ada siap untuk dicetak. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

✓ **Form Data Master Jenis Barang**

No Layar : PA-23

Nama Layar : Form Data Master Jenis Barang

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data jenis barang dan detail jenis barang



Gambar 4.45 Antarmuka Form Data Master Jenis Barang

Keterangan :

Form yang digunakan untuk menginputkan data master jenis barang. Kode Jenis dan Kode Detail pada awalnya tidak aktif, kode akan otomatis terisi jika menekan tombol Simpan.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *TextBox*, tempat untuk memasukkan nilai pada item Jenis Barang dan Detail Jenis Barang, juga digunakan untuk memasukkan jenis barang atau detail jenis barang yang akan dicari

c. *DataGrid*, tempat untuk menginputkan data master jenis barang atau detail jenis barang yang akan disimpan ke *database*

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

e. *DataCombo*, jika di klik akan menampilkan data-data Detail Jenis Barang yang sudah ada di dalam *database*

JenisBarang

Algoritma :

Nama Operasi : TambahDataMasterMerkBarang

Declare

KdJenisBarang

Jenis Barang

Open koneksi

IF koneksi berhasil THEN

pilih database

IF tekan tombol **Simpan** THEN

IF nama jenis barang belum ada THEN

Masukkan data-data nya

Insert Into [JenisBarang]

Tampil pesan “Penyimpanan Jenis Barang Berhasil”

ELSE nama jenis barang sudah ada THEN

Tampil pesan “Nama Jenis yang Anda masukkan sudah Ada”

ELSE

ELSE

IF koneksi berhasil THEN

pilih database

IF tekan tombol **Hapus** THEN

```
        IF nama jenis barang sudah dipilih THEN
          Delete [JenisBarang] Where KdJenisBarang=@KdJenisBarang
        ELSE {nama jenis barang sudah ada}
          Tampil Pesan “Pilih Data Yang Akan Di Hapus”
       ELSE
      ELSE

close_koneksi
```

```
      DetailJenisBarang
      Algoritma :
      Nama Operasi          : TambahDataMasterTypeBarang
      Declare
        KdDetailJenisBarang
        DetailJenisBarang
        KdJenisBarang

      Open koneksi
      IF koneksi berhasil THEN
          pilih database
       IF tekan tombol Simpan THEN
        IF nama detail jenis barang belum ada THEN
          Masukkan data-data nya
```

```
            Insert Into [DetailJenisBarang]
            Tampil pesan “Penyimpanan Detail Jenis Barang Berhasil”
          ELSE nama detail jenis barang & jenis barang sudah ada THEN
             Tampil pesan “Nama Detail Jenis yang Anda masukkan
                        sudah Ada”
       ELSE


       IF koneksi berhasil THEN
            pilih database
        IF tekan tombol Hapus THEN
          IF nama detail jenis barang sudah dipilih THEN
            Delete [DetailJenisBarang] Where KdDetailJenisBarang=
              @KdDetailJenisBarang
          ELSE {nama detail jenis barang harus dipilih}
            Tampil Pesan “Pilih Data Yang Akan Di Hapus”
        ELSE
    ELSE

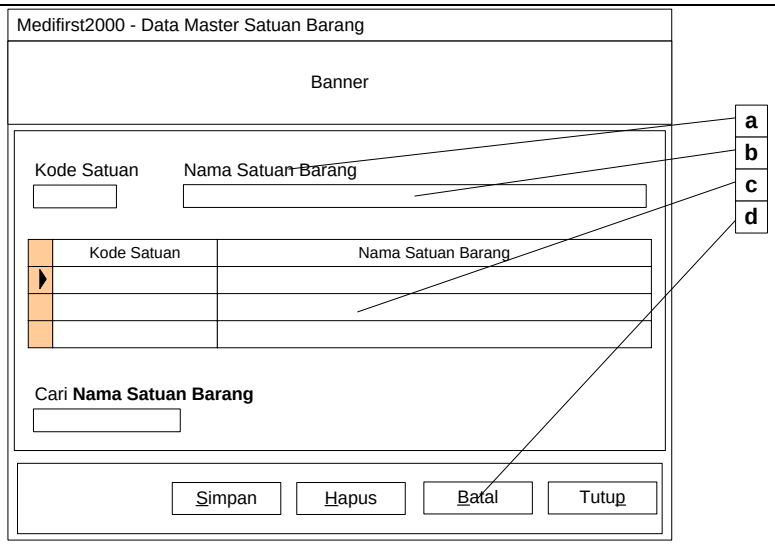

close_koneksi
```

✓ **Form Data Master Satuan Barang**

No Layar : PA-24

Nama Layar : Form Data Master Satuan Barang

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data satuan barang



Gambar 4.46 Antarmuka Form Data Master Satuan Barang

SatuanJmlBarang

Algoritma :

Nama Operasi : TambahDataMasterSatuanBarang

Declare

KdSatuan

Satuan

Open koneksi

IF koneksi berhasil THEN

```
        pilih database
      IF tekan tombol Simpan THEN
       IF nama satuan barang belum ada THEN
         Masukkan data-data nya
         Insert Into [SatuanBarang]
         Tampil pesan “Penyimpanan Satuan Barang Berhasil”
       ELSE nama satuan barang sudah ada THEN
          Tampil pesan “Nama Satuan Barang yang Anda masukkan
                sudah Ada”
      ELSE
     ELSE


     IF koneksi berhasil THEN
         pilih database
      IF tekan tombol Hapus THEN
       IF nama satuan barang sudah dipilih THEN
         Delete [SatuanBarang] Where KdStuan=@KdSatuan
       ELSE {nama satuan barang harus dipilih}
         Tampil Pesan “Pilih Data Yang Akan Di Hapus”
      ELSE
     ELSE

close_koneksi
```

Keterangan :

Form yang digunakan untuk menginputkan data master satuan barang. Kode Satuan pada awalnya tidak aktif, kode akan otomatis terisi jika menekan tombol Simpan.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *TextBox*, tempat untuk memasukkan nilai pada item Nama Satuan Barang dan digunakan untuk memasukkan nama satuan barang yang akan dicari

c. *DataGrid*, tempat untuk menginputkan data master satuan barang yang akan disimpan ke *database*

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

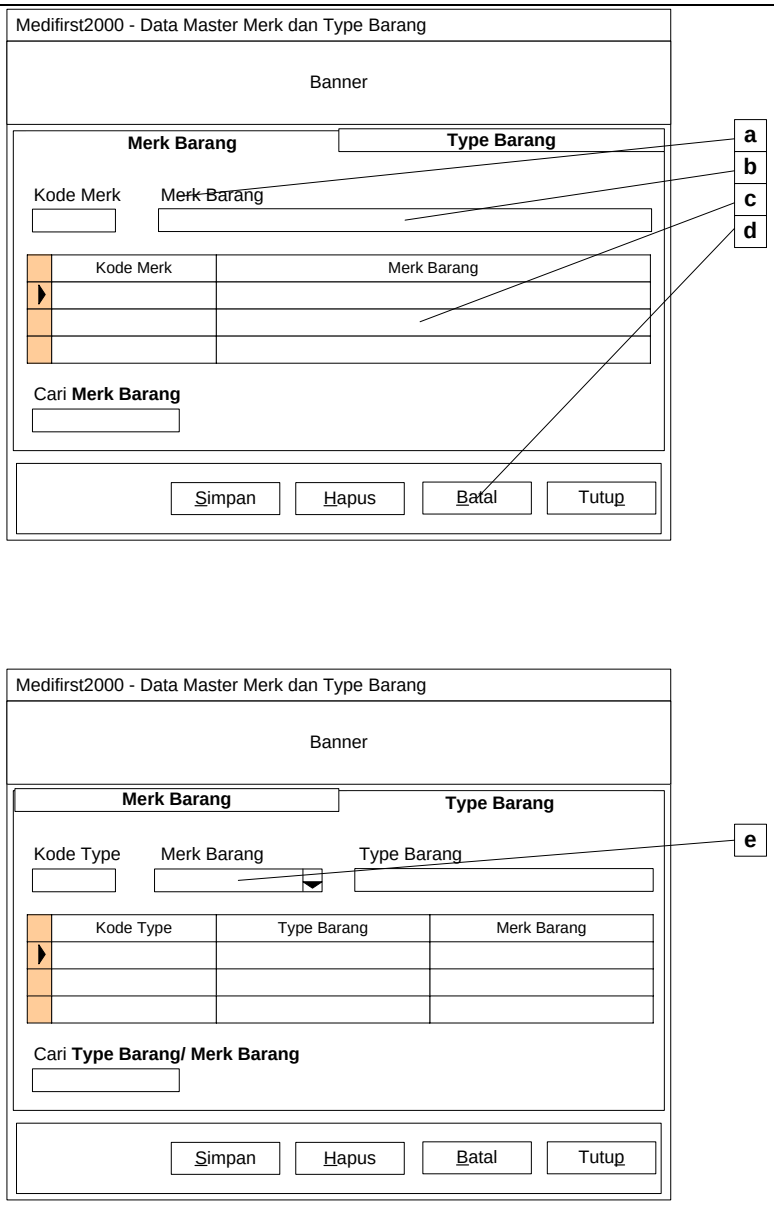
✓ **Form Data Master Merk dan Type Barang**

No Layar : PA-25

Nama Layar : Form Terima Barang Langsung

Deskripsi : Tampilan antarmuka yang digunakan untuk

menginputkan data merk dan type barang



Gambar 4.47 Antarmuka Form Data Master Merk dan Type Barang

MerkBarang

Algoritma :

Nama Operasi : TambahDataMasterMerkBarang

Declare

KdMerk

NamaMerk

Open koneksi

IF koneksi berhasil THEN

pilih database

IF tekan tombol **Simpan** THEN

IF nama merk barang belum ada THEN

Masukkan data-data nya

Insert Into [MerkBarang]

Tampil pesan “Penyimpanan Merk Barang Berhasil”

ELSE nama merk barang sudah ada THEN

Tampil pesan “Nama Merk Barang yang Anda masukkan

sudah Ada”

ELSE

ELSE

IF koneksi berhasil THEN

pilih database

IF tekan tombol **Hapus** THEN

IF nama merk barang sudah dipilih THEN

```
            Delete [MerkBarang] Where KdMerk=@KdMerk
          ELSE {nama merk barang harus dipilih}
            Tampil Pesan “Pilih Data Yang Akan Di Hapus”
        ELSE
      ELSE

close_koneksi
```

```
      TypeBarang
      Algoritma :
      Nama Operasi          : TambahDataMasterTypeBarang
      Declare
        Kdtype
        NamaType
        KdMerk

      Open koneksi
      IF koneksi berhasil THEN
          pilih database
       IF tekan tombol Simpan THEN
        IF nama type barang belum ada THEN
          Masukkan data-data nya
          Insert Into [TypeBarang]
```

```
            Tampil pesan “Penyimpanan Type Barang Berhasil”
          ELSE nama type barang AND merk barang sudah ada THEN
              Tampil pesan “Nama Type Barang yang Anda masukkan
                        sudah Ada”
        ELSE
       ELSE


       IF koneksi berhasil THEN
           pilih database
        IF tekan tombol Hapus THEN
          IF nama type barang sudah dipilih THEN
            Delete [TypeBarang] Where KdType=@KdType
          ELSE {nama type barang harus dipilih}
            Tampil Pesan “Pilih Data Yang Akan Di Hapus”
         ELSE
       ELSE
close_koneksi
```

Keterangan :

Form yang digunakan untuk menginputkan data master merk dan type barang. Kode Merk dan Kode Type pada awalnya tidak aktif, kode akan otomatis terisi jika menekan tombol Simpan.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *TextBox*, tempat untuk memasukkan nilai pada item Merk Barang dan type Brang, juga digunakan untuk memasukkan merk barang atau type barang yang akan dicari

c. *DataGrid*, tempat untuk menginputkan data master merk barang atau type barang yang akan disimpan ke *database*

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

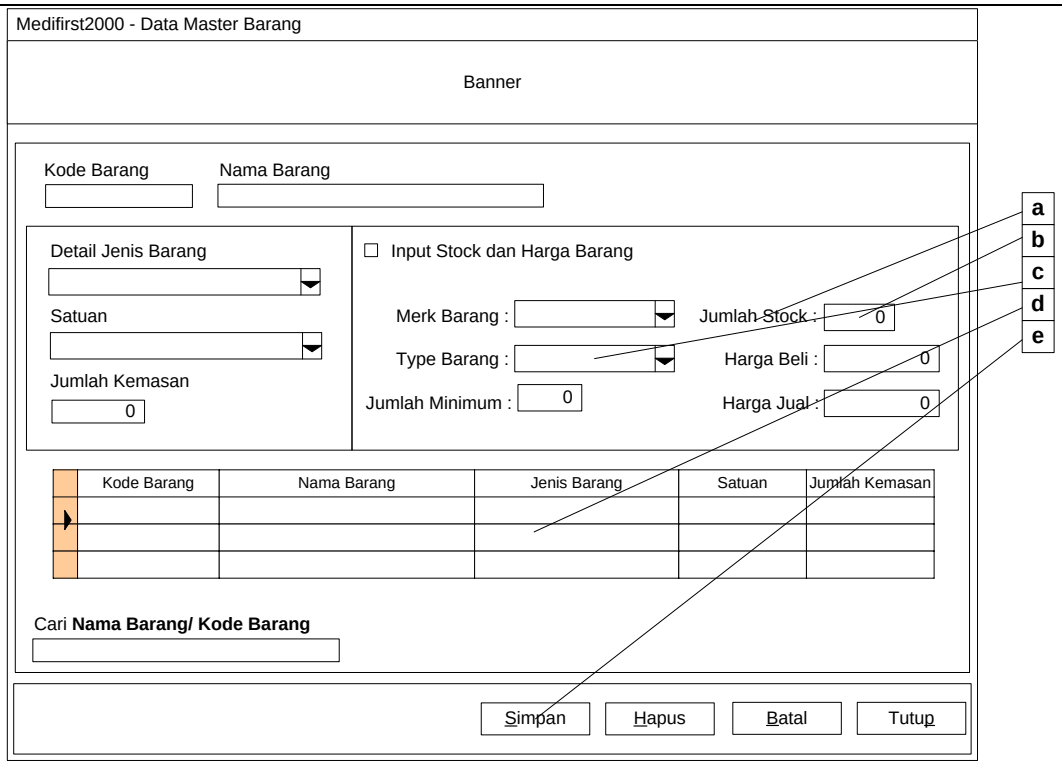
e. *DataCombo*, jika di klik akan menampilkan data-data Merk Barang yang sudah ada di dalam *database*

✓ **Form Master Barang**

No Layar : PA-26

Nama Layar : Form Master Barang

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data master barang



Gambar 4.48 Antarmuka Form Master Barang

MasterBarang,StockBarang

Algoritma :

Nama Operasi : TambahDataMasterBarang

Declare

KdBarang NamaBarang

KdDetailJenisBarang KdSatuan

JmlKemasan KdMerk

KdType Harga Beli

Harga Jual JmlMinimum

JmlStock

```
Open koneksi
IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Simpan THEN
   IF nama barang belum ada THEN
     Masukkan data-data nya
     Insert Into [MasterBarang] & [Stockbarang]
     Tampil pesan “Penyimpanan MAster Barang Berhasil”
   ELSE
     IF nama barang sudah ada THEN
       IF detail jenis barang & satuan sama THEN
         Tampil pesan “Nama Master Barang yang Anda masukkan
                  sudah Ada”
       ELSE {detail & satuan berbeda}
         Masukkan data-data nya
         Insert Into [MasterBarang] & [Stockbarang]
         Tampil pesan “Penyimpanan MAster Barang Berhasil”
     ELSE
 ELSE
ELSE
IF koneksi berhasil THEN
    pilih database
```

```
IF tekan tombol Hapus THEN
  IF nama master barang sudah dipilih THEN
    Delete From [MasterBarang] Where KdBarang=@KdBarang
  ELSE {nama master barang harus dipilih}
    Tampil Pesan "Pilih Data Yang Akan Di Hapus"
 ELSE
ELSE

close_koneksi
```

Keterangan :

Form yang digunakan untuk menginputkan data master barang. Awalnya Kode Barang tidak aktif, kode barang akan tampil setelah menekan tombol Simpan. Isikan data-data barang sesuai dengan item yang diminta.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *TextBox*, tempat untuk memasukkan nilai pada item-item data master barang yang diminta dan untuk menampilkan Kode Barang jika sudah menekan tombol Simpan

c. *DataCombo*, jika di klik akan menampilkan data-data Detail Jenis Barang, Merk Barang, Type Barang, dan Satuan yang data-datanya sudah ada di dalam *database*

d. *DataGrid*, tempat untuk menampilkan data master barang yang baru diinputkan dan yang akan ada di dalam *database*

e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

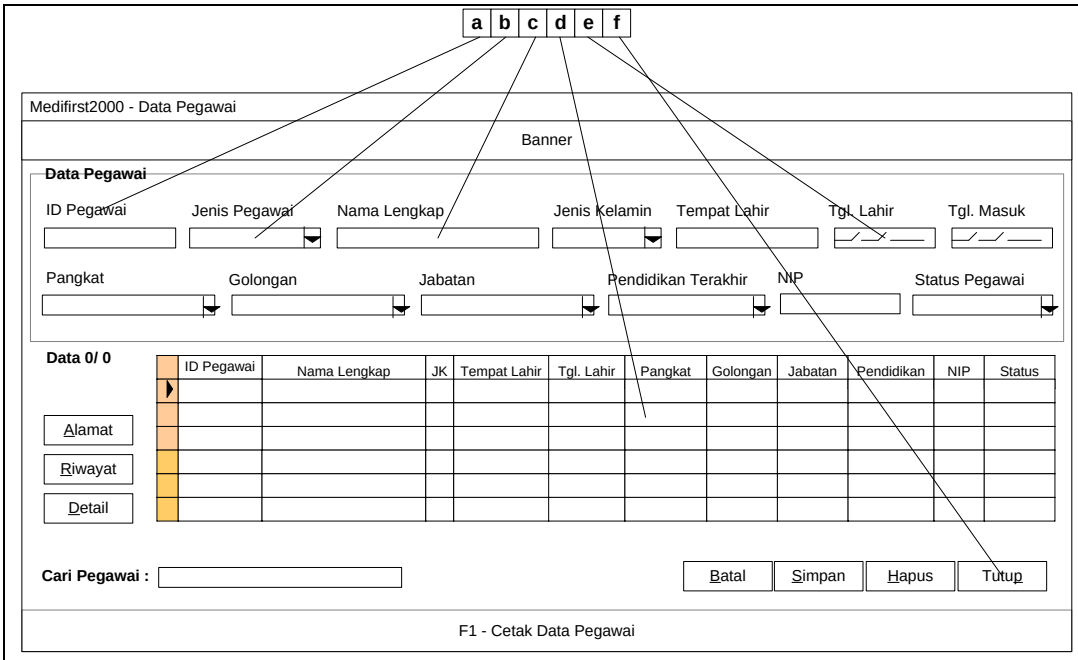
f. *CheckBox*, jika diklik maka data Merk Barang, Type Barang, Jumlah Minimum, Jumlah Stock, Harga Beli dan Harga Jual menjadi aktif dan data-datanya dapat diisi

✓ **Form Data Pegawai**

No Layar : PA-27

Nama Layar : Form Data Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pegawai



Gambar 4.49 Antarmuka Form Data Pegawai

DataPegawai,DataCurrentPegawai

Algoritma :

Nama Operasi : TamabahDataPegawai

Declare

IDPegawai JenisPegawai

NamaLengkap JenisKelamin

TempatLahir Tgl.Lahir

Tgl.Masuk Pangkat

Golongan Jabatan

Pendidikan NIP

StatusPegawai

```
Open koneksi
IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Simpan THEN
   IF salah satu data masih kosong THEN
     Masukkan data-data nya
     Insert Into [DataPegawai]&[DataCurrentPegawai]
     Tampil pesan “Penyimpanan DataPegawai Berhasil”
   ELSE {data sudah lengkap}
     Masukkan data-data nya
      Insert Into [DataPegawai]&[DataCurrentPegawai]
      Tampil pesan “Penyimpanan DataPegawai Berhasil”
ELSE
ELSE
IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Hapus THEN
   IF DataPegawai sudah dipilih THEN
     Delete From [DataPegawai] Where IDPegawai=@IDPegawai
   ELSE {data pegawai belum dipilih}
     Tampil Pesan “Pilih Data Yang Akan Di Hapus”
  ELSE
```

```
    ELSE


close_koneksi
```

Keterangan :

Form di atas digunakan utnuk menginputkan data pegawai. ID Pegawai dalam keadaan tidak aktif dan akan terisai otomatis setelah menekan tombol Simpan.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

c. *TextBox*, tempat untuk memasukkan nilai pada item-item yang akan diinputkan

d. *DataGrid*, tempat untuk menampilkan data pegawai yang baru diinputkan dan yang sudah ada di dalam *database*

e. *MaskEdBox*, tempat untuk menulisakan tanggal yang formatnya DD/ MM/ YYY

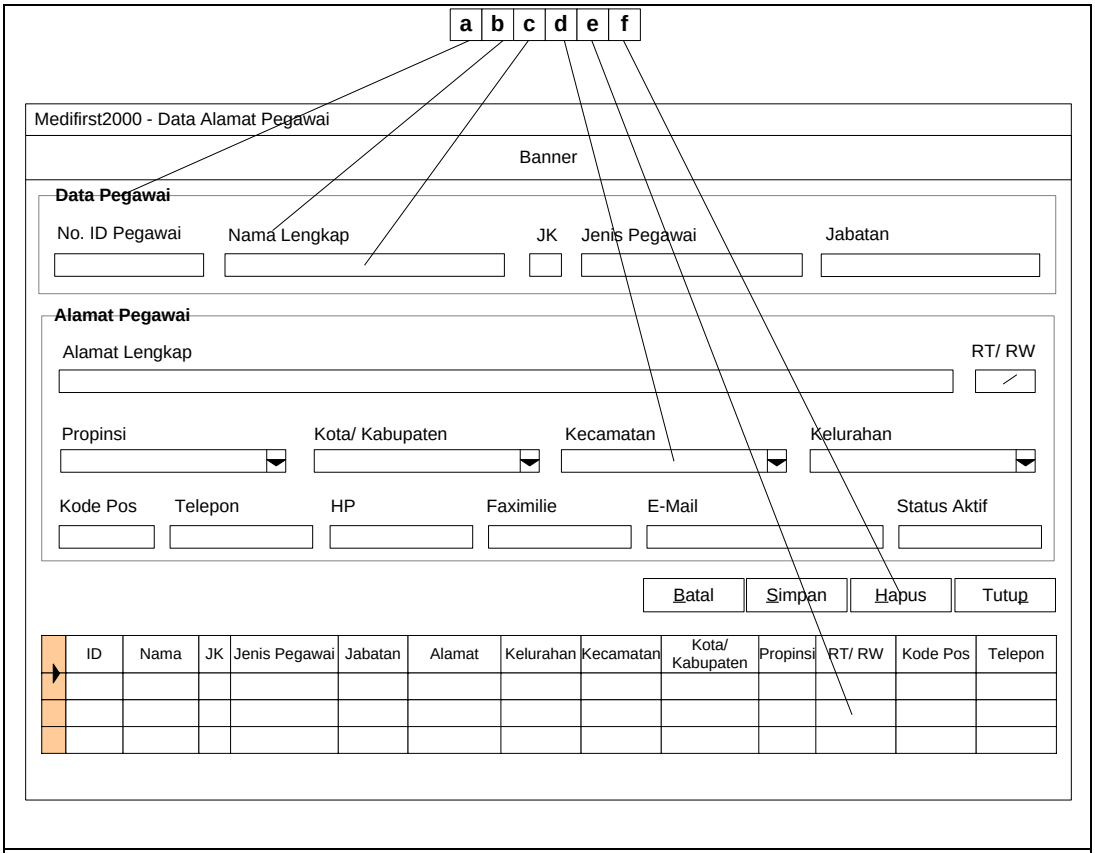
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

- Tombol Alamat

No Layar : PA-28

Nama Layar : Form Data Alamat Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan alamat pegawai



Gambar 4.50 Antarmuka Form Data Alamat Pegawai

DataPegawai,DataCurrentPegawai,Propinsi,KotaKabupaten,

Kecamatan,Kelurahan

Algoritma :

```
Nama Operasi          : InsertAlamatPegawai
Declare
  IDPegawai           JenisPegawai
  NamaLengkap         JenisKelamin
  Jabatan             Alamat
  Propinsi            KotaKabupaten
  Kecamatan           Kelurahan
  RTRW                KodePos
  Telepon             Hp
  Faximilie           E-mail
  StatusAktif

Open koneksi
IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Simpan THEN
  IF salah satu data masih kosong THEN
    ampil pesan “Data Belum Lengkap”
  ELSE {data sudah lengkap}
    Masukkan data-data nya
     Insert Into [DataPegawai],[DataCurrentPegawai],
        [Propinsi],[KotaKabupaten],[Kecamatan],[Kelurahan]
     Tampil pesan “Penyimpanan DataPegawai Berhasil”
```

```
    ELSE
    ELSE
    IF koneksi berhasil THEN
        pilih database
     IF tekan tombol Hapus THEN
      IF AlamatPegawai sudah dipilih THEN
        Delete From [DataPegawai] Where IDPegawai=@IDPegawai
      ELSE {data pegawai belum dipilih}
        Tampil Pesan "Pilih Data Yang Akan Di Hapus"
     ELSE
    ELSE

close_koneksi
```

Keterangan :

Form di atas digunakan utnuk menginputkan data alamat pegawai. Kolom Data Pegawai dalam keadaan tidak aktif dan menampilkan data-data pasien sesuai dengan yang dipilih.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

c. *TextBox*, tempat untuk memasukkan nilai pada item-item yang akan diinputkan

d. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

e. *DataGrid*, tempat untuk menampilkan data alamt pegawai yang diinputkan dan disimpan di dalam *database*

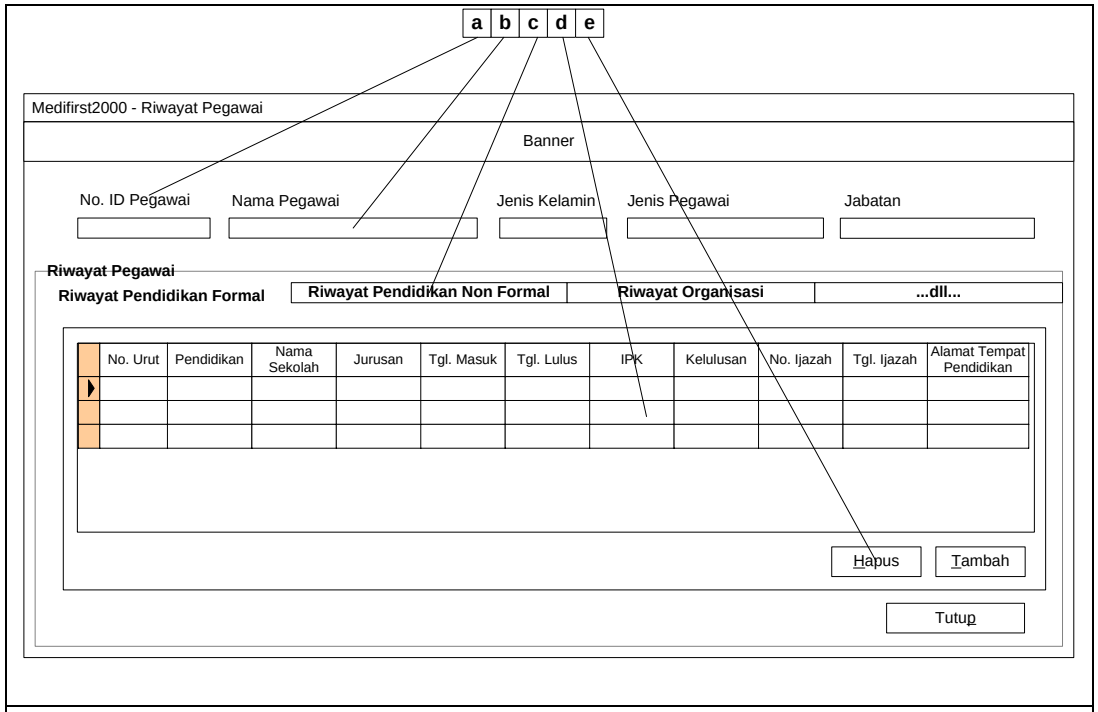
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

- Tombol Riwayat

  No Layar : PA-29

  Nama Layar : Form Riwayat Pegawai

  Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat pegawai



Gambar 4.51 Antarmuka Form Riwayat Pegawai

Keterangan :

Form di atas digunakan untuk menampilkan data-data riwayat pegawai. Pada form di atas No. ID Pegawai, Nama Pegawai, Jenis Kelamin, Jenis Pegawai, dan Jabatan terisi data-data sesuai dengan data pegawai yang dipilih dan dalam keadaan tidak aktif. Form tersebut terdiri dari 14 Tab, yaitu : Tab Riwayat Pendidikan Formal, Riwayat Pendidikan Non Formal, Riwayat Organisasi, Riwayat Perjalanan Dinas, Riwayat Golongan, Riwayat Hukuman, Riwayat Pangkat, Riwayat Extra Pelatihan, Riwayat Prestasi, Riwayat Pekerjaan, Riwayat Gaji, Riwayat Tempat Bertugas, Riwayat Keluarga, dan Riwayat Status.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item

b. *TextBox*, tempat untuk memasukkan data-data yang akan diinputkan

c. *SsTab*, digunakan untuk membuat 14 form pengisian data dalam satu tampilan form

d. *DataGrid*, tempat untuk menampilkan data pelayanan tindakan yang sudah dilakukan oleh pasien

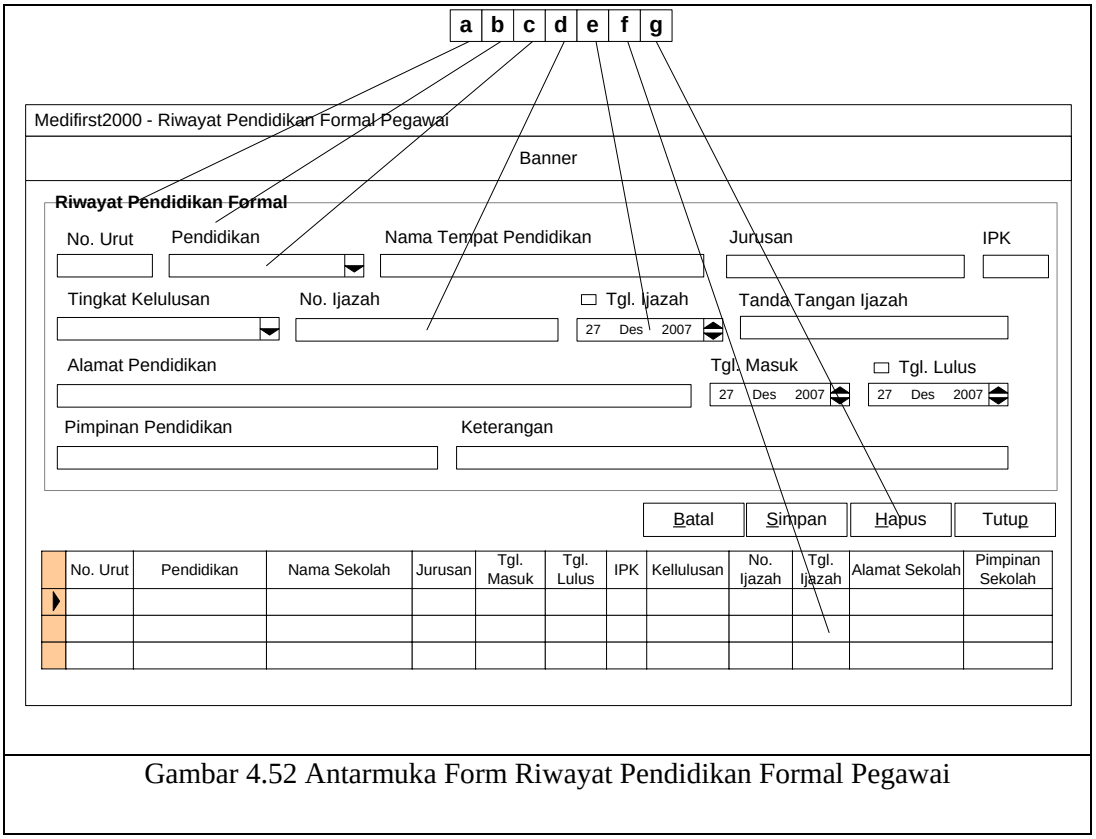
e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Hapus digunakan untuk menghapus data yang sudah ada di DataGrid. Tombol Tambah digunakan untuk menambahkan data riwayat pendidikan formal, maka akan tampil form **Riwayat Pendidikan Formal Pegawai** seperti di bawah ini :

- **Riwayat Pendidikan Formal Pegawai**

No Layar : PA-30

Nama Layar : Form Riwayat Pendidikan Formal Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat pendidikan formal pegawai



Gambar 4.52 Antarmuka Form Riwayat Pendidikan Formal Pegawai

Keterangan :

Form di atas digunakan untuk menginputkan data riwayat pendidikan formal setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. Ijazah dan Tgl. Lulus dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

c. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

d. *TextBox*, tempat untuk memasukkan nilai pada item-item yang akan diinputkan

e. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DDMMYYYY

f. *DataGrid*, tempat untuk menampilkan data riwayat pendidika formal pegawai yang diinputkan dan disimpan di dalam *database*

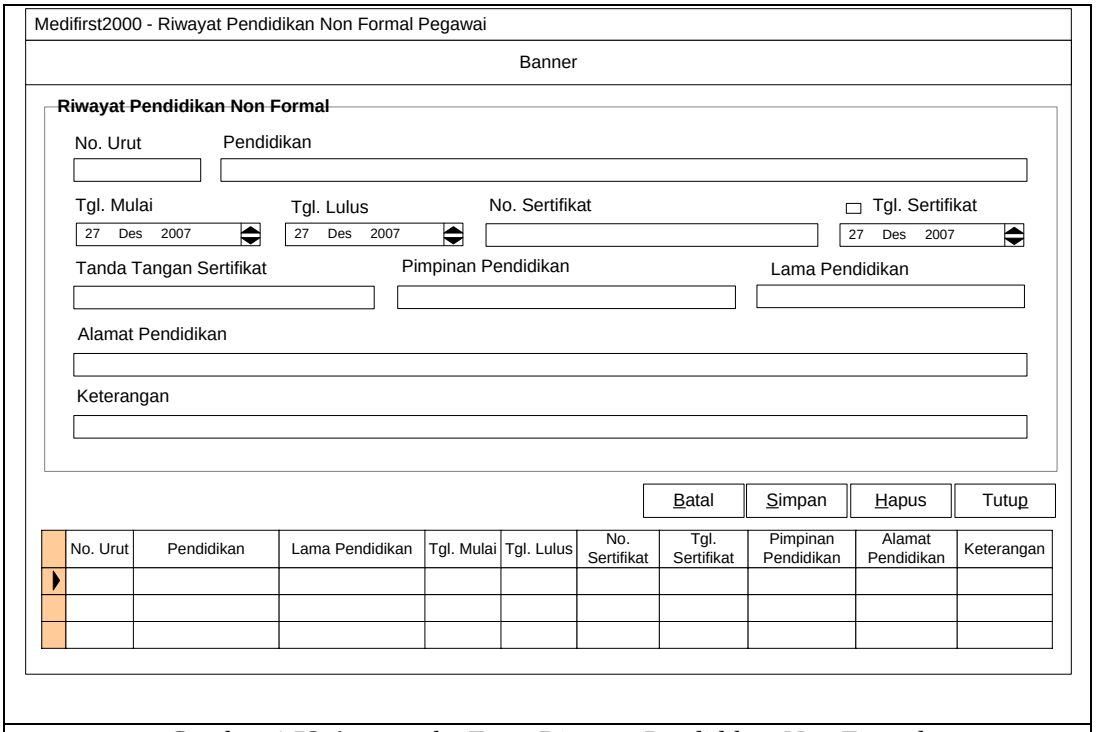
g. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

- **Riwayat Pendidikan Non Formal**

No Layar : PA-31

Nama Layar : Form Riwayat Pendidikan Non Formal

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat pendidikan non formal



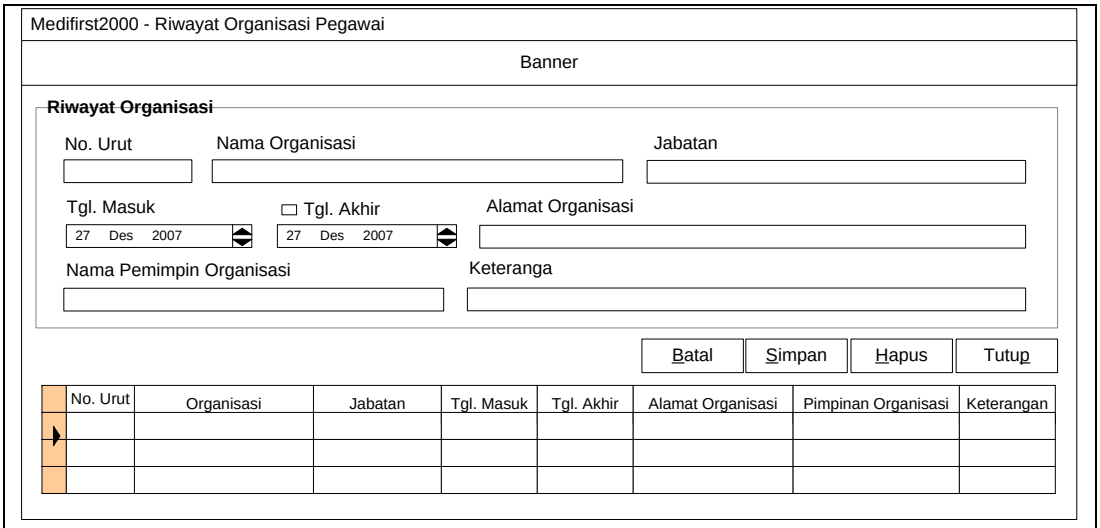
Gambar 4.53 Antarmuka Form Riwayat Pendidikan Non Formal

Keterangan :

Form di atas digunakan untuk menginputkan data riwayat pendidikan non formal setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. Sertifikat dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Organisasi Pegawai**

No Layar	:	PA-32
Nama Layar	:	Form Riwayat Organisasi Pegawai
Deskripsi	:	Tampilan antarmuka yang digunakan untuk menginputkan data riwayat organisasi pegawai



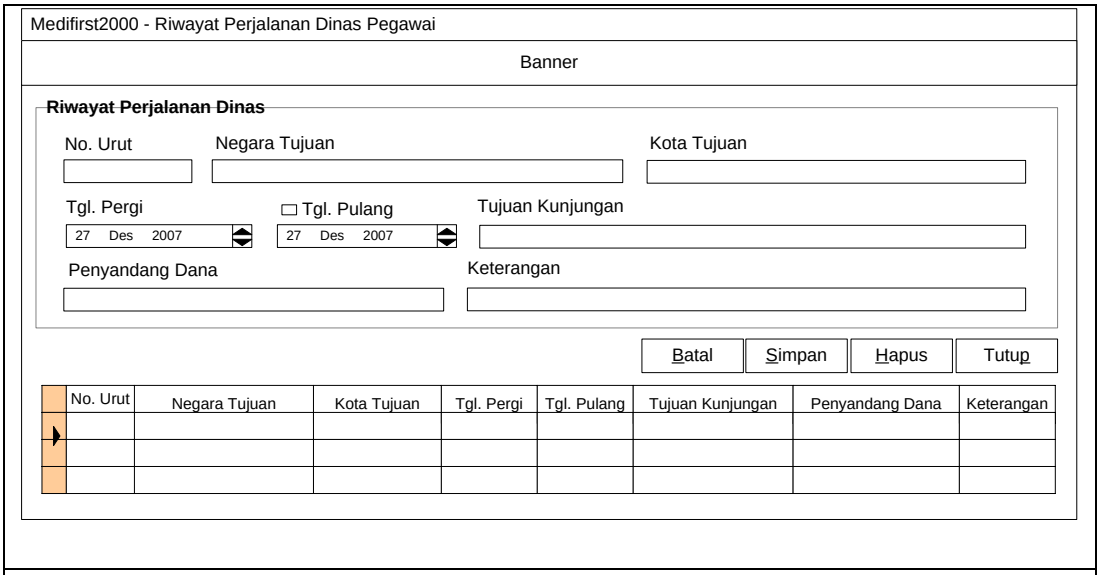
Gambar 4.54 Antarmuka Form Riwayat Organisasi Pegawai

Keterangan :

Form di atas digunakan untuk menginputkan data riwayat organisasi yang pernah dan sedang diikuti oleh setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. Akhir dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Perjalanan Dinas Pegawai**

No Layar	:	PA-33
Nama Layar	:	Form Riwayat Perjalanan Dinas Pegawai
Deskripsi	:	Tampilan antarmuka yang digunakan untuk menginputkan data riwayat perjalanan dinas pegawai



Gambar 4.55 Antarmuka Form Riwayat Perjalanan Dinas Pegawai

Keterangan :

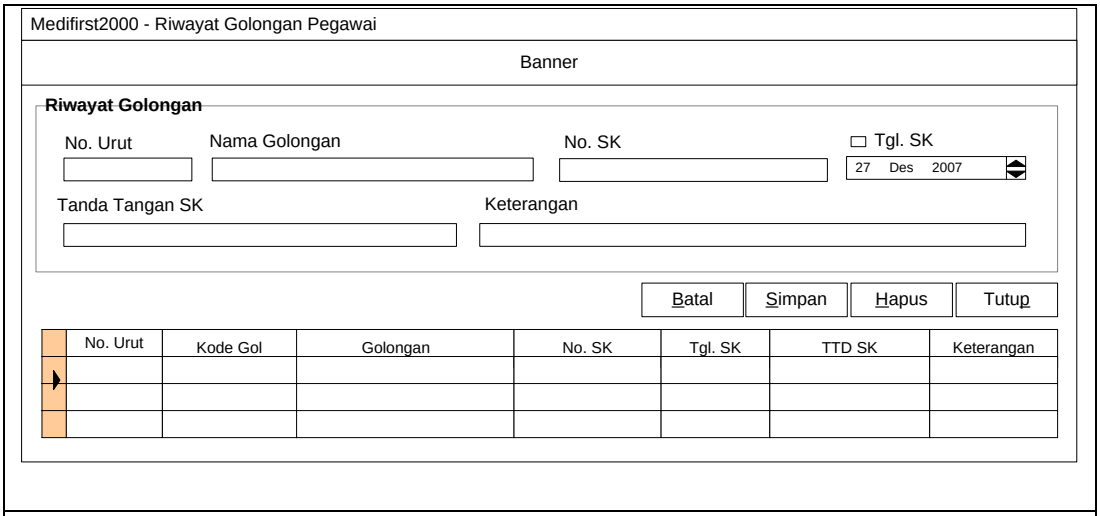
Form di atas digunakan untuk menginputkan data riwayat perjalanan dinas yang pernah lakukan oleh setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. Pulang dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Golongan Pegawai**

No Layar : PA-34

Nama Layar : Form Riwayat Golongan Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat golongan pegawai



Gambar 4.56 Antarmuka Form Riwayat Golongan Pegawai

Keterangan :

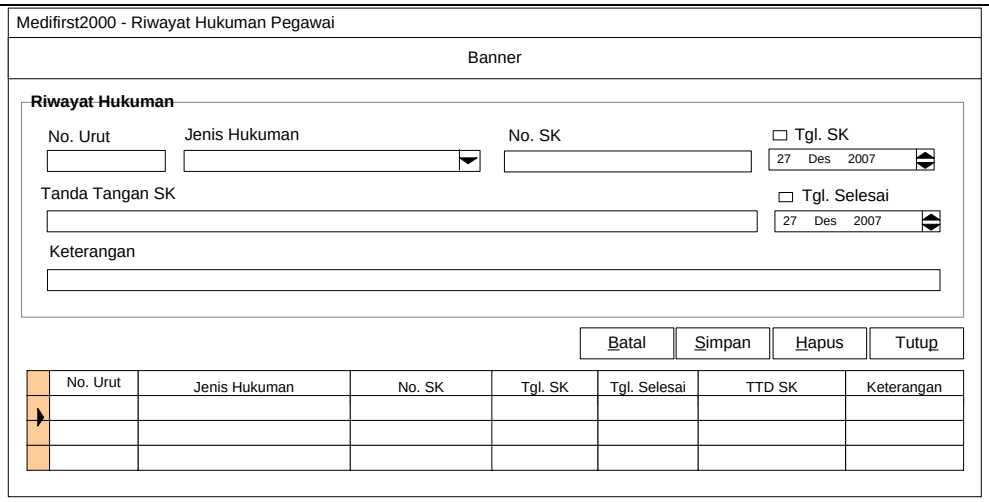
Form di atas digunakan untuk menginputkan data riwayat golongan setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. SK dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Hukuman Pegawai**

No Layar : PA-35

Nama Layar : Form Riwayat Hukuman Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat hukuman pegawai



Gambar 4.57 Antarmuka Form Riwayat Hukuman Pegawai

Keterangan :

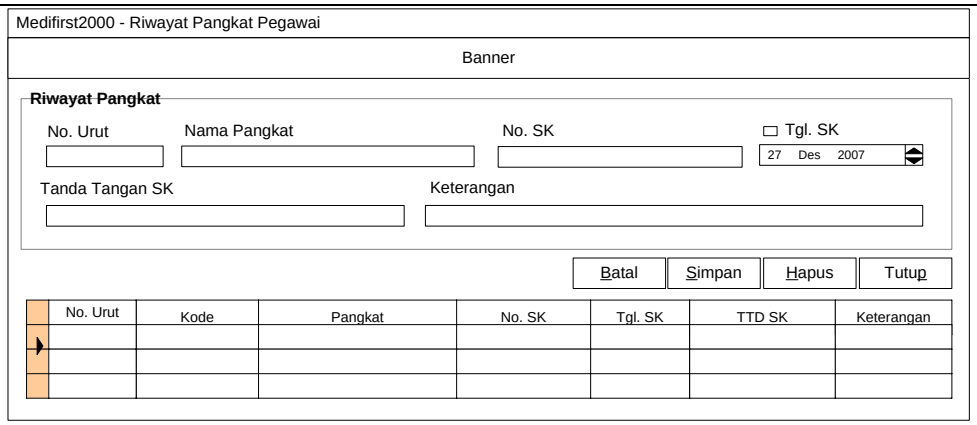
Form di atas digunakan untuk menginputkan data riwayat hukuman setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. SK dan Tgl. Selesai dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Pangkat Pegawai**

No Layar : PA-36

Nama Layar : Form Riwayat Pangkat Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat pangkat pegawai



Gambar 4.58 Antarmuka Form Riwayat Pangkat Pegawai

Keterangan :

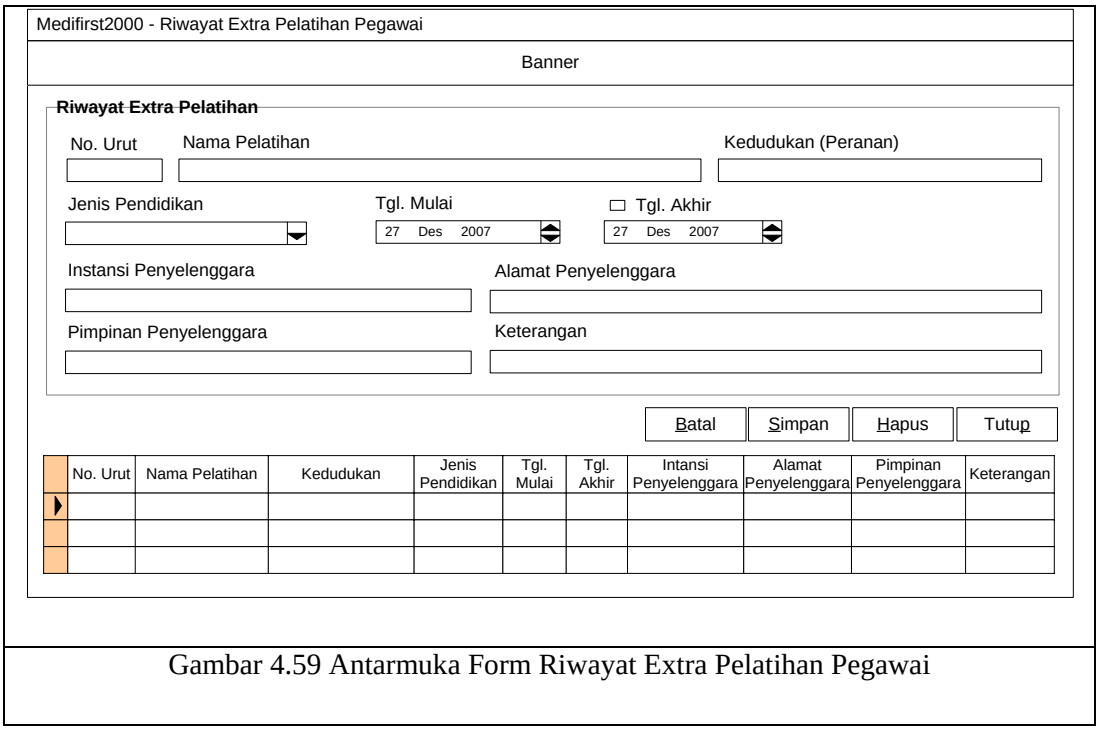
Form di atas digunakan untuk menginputkan data riwayat pangkat setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. SK dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Extra Pelatihan Pegawai**

No Layar : PA-37

Nama Layar : Form Riwayat Extra Pelatihan Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat extra pelatihan pegawai



Gambar 4.59 Antarmuka Form Riwayat Extra Pelatihan Pegawai

Keterangan :

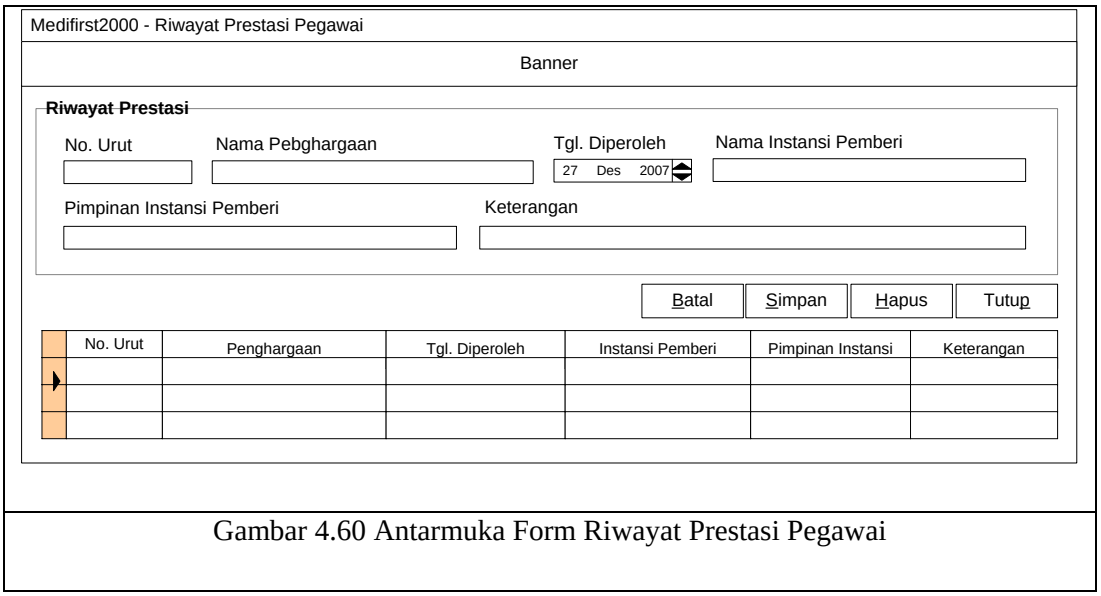
Form di atas digunakan untuk menginputkan data riwayat extra pelatihan setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. Akhir dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Prestasi Pegawai**

No Layar : PA-38

Nama Layar : Form Riwayat Prestasi Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat prestasi pegawai



Gambar 4.60 Antarmuka Form Riwayat Prestasi Pegawai

Keterangan :

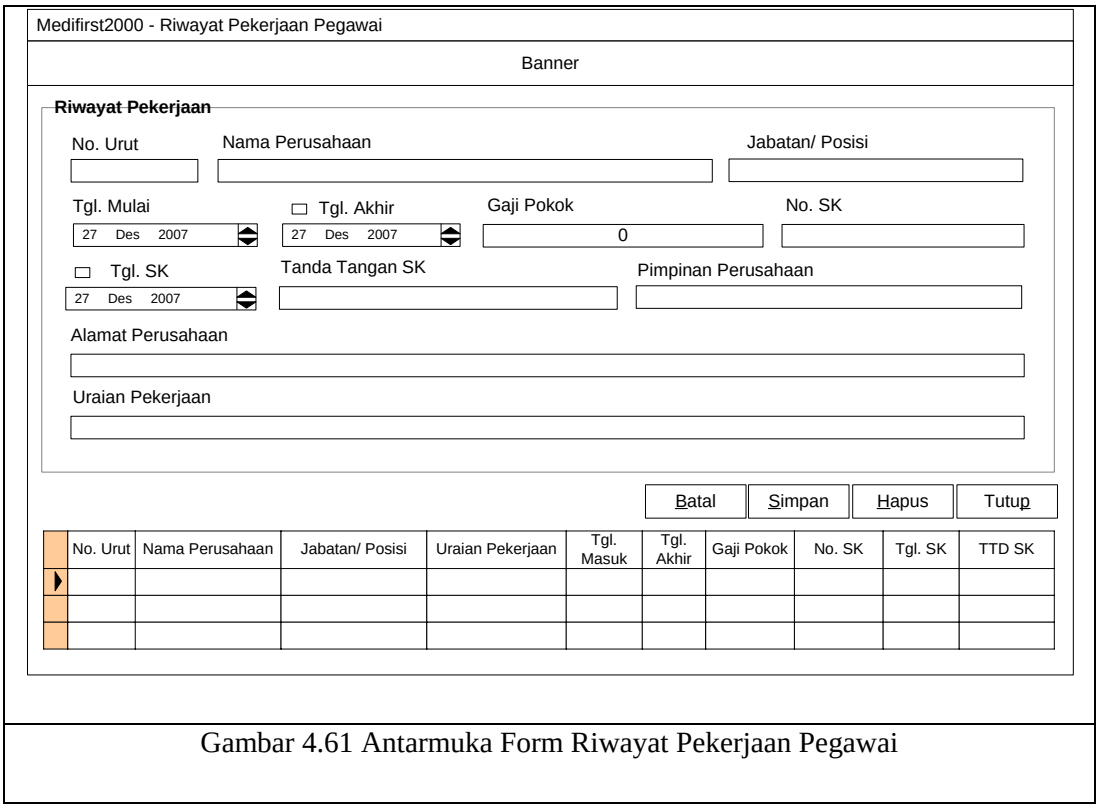
Form di atas digunakan untuk menginputkan data riwayat prestasi setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan.

➢ **Riwayat Pekerjaan Pegawai**

No Layar : PA-39

Nama Layar : Form Riwayat Pekerjaan Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat pekerjaan pegawai



Gambar 4.61 Antarmuka Form Riwayat Pekerjaan Pegawai

Keterangan :

Form di atas digunakan untuk menginputkan data riwayat pekerjaan setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah

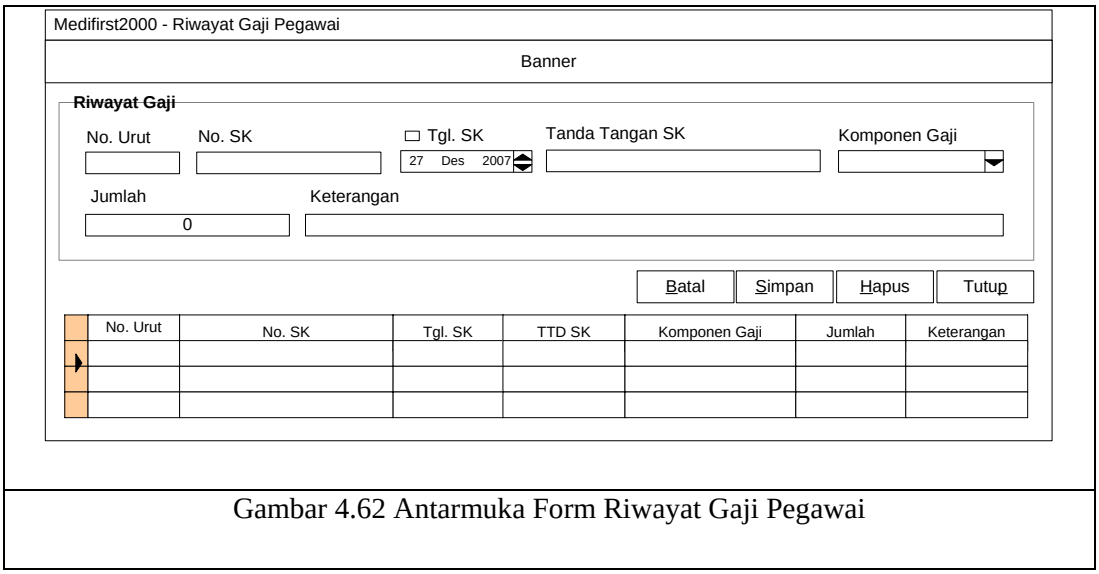
menekan tombol Simpan. Tgl. Akhir dan Tgl. SK dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Gaji Pegawai**

No Layar : PA-40

Nama Layar : Form Riwayat Gaji Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat gaji pegawai



Gambar 4.62 Antarmuka Form Riwayat Gaji Pegawai

Keterangan :

Form di atas digunakan untuk menginputkan data riwayat gaji setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan

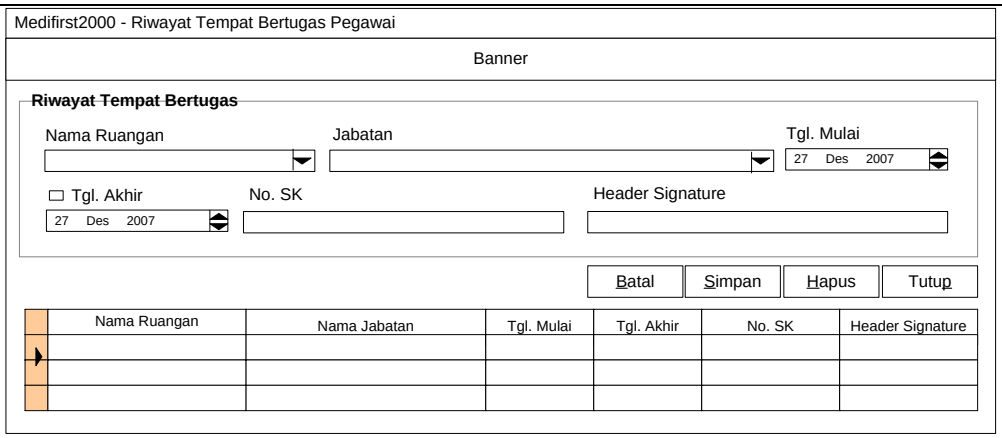
tombol Simpan. Tgl. SK dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Tempat Bertugas Pegawai**

No Layar : PA-41

Nama Layar : Form Riwayat Tempat Bertugas Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat tempat bertugas pegawai



Gambar 4.63 Antarmuka Form Riwayat Tempat Bertugas Pegawai

Keterangan :

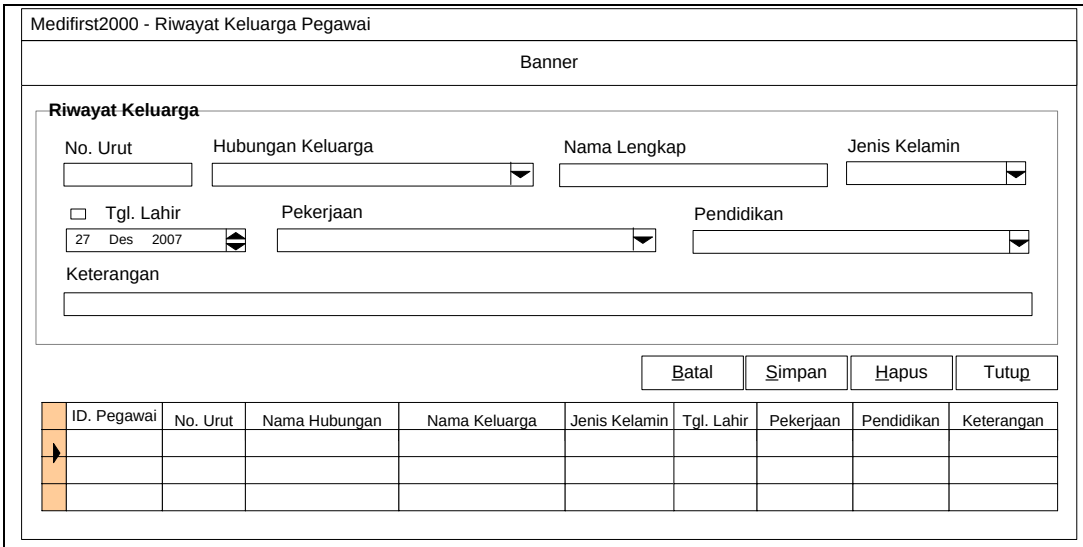
Form di atas digunakan untuk menginputkan data riwayat tempat bertugas setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. Akhir dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Keluarga Pegawai**

No Layar : PA-42

Nama Layar : Form Riwayat Keluarga Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat keluarga pegawai



Gambar 4.64 Antarmuka Form Riwayat Keluarga Pegawai

Keterangan :

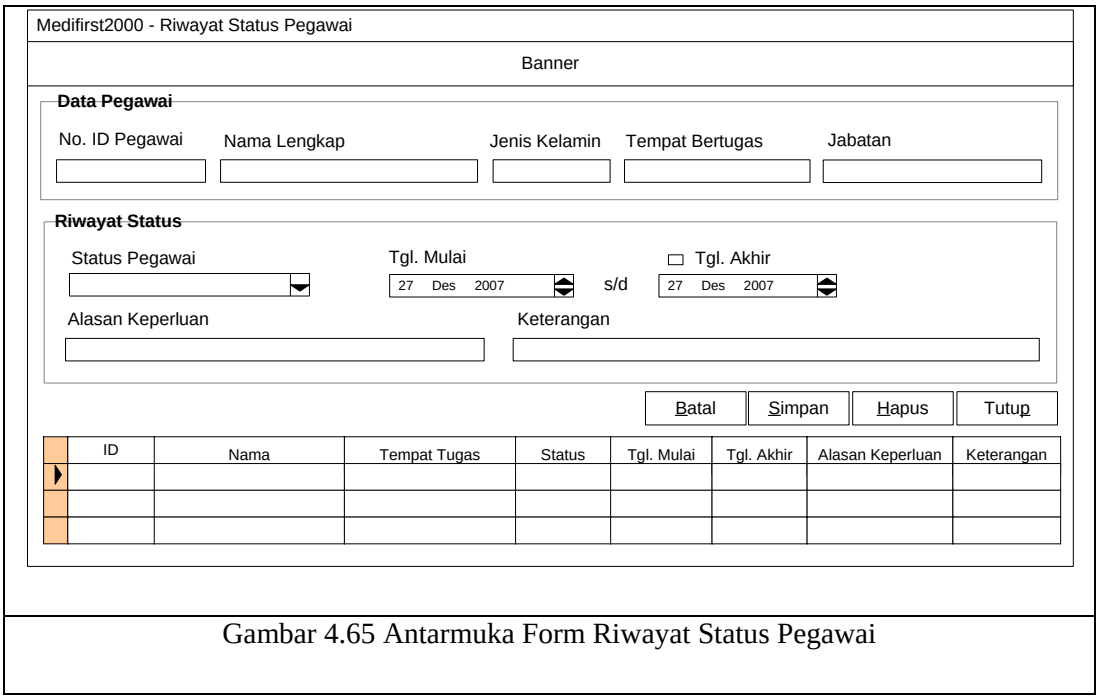
Form di atas digunakan untuk menginputkan data riwayat keluarga setiap pegawai. No. urut dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Tgl. Lahir dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- **Riwayat Status Pegawai**

No Layar : PA-43

Nama Layar : Form Riwayat Status Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data riwayat status pegawai



Gambar 4.65 Antarmuka Form Riwayat Status Pegawai

Keterangan :

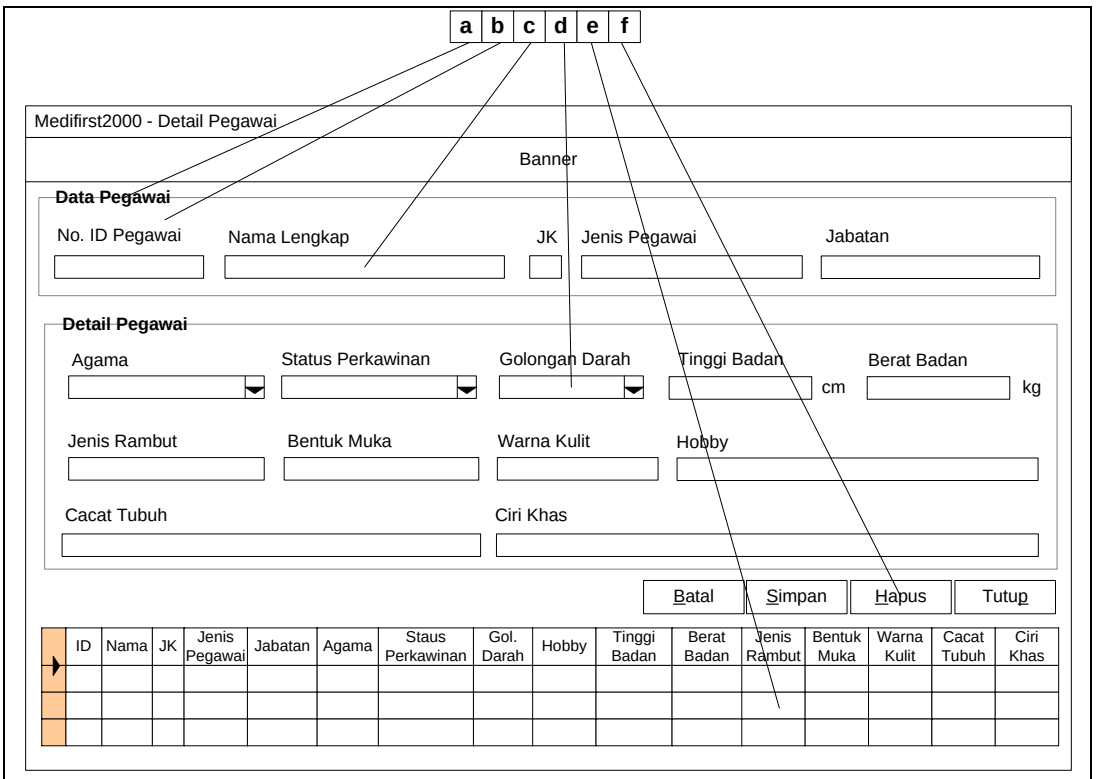
Form di atas digunakan untuk menginputkan data riwayat status setiap pegawai. No. ID Pegawai, Nama Lengkap, Jenis Kelamin, dan Jabtan dalam keadaan tidak aktif sesuai dengan data pegawai yang dipilih. Tgl. Akhir dalam keadaan tidak aktif, jika akan mengisi tanggalnya checklist pada checkbox disampingnya.

- Tombol Detail

No Layar : PA-44

Nama Layar : Form Detail Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data detail pegawai



Gambar 4.66 Antarmuka Form Detail Pegawai

Keterangan :

Form di atas digunakan utnuk menginputkan data detail pegawai. Kolom Data Pegawai dalam keadaan tidak aktif dan menampilkan data-data pasien sesuai dengan yang dipilih.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

c. *TextBox*, tempat untuk memasukkan nilai pada item-item yang akan diinputkan

d. *DataCombo*, jika di klik akan menampilkan data-data yang sudah ada di dalam *database*

e. *DataGrid*, tempat untuk menampilkan data detail pegawai yang diinputkan dan disimpan di dalam *database*

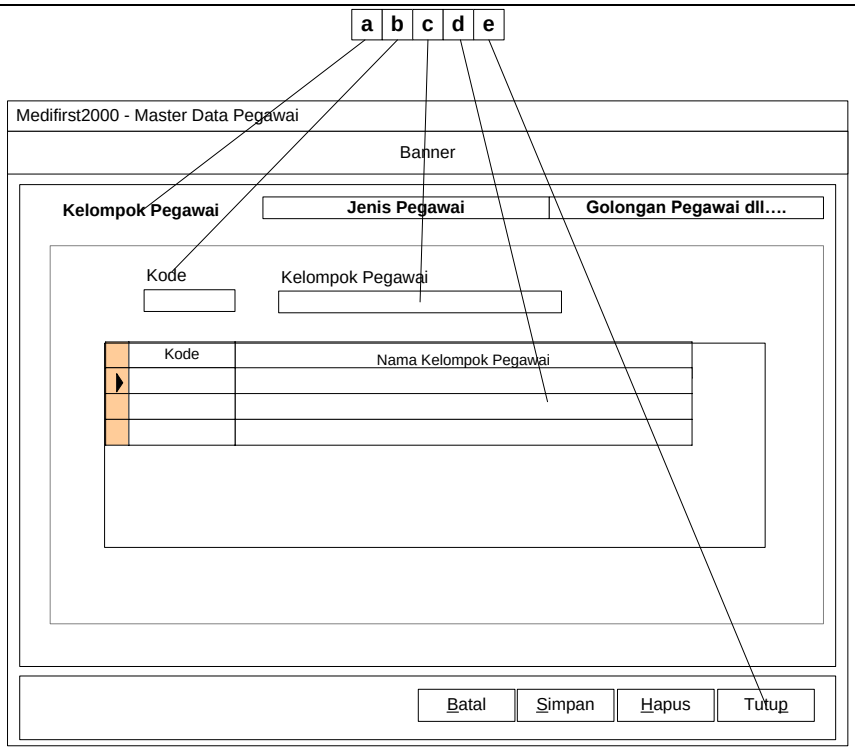
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

✓ **Form Master Data Pegawai**

No Layar : PA-45

Nama Layar : Form Master Data Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan master data pegawai



Gambar 4.67 Antarmuka Form Master Data Pegawai

Keterangan :

Form di atas digunakan untuk menampilkan dan menginputkan data-data master kelompok pegawai. Kode dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan. Form tersebut terdiri dari 6 Tab, yaitu : Tab Kelompok Pegawai, Jenis Pegawai, Golongan Pegawai, Pangkat Pegawai, Jenis Jabatan, dan Jabatan.

a. *SsTab*, digunakan untuk membuat 6 form pengisian data dalam satu tampilan form

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

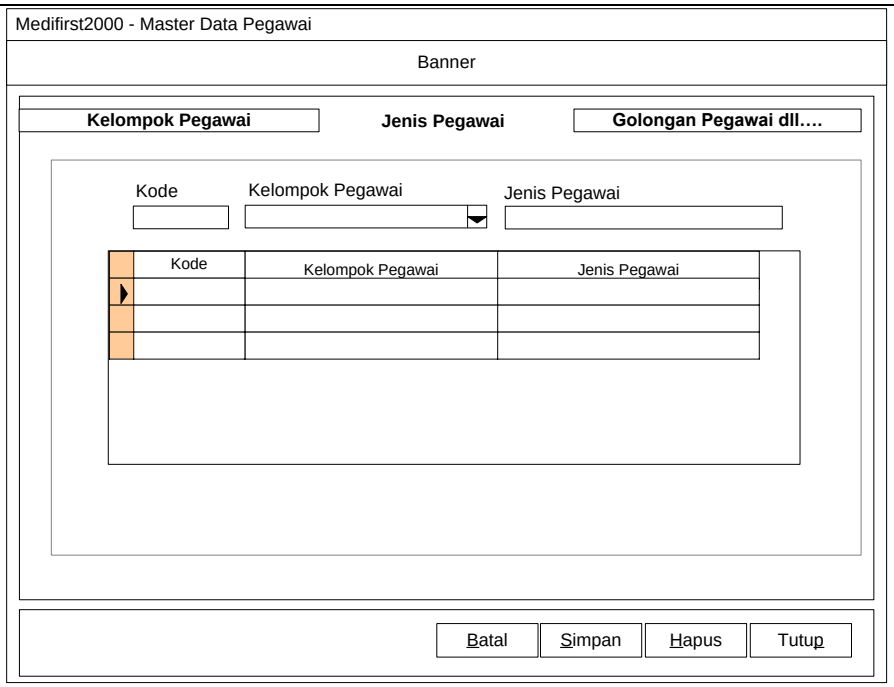
c. *TextBox*, tempat untuk memasukkan data-data yang akan diinputkan

d. *DataGrid*, tempat untuk menampilkan data pelayanan tindakan yang sudah dilakukan oleh pasien

e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

- **Jenis Pegawai**

No Layar	:	PA-46
Nama Layar	:	Form Master Data Pegawai
Deskripsi	:	Tampilan antarmuka yang digunakan untuk menginputkan data jenis pegawai



Gambar 4.68 Antarmuka Form Master Data Pegawai

Keterangan :

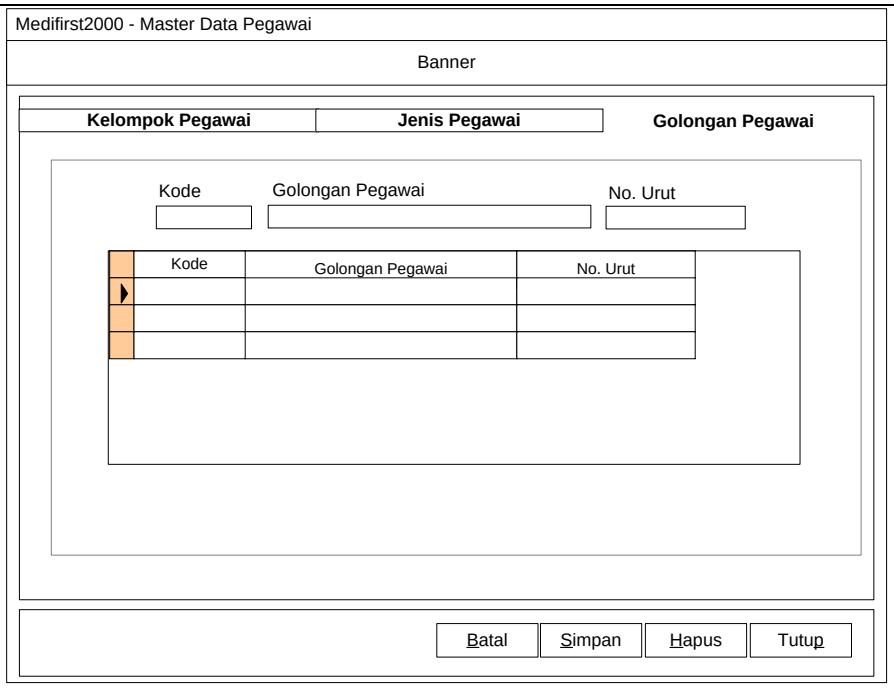
Form di atas digunakan untuk menampilkan dan menginputkan data jenis pegawai. Kode dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan.

- **Golongan Pegawai**

No Layar : PA-47

Nama Layar : Form Master Data Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data golongan pegawai



Gambar 4.69 Antarmuka Form Master Data Pegawai

Keterangan :

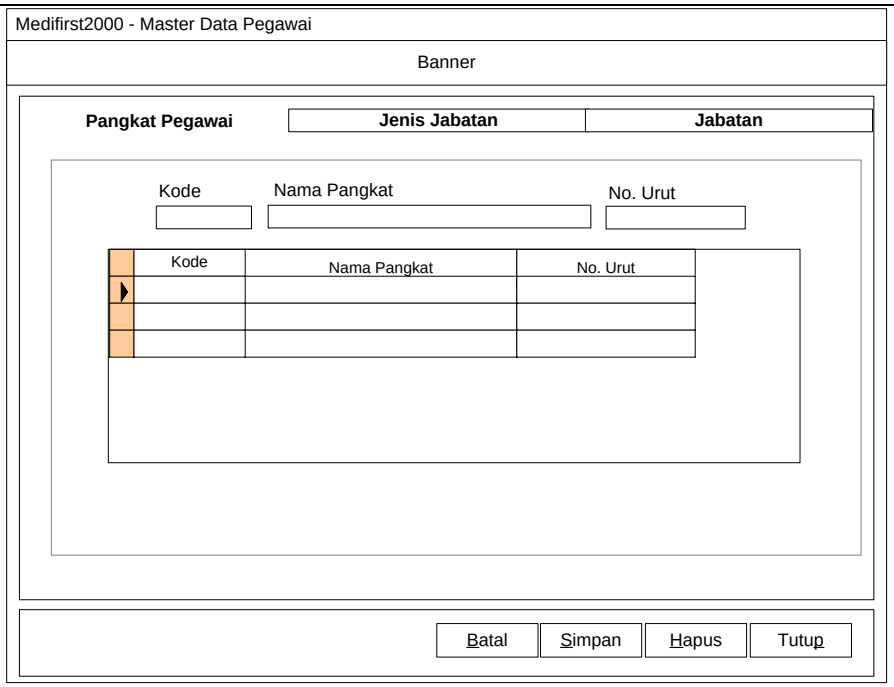
Form di atas digunakan untuk menampilkan dan menginputkan data golongan pegawai. Kode dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan.

- **Pangkat Pegawai**

No Layar : PA-48

Nama Layar : Form Master Data Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data pangkat pegawai



Gambar 4.70 Antarmuka Form Master Data Pegawai

Keterangan :

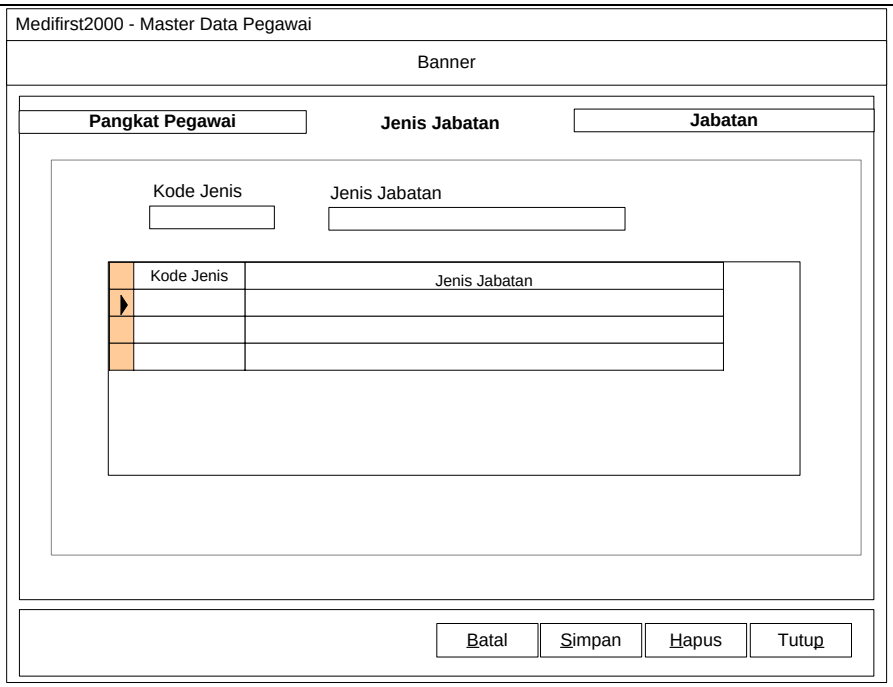
Form di atas digunakan untuk menampilkan dan menginputkan data pangkat pegawai. Kode dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan.

- **Jenis Jabatan**

No Layar : PA-49

Nama Layar : Form Master Data Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data jenis jabatan



Gambar 4.71 Antarmuka Form Master Data Pegawai

Keterangan :

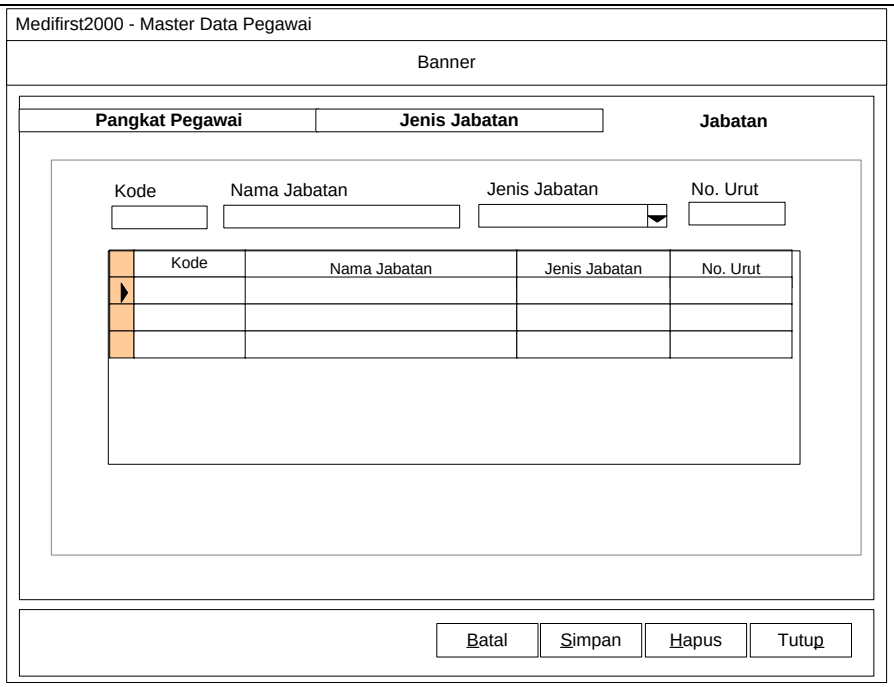
Form di atas digunakan untuk menampilkan dan menginputkan data jenis jabatan. Kode dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan.

- **Jabatan**

No Layar : PA-50

Nama Layar : Form Master Data Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data jabatan



Gambar 4.72 Antarmuka Form Master Data Pegawai

Keterangan :

Form di atas digunakan untuk menampilkan dan menginputkan data jabatan. Kode dalam keadaan tidak aktif, akan otomatis terisi nomor setelah menekan tombol Simpan.

JenisPegawai

Algoritma :

Nama Operasi : TamabahJenisPegawai

```
Declare
  KdJenisPegawai        JenisPegawai


Open koneksi
IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Simpan THEN
   IF JenisPegawai sudah ada THEN
     Akan tampil informasi “Data Jenis Pegawai Sudah Ada”
   ELSE {JenisPegawai belum ada}
     Masukkan data-data nya
     Insert Into [JenisPegawai]
     Tampil pesan “Penyimpanan Jenis Pegawai Berhasil”
ELSE
ELSE
IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Hapus THEN
   IF JenisPegawai sudah dipilih THEN
     Delete From [JenisPegawai]
     Where KdJenisPegawai=@KdJenisPegawai
   ELSE {jenis pegawai belum dipilih}
     Tampil Pesan “Pilih Data Yang Akan Di Hapus”
```

```
        ELSE

      ELSE



close_koneksi
```

Keterangan :

Form di atas digunakan utnuk menginputkan data jenis pegawai. Kode Jenis Pegawai dalam keadaan tidak aktif dan akan terisai otomatis setelah menekan tombol Simpan.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *TextBox*, tempat untuk memasukkan nilai pada item-item yang akan diinputkan

c. *DataGrid*, tempat untuk menampilkan data jenis pegawai yang baru diinputkan dan yang sudah ada di dalam *database*

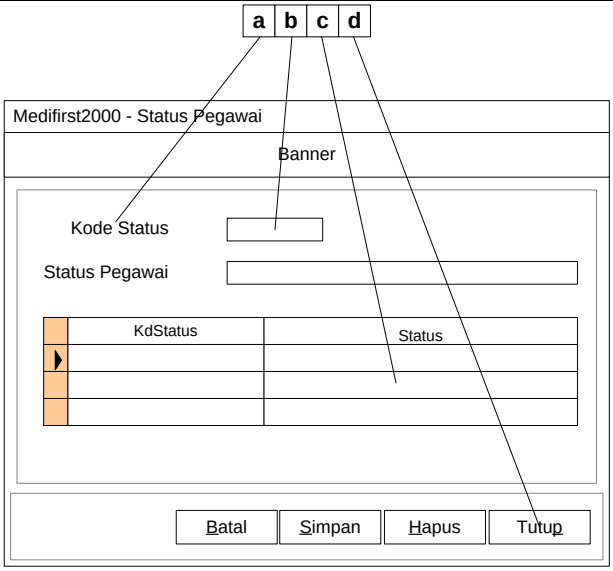
d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

✓ **Form Status Pegawai**

No Layar : PA-51

Nama Layar : Form Status Pegawai

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data status pegawai



Gambar 4.73 Antarmuka Form Status Pegawai

StatusPegawai

Algoritma :

Nama Operasi : TamabahStatusPegawai

Declare

KdStatus Status

```
Open koneksi
IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Simpan THEN
  IF Status sudah ada THEN
    Akan tampil informasi “Data Status Pegawai Sudah Ada”
  ELSE {Status belum ada}
    Masukkan data-data nya
    Insert Into [KdStatus]
    Tampil pesan “Penyimpanan Status Pegawai Berhasil”
 ELSE
 ELSE
 IF koneksi berhasil THEN
    pilih database
 IF tekan tombol Hapus THEN
  IF StatusPegawai sudah dipilih THEN
    Delete From [StatusPegawai] Where KdStatus=@KdStatus
  ELSE {status pegawai belum dipilih}
    Tampil Pesan “Pilih Data Yang Akan Di Hapus”
 ELSE
 ELSE

close_koneksi
```

Keterangan :

Form di atas digunakan utnuk menginputkan data status pegawai. Kode Status Pegawai dalam keadaan tidak aktif dan akan terisai otomatis setelah menekan tombol Simpan.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *TextBox*, tempat untuk memasukkan nilai pada item-item yang akan diinputkan

c. *DataGrid*, tempat untuk menampilkan data status pegawai yang baru diinputkan dan yang sudah ada di dalam *database*

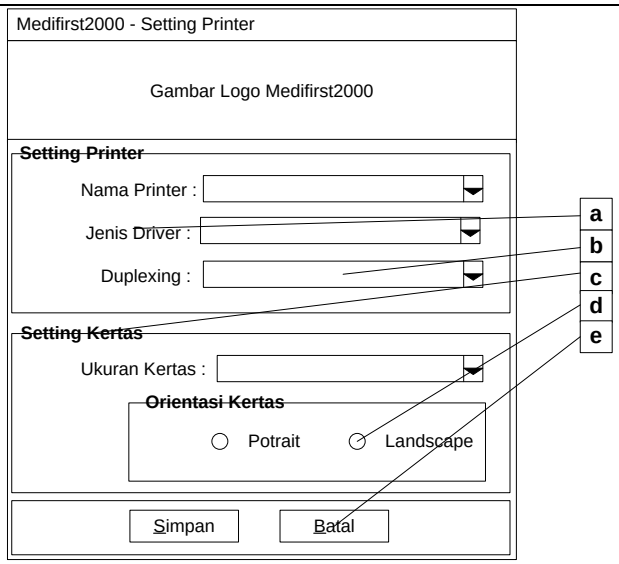
d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database dan data tersebut akan ditampilkan di DataGrid. Tombol Hapus, digunakan untuk menghapus data yang sudah ada. Tombol Batal, jika di klik maka data-data yang diinputkan akan menjadi kosong. Tombol Tutup, jika diklik akan kembali ke form Menu Utama

✓ **Form Setting Printer**

No Layar : PA-52

Nama Layar : Form Terima Barang Langsung

Deskripsi : Tampilan antarmuka yang digunakan untuk melakukan penyettingan printer



Gambar 4.74 Antarmuka Form Setting Printer

Algoritma :

Nama Operasi : SettingPrinter

Open koneksi

IF koneksi berhasil THEN

pilih database

IF nama printer AND jenis driver AND duplexing AND ukuran kertas AND orientasi kertas dipilih THEN

Tampil pesan “Penyettingan Printer Berhasil”

ELSE

```
        Salah satu pilihan tidak boleh ada yang kosong
    ELSE
close_koneksi
```

Keterangan :

Form yang digunakan untuk menyetting printer yang akan digunakan untuk mencetak laporan.

a. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

b. *DataCombo*, jika di klik akan menampilkan data-data nama Supplier yang sudah ada di dalam *database*

c. *Frame,* digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

d. *RadioButton*, digunakan untuk memilih salah satu pilihan yang sudah ada

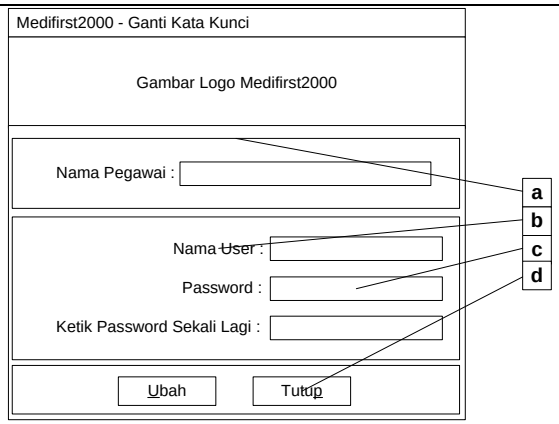
e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Simpan, jika diklik data yang diinputkan akan di simpan dan laporan yang akan dicetak akan sesuai dengan settingan yang sudah dipilih. Tombol Batal, jika di klik maka data-data seperti di kolom Setting Printer dan Setting Kertas menjadi kosong dan akan kembali ke form Menu Utama

✓ **Form Ganti Kata Kunci**

No Layar : PA-53

Nama Layar : Form Ganti Kata Kunci

Deskripsi : Tampilan antarmuka yang digunakan untuk mengganti password pegawai yang sudah ada



Gambar 4.75 Antarmuka Form Ganti Kata Kunci

DataPegawai,Login

Algoritma :

Nama Operasi : UbahKataKunci

Declare

NamaUser Password

Open koneksi

IF koneksi berhasil THEN

```
        pilih database
      IF salah satu item tidak diisi THEN
         Tampil pesan
      ELSE {semua item terisi}
        Masukkan data yang baru
        UPDATE Login SET IdPegawai = @IDPegawai, UserName =
         @UserName,Password = @Password
         WHERE IdPegawai = @IdPegawai
    ELSE
close_koneksi
```

Keterangan :

Form yang digunakan untuk mengubah kata kunci pada saat Anda akan masuk ke program aplikasi.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item yang akan diinputkan

c. *TextBox*, tempat untuk memasukkan nilai pada item-item data ganti kata kunci

d. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Ubah, jika diklik data yang diinputkan akan di simpan dan dirubah

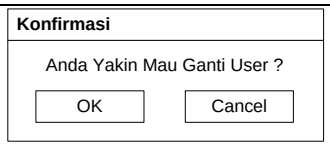
sesuai dengan data yang terakhir Anda masukkan. Tombol Tutup, jika di klik akan kembali ke form Menu Utama

✓ **Form Log Off...Analist/ Programmer**

No Layar : PA-54

Nama Layar : Form

Deskripsi : Tampilan antarmuka yang digunakan untuk



Gambar 4.76 Antarmuka Form

Keterangan :

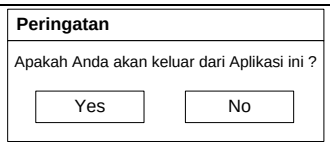
Konfirmasi di atas ditampilkan menggunakan Message Box yang sudah disediakan oleh aplikasi VB. Jika menekan Ok maka akan masuk ke form login aplikasi

✓ **Form Keluar Aplikasi**

No Layar : PA-55

Nama Layar : Form

Deskripsi : Tampilan antarmuka yang digunakan untuk



Gambar 4.77 Antarmuka Form

Keterangan :

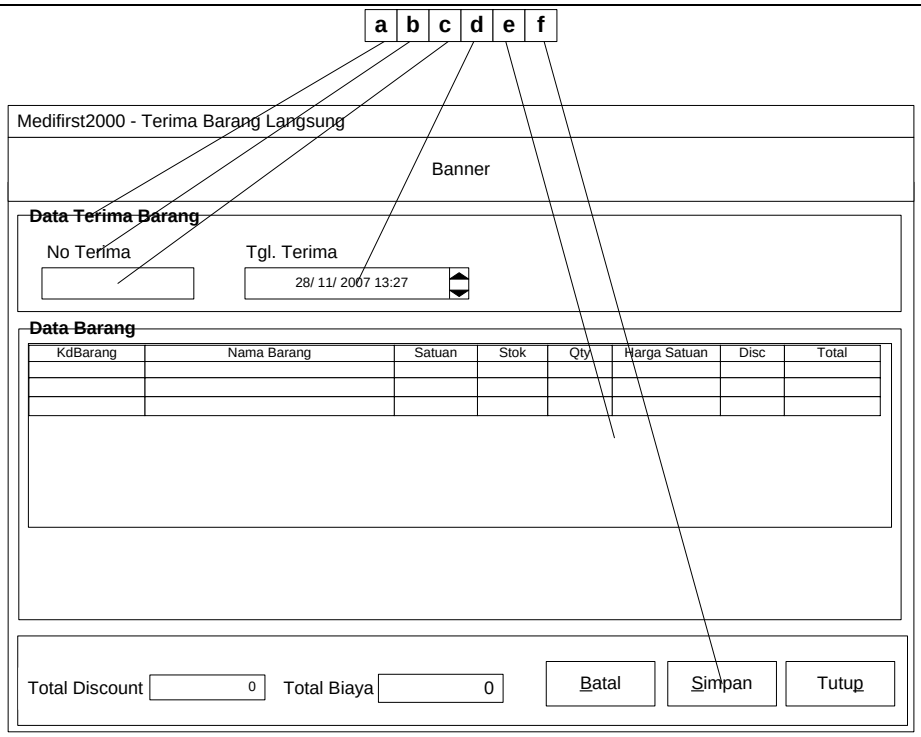
Peringatan di atas ditampilkan menggunakan Message Box yang sudah disediakan oleh aplikasi VB. Jika menekan Yes maka akan keluar dari aplikasi

✓ **Form Penerimaan Barang**

No Layar : PA-56

Nama Layar : Form Terima Barang Langsung

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan data barang yang diterima



Gambar 4.78 Antarmuka Form Terima Barang Langsung

MasterBarang,StockBarang,Satuan,DetailTerimaBarang

Algoritma :

Nama Operasi : SimpanTerimaBarang

Declare

NoTerima Tgl.Terima

KdBarang NamaBarang

Satuan Stok

JmlTerima HargaSatuan

Disc

```
If NamaBarang sudah ada Then
  Data barang yang akan diterima dapat diinputkan
  Insert INTO [MasterBarang],[StockBarang],[Satuan],
        [DetailTerimaBarang]
  Jumlah barang yang diterima akan meng-update jumlah
  barang di tabel StockBarang
  Jumlah StockBarang akan bertambah
  Else { NamaBarang tidak ada }
    Cek tabel MasterBarang
   Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menginputkan data barang yang sudah diterima. Masukkan Tgl. Terima barang, No Terima dalam keadaan tidak aktif dan akan terisi otomatis setelah menekan tombol Simpan. Untuk manginputkan data barang, pada DataGrid di kolom Nama Barang ketikkan atau masukkan salah satu huruf nama barang yang akan diinputkan lalu tekan Enter, maka KdBarang, Satuan, Stok, Harga Satuan, dan Disc akan terisi sesuai dengan Nama Barang yang sudah dipilih. Untuk besarnya Total akan terisi otomatis sesuai dengan Qty*Harga Satuan. Total Discount dan Total Biaya akan terisi otomatis sesuai dengan besar totalnya.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

c. *TextBox*, tempat untuk menampilkan No Terima, Total Discount, dan Total Biaya

d. *DTPicker*, digunakan untuk menampilkan tanggal dan waktu pada saat menginputkan data. Formatnya DD/ MM/ YYYY HH:SS

e. *DataGrid*, tempat untuk menginputkan data barang yang sudah diterima

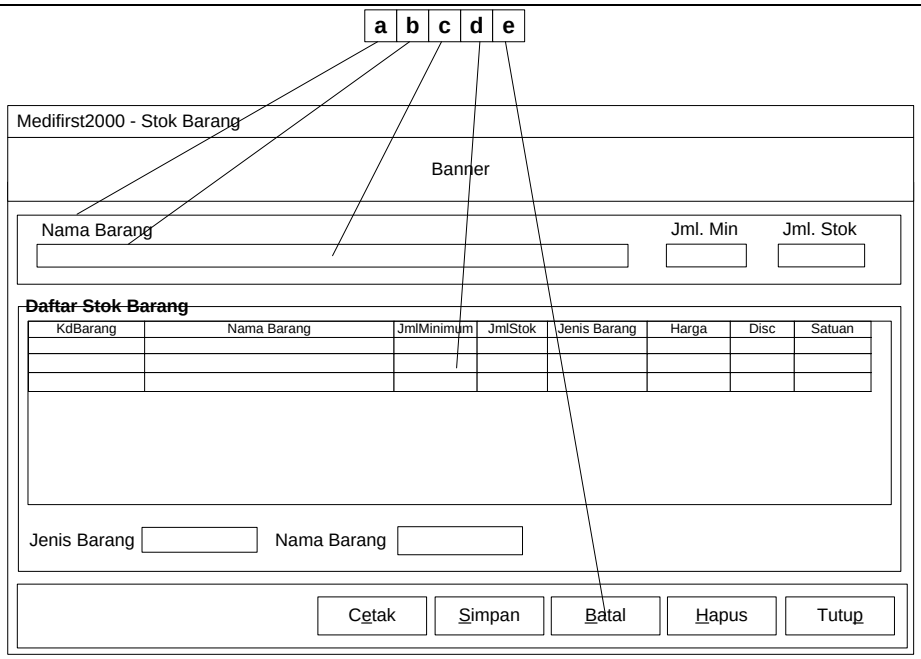
f. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Batal, jika diklik data yang sudah terisi di DataGrid akan menjadi kosong kembali. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

✓ **Form Stok Barang**

No Layar : PA-57

Nama Layar : Form Stok Barang

Deskripsi : Tampilan antarmuka yang digunakan untuk menginputkan stok barang



Gambar 4.79 Antarmuka Form Stok Barang

MasterBarang,StockBarang,Satuan,DetailJenisBarang,

JenisBarang

Algoritma :

Nama Operasi : SimpanStockBarang

Declare

KdBarang NamaBarang

JmlMin JmlStock

JenisBarang HargaSatuan

Disc Satuan

If NamaBarang,JmlMin, atau JmlStock tidak di isi Then

```
    Akan tampil pesan Data harus diisi
    Else { NamaBarang,JmlMin, atau JmlStock di isi}
        Data yang dimasukkan akan disimpan
        Insert INTO [StockBarang]
      Endif
Endif


    Nama Operasi        : HapusStockBarang
    Declare
      KdBarang          NamaBarang
      JmlMin            JmlStock
      JenisBarang       HargaSatuan
      Disc              Satuan


If DataStockBarang sudah dipilih Then
   Data yang dipilih akan dihapus
   Delete From [StockBarang]
   Else { DataStockBarang blm dipilih}
       Akan tampil pesan Data yang akan dihapus Belum Dipilih
      Endif
Endif
```

Keterangan :

Form di atas digunakan untuk menginputkan data Stok Barang dan dapat digunakan untuk mencari data stok barang. Ketikkan Nama Barang yang akan diinputkan dengan menuliskan namanya atau salah satu huruf maka akan ditampilkan data-data barang yang sudah ada seperti KdBarang, Nama Barang, dan Jenis Barang.

a. *Frame*, digunakan untuk mengelompokkan data-data agar terlihat lebih rapih

b. *Label*, digunakan untuk memberikan keterangan kepada item-item

c. *TextBox*, tempat untuk menginputkan data-data stok barang

d. *DataGrid*, tempat untuk menampilkan data stok barang

e. *CommandButton*, digunakan untuk melakukan operasi pada form tersebut. Tombol Cetak, jika diklik akan mencetak data stok barang yang ada. Tombol Batal, jika diklik data yang sudah terisi akan menjadi kosong kembali. Tombol Simpan, jika diklik data yang diinputkan akan di simpan ke database. Tombol Tutup, jika diklik akan kembali ke form Menu Utama.

4.3 Model Perancangan

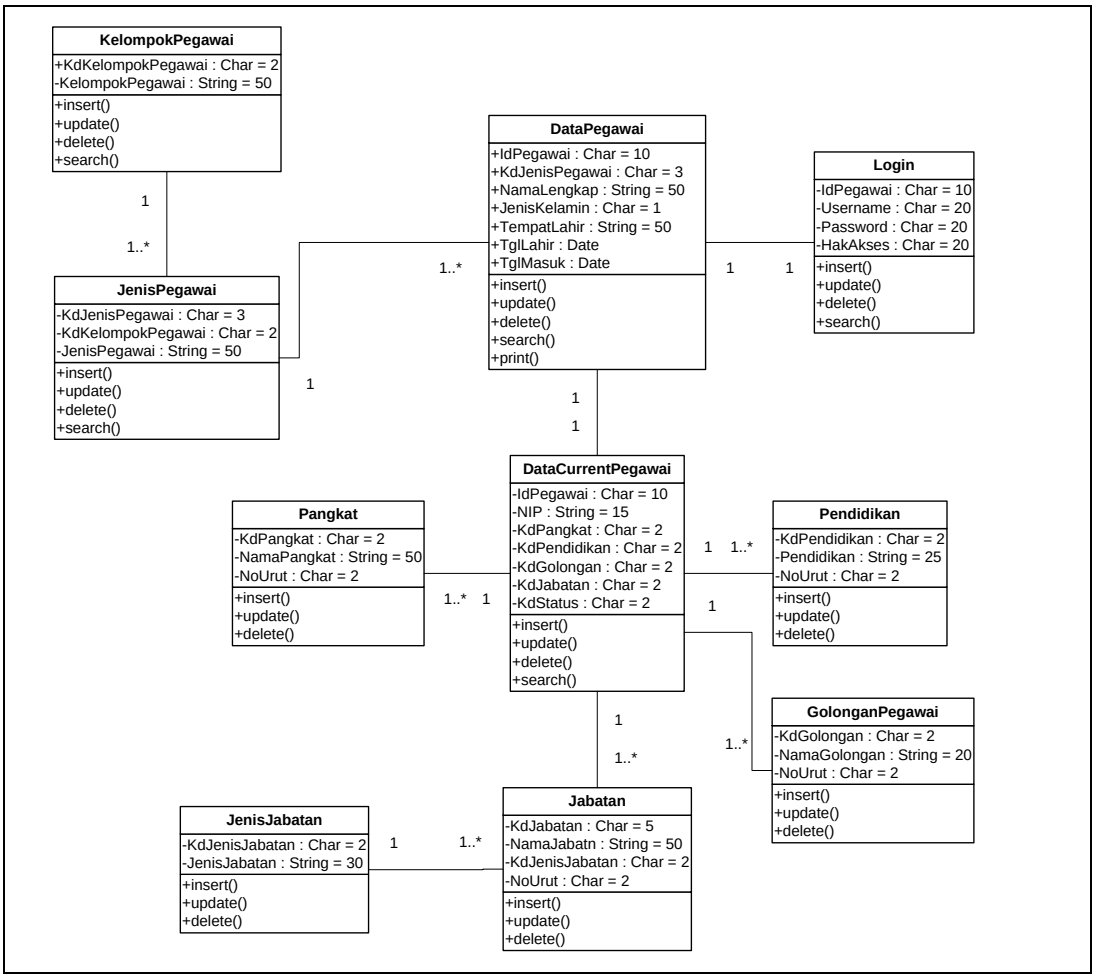
4.3.1 Realisasi *Use case* Tahap Perancangan

Setelah dijelaskan dalam model analisis, sistem yang dibangun digambarkan ke dalam model desain, yang menggambarkan software dalam bentuk objek-objek, tetapi dalam tingkat yang lebih mendekati *source code*.

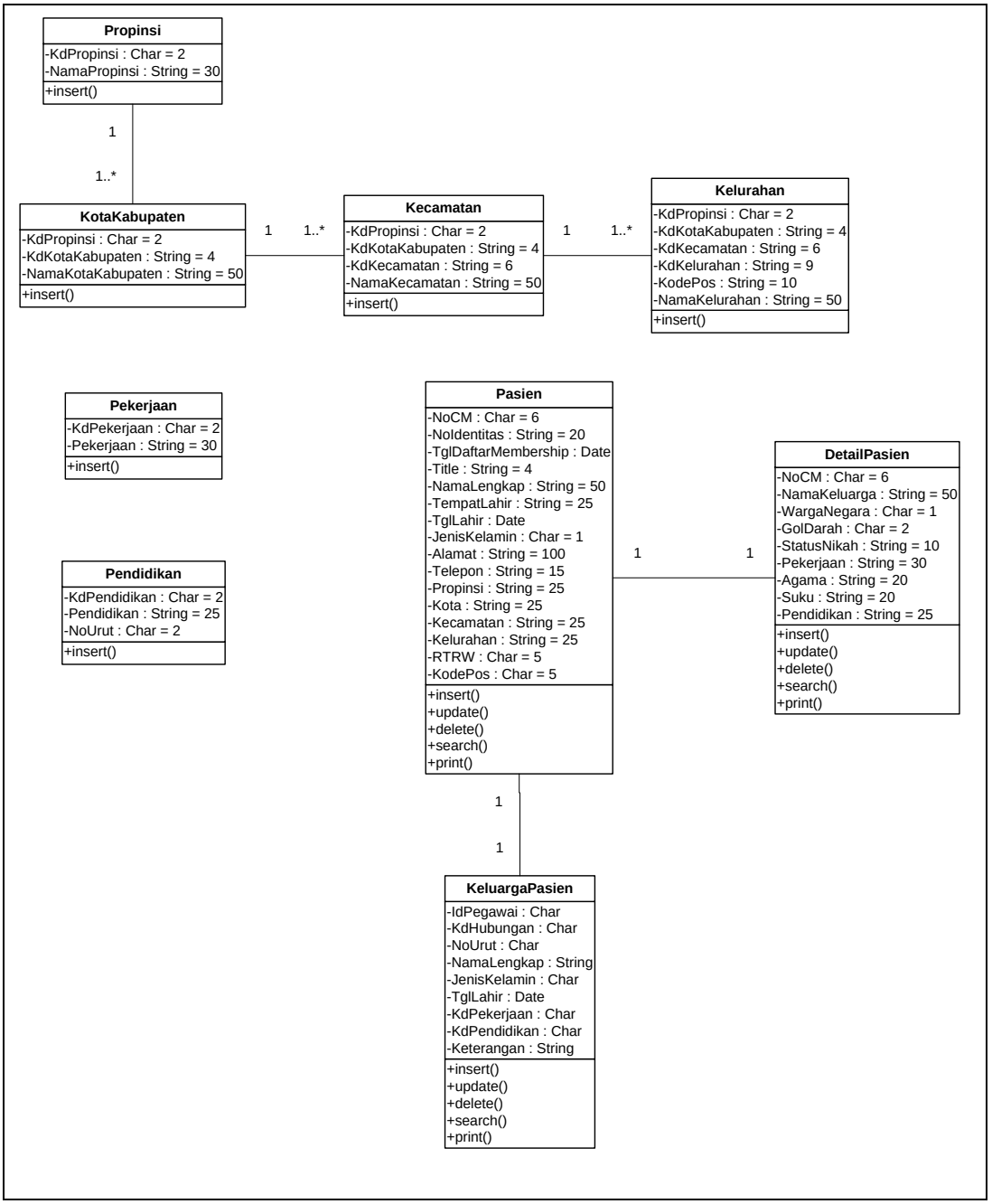
- ***Class Diagram* Model Perancangan**

Menggambarkan struktur kelas desain di dalam sistem. *Class Diagram* dalam sistem ini terbagi ke dalam 6 macam, yaitu Class diagram DataPegawai&Login, Class Diagram Alamat&DataPasien, Class Diagram Registrasi Laboratorium, Class Diagram Pelayanan Tindakan, Class Diagram Transaksi Pelayanan Pasien, dan Class Diagram Hasil Periksa Laboratorium

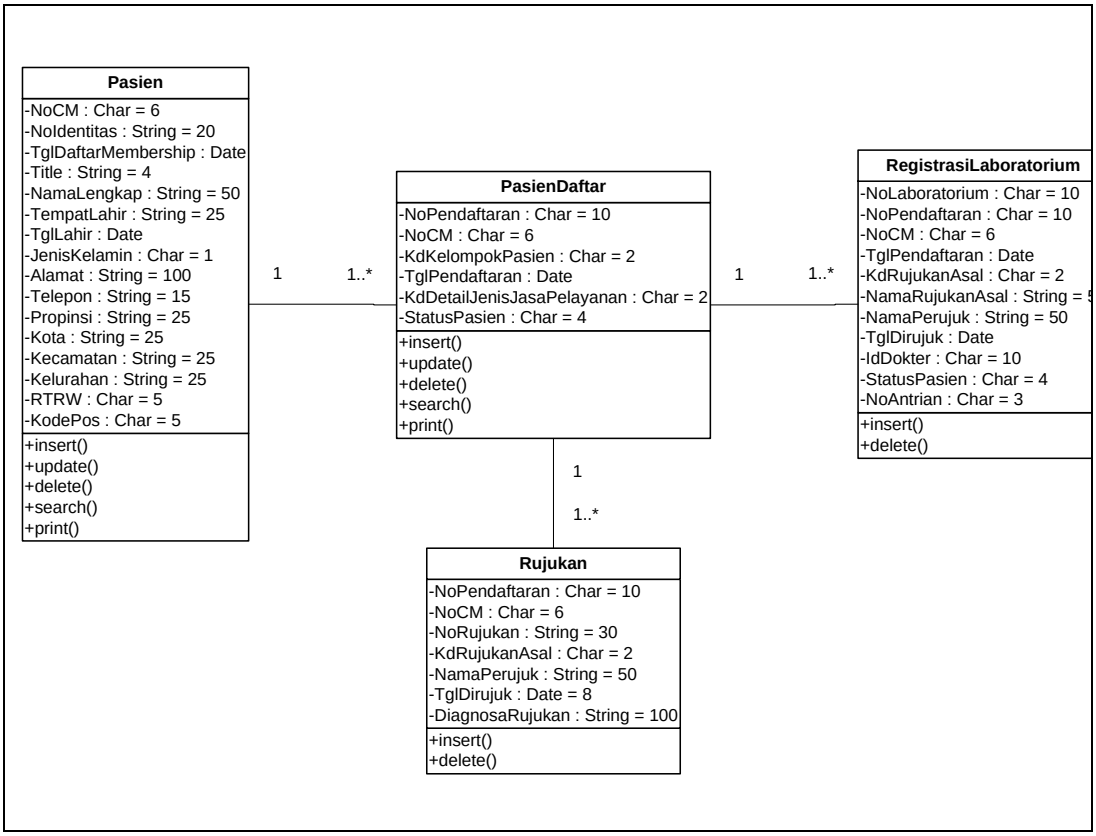
- **Class Diagram DataPegawai&Login**



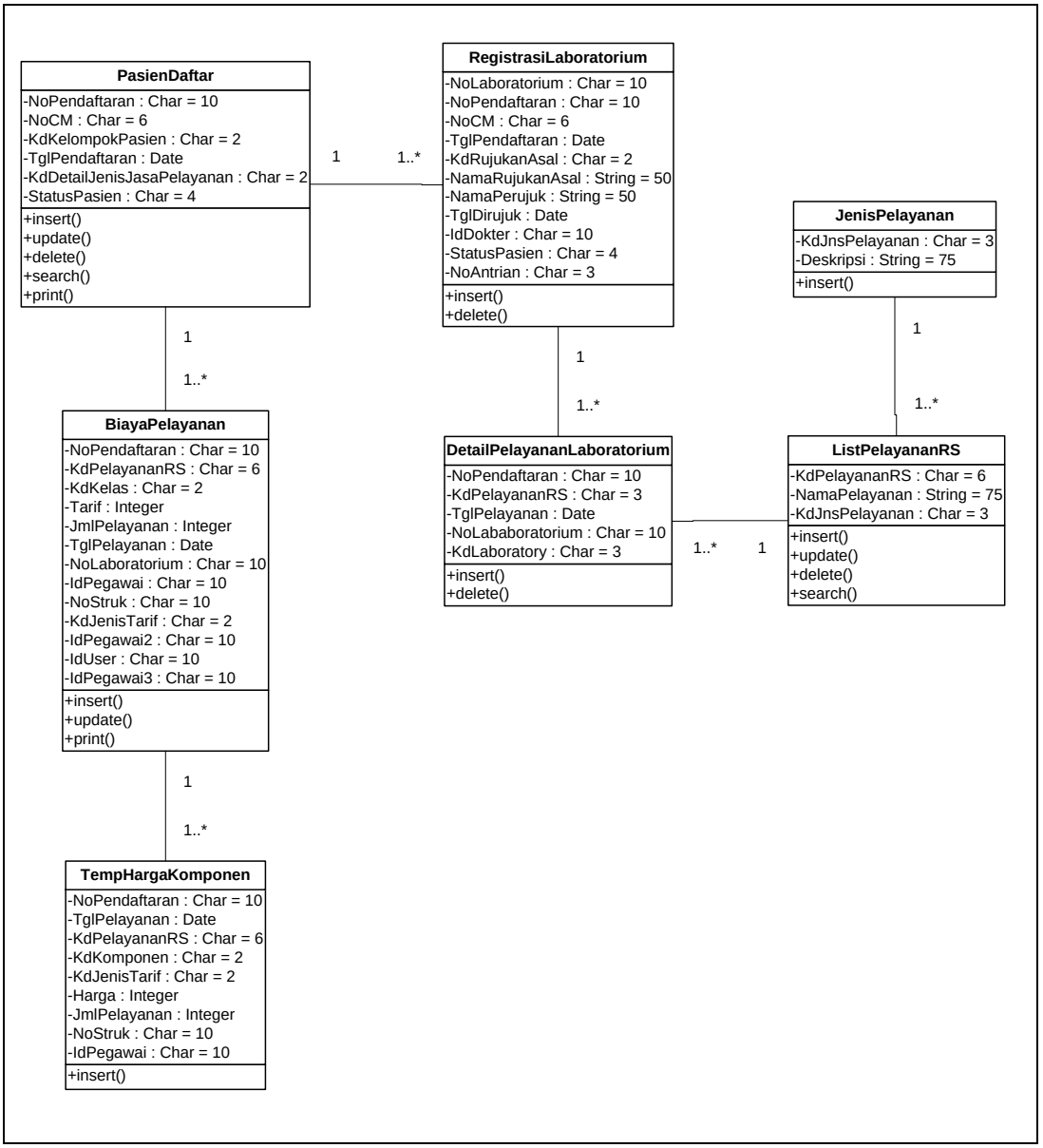
- **Class Diagram Alamat&DataPasien**



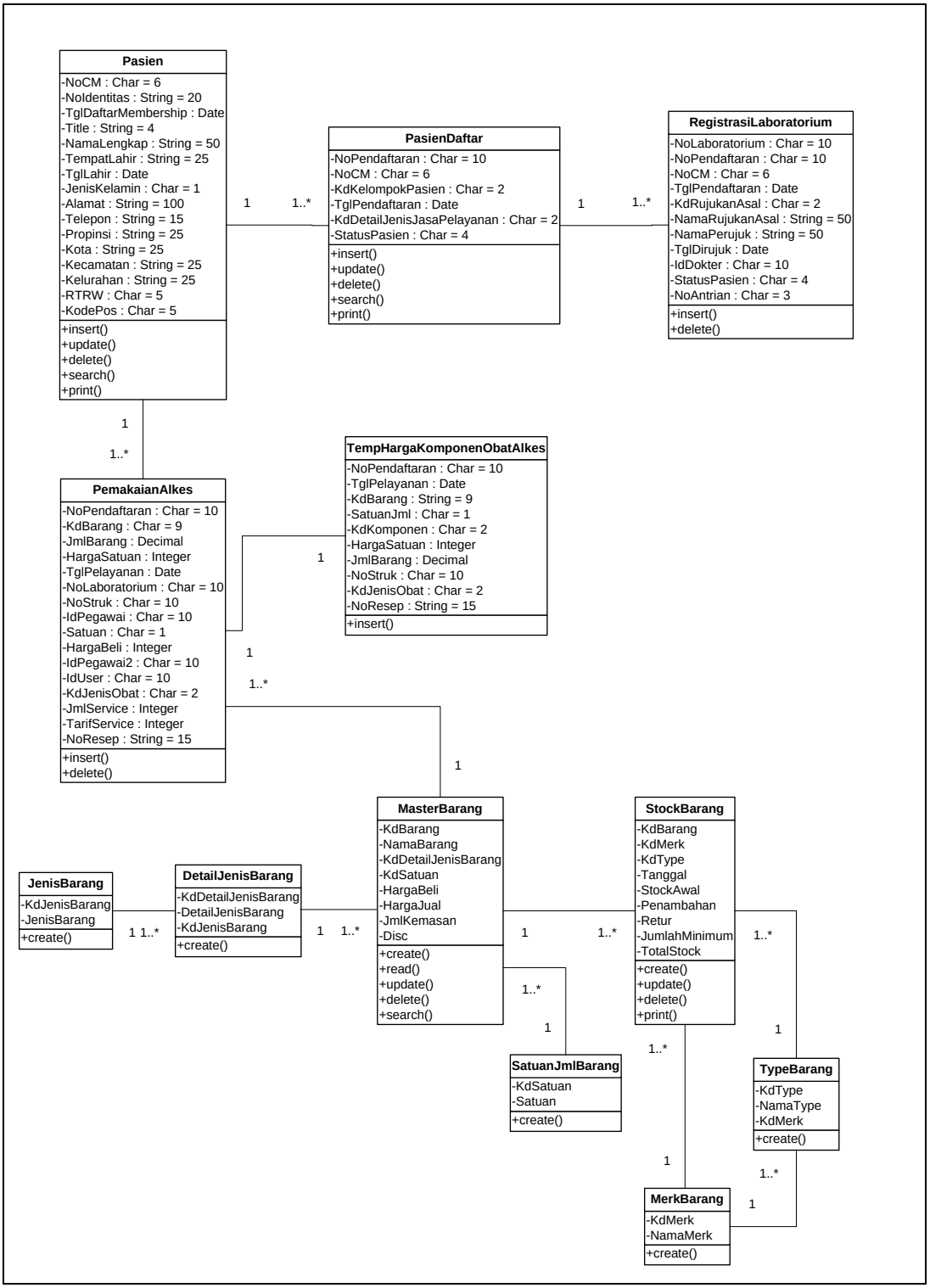
- **Class Diagram Registrasi Laboratorium**



- **Class Diagram Pelayanan Tindakan**



- **Class Diagram Transaksi Pelayanan**



- **Class Diagram Hasil Periksa Laboratorium**

