BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Sistem

Perancangan sistem adalah tahapan setelah analisis dari siklus pengembangan sistem yang didefinisikan dari kebutuhan-kebutuhan fungsional dan persiapan untuk membangun impelemntasi yang menggambarkan bagaimana suatu sistem dibentuk, yang dapat berupa penggambaran, perancangan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi juga menyangkut konfigurasi dari komponen-komponen perangkat keras dan perangkat lunak.

4.1.1. Tujuan Perancangan Sistem

Berdasarkan hasil analisis secara menyeluruh, maka dirancanglah Sistem Informasi Klinik Berbasis Aplikasi Web yang akan digunakan sebagai solusi yang ditawarkan dalam memperbaiki dan mengembangkan sistem yang sedang berjalan di Klinik Drg. Deni S.L. Dengan dirancangnya sistem yang berdasarkan kebutuhan informasi dari pihak yang bersangkutan maka diharapkan sistem informasi berbasis web aplikasi ini dapat berguna dan membantu dalam melaksanakan kegiatan nya dengan mudah.

4.1.2. Gambaran Umum Sistem Yang Diusulkan

Sistem yang dibuat adalah sistem informasi klinik, sistem ini dibuat dengan menggunakan data base pada sebuah komputer. Semua proses pengolahan data

yang terjadi di klinik dapat diinput, diproses, dilaporkan, dan disimpan kedalam data base hardisk komputer.

Dalam sistem informasi klinik berbasis web aplikasi ini petugas memiliki hak akses mengedit, menghapus dan mengupdate data sedangkan gamabaran dari hak akses yang diberikan berdasarkan tugas yang diberikan pada sistem informasi klinik Drg. Deni S.L berbasis web adalah sebagai berikut :

Tabel 4.1 Tugas Berdasarkan Pengguna

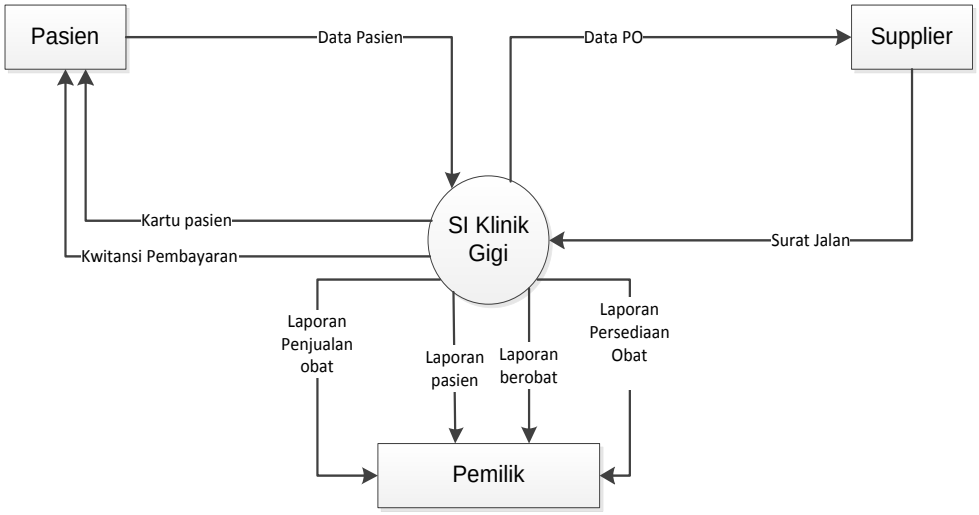
Pengguna	Tugas	Keterangan
Petugas	Kelola pendaftaran Kelola Persedian Barang Kelola Laporan	- Mengolah data pasien - Mengolah data pembelian, penerimaan dan retur barang - Mencetak laporan
Asisten	Kelola Pemeriksaan	- Mengolah data rekamedis
Kasir	Kelola Resep & Pembayaran	- Mengolah data resep - Mencetak kwitansi pembayaran

4.1.3. Perancangan Prosedur Yang Diusulkan

Perancangan prosedur merupakan awal dari pembuatan sistem yang akan dibuat, dimana dapat dilihat proses-proses apa saja yang nantinya diperlukan dalam pembuatan suatu sistem.

4.1.3.1.Diagram Konteks

Diagram konteks di bawah ini adalah diagram aliran data level 0 untuk menggambarkan aliran data yang terjadi pada sistem yang diusulkan.



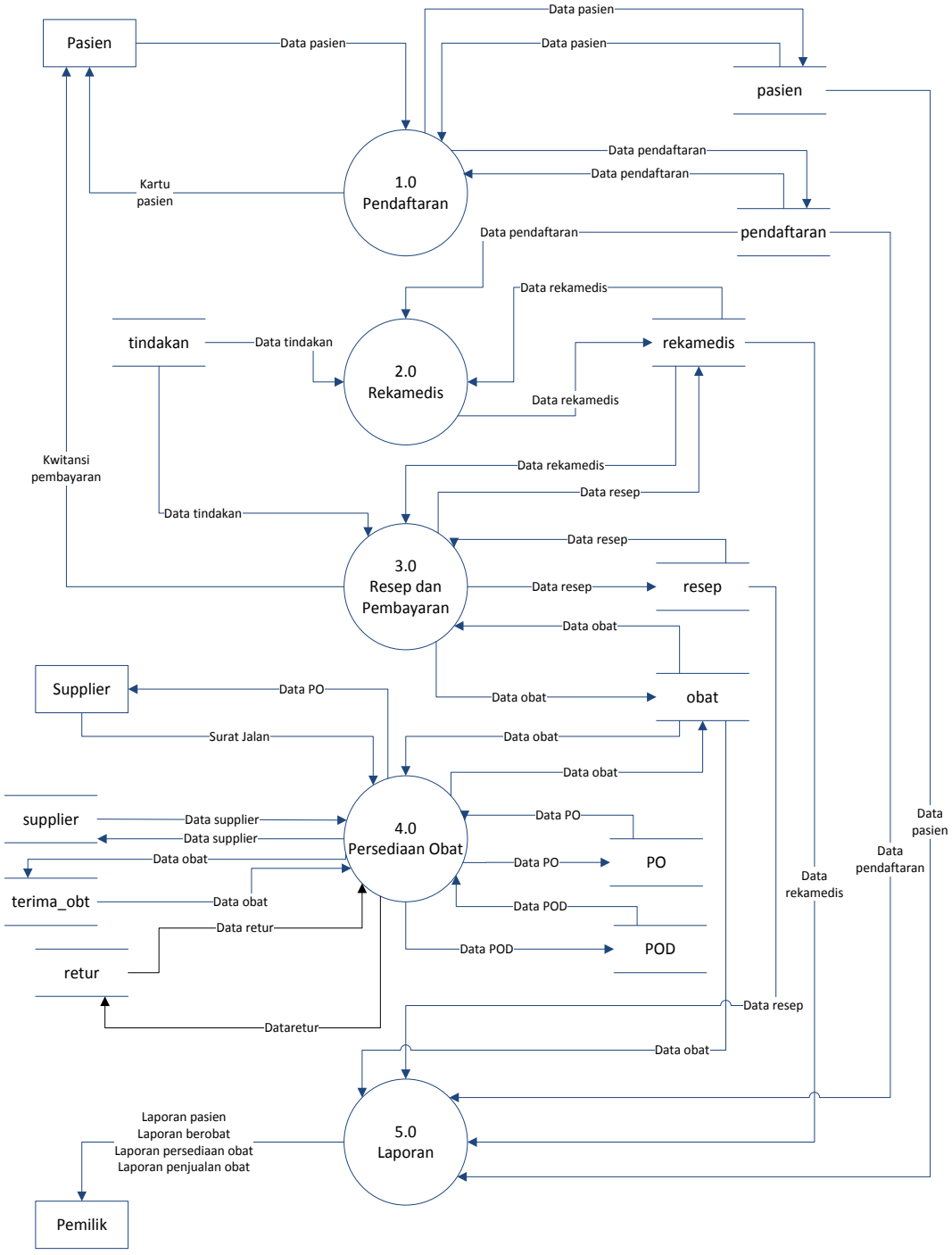
Gambar 4.1 Diagram Konteks yang diusulkan

4.1.3.2. DFD (Data Flow Diagram)

Data flow diagram merupakan uraian yang lebih terperinci berdasarkan diagram konteks yang telah dibuat sebelumnya. Di bawah ini akan diuraikan menjadi dua level data flow diagram yang sesuai dengan kebutuhan yang ada.

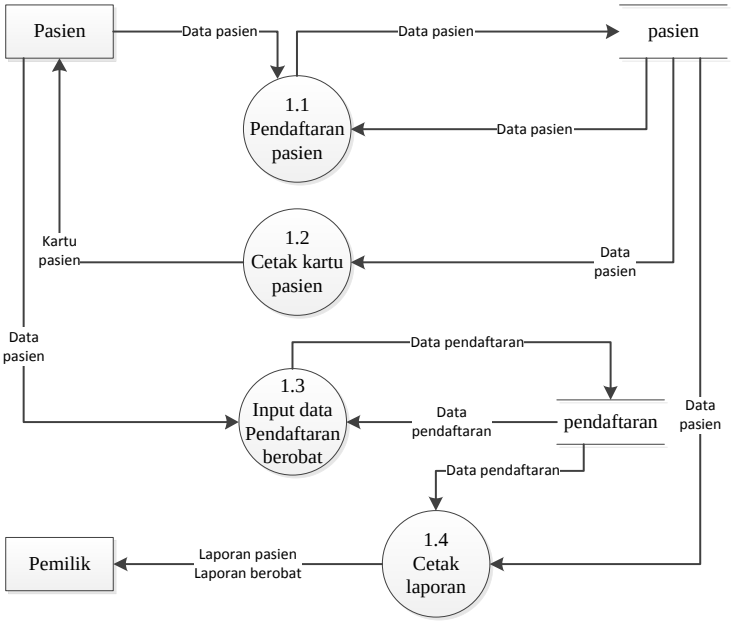
4.1.3.2.1. DFD Level 1 Klinik Drg Deni S.L

DFD level 1 merupakan data flow diagram tingkat pertama dari sistem yang diusulkan.



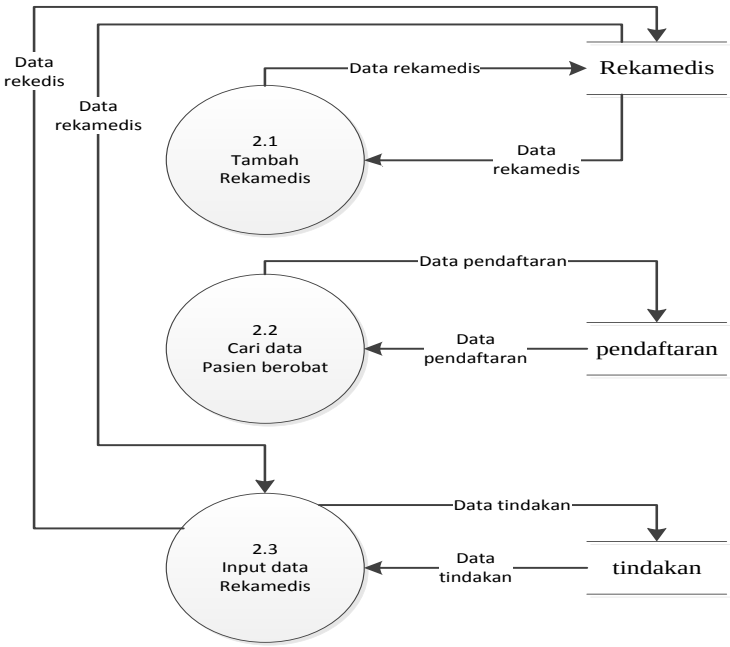
Gambar 4.2 DFD Level 1 yang diusulkan

4.1.3.2.2. DFD Level 2 Proses 1 Pendaftaran Klinik Drg. Deni S.L.



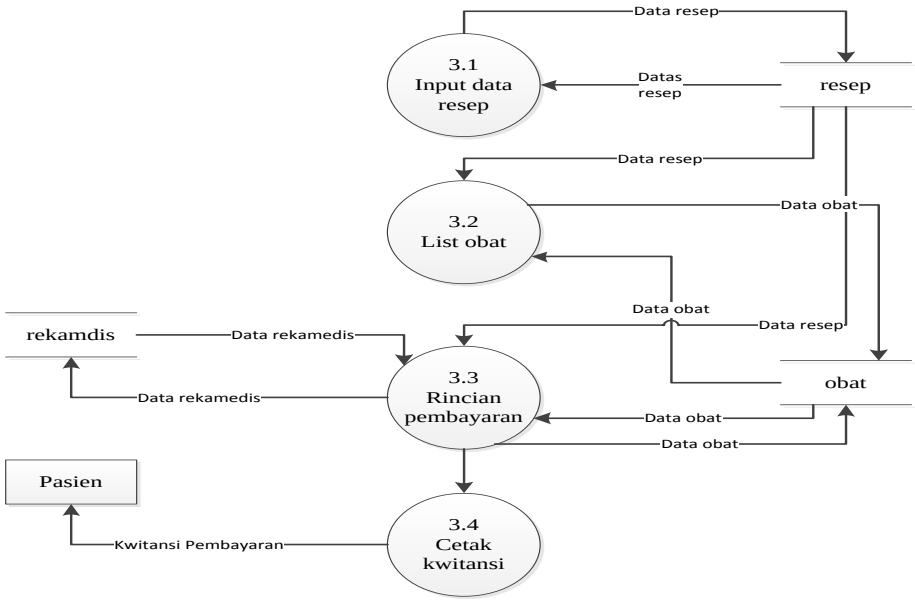
Gambar 4.3 Level 2 Proses Pendaftaran pasien yang diusulkan

4.1.3.2.3. DFD Level 2 Proses 2 Rekamedis Klinik Drg Deni S.L.



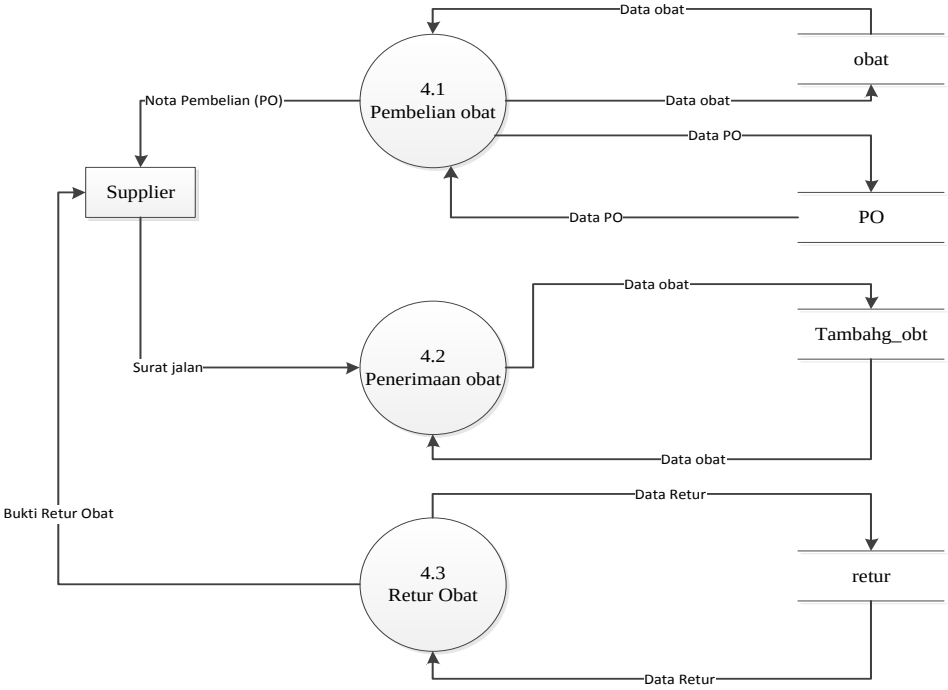
Gambar 4.4 Level 2 Proses 3 Rekamedis yang diusulkan.

4.1.3.2.4. DFD Level 2 Proses 3 Pembayaran Klinik Drg. Deni S.L.



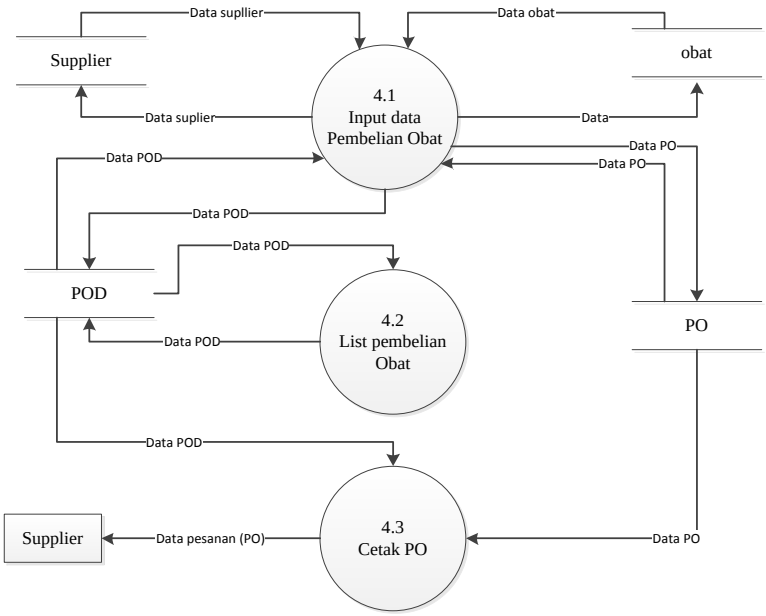
Gambar 4.5 Level 2 Proses 4 Pembayaran yang diusulkan.

4.1.3.2.5. DFD Level 2 Proses 4 Persediaan Obat Klinik Drg. Deni S.L.



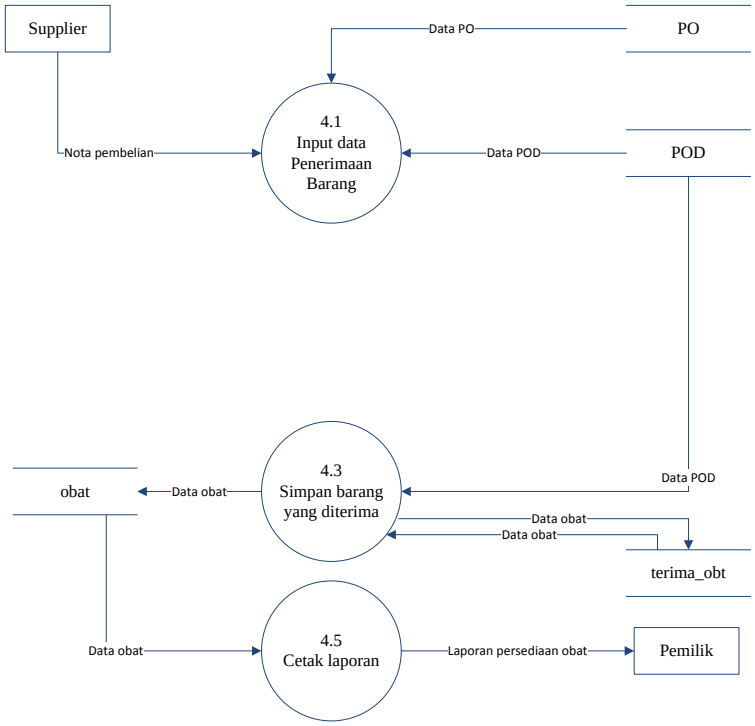
Gambar 4.6 Level 2 Proses 5 Persedian Obat yang diusulkan.

4.1.3.2.6. DFD Level 3 Proses 4 Pembelian Barang.



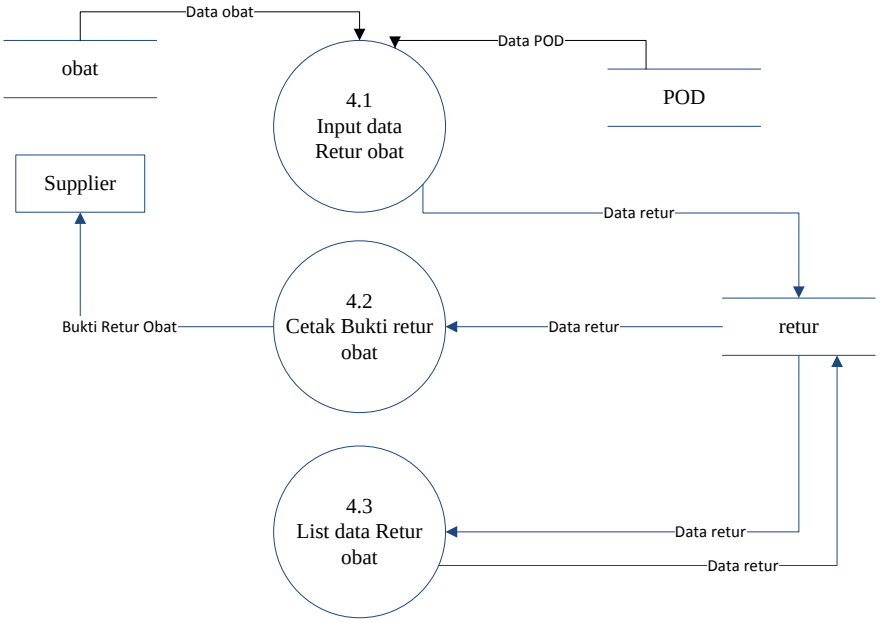
Gambar 4.7 Level 3 Proses 4 Pembelian Barang

4.1.3.2.7. DFD Level 3 Proses 4 Penerimaan Barang.



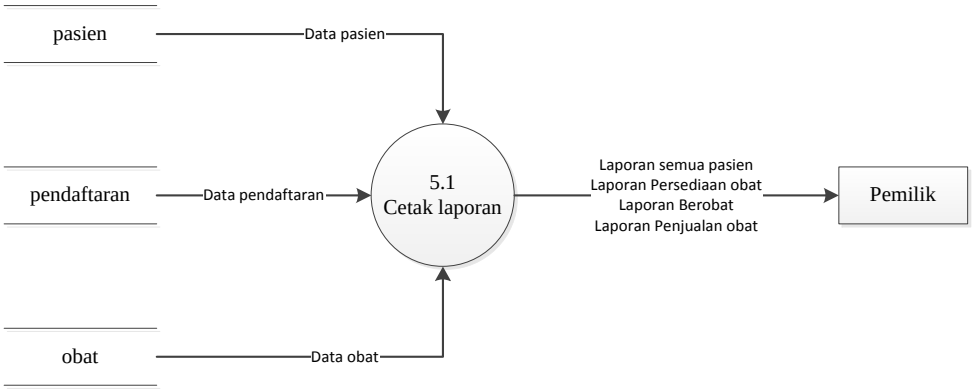
Gambar 4.8 Level 3 Proses 4 Penerimaan Barang

4.1.3.2.8. DFD Level 3 Proses 4 Retur Obat



Gambar 4.9 Level 3 Proses 4 Retur Obat

4.1.3.2.9. DFD Level 2 Proses 5 Laporan Klinik Drg Deni S.L.



Gambar 4.10 Level 2 Proses 6 Laporan yang diusulkan.

4.1.3.3.Kamus Data

Kamus data merupakan sebuah alat yang dapat membantu dalam perancangan sebuah sistem karena dapat menjelaskan atau mengidentifikasikan setiap *field* atau file yang dipakai oleh sistem.

1. Nama Arus Data : Data Pasien

   Alias : -

   Aliran Data : Pasien – proses 1.1 - Proses 1.2 - proses 1.3

   Pasien – Proses 2.1 – proses 2.2 – proses 2.3

   Pasien – proses 3.1

   Struktur Data : id_pasien, tgl_daftar, no_ktp, nama_pasien, tempat_lahir, tgl_lahir, jns_kelamin, agama, alamat,kota, stat_kawin, pekerjaan, no_hp

2. Nama Arus Data : Data pendaftaran Pasien berobat

   Alias : -

   Aliran Data : pasien – proses 2.1 – proses 2.3

   Pasien – proses 3.1

   Struktur Data : tgl_pendaftaran, id_pasien, id_pendaftaran,no_antrian, nama_pasien

3. Nama Arus Data : Data Rekam Medis

   Alias : -

   Aliran Data : rekam medis – proses 3.1– proses 3.2 – proses 3.3

   rekam medis – proses 4.1 – proses 4.2

   Struktur Data : no_rekamedis, id_pasien, tgl_rekamedis, umur, no_gigi, id_tindakan, diagnosa, tarif, stat

4. Nama Arus Data : Data pembayaran Dan Resep

   Alias : -

   Aliran Data : resep – proses 4.1 – proses 4.1 – proses 4.3 –

proses 4.4

Struktur Data : id_resep, no_rekamedis, tgl_resep, total

4.1.4. Perancangan Basis Data

Perancangan basis data merupakan perancangan yang digunakan untuk pembuatan dan penyimpanan data ke dalam sistem yang terdiri dari beberapa file *database*. Pada perancangan basis data yang akan dibahas, diantaranya adalah normalisasi, relasi tabel, entity relationship diagram, struktur file, dan kodifikasi.

4.1.4.1. Normalisasi

Normalisasi adalah suatu proses dimana elemen-elemen data dikelompokkan menjadi tabel-tabel, dimana dalam tabel tersebut terdapat entity-entiti dan relasi antar entity tersebut. Dalam proses normalisasi, field kunci memegang peranan penting dalam pembuatan tabel yang berisi entiti dan relasinya.

Berikut adalah bentuk normalisasi dari sistem yang dirancang :

1. **Bentuk Tidak Normal (Unnormal)**

kode_bulan*, nama_bulan, id_dokter*, nama_dokter, tempat_lahir, tgl_lahir, jns-kel, agama, alamat, kota, stat_kawin, no_hp, speasialis, id_hari*, nama_hari, id_jadwal*, id_hari**, jam, id_menu*, nama_menu, link, aktif, urutan, akses_admin, akses_petugas, akses_dokter, id_jns_obt*, nama_jns_obat, id_obat*, id_jns_obt**, nama-obt, satuan, stok, harga, ket, id_pasien*, tgl_daftar,

no_ktp, nama_pasien, tempat_lahir, tgl_lahir, jns_kel, agama, alamat, kota, stat_kawin, pekerjaan, no_hp, tgl_pendaftaran, id_pasien**, id_pendaftaran*, no_antran, nama_pasien, status_berobat, no_po*, tgl_po, tot, status, no_po**, id_obt**, id_supplier**, jlm, sub, no_rekamedis*, id_pasien**, tgl_rekamedis, umur, no_gigi, id_tindakan*, diagnosa, tarif, stat, id_resep*, no_rekamedis**, tgl_resep, total, id_resep**, id_obt**, qty, subtot, no_retur*, tgl_retur, id_supplier**, id_obat**, stok, jml_retur, ustok, id_supplier*, npwp, an_npwp, nama_sup, alamat, kota, telepon, fax, email, no_terima*, tgl_terima, no_po**, id_supp, keterangan, no_terima**, id_obt**, jumlah, stok, id_tindakan*, nama_tindakan, tarif.

2. Bentuk Normal Pertama (1NF)

kode_bulan*, nama_bulan, id_dokter*, nama_dokter, tempat_lahir, tgl_lahir, jns_kel, agama, alamat, kota, stat_kawin, no_hp, speasialis, id_hari*, nama_hari, id_jadwal*, jam, id_menu*, nama_menu, link, aktif, urutan, akses_admin, akses_petugas, akses_dokter, id_jns_obt*, nama_jns_obat, id_obat*, nama_obt, satuan, stok, harga, ket, id_pasien*, tgl_daftar, no_ktp, nama_pasien, pekerjaan, tgl_pendaftaran, id_pendaftaran*, no_antrian, tgl_po, tot, status, no_po*, id_obt**, id_supplier**, jml, sub, no_rekamedis*, tgl_rekamedis, umur, no_gigi, id_tindakan**, diagnosa, tarif, stat, id_resep*, tgl_resep, total, qty, subtot, no_retur*, tgl_retur, jml_retur, ustok, npwp, an_npwp, nama_sup, telepon, fax, email, no_terima*, tgl_terima, id_supp, keterangan, jumlah, nama_tindakan, tarif.

3. Bentuk Normal Kedua (2NF)

bulan = { kode_bulan*, nama_bulan}

dokter = { id_dokter*, nama_dokter, tempat_lahir, tgl_lahir, jns_kel, agama, alamat, kota, stat_kawin, no_hp, speasialis}

hari = { id_hari*, nama_hari}

jadwal = { id_jadwal*, id_hari**, jam}

menu = { id_menu*, nama_menu, link, aktif, urutan, akses_admin, akses_petugas, akses_dokter}

Jns_obat = { Id_jns_obt*, nama_jns_obat}

Obat = {id_obat*, id_jns_obt**, nama-obt, satuan, stok, harga, Ket}

Pasien = {id_pasien*, tgl_daftar, no_ktp, nama_pasien, tempat_lahir, tgl_lahir, jns_kel, agama, alamat, kota, stat_kawin, pekerjaan, no_hp}

Pendaftaran ={tgl_pendaftaran, id_pasien*, id_pendaftaran**, no_antran, nama_pasien}

Po = { no_po*, tgl_po, tot, status}

Pod = { no_po**, id_obt**, id_supplier**, jlm, sub}

Rekam medis ={no_rekamedis*, id_pasien**, tgl_rekamedis, umur, no_gigi, id_tindakan**, diagnosa, tarif, stat}

Resep = { id_resep*, no_rekamedis**, tgl_resep, total}

Resep_d = { id_resep**, id_obt**, qty, subtot}

Retur = { no_retur*, tgl_retur, jml_retur, ustok}

Supplier = {id_supplier*, npwp, an_npwp, nama_sup, alamat, kota, telepon, fax, email}

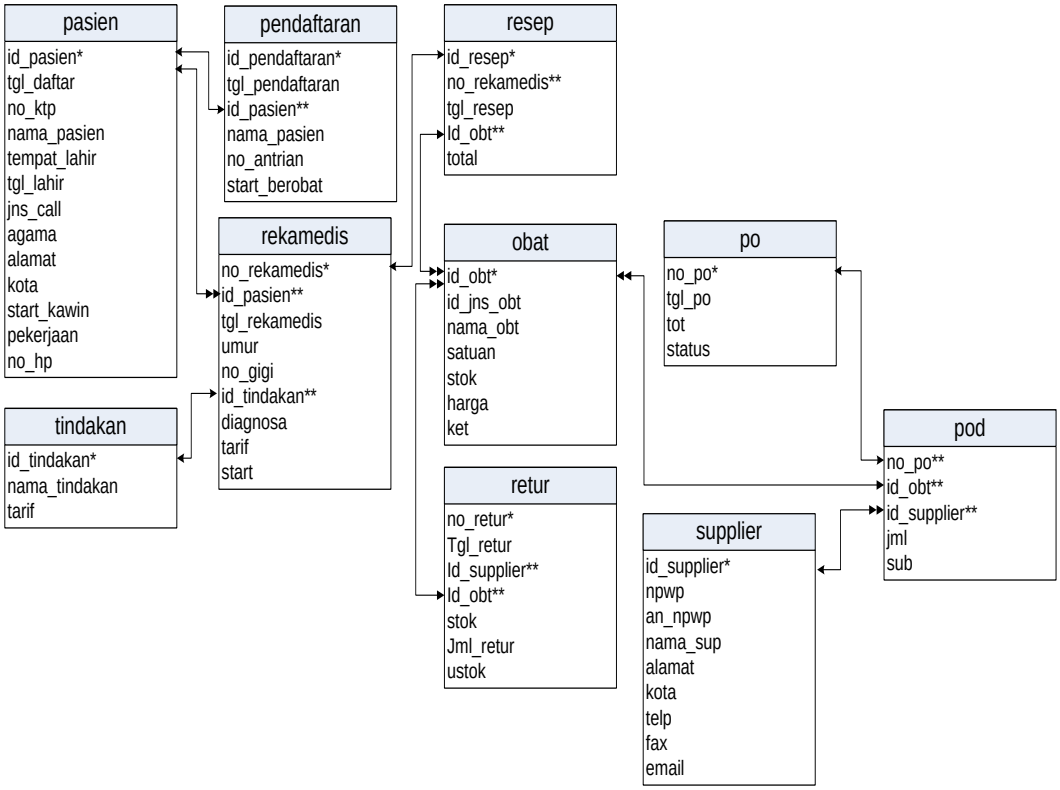
Terima_obt = { no_terima*, tgl_terima, no_po**, id_supp, keterangan}

Terima_obtd = { no_terima**, id_obt**, jumlah, stok}

Tindakan = { id_tindakan*, nama_tindakan, tarif}

4.1.4.2.Relasi Tabel

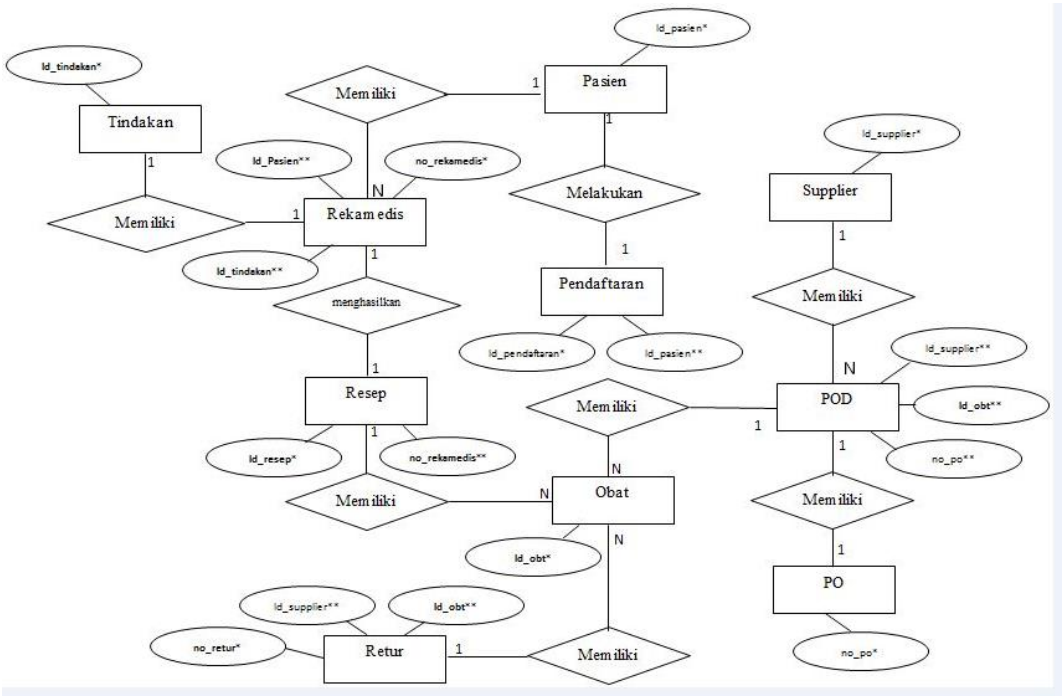
Tabel relasi merupakan hubungan yang terjadi pada suatu tabel dengan tabel lainnya. Masing – masing tabel yang dihubungkan oleh kunci primer di salah satu tabel, dan kunci tamu di tabel yang lain yang merupakan salinan dari kunci primer yang ada pada tabel sebelumnya. Berikut adalah relasi tabel yang telah dirancang :



Gambar 4.11 Relasi Tabel Klinik Drg Deni S.L.

4.1.4.3. ERD (Entity Relationship Diagram)

Entity Relationship Diagram (ERD) berguna untuk memodelkan sistem yang nantinya akan dikembangkan basis datanya. gambaran dari hubungan antara entitas yang terlibat di dalam sistem, dan masing - masing entitas tersebut memiliki kunci primer.Berikut adalah ERD dari perancangan sistem informasi klinik Drg. Deni S.L.



Gambar 4.12 ERD *(Entity Relationship Diagram)*Klinik Drg. Deni S.L.

4.1.4.4.Struktur File

Struktur filedigunakan untuk perancangan Sistem karena fileini akan menentukan struktur fisik basis data dan jenis data.Arsitektur yang menjelaskan field – field yang membentuk suatu file data yang terlibat di dalam siste informasi.

Struktur file pada basis data yang diusulkan sebagai berikut :

1. **Struktur File Data Bulan.**

   Key field :kode_bulan

   Fungsi :Mencatat data bulan

Tabel 4.2 File Data Bulan

No	Nama Field	Type	Length	Keterangan
1	kode_bulan	varchar	2	Primary key
2	nama_bulan	varchar	10	

2. **Struktur File Data Dokter.**

   Key field : id_dokter

   Fungsi : Mencatat data dokter

Tabel 4.3 File Data Dokter

No	Nama Field	Type	Length	Keterangan
1	Id_dokter	Varchar	5	Primary key
2	Nama_dokter	Varchar	30	
3	Tempat_lahir	Varchar	15	
4	Tgl_lahir	Date		
5	Jns_kel	Varchar	10	

6	Agama	Varchar	10	
7	Alamat	Text		
8	Kota	Varchar	15	
9	Stat_kawin	Varchar	10	
10	No_hp	Varchar	15	
11	spesialis	Varchar	15	

3. Struktur File Data Hari.

Key field : id_hari

Fungsi : Mencatat data hari

Tabel 4.4 File Data Hari

No	Nama Field	Type	Length	Keterangan
1	id_hari	varchar	5	Primary key
2	nama_hari	varchar	15	

4. Struktur File Data Jadwal.

Key field : id_jadwal

Fungsi : Mencatat data jadwal

Tabel 4.5 File Data Jadwal

No	Nama Field	Type	Length	Keterangan
1	id_jadwal	Int	3	Primary key
2	id_hari	Int	2	
3	nama_hari	Varchar	15	
4	jam	Varchar	15	

5. Struktur File Data Jenis Obat.

Key field : id_jns_obt

Fungsi : Mencatat data jenis obat

Tabel 4.6 File Data Jenis Obat

No	Nama Field	Type	Length	Keterangan
1	id_jns_obt	int	3	Primary key
2	nama_jns-obt	varchar	20	

6. Struktur File Data Menu.

Key field : id_menu

Fungsi : Mencatat data menu

Tabel 4.7 File Data Menu

No	Nama Field	Type	Length	Keterangan
1	id_menu	int	5	Primary key
2	akses_admin	enum	-	
3	akses_dokter	enum	-	
4	akses_petugas	enum	-	
5	aktif	enum	-	
6	urutan	int	5	
7	nama_menu	varchar	20	
8	link	varchar	30	

7. Struktur File Data Obat.

Key field : id_obt

Fungsi : Mencatat data obat

Tabel 4.8 File Data Obat

No	Nama Field	Type	Length	Keterangan
1	id_obt	Varchar	5	Primary key
2	id_jns_obt	Int	3	
3	nama_obt	Varchar	25	
4	satuan	Varchar	14	
5	stok	Varchar	9	
6	harga	Varchar	9	
7	ket	Text	-	

8. Struktur File Data Pasien.

Key field : id_pasien

Fungsi : Mencatat data pasien

Tabel 4.9 File Data Pasien

No	Nama Field	Type	Length	Keterangan
1	id_pasien	Varchar	10	Primary key
2	tgl_daftar	Date	-	
3	no_ktp	Varchar	25	
4	nama_pasien	Varchar	30	
5	Tempat_lahir	Varchar	10	
6	Tgl_lahir	Date	-	
7	Jns_kelamin	Varchar	10	
8	Agama	Varchar	10	
9	Alamat	Text	-	
10	Kota	Varchar	50	
11	Stat_kawin	Varchar	10	
12	Pekerjaan	Varchar	15	
13	No_hp	Varchar	14	

9. Struktur File Data Pendaftaran.

Key field : id_pendaftaran

Fungsi : Mencatat data pendaftaran

Tabel 4.10 File Data Pendaftaran

No	Nama Field	Type	Length	Keterangan
1	id_pendaftaran	Varchar	11	Primary key
2	tgl_pendaftaran	Date	-	
3	id_pasien	Varchar	10	
4	no_antrian	Varchar	3	
5	nama_pasien	Varchar	40	

10. Struktur File Data Petugas.

Key field : id_petugas

Fungsi : Mencatat data petugas

Tabel 4.11 File Data Petugas

No	Nama Field	Type	Length	Keterangan
1	id_petugas	Int	10	Primary key
2	username	Varchar	15	

3	password	Varchar	25	
4	nama_petugas	Varchar	30	
5	Tmpt_lahir	Varchar	10	
6	Tgl_lahir	Date	-	
7	Jns_kelamin	Varchar	10	
8	Agama	Varchar	10	
9	Alamat	Text	-	
10	Kota	Varchar	50	
11	Stat_kawin	Varchar	10	
12	No_hp	Varchar	15	
13	status	Varchar	14	

11. Struktur File Data Po.

Key field : no_po

Fungsi : Mencatat data po

Tabel 4.12 File Data PO

No	Nama Field	Type	Length	Keterangan
1	no_po	Varchar	14	Primary key
2	tgl_po	Date	-	
3	tot	Varchar	15	
4	status	Varchar	1	

12. Struktur File Data Pod.

Key field : no_po

Fungsi : Mencatat data pod

Tabel 4.13 File Data POD

No	Nama Field	Type	Length	Keterangan
1	no_po	Varchar	14	Primary key
2	id_obt	Varchar	5	
3	id_supplier	Varchar	15	

4	Jml	Varchar	4	
5	Sub	Varchar	14	

13. Struktur File Data Rekamedis.

Key field : no_rekamedis

Fungsi : Mencatat data rekamedis

Tabel 4.14 File Data Rekamedis

No	Nama Field	Type	Length	Keterangan
1	no_rekamedis	Varchar	12	Primary key
2	id_pasien	Varchar	10	
3	tgl_rekamedis	Date	-	
4	umur	Varchar	3	
5	no_gigi	Varchar	8	
6	id_tindakan	Int	4	
7	diagnosa	Text	-	
8	tarif	Varchar	9	
9	stat	Varchar	1	

14. Struktur File Data Resep.

Key field : id_resep

Fungsi : Mencatat data resep

Tabel 4.15 File Data Resep

No	Nama Field	Type	Length	Keterangan
1	id_resep	Varchar	14	Primary key
2	no_rekamedis	Varchar	14	
3	tgl_resep	Date	-	
4	total	Varchar	9	

15. Struktur File Data Resep Diterima.

Key field : id_resep

Fungsi : Mencatat data resep diterima

Tabel 4.16 File Data Resep Diterima

No	Nama Field	Type	Length	Keterangan
1	id_resep	Varchar	14	Primary key
2	id_obt	Varchar	14	
3	qyt	Varchar	9	

4	subtot	Varchar	9	

16. Struktur File Data Supplier.

Key field : no_supplier

Fungsi : Mencatat data supplier

Tabel 4.17 File Data Supplier

No	Nama Field	Type	Length	Keterangan
1	id_supplier	Varchar	6	Primary key
2	npwp	Varchar	30	
3	an_npwp	Varchar	50	
4	nama_sup	Varchar	40	
5	alamat	Text	-	
6	kota	Varchar	14	
7	telpon	Varchar	14	
8	fax	Varchar	14	
9	email	Varchar	40	

17. Struktur File Data Terima obat.

Key field : no_terima

Fungsi : Mencatat data Terima obat

Tabel 4.18 File Data Terima Obat

No	Nama Field	Type	Length	Keterangan
1	no_terima	Varchar	12	Primary key
2	tgl_terima	Date	-	
3	No_po	Varchar	12	
4	Id_supp	Varchar	6	
5	keterangan	Text	-	

18. Struktur File Data Terima stok obat.

Key field : no_terima

Fungsi : Mencatat data Terima stok obat

Tabel 4.19 File Data Terima Stok Obat.

No	Nama Field	Type	Length	Keterangan
1	no_terima	Varchar	12	Primary key
2	id_obat	Varchar	5	
3	jumlah	Varchar	9	

4	stok	Varchar	9	

19. Struktur File Data Tindakan.

Key field : id_tindakan

Fungsi : Mencatat data tindakan

Tabel 4.20 File Data Tindakan

No	Nama Field	Type	Length	Keterangan
1	id_tindakan	Int	2	Primary key
2	nama_tindakan	Varchar	40	
3	tarif	Varchar	12	

20. Struktur File Data Retur

Key field : no_retur

Fungsi : Mencatat data retur

Tabel 4.21 File Data Retur

No	Nama Field	Type	Length	Keterangan
1	no_retur	Varchar	12	Primary key
2	tgl_retur	Date	-	
3	id_supplier	Varchar	6	

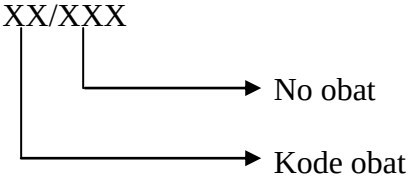
4	id_obt	Varchar	5	
5	stok	Varchar	9	
6	jml_retur	Varchar	9	
7	ustok	Varchar	9	

4.1.4.5.Kodifikasi

Kodifikasi berfungsi untuk pengklasifikasian data agar lebih mudah dalam proses memasukan data, menyusun data, ataupun menghapus data dari sistem yang ada. Untuk lebih jelasnya berikut adalah pengkodifikasian dalam sistem yang diusulkan.

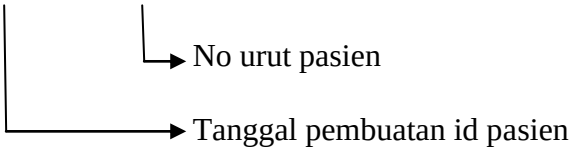
Berikut adalah kode – kode yang digunakan dalam sistem informasi klinik Drg. Deni S.L.

id_obt : XX/XXX



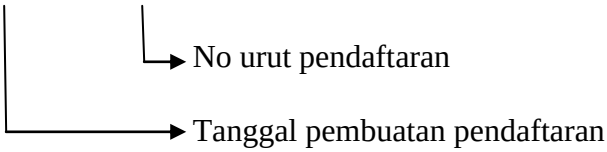
Contoh : OB001

id_pasien : XXXXXXXX/XX



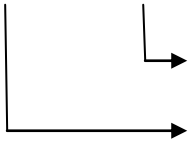
Contoh : 2014050601

id_ pendaftaran : XXXXXXXX/XX



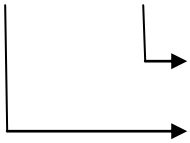
Contoh : 2014220601

no_po : XXXXXXXX/XXXX



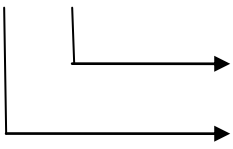
Contoh : 201418060001

no_rekamedis : XXXXXXXX/XXX



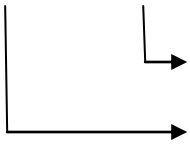
Contoh : 20141806008

id_supplier :XXX/XXX



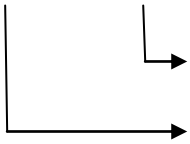
Contoh :SUP001

no_terima : XXXXXXXX/XXXX



Contoh : 2014113060002

no_retur : XXXXXXXX/XXXX



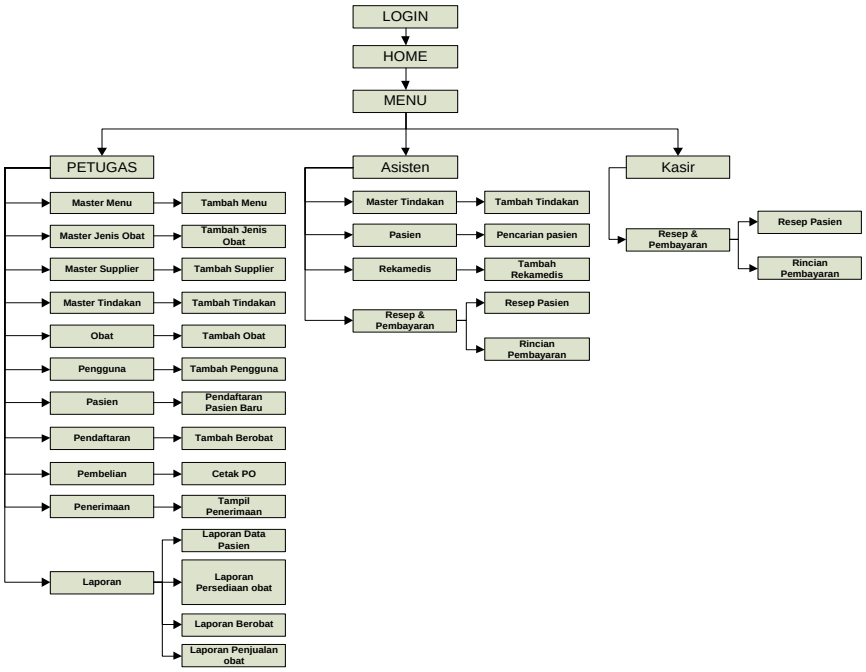
Contoh : 2014113060004

4.2. Perancangan Antar Muka

Perancangan antar muka dimaksudkan untuk memperlihatkan bagaimanakah bentuk dari perangkat lunak yang akan dibangun nantinya berdasarkan struktur sistem yang telah dibuat. Perancangan antar muka ini meliputi perancangan struktur menu, perancangan inputdan perancangan output.

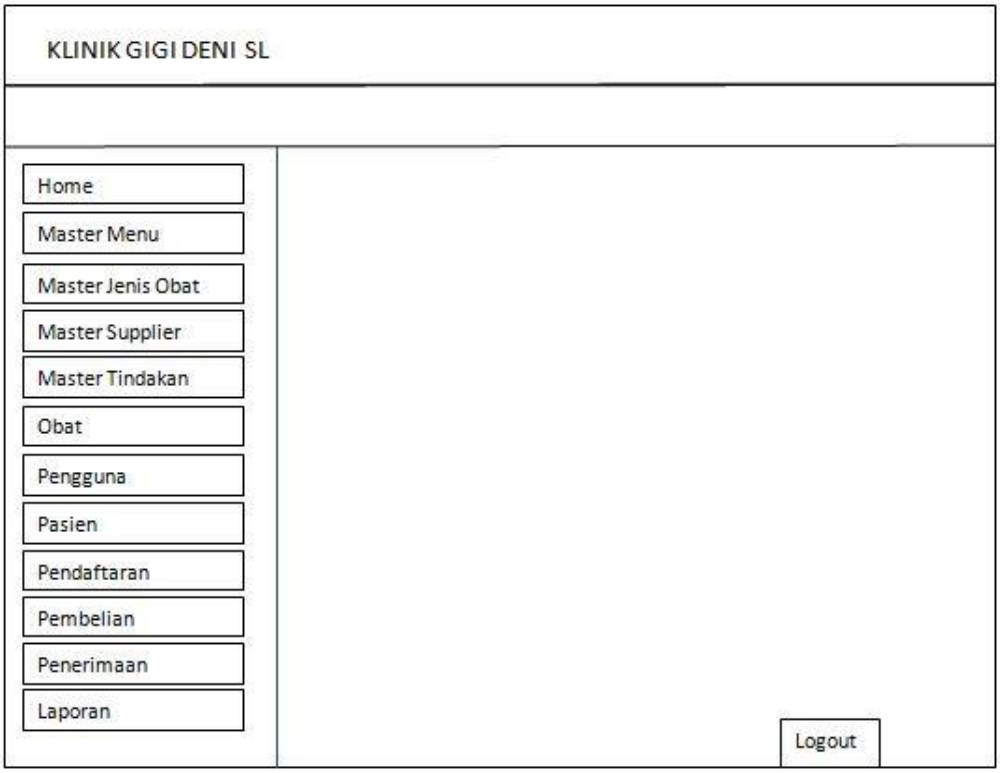
4.2.1. Struktur Menu

Struktur menu adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program sehingga pada saat menjalankan komputer pemakai (*user*) tidak mengalami kesulitan dalam memilih menu-menu yang digunakan. Struktur menu dibuat sebagai gambaran mengenai skema program yang akan dirancang, Berikut adalah struktur menu perancangan sistem informasi klinik Drg. Deni S.L :



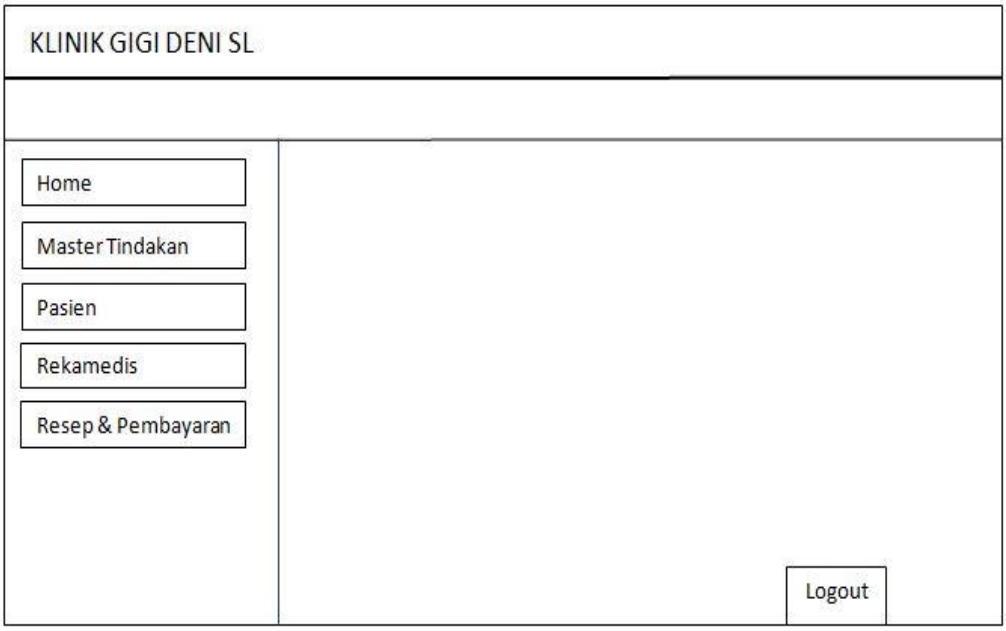
Gambar 4.13 Struktur Menu Klinik Drg. Deni S.L.

1. Menu Petugas



Gambar 4.14 Struktur Menu Petugas

2. Menu Asisten Dokter



Gambar 4.15 Struktur Menu AsistenDokter

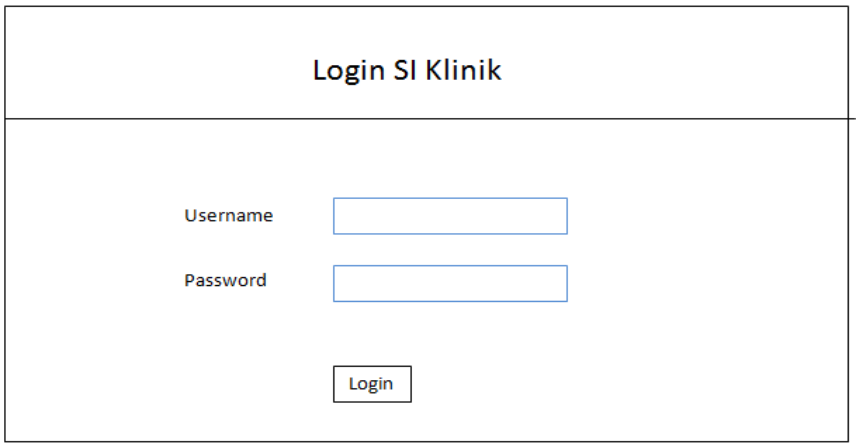
4.2.2. Perancangan Input

Perancangan input merupakan desain yang dirancang untuk menerima masukan dari pengguna sistem. Untuk lebih jelas mengenai bentuk-bentuk yang dirancang, berikut adalah gambarnya :

4.2.2.1. Perancangan Input Petugas

1. **Desain input login**

Tampilan Form Login Userini digunakan untuk masuk kedalam sistem, yang dilakukan oleh petugas pendaftaran, kasir dan asisten dokter. Didalam Form Login ini terdapat username dan passwordyang harus diisi dengan benar.



Gambar 4.16 Desain Halaman Login

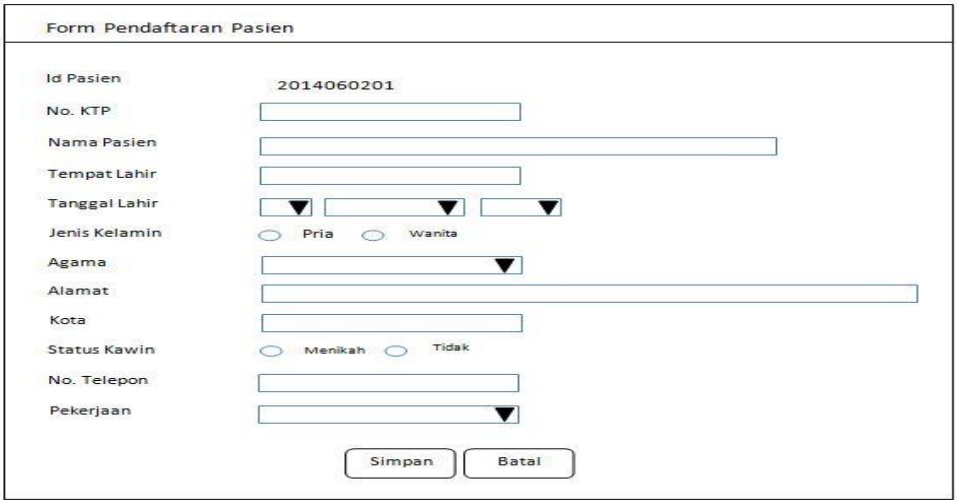
Penjelasan tombol :

a. Login

Untuk loginkedalam sistem, jika username dan password masih salah maka user tidak di izinkan untuk masuk kedalam sistem dan akan terus ditampilkan form login tersebut.

2. Desain InputPendaftaran Pasien

Tampilan Formpendaftaran pasien ini digunakan untuk pasien yang akanmendaftar sebagai pasien baru. Dan dapat digunakan oleh dokter untuk pencarian riwayat rekamedis pasien.



Gambar 4.17 Desain Input Pendaftaran Pasien

Penjelasan tombol :

a. Simpan

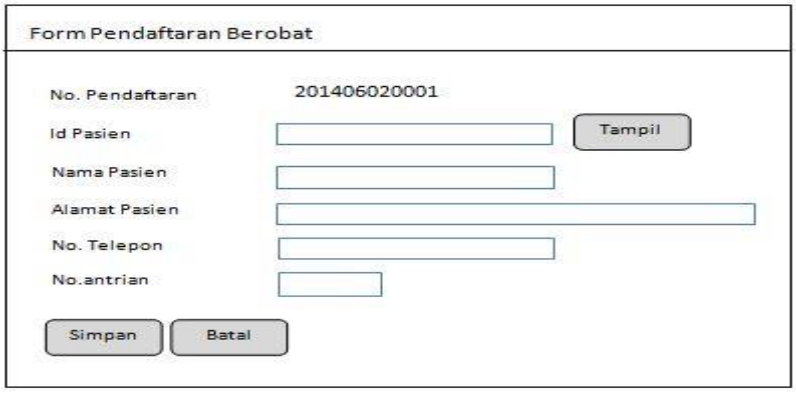
Digunakan untuk menyimpan dan mengirimkan data calon pasienyang telah diisikan pada formpendaftaran pasien kedalam sistem.

b. Batal

Digunakan untuk membatalkan data yang telah diisikan pada form pendaftaran pasien.

3. Desain Input Pendaftaran Berobat

Tampilan Formpendaftaran berobat ini digunakan untuk pasien yang ingin mendaftar saat akan berobat.



Gambar 4.18 Desain Input Pendaftaran Berobat

Penjelasan tombol :

a. Tampil

   Digunakan untuk menampilkan data pasien pada form secara automatis setelah mengetikan nomor id pasien kedalam sistem

b. Simpan

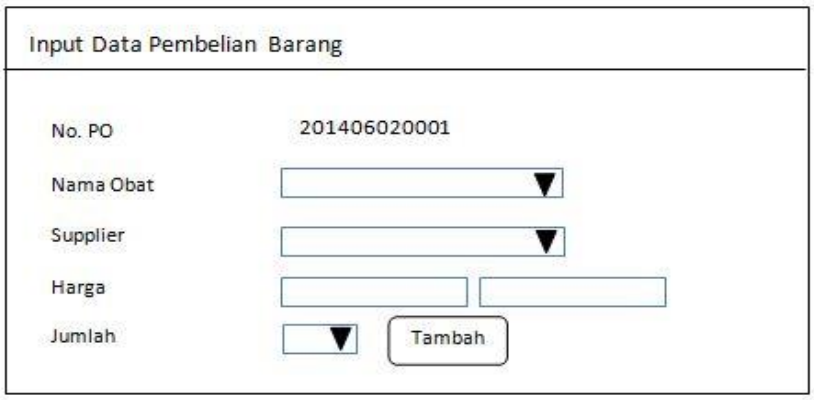
   Digunakan untuk menyimpan dan mengirimkan data pasien yang akan berobat,yang telah diisikan oleh prosespenggunaan tombol tampil pada form.

c. Batal

   Digunakan untuk membatalkan data yang telah diisikan pada form pendaftaran berobat.

4. Desain Input Data Pembelian Barang

Tampilan Forminput data pembelian ini digunakan petugas untuk membuat daftar list obat yang akandibeli.



Gambar 4.19 Desain Input Data Pembelian Barang

Penjelasan tombol :

a. Simpan

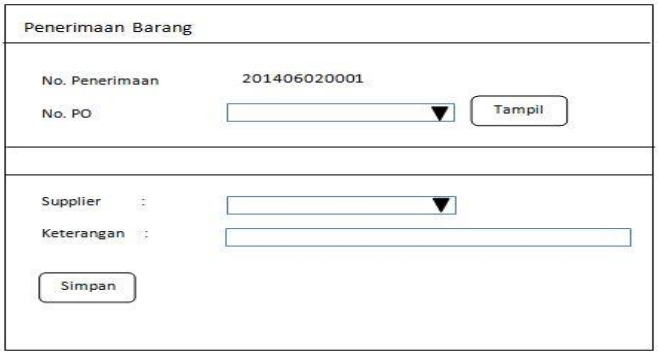
Digunakan untuk menyimpan dan mengirimkan data pembelian barangyang telah diisikan pada formpembelian barang kedalam sistem.

b. Batal

Digunakan untuk membatalkan data yang telah diisikan pada form pembelian barang.

5. Desain Input Data Penerimaan Barang.

Tampilan Forminput data penerimaan barang ini digunakan petugas untuk melihat barang yang dibeli dan kemudian menyimpan stok barang kedalam data obat.Dan dapat digunakan oleh administrator.



Gambar 4.20 Desain Input Penerimaan Barang

Penjelasan tombol :

a. Tampil

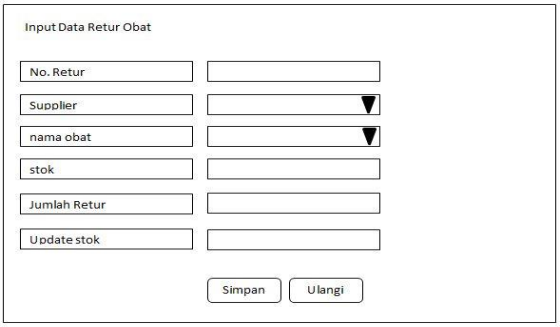
Digunakan untuk menampilkan data pembelian barangyang telah diisikan pada formpenerimaan barang kedalam sistem.

b. Simpan

Digunakan untuk mengirimkan dan menyimpan data pembelian barang pada stok obat yang telah diisikan pada form penerimaan barang.

6. Desain Input Retur Barang

Form input data retur obat yang digunakan oleg petugas untuk pembuatan retur obat.



Gambar 4.21 Desain Input Data Retur Obat

Penjelasan tombol :

a. Simpan

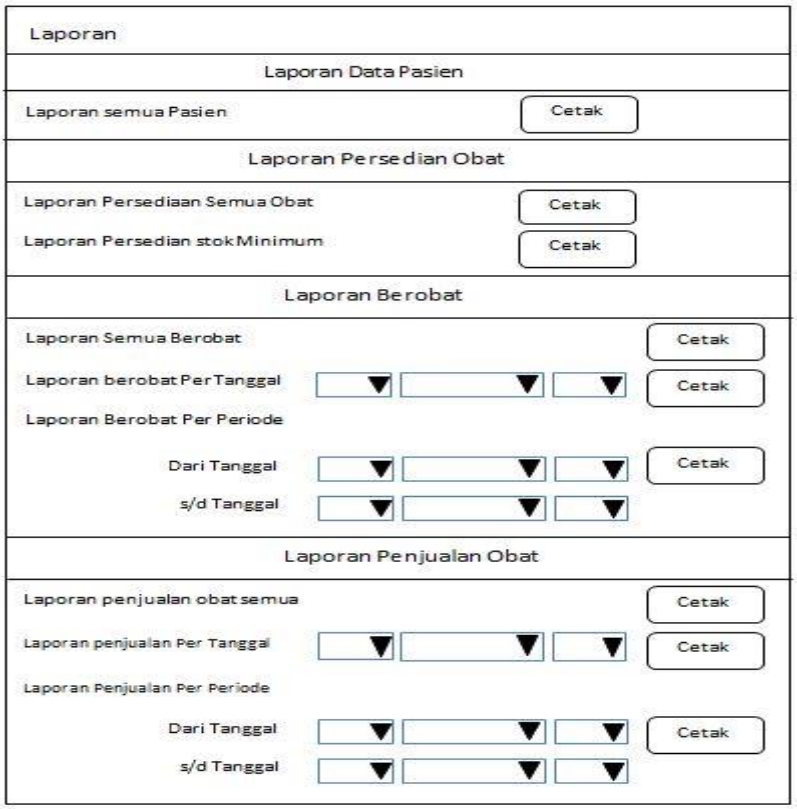
Digunakan untuk menampilkan data retur obat yang telah diisikan pada form input data retur obat.

b. Ulangi

Digunakan untuk mengosongkan kembali field pada form input data retur obat.

7. Desain Input Laporan.

Tampilan Forminput data laporan ini digunakan petugas untuk membuat laporan.



Gambar 4.22 Desain Input Laporan

Penjelasan tombol :

a. Cetak 1

   Digunakan untuk menampilkan dan mencetak laporan semua pasien.

b. Cetak 2

   Digunakan untuk menampilkan dan mencetak laporan semua persediaan obat.

c. Cetak 3

   Digunakan untuk menampilkan dan mencetak laporan persediaan stok minimum.

d. Cetak 4

   Digunakan untuk menampilkan dan mencetak laporan semua berobat.

e. Cetak 5

   Digunakan untuk menampilkan dan mencetak laporan berobat per tanggal.

f. Cetak 6

   Digunakan untuk menampilkan dan mencetak laporan berobat per periode.

g. Cetak 7

   Digunakan untuk menampilkan dan mencetak laporan penjualan semua obat.

h. Cetak 8

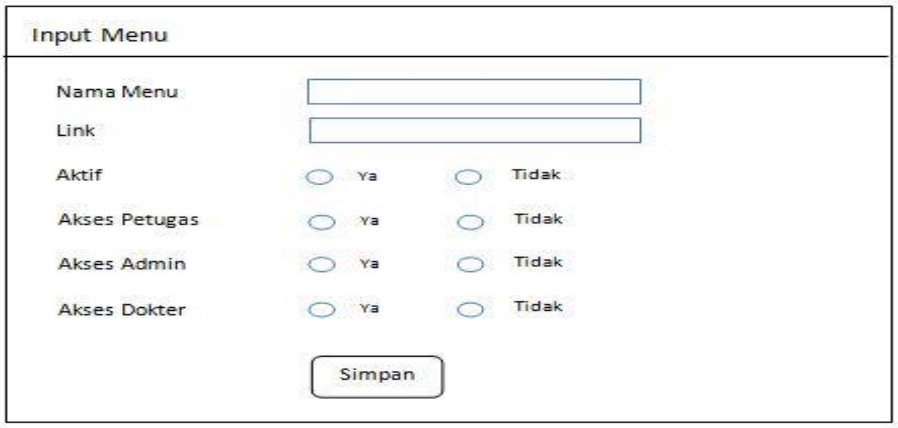
   Digunakan untuk menampilkan dan mencetak laporan penjualan obat per tanggal.

i. Cetak 9

Digunakan untuk menampilkan dan mencetak laporan penjualan obat per periode.

8. Desain Input Menu.

Tampilan Forminput menu ini digunakan oleh petugas untuk pembuatan dan menyimpan menu baru.



Gambar 4.23 Desain Input Menu

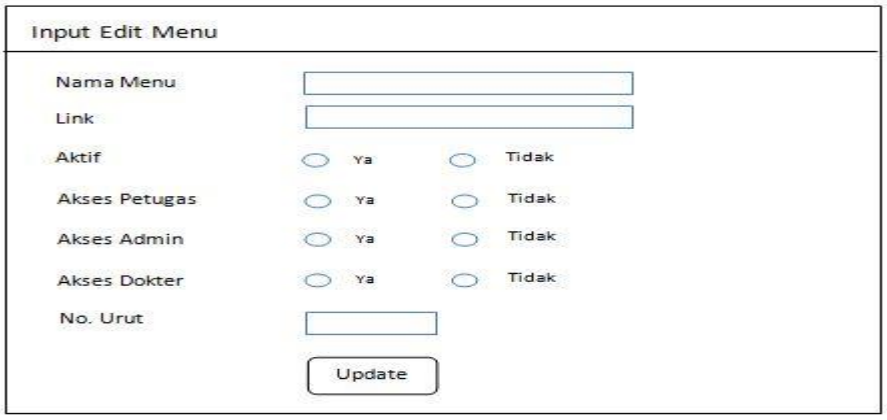
Penjelasan Tombol :

a. Simpan

Digunakan untuk menyimpan dan mengirimkan data menu baru yang telah diisikan kedalam form input menu kedalam sistem.

9. Desain Input Edit Menu.

Tampilan Forminput edit menu ini digunakan oleh petugas untuk mengedit menu yang sudah ada.



Gambar 4.24 Desain Input Edit Menu.

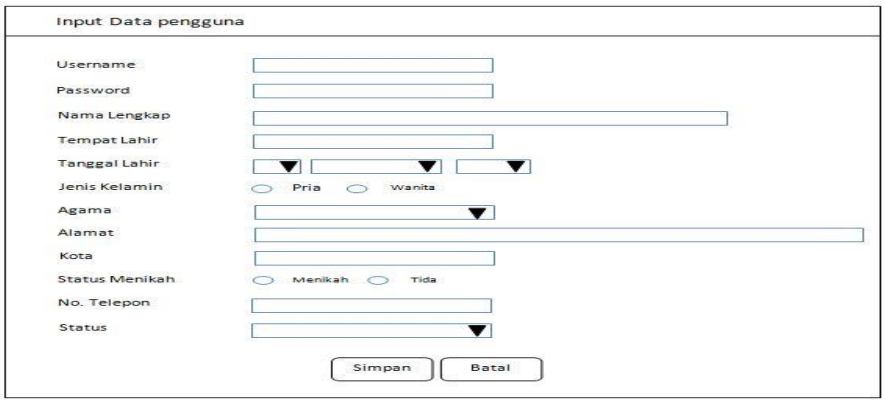
Penjelasan tombol :

a. Update

Digunakan untuk menyimpan dan mengirimkan data menu yang telah diperbaharui,yang telah diisikan pada forminput edit menu kedalam sistem.

10. Desain Input Data Pengguna.

Tampilan Forminput data pengguna ini digunakan oleh petugas untuk pembuatan dan menyimpan data pengguna baru.



Gambar 4.25 Desain Input Data Pengguna.

Penjelasan tombol :

a. Simpan

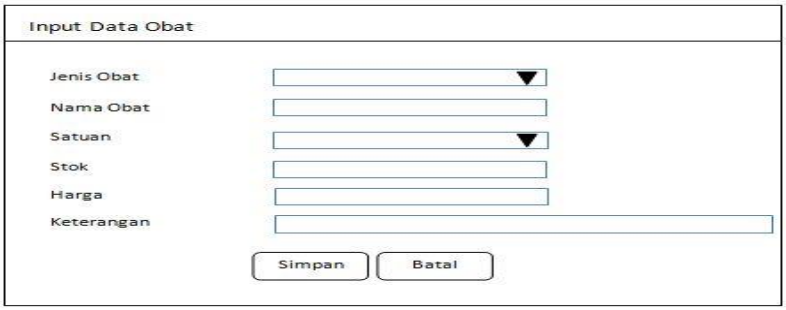
Digunakan untuk menyimpan dan mengirimkan data penggunayang telah diisikan pada forminput data pengguna kedalam sistem.

b. Batal

Digunakan untuk membatalkan data yang telah diisikan pada form input data pengguna.

11. Desain Input Data Obat.

Tampilan Forminput data obat ini digunakan oleh petugas untuk pembuatan dan menyimpan data Obat.



Gambar 4.26 Desain Input Data Obat.

Penjelasan tombol :

a. Simpan

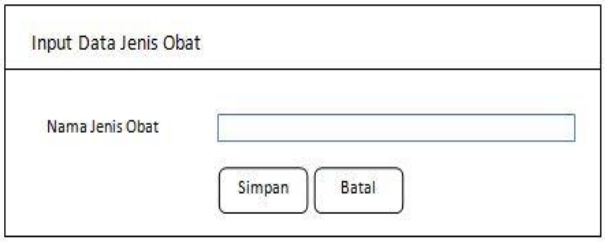
Digunakan untuk menyimpan dan mengirimkan data obatyang telah diisikan pada forminput data Obat kedalam sistem.

b. Batal

Digunakan untuk membatalkan data yang telah diisikan pada form input data Obat.

12. Desain Input Data Jenis Obat.

Tampilan Forminput data jenis obat ini digunakan oleh petugas untuk pembuatan dan menyimpan data jenis Obat.



Gambar 4.27 Desain Input Data Jenis Obat.

Penjelasan tombol :

a. Simpan

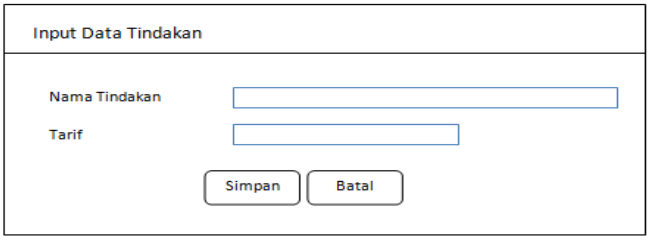
Digunakan untuk menyimpan dan mengirimkan data jenis Obatyang telah diisikan pada forminput data jenis obat kedalam sistem.

b. Batal

Digunakan untuk membatalkan data yang telah diisikan pada form input data jenis obat.

13. Desain Input Data Tindakan.

Tampilan Forminput data tindakan ini digunakan oleh petugas untuk pembuatan dan menyimpan data tindakan.Dan bisa juga digunakan oleh dokter.



Gambar 4.28 Desain Input Data Tindakan.

Penjelasan tombol :

a. Simpan

   Digunakan untuk menyimpan dan mengirimkan data tindakanyang telah diisikan pada forminput data tindakan kedalam sistem.

b. Batal

   Digunakan untuk membatalkan data yang telah diisikan pada form input data tindakan.

14. Desain Input Data Supplier.

Tampilan Forminput data supplier ini digunakan oleh petugas untuk pembuatan dan menyimpan data supplier.



Gambar 4.29 Desain Input Data Supplier.

Penjelasan tombol :

a. Simpan

   Digunakan untuk menyimpan dan mengirimkan data supplieryang telah diisikan pada forminput data supplier kedalam sistem.

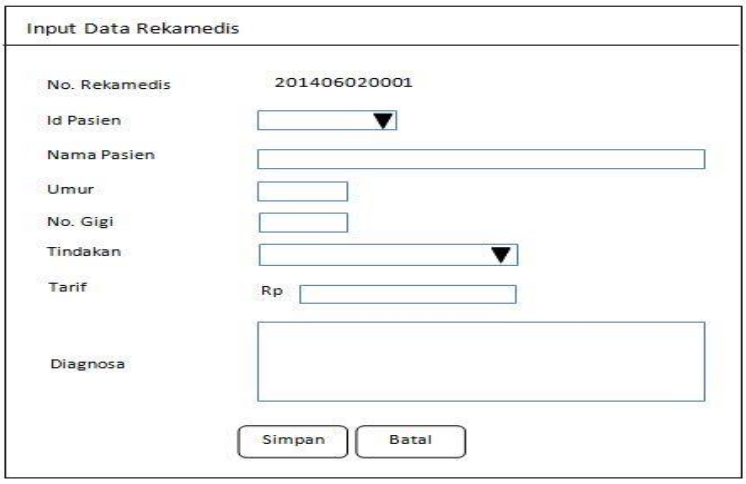
b. Batal

Digunakan untuk membatalkan data yang telah diisikan pada form input data supplier.

4.2.2.2. Perancangan Input Asisten Dokter

1. Desain Input Rekamedis.

Tampilan Forminput data rekamedis ini digunakan oleh dokter untuk mengisi data rekamedis.



Gambar 4.30 Desain Input Data Rekamedis

Penjelasan tombol :

a. Simpan

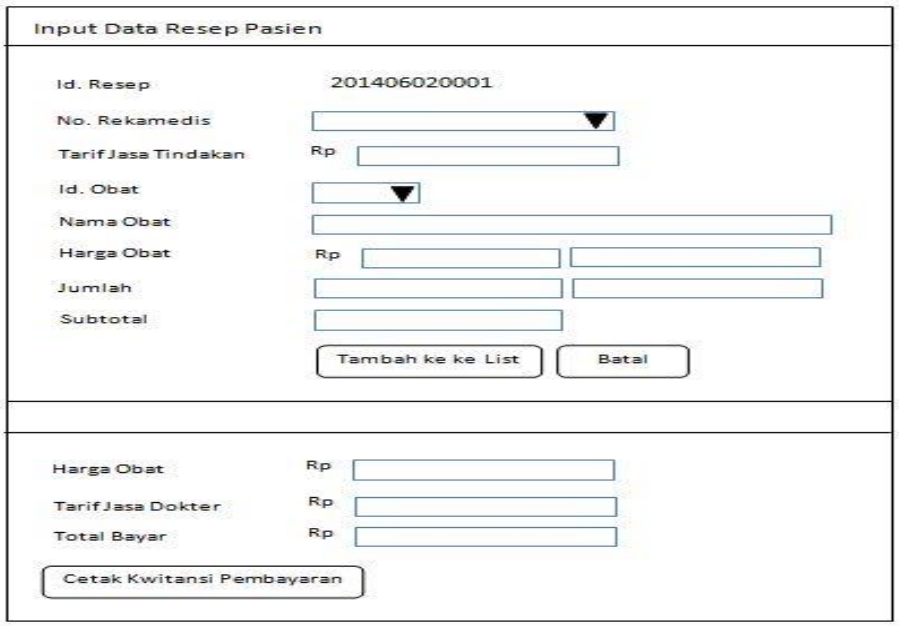
Digunakan untuk menyimpan dan mengirimkan data rekamedisyang telah diisikan pada forminput data rekamedis kedalam sistem.

b. Batal

Digunakan untuk membatalkan data yang telah diisikan pada form input data rekamedis.

2. Desain Input Resep dan Pembayaran.

Tampilan Forminput data resep dan pembayaran ini digunakan oleh kasir untuk pembuatan resep dan penghitungan total rincian pembayaran biaya pengobatan.



Gambar 4.31 Desain Input data Resep dan Pembayaran

Penjelasan tombol :

a. Tambah Ke List

Digunakan untuk menyimpan dan mengirimkan data resep obat ke list obat yang telah diisikan pada forminput data resep pasien kedalam sistem.

b. Batal

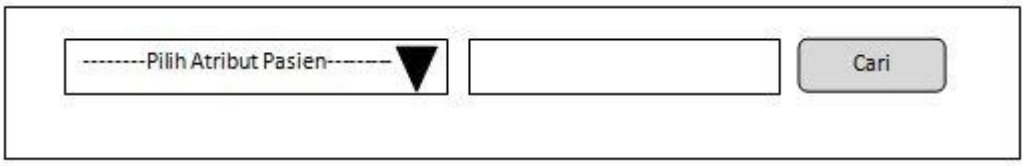
Digunakan untuk membatalkan data yang telah diisikan pada form input data resep pasien.

c. Cetak Kwitansi Pembayaran

Digunakan untuk menampilkan dan mencetak kwitansi pembayaran yang telah diisikan pada formrincian pembayaran kedalam sistem.

3. Desain Input Pencarian Data Riwayat Medis Pasien

Form input pencarian data riwayat medis pasien ini digunakan oleh asisten dokter untuk mengetahui riwayat pengobatan pasien pada bagian sub menu pasien. Petugas juga dapat menggunakan menu pencarian riwayat pasien tersebut.



Gambar 4.32 Desain Input Pencarian Data Riwayat Medis Pasien.

Penjelasan tombol :

a. Cari

Digunakan untuk mencari dan menampilkan data riwayat medis pasien berdasarakan id pasien atau nama pasien yang telah diisikan pada forminput pencarian datariwayat medis pasien kedalam sistem.

4.2.3. Perancangan Output

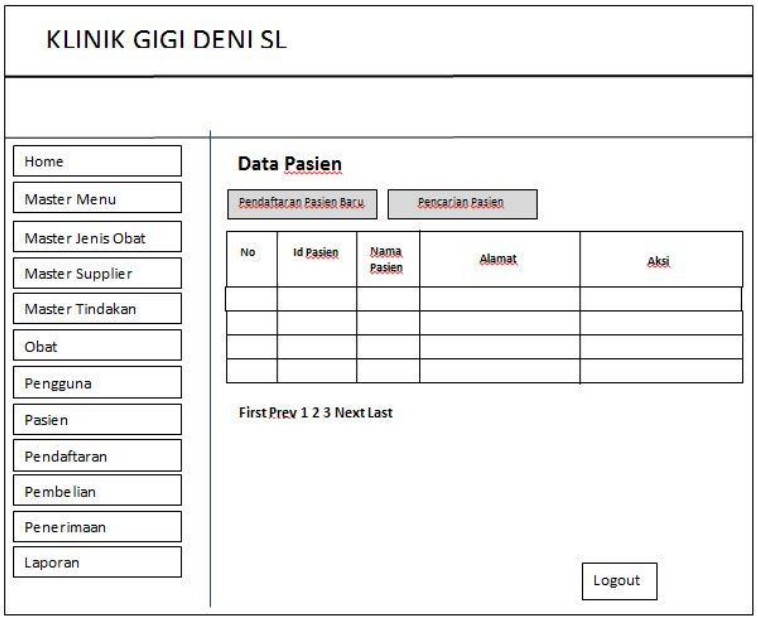
Tampilan outputpada rancangan ini berupa kumpulan data-data yang telah dimasukan melalui inputan oleh pengguna yakni petugas, dokter dan admin.

Berikut adalah desain output dari prosess yang dilakukan berdasarkan penggunanya :

1. **Output dari proses input oleh petugas.**

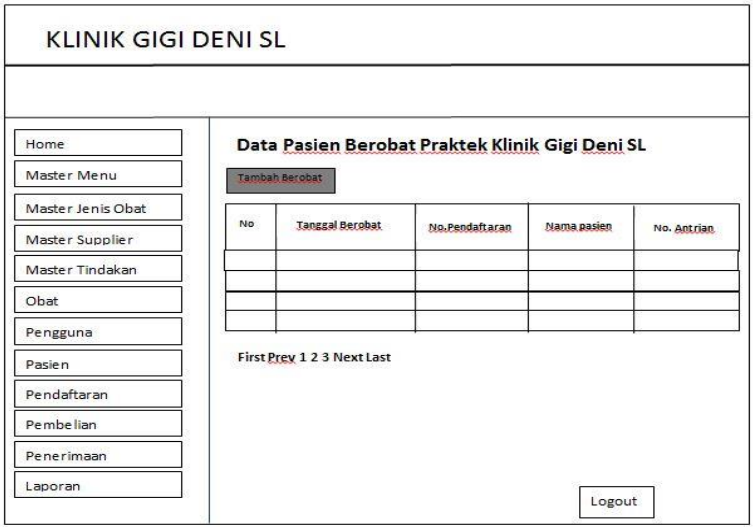
Output yang ada adalah output pendaftaran pasien, output pendaftaran berobat, output pembelian barang, output penerimaan barang, output master menu, output data pengguna, output data obat, output data jenis obat, output data tindakan dan output data supplier.Beberapa data tersebut untuk dijadikan arsip oleh petugasatau sebagai laporan untuk diberikan kepada pemilik.

1. **Desain output pendaftaran pasien**



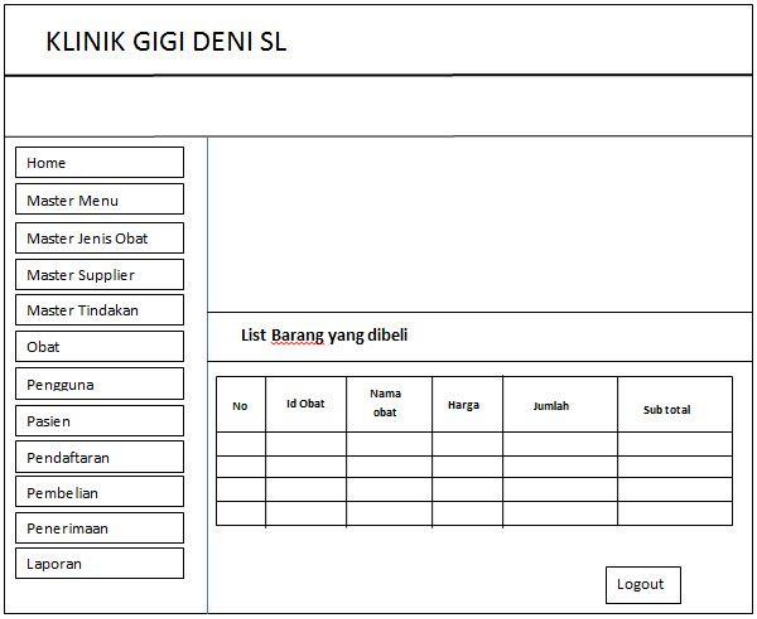
Gambar 4.33 Desain Output Pendaftaran Pasien.

2. Desain output pendaftaran berobat



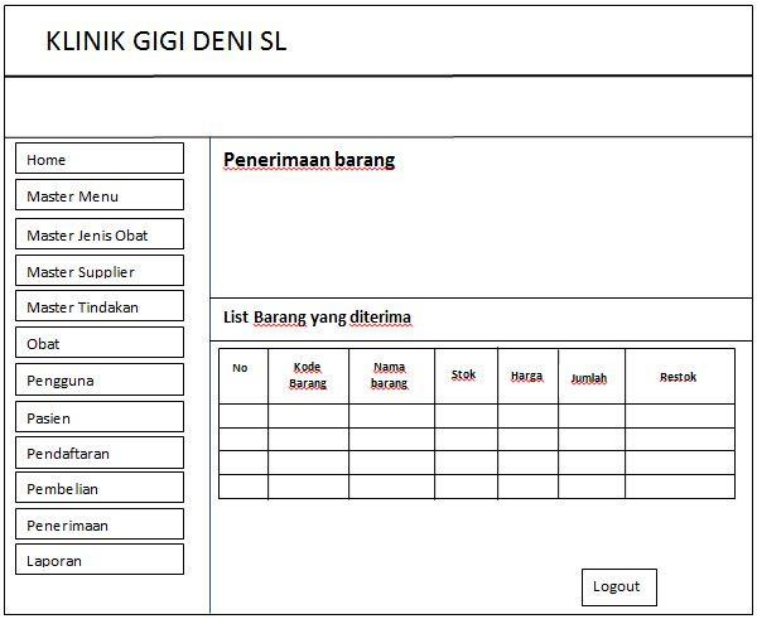
Gambar 4.34 Desain Output Pendaftaran Berobat.

3. Desain output pembelian barang



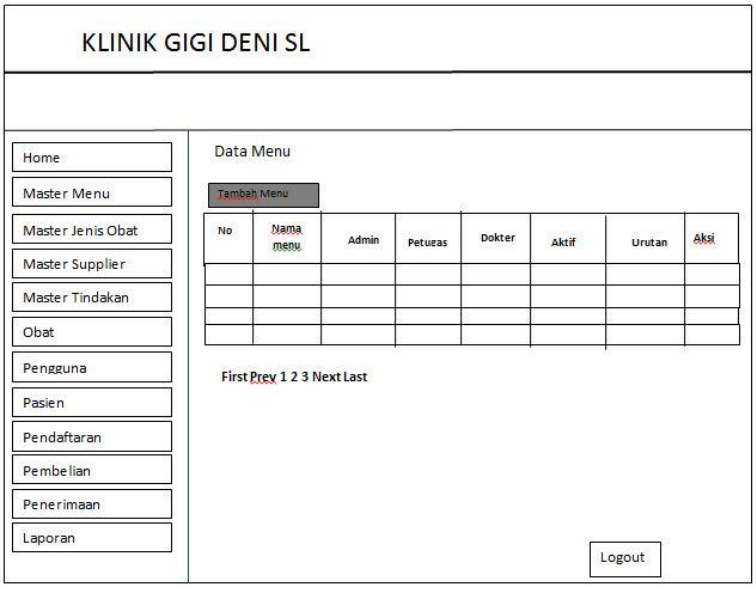
Gambar 4.35 Desain Output Pembelian Barang.

4. Desain output penerimaan barang



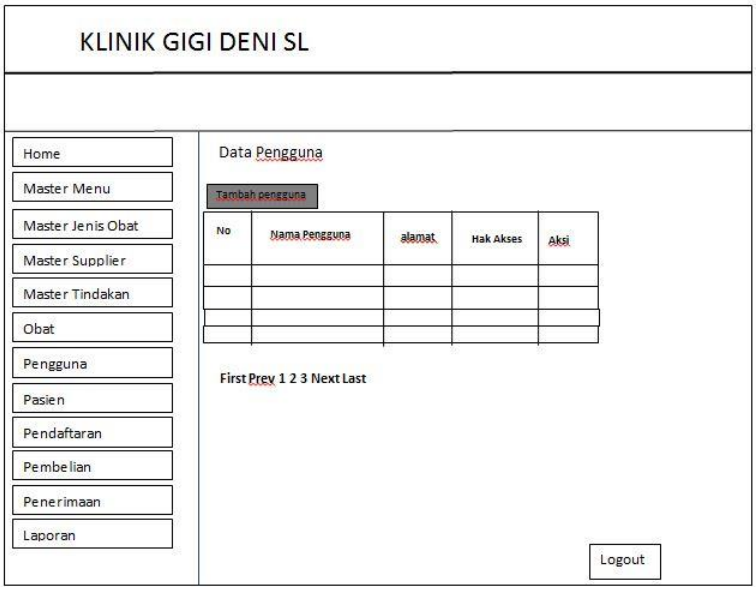
Gambar 4.36 Desain Output Penerimaan Barang.

5. Desain output data menu



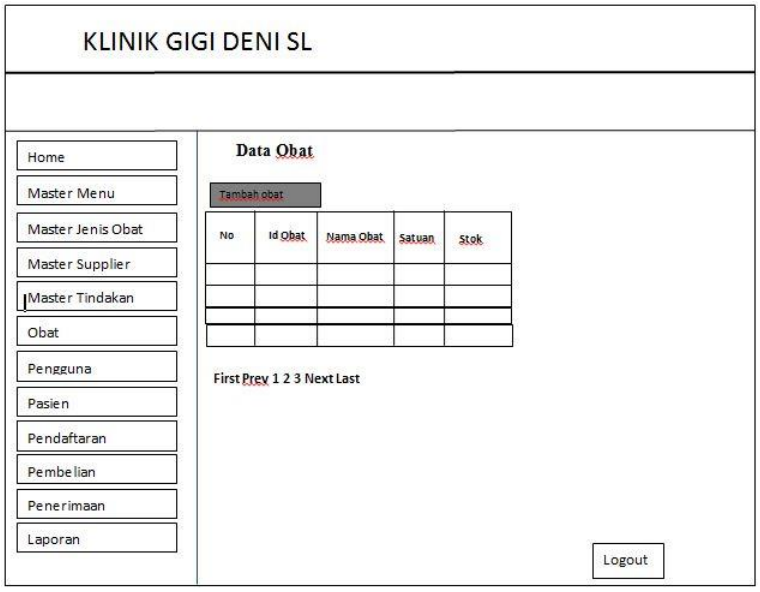
Gambar 4.37 Desain Output Data Menu.

6. Desain output data pengguna



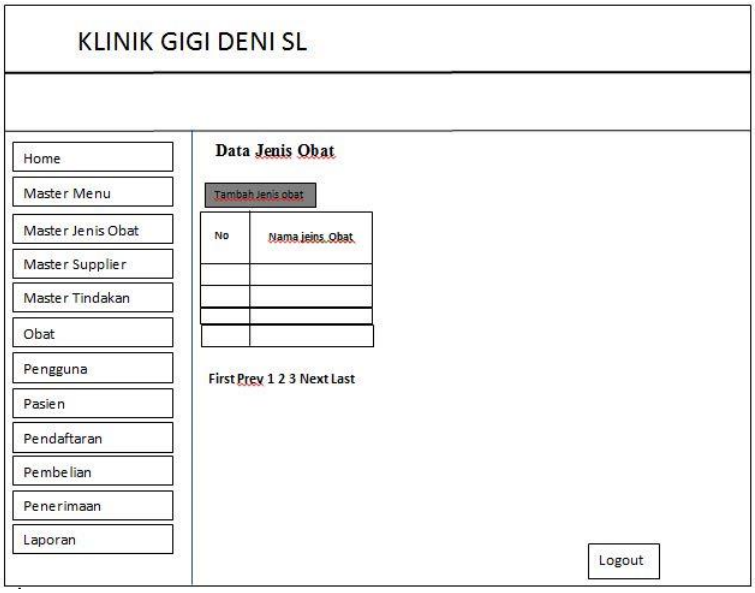
Gambar 4.38 Desain Output Data Pengguna.

7. Desain output data obat.



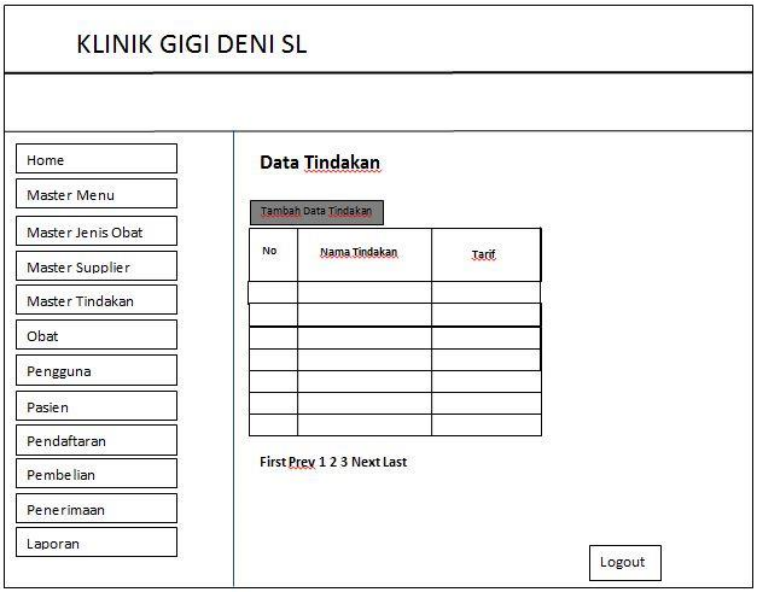
Gambar 4.39 Desain Output Data Obat.

8. Desain output data jenis obat



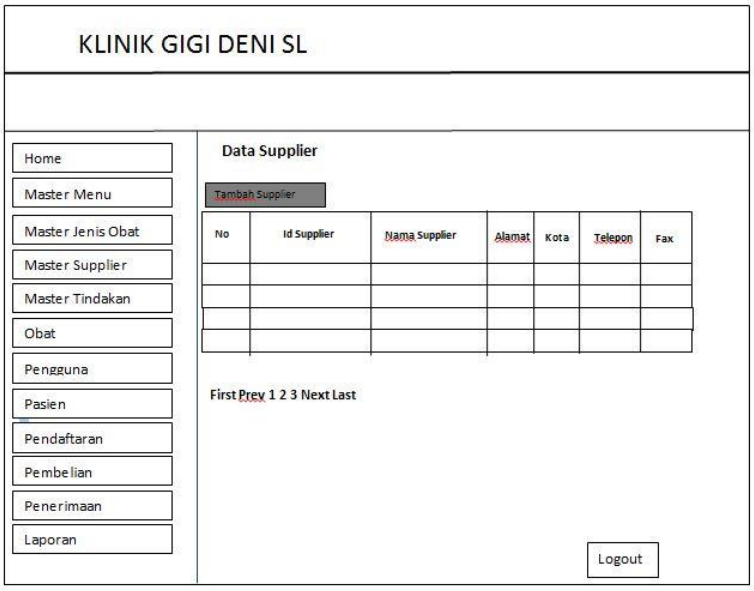
Gambar 4.40 Desain Output Data Jenis Obat.

9. Desain output data tindakan.



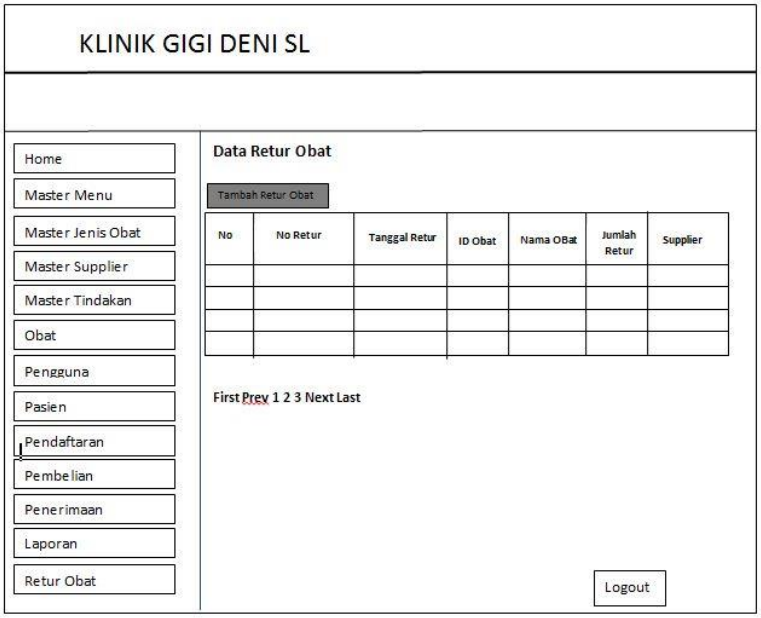
Gambar 4.41 Desain Output Data Tindakan

10. Desain output data supplier.



Gambar 4.42 Desain Output Data Supplier

11. Desain Output Data Retur Obat



Gambar 4.43 Desain Output Data Retur Obat

Adapun beberapa desain output yang dapat dicetak, berikut beberapa desain output yang dapat dicetak dari hasil prosesinput :

a. Kartu Pasien



Gambar 4.44 Desain Output Cetak Kartu pasien

b. Nota Pemesanan (PO)



Gambar 4.45 Desain Output Cetak Nota Pemesanan (PO)

c. Kwitansi



Gambar 4.46 Desain Output Cetak Kwitansi Pembayaran

d. Bukti Retur Obat



Gambar 4.47 Desain Output Cetak Bukti Retur Obat

e. Laporan

LAPORAN DATA PASIEN
KLINIK GIGI DENI S.L

No	Id Pasien	Nama Pasien	Pendaftaran	Alamat

Petugas

(xxxxxxxxxxx)

Gambar 4.48 Desain Output Cetak Laporan Pasien

LAPORAN DATA PERSEDIAN OBAT
KLINIK GIGI DENI S.L

No	Id Obat	Nama Obat	Harga	Stok

Petugas

(xxxxxxxxxxx)

Gambar 4.49 Desain Output Cetak Laporan Persedian Obat

LAPORAN DATA BEROBAT
KLINIK GIGI DENI S.L

No	No Pendaftaran	Tanggal	Id Pasien	Nama Pasien	No Antrian

Petugas

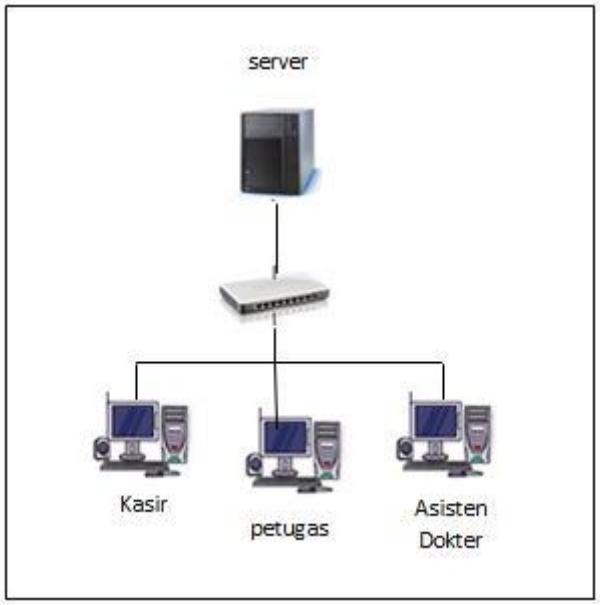
(xxxxxxxxxxx)

Gambar 4.50 Desain Output Cetak Laporan Berobat

4.3. Perancangan Arsitektur Jaringan

Sistem operasi Microsoft Windows 7memberikan beberapa kemudahan dalam membangun suatu jaringan berskala kecil karena dalam sistem operasi Microsoft Windows 7terdapat berbagai fitur baru yang menunjang pembangunan jaringan komputer berskala kecil. Pembangunan jaringan komputer berskala kecil membutuhkan beberapa perangkat keras jaringan yang harus dimiliki oleh komputer. Perangkat keras jaringan yang dibutuhkan adalah sebagai berikut :

- Kabel UTPdengan konfigurasi straight. Kabel UTP straightdigunakan untuk menghubungkan banyak komputer dengan menggunakan media penghubung seperti switch atau hub.
- Switch atau hub sebagai media penghubung antara komputer satu dengan komputer lainnya.
- Jaringan komputer yang diterapkan menggunakan topologi star.



Gambar 4.51 Desain Arsitektur Jaringan pada Klinik Drg. Deni S.L

4.4 Implementasi

Implementasi Sistem Informasi klinik Drg. Deni S.L dilakukan dengan menggunakan bahasa dan pemrograman *PHP*, dengan Basis data yang digunakan ialah *MySQL*, dengan versi 5.0. Aplikasi *PHP* tersebut dapat dijalankan pada berbagai *platform* sistem operasi dan perangkat keras, tetapi implementasi dan pengujian sepenuhnya hanya dilakukan di perangkat keras *PC (Personal Computer)* dengan sistem operasi *Microsoft Windows 7.*

4.4.1. Batasan Implementasi

Dalam mengimplementasikan perangkat lunak, pemantauan dan pengendalian aplikasi ini, ada beberapa hal yang menjadi batasan implementasi yaitu :

1. Dalam sistem informasi klinik berbasis *PHP* ini hanya sebatas pengelolaan pendaftaran, rekamedis, pembayaran, persedian obat dan laporan.
2. Basis data yang digunakan dalam pengimplementasian ini adalah *MySQL*. Pengaksesan basis data di *server* dilakukan melalui *Localhost.*
3. Sistem ini hanya berupa perancangan dan belum biasa langsung diterapkan menggantikan sistem yang lama.

4.4.2. Implementasi Perangkat Lunak

Untuk pengembangan perangkat lunak ini digunakan *Apache* (2.2.8), *PHP* (5.2.3) dan *MySQL* (5.0.18), Apache dipilih sebagai perangkat lunak untuk *server*, *PHP* dipilih sebagai perangkat lunak pengembangan karena menyediakan fasilitas

yang memadai untuk membuat perangkat lunak. Sementara *MySQL* digunakan sebagai perangkat lunak pengembang dalam pembuatan basis data.

4.4.3. Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain :

1. *Server*
   - Menggunakan minimal *Prossesor Intel Pentium* atau yang sekelasnya.
   - Menggunakan *RAM* minimal 512 MB
   - Tersedia *Hard Drive*, untuk media penyimpanan, minimal 10MB untuk *server* diluar basis data
   - *LAN Card*
   - *Mouse, Keyboard* dan *Monitor* sebagai peralatan antar muka
2. *User*
   - Menggunakan minimal *Prossesor Intel Pentium* atau sekelasnya.
   - Menggunakan *RAM* minimal 512 MB
   - *LAN Card*
   - *Mouse, Keyboard* dan *Monitor* sebagai peralatan antarmuka.

4.4.4. Implementasi Basis Data (Sintaks SQL)

Pembuatan basis data dilakukan dengan menggunakan *SQL*, dimana DBMS yang digunakan adalah *MySQL,* Implementasi basis datanya dalam *SQL* adalah sebagai berikut :

```
Database: `dbklinik`

-- ------------------------------------------------------

----------Table structure for table `bulan`

CREATE TABLE IF NOT EXISTS `bulan`(

`kode_bulan` varchar(2) character set latin1 NOT NULL,

`nama_bulan` varchar(10) character set latin1 NOT NULL,

PRIMARYKEY(`kode_bulan`)ENGINE=MyISAMDEFAULT

CHARSET=latin1 COLLATE=latin1_general_ci;

-- ------------------------------------------------------

----------Table structure for table `dokter`

CREATE TABLE IF NOT EXISTS `dokter` (

  `id_dokter` varchar(5) NOT NULL,

  `nama_dokter` varchar(30) NOT NULL,

  `tempat_lahir` varchar(15) NOT NULL,

  `tgl_lahir` date NOT NULL,

  `jns_kel` varchar(10) NOT NULL,

  `agama` varchar(10) NOT NULL,

  `alamat` text NOT NULL,

  `kota` varchar(15) NOT NULL,

  `stat_kawin` varchar(10) NOT NULL,

  `no_hp` varchar(15) NOT NULL,

  `spesialis` varchar(15) NOT NULL,

  PRIMARY KEY  (`id_dokter`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- ----------------------------------------------------

------------Table structure for table `hari`

CREATE TABLE IF NOT EXISTS `hari` (

  `id_hari` int(2) NOT NULL auto_increment,

  `nama_hari` varchar(15) NOT NULL,

  PRIMARY KEY  (`id_hari`)

) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;

-- ----------------------------------------------------

----------Table structure for table `jadwal`

CREATE TABLE IF NOT EXISTS `jadwal` (

  `id_jadwal` int(3) NOT NULL auto_increment,

  `id_hari` int(2) NOT NULL,

  `nama_hari` varchar(15) NOT NULL,

  `jam` varchar(13) NOT NULL,

  PRIMARY KEY  (`id_jadwal`)

) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;

-- ----------------------------------------------------

-----------Table structure for table `jns_obat`

CREATE TABLE IF NOT EXISTS `jns_obat` (

  `id_jns_obt` int(3) NOT NULL auto_increment,

  `nama_jns_obt` varchar(20) NOT NULL,

  PRIMARY KEY  (`id_jns_obt`)
```

```
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;

-- ------------------------------------------------------

------------Table structure for table `menu`

CREATE TABLE IF NOT EXISTS `menu` (
  `id_menu` int(5) NOT NULL auto_increment,
  `nama_menu` varchar(20) NOT NULL,
  `link` varchar(50) NOT NULL,
  `aktif` enum('Ya','Tidak') NOT NULL default 'Ya',
  `urutan` int(5) NOT NULL,
  `akses_admin` enum('Ya','Tidak') NOT NULL default 'Ya',
  `akses_petugas` enum('Ya','Tidak') NOT NULL default 'Tidak',
  `akses_dokter` enum('Ya','Tidak') NOT NULL default 'Tidak',
  PRIMARY KEY  (`id_menu`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;

-- ------------------------------------------------------

-----------------Table structure for table `obat`

CREATE TABLE IF NOT EXISTS `obat` (
  `id_obt` varchar(5) NOT NULL,
  `id_jns_obt` int(3) NOT NULL,
  `nama_obt` varchar(25) NOT NULL,
  `satuan` varchar(15) NOT NULL,
  `stok` varchar(9) NOT NULL,
  `harga` varchar(9) NOT NULL,
```

```
  `ket` text NOT NULL,
  PRIMARY KEY  (`id_obt`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
---------- Table structure for table `pasien`
CREATE TABLE IF NOT EXISTS `pasien` (
  `id_pasien` varchar(10) NOT NULL,
  `tgl_daftar` date NOT NULL,
  `no_ktp` varchar(20) NOT NULL,
  `nama_pasien` varchar(30) NOT NULL,
  `tempat_lahir` varchar(15) NOT NULL,
  `tgl_lahir` date NOT NULL,
  `jns_kel` varchar(10) NOT NULL,
  `agama` varchar(10) NOT NULL,
  `alamat` text NOT NULL,
  `kota` varchar(15) NOT NULL,
  `stat_kawin` varchar(10) NOT NULL,
  `pekerjaan` varchar(15) NOT NULL,
  `no_hp` varchar(15) NOT NULL,
  PRIMARY KEY  (`id_pasien`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
```

```
----------Table structure for table `pendaftaran`
CREATE TABLE IF NOT EXISTS `pendaftaran` (
  `id_pendaftaran` varchar(11) NOT NULL,
  `tgl_pendaftaran` date NOT NULL,
  `id_pasien` varchar(10) NOT NULL,
  `nama_pasien` varchar(30) NOT NULL,
  `no_antrian` varchar(3) NOT NULL,
  `stat_berobat` varchar(2) NOT NULL,
  PRIMARY KEY  (`id_pendaftaran`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
----------Table structure for table `petugas`
CREATE TABLE IF NOT EXISTS `petugas` (
  `id_petugas` int(2) NOT NULL auto_increment,
  `username` varchar(15) NOT NULL,
  `password` varchar(15) NOT NULL,
  `nama_petugas` varchar(30) NOT NULL,
  `tmpt_lahir` varchar(15) NOT NULL,
  `tgl_lahir` date NOT NULL,
  `jns_kelamin` varchar(10) NOT NULL,
  `agama` varchar(10) NOT NULL,
  `alamat` text NOT NULL,
  `kota` varchar(15) NOT NULL,
```

```
  `stat_kawin` varchar(10) NOT NULL,
  `no_hp` varchar(15) NOT NULL,
  `status` varchar(10) NOT NULL,
  PRIMARY KEY  (`id_petugas`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
-- --------------------------------------------------------
-- Table structure for table `po`
CREATE TABLE IF NOT EXISTS `po` (
  `no_po` varchar(12) NOT NULL,
  `tgl_po` date NOT NULL,
  `tot` varchar(9) NOT NULL,
  `status` varchar(1) NOT NULL,
  PRIMARY KEY  (`no_po`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
-- Table structure for table `pod`
CREATE TABLE IF NOT EXISTS `pod` (
  `no_po` varchar(12) NOT NULL,
  `id_obt` varchar(5) NOT NULL,
  `id_supplier` varchar(6) NOT NULL,
  `jml` varchar(9) NOT NULL,
  `sub` varchar(9) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-- ----------------------------------------------------

-- Table structure for table `rekamedis`

CREATE TABLE IF NOT EXISTS `rekamedis` (

  `no_rekamedis` varchar(12) NOT NULL,

  `id_pasien` varchar(10) NOT NULL,

  `tgl_rekamedis` date NOT NULL,

  `umur` varchar(2) NOT NULL,

  `no_gigi` varchar(6) NOT NULL,

  `id_tindakan` int(2) NOT NULL,

  `diagnosa` text NOT NULL,

  `tarif` varchar(9) NOT NULL,

  `stat` varchar(1) NOT NULL,

  PRIMARY KEY  (`no_rekamedis`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- ----------------------------------------------------

-- Table structure for table `resep`

CREATE TABLE IF NOT EXISTS `resep` (

  `id_resep` varchar(12) NOT NULL,

  `no_rekamedis` varchar(12) NOT NULL,

  `tgl_resep` date NOT NULL,

  `total` varchar(9) NOT NULL,

  PRIMARY KEY  (`id_resep`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-- ----------------------------------------------------

-- Table structure for table `resep_d`

CREATE TABLE IF NOT EXISTS `resep_d` (

  `id_resep` varchar(12) NOT NULL,

  `id_obt` varchar(5) NOT NULL,

  `qty` varchar(9) NOT NULL,

  `subtot` varchar(9) NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- ----------------------------------------------------

-- Table structure for table `retur`

CREATE TABLE IF NOT EXISTS `retur` (

  `no_retur` varchar(12) NOT NULL,

  `tgl_retur` date NOT NULL,

  `id_supplier` varchar(6) NOT NULL,

  `id_obt` varchar(5) NOT NULL,

  `stok` varchar(9) NOT NULL,

  `jml_retur` varchar(9) NOT NULL,

  `ustok` varchar(9) NOT NULL,

  PRIMARY KEY  (`no_retur`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- ----------------------------------------------------

-- Table structure for table `supplier`

CREATE TABLE IF NOT EXISTS `supplier` (
```

```
  `id_supplier` varchar(6) NOT NULL,
  `npwp` varchar(20) NOT NULL,
  `an_npwp` varchar(30) NOT NULL,
  `nama_sup` varchar(30) NOT NULL,
  `alamat` text NOT NULL,
  `kota` varchar(15) NOT NULL,
  `telpon` varchar(15) NOT NULL,
  `fax` varchar(15) NOT NULL,
  `email` varchar(30) NOT NULL,
  PRIMARY KEY  (`id_supplier`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
-- Table structure for table `terima_obt`
CREATE TABLE IF NOT EXISTS `terima_obt` (
  `no_terima` varchar(12) NOT NULL,
  `tgl_terima` date NOT NULL,
  `no_po` varchar(12) NOT NULL,
  `id_supp` varchar(6) NOT NULL,
  `keterangan` text NOT NULL,
  PRIMARY KEY  (`no_terima`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;\
-- --------------------------------------------------------
```

```
-- Table structure for table `terima_obtd`
CREATE TABLE IF NOT EXISTS `terima_obtd` (
  `no_terima` varchar(12) NOT NULL,
  `id_obt` varchar(5) NOT NULL,
  `jumlah` varchar(9) NOT NULL,
  `stok` varchar(9) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
-- Table structure for table `tindakan`
CREATE TABLE IF NOT EXISTS `tindakan` (
  `id_tindakan` int(2) NOT NULL auto_increment,
  `nama_tindakan` varchar(40) NOT NULL,
  `tarif` varchar(9) NOT NULL,
  PRIMARY KEY  (`id_tindakan`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

4.4.5. Implementasi Antar Muka

Dalam *PHP* (5.2.3.), implementasi antarmuka dilakukan dengan sebuah *webpage* yang berekstensi *.PHP*.Setiap halaman dalam perangkat lunak dibuat *file* program yang ditulis dengan berekstensi *php*. Berikut ini adalah implementasi dari setiap halaman yang dibuat dan dibedakan berdasarkan Hak akses.

a. Implementasi Halaman Utama Petugas

Implementasi halaman yang terdapat pada halaman utama user dibuat dalam bentuk *file* program dengan berekstensi *PHP* dapat dilihat pada tabel 4.2 berikut :

Tabel 4.22 Implementasi Sub Menu Petugas

Sub Menu	Deskripsi	Nama File
Login	File program untuk menangani user login	index.php
Home	File program untuk melihat data beranda.	media.php
Master Menu	File program untuk tambah, edit, hapus menu.	menu.php
Master Jenis obat	File program untuk tambah data jenis obat	jenis.php
Master Supplier	File program untuk menambah data supplier.	supplier.php
Master Tindakan	File program untuk melihat dan tambah data tindakan.	tindakan.php
Obat	File program untuk tambah data obat	obat.php
Pengguna	File program untuk tambah, edit, hapus pengguna.	petugas.php
Pasien	File program untuk melihat data pasien dan tambah data pasien.	pasien.php
Pendaftaran	File program untuk menambah dan melihat daftar antrian berobat.	pendaftaran.php

Pembelian	File program untuk melihat data pembelian barang.	pembelian.php
Penerimaan	File program untuk melihat data penerimaan. barang.	penerimaan.php
laporan	File program untuk melihat data laporan.	laporan.php

b. Implementasi Halaman Utama AsistenDokter

Implementasi halaman yang terdapat pada halaman utama Asistendokter dibuat dalam bentuk file program dengan berekstensi PHP dapat dilihat pada tabel 4.4 berikut :

Tabel 4.23. Implementasi Sub Menu Asisten Dokter

Submenu	Deskripsi	Nama File
Home	File program untuk melihat data beranda.	media.php
Master Tindakan	File program untuk melihat dan tambah data tindakan.	tindakan.php
Pasien	File program untuk melihat data pasien dan tambah data pasien.	pasien.php
Rekamedis	File program untuk menambah data rekamedis pasien.	rekamedis.php
Resep &Pembayaran	File program untuk melihat resep obat dan rincian pembayaran.	resep.php

4.4.6. Implementasi Instalasi Program

Kegiatan pembuatan program terdiri dari pembentukan program-program yang akan saling mendukung dalam pembuatan aplikasi koperasi. Untuk keperluan menampung data di server, kita memerlukan sebuah database. MySql merupakan sebuah program aplikasi untuk membuat database.. MySql banyak digunakan karena kemampuannya dalam menampung data yang cukup besar dan proses loading yang cepat dalam pemanggilan data.

Implementasinya yaitu :

1. Instalkan program Apache 2.0, MySQL server 5.0, MySQL front, PHP di Drive C:
2. Copy folder sisp pada C:/Apache2/htdocs/klinik untuk programnya
3. Copy folder dbklinik pada C:\xampp\mysql\data untuk databasenya.
4. Ketikan pada URL browser yang dimiliki yaitu localhost\klinik.
5. Jika berhasil maka akan masuk pada tampilan awal login user.

Dalam pembahasan kali ini akan membuat database dengan menggunakan Phpmyadmin yang sudah built-in (sudah terintegrasi dengan aplikasi server).

4.4.7. Penggunaan Program

Dalam penggunaan Program ini langkah awalnya adalah memanggil tampilan awal sistem informasi Klinik dengan mengetikan alamat localhost/klinik. Maka akan tampil halaman Login Untuk Petugas dan asisten Dokter.

4.4.7.1.Tampilan Login

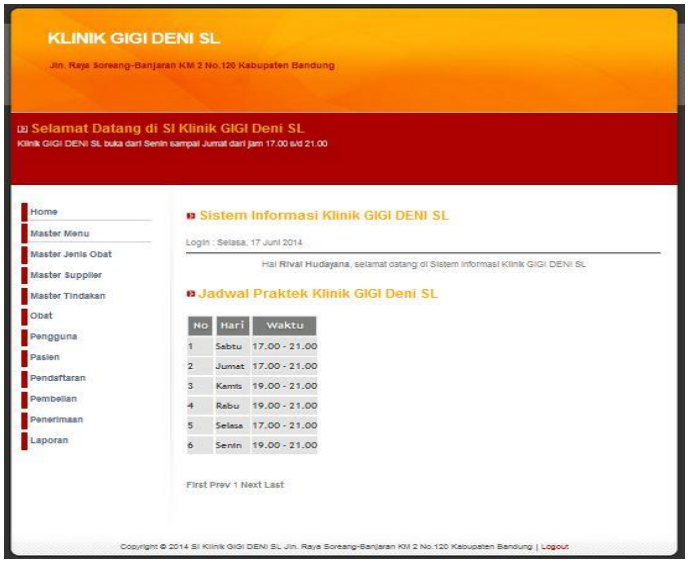
Merupakan Tampilan awal pada saat pengguna masuk ke dalam Aplikasi



Gambar 4.52 Tampilan Halaman Login

Pada tampilan diatas terlihat form login yang harus diisikan oleh para pengguna untuk dapat masuk ke dalam sistem aplikasi Klinik.

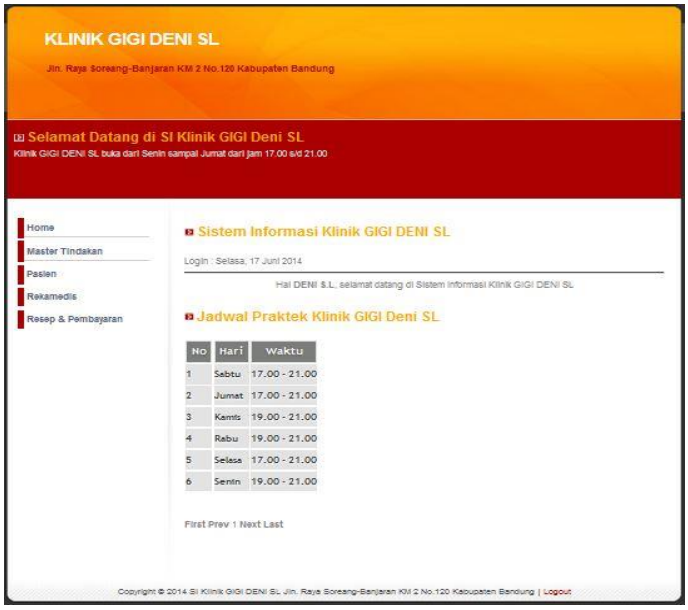
4.4.7.2.Tampilan Halaman Home Petugas



Gambar 4.53 Tampilan Menu Home Petugas

Pada tampilan di atas terlihat beberapa menu di bagian kiri layar, seperti Home, Master Menu, Master Jenis Obat, Master Supplier, Master Tindakan, Obat, Pengguna, Pasien, Pendaftaran, Pembelian, Penerimaandan Laporan.

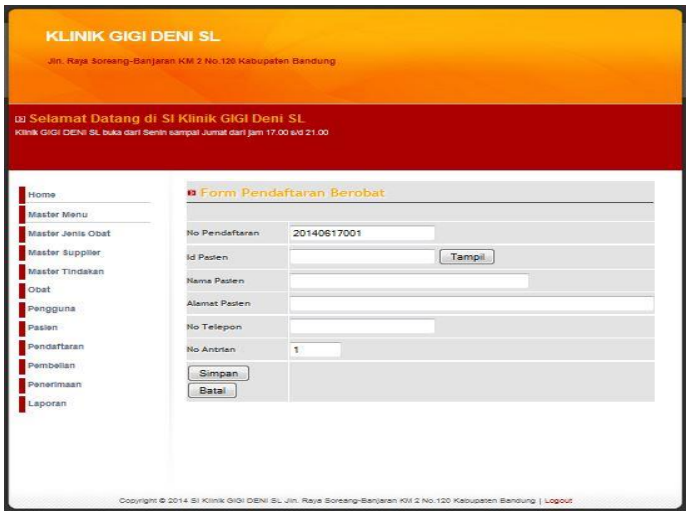
4.4.7.3.Tampilan Halaman Home Asisten Dokter



Gambar 4.54 Tampilan Menu Home Dokter

Pada tampilan di atas terlihat beberapa menu di bagian kiri layar, seperti Home, Master Tindakan, Pasien,Rekamedis,Resep &Pembayaran dan Laporan.

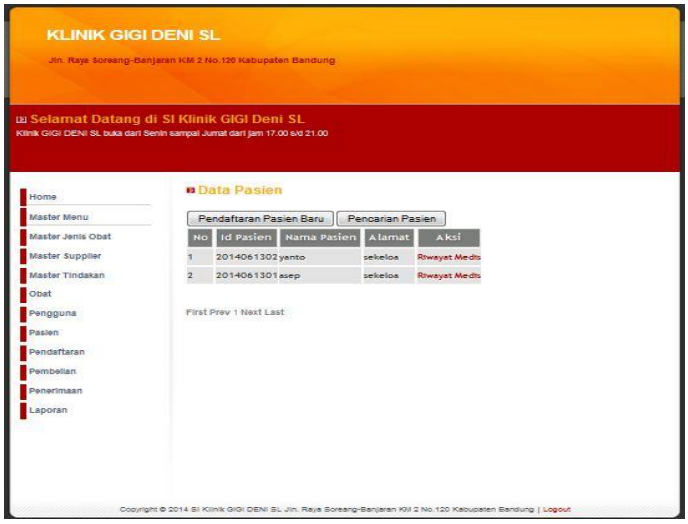
4.4.7.4.Tampilan Halaman Pendaftaran Pasien



Gambar 4.55 Tampilan Form Pendaftaran Pasien

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Id Pasien, No. KTP, Nama Pasien, Tempat Lahir, tanggal Lahir, Jenis kelamin, Agama, Alamat, Kota, Status Kawin, No.Telepon, Pekerjaan, Simpan, Batal dan Logout.

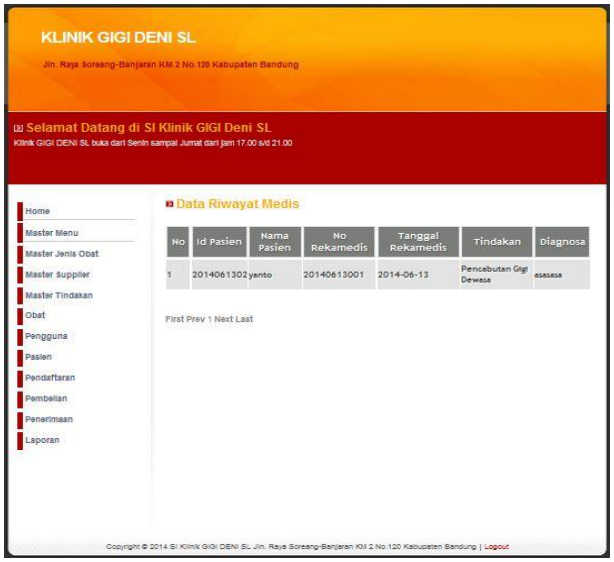
4.4.7.5.Tampilan List Daftar Data Pasien



Gambar 4.56 Tampilan List Daftar Data Pasien

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Pendaftaran Pasien Baru, Pencarian Pasien, No, Id Pasien, Nama Pasien, Alamat, Aksi (pada kolom aksi terdapat link riwayat pasien, untuk melihat riwayat pengobatan pasien yang biasanya dilakukan oleh dokter), Frist, Prev, Next, Last dan Logout.

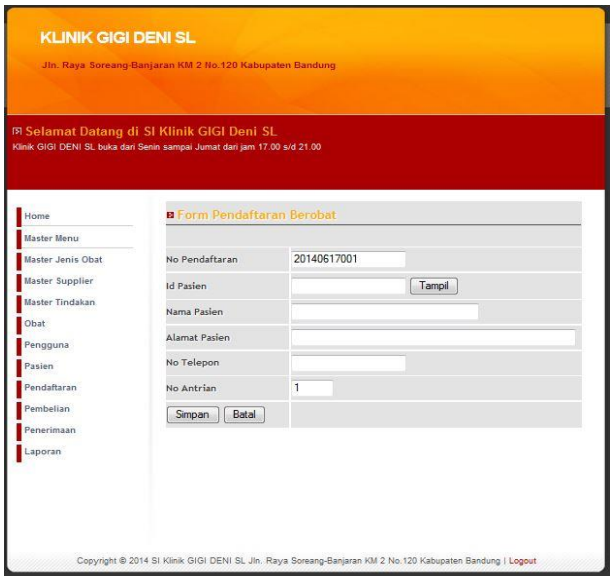
4.4.7.6.Tampilan List Daftar Data Riwayat Medis



Gambar 4.57 Tampilan List Daftar Data Riwayat Medis.

Pada tampilan di atas bagian sebelah kanan pada layar terdapat data riwayat medis, yang berisi No, Id Pasien, Nama Pasien, No. Rekamedis, Tanggal Rekamedis, Tindakan, Diagnosa, Frist, Prev, Next, Last dan Logout.

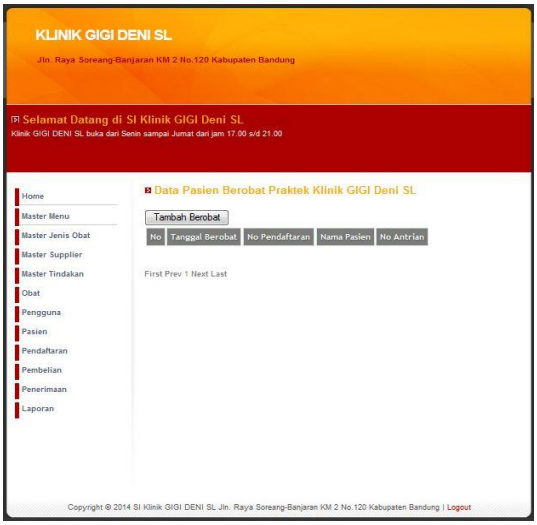
4.4.7.7.Tampilan Pendaftaran Berobat



Gambar 4.58 Tampilan Form Pendaftaran Berobat.

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu No. Pendaftaran, Id Pasien, Nama Pasien, Alamat Pasien, No. Telepon, No. Antrian, Simpan, Batal dan Logout.

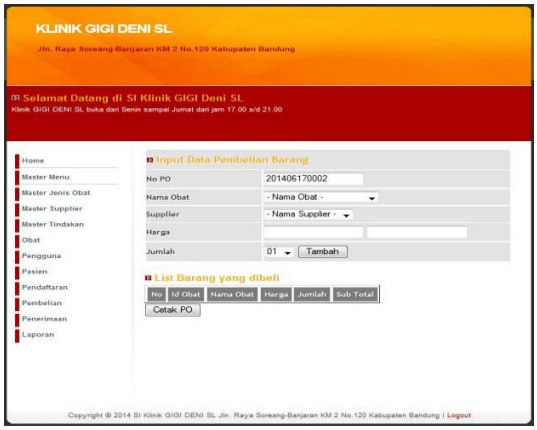
4.4.7.8.Tampilan List Daftar Pasien Berobat



Gambar 4.59 Tampilan List Daftar Pasien Berobat.

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Tambah Berobat, No, Tanggal Berobat, No. Pendaftaran, Nama Pasien, No. Antrian, Frist, Prev, Next, Last dan Logout.

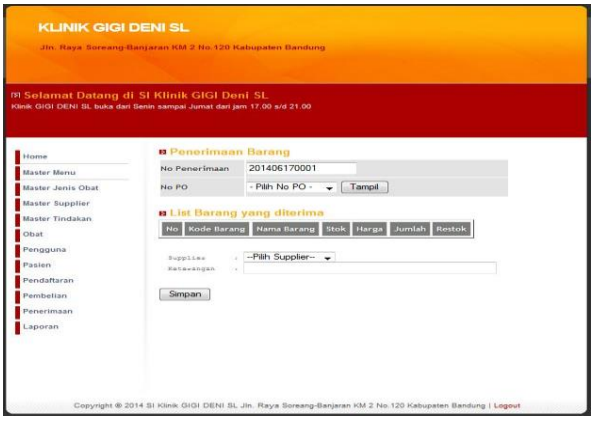
4.4.7.9.Tampilan Pembelian Barang dan List Barang yang Dibeli



Gambar 4.60 Tampilan Pembelian Barang dan Daftar List Barang yang Dibeli

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu No.PO, Nama Obat, Supplier, Harga, Jumlah, Tambah, No, Id.Obat, Sub Total, Cetak PO dan Logout.

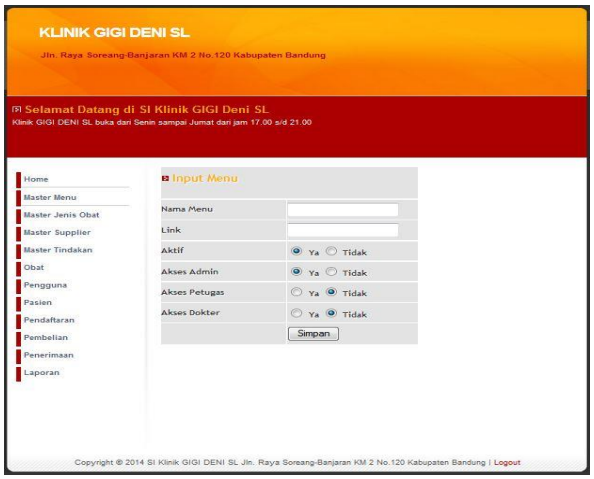
4.4.7.10.Tampilan Penerimaan Barang dan List Barang yang Diterima



Gambar 4.61 Tampilan Penerimaan Barang dan Daftar List Barang yang Diterima

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu No.Penerima, No.PO, Tampil, No, Kode Barang, Nama Barang, Stok, Harga, Jumlah, Restok, Supplier, Keterangan, Simpan dan Logout.

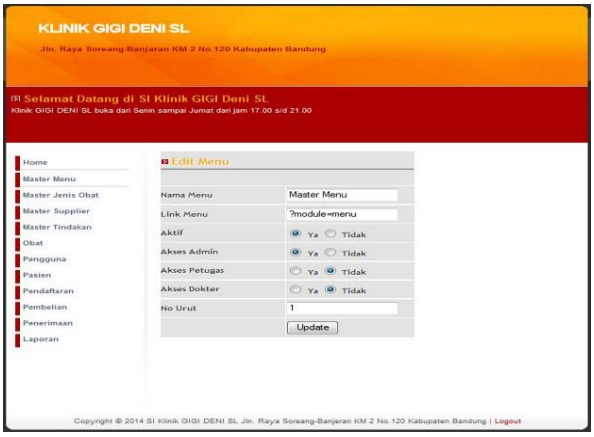
4.4.7.11. Tampilan Input Data Menu



Gambar 4.62 Tampilan Input Data Menu

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Nama Menu, Link, Aktif, Akses Admin, Akses Petugas, Akses Dokter, Simpan dan Logout.

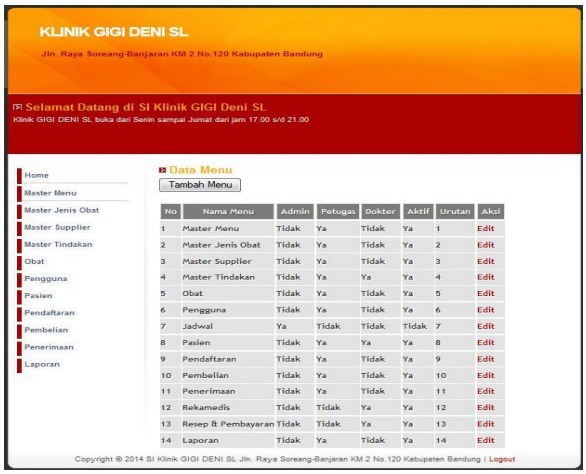
4.4.7.12. Tampilan Input Edit Data Menu



Gambar 4.63 Tampilan Input Edit Data Menu

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu, Nama Menu, Link, Aktif, Akses Admin, Akses Petugas, Akses Dokter, No.Urut, Update dan Logout.

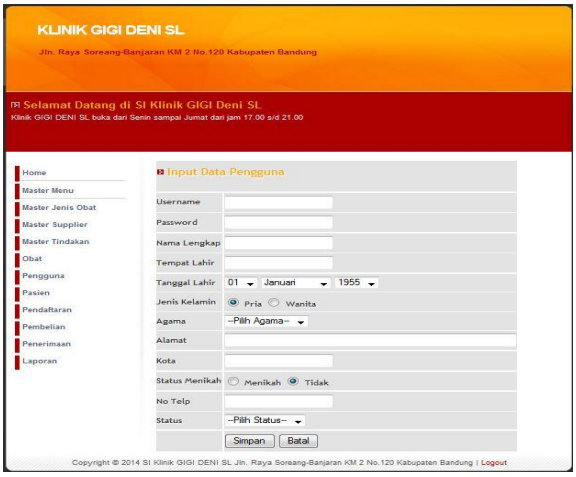
4.4.7.13. Tampilan List Daftar Menu



Gambar 4.64 Tampilan List Daftar Menu

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Tambah Menu, No, Nama Menu, Admin, Petugas, Dokter, Aktif, Urutan, Aksi dan Logout.

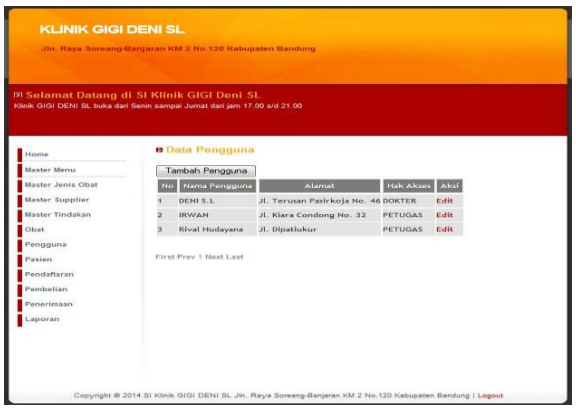
4.4.7.14. Tampilan Input Pengguna



Gambar 4.65 Tampilan Input Data Pengguna

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Username, Password, Nama Lengkap, Tempat Lahir, Tanggal Lahir, Jenis Kelamin, Agama, Alamat, Kota, Status Menikah, No.Telepon, Status, Simpan, Batal dan Logout.

4.4.7.15. Tampilan List Daftar Pengguna



Gambar 4.66 Tampilan List Daftar Pengguna

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Tambah Pengguna, No, Nama Pengguna, Alamat, Hak Akses, Aksi, Frist, Prev, Next, Last dan Logout.

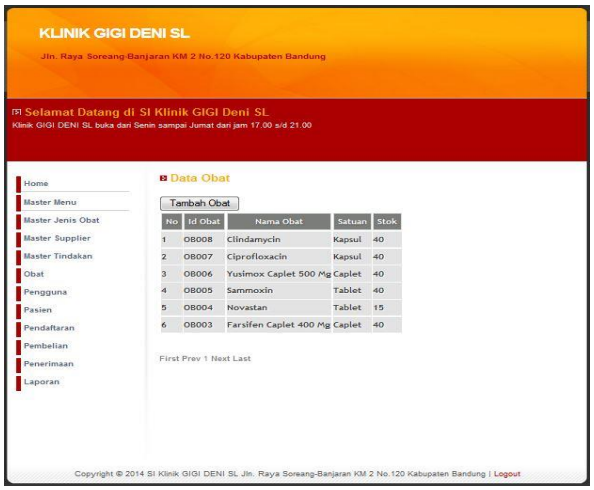
4.4.7.16. Tampilan Input Data Obat



Gambar 4.67 Tampilan Input Data Obat

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Jenis Obat, Nama Obat, Satuan, Stok, Harga, Keterangan, Simpa, Batal dan Logout.

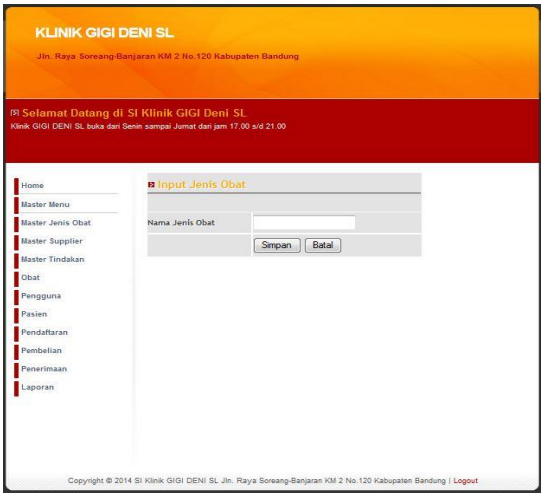
4.4.7.17. Tampilan List Daftar Obat



Gambar 4.68 Tampilan List Daftar Obat

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Tambah Obat, No, Id Obat, Nama Obat, Satuan, Stok, Frist, Prev, Next, Last dan Logout.

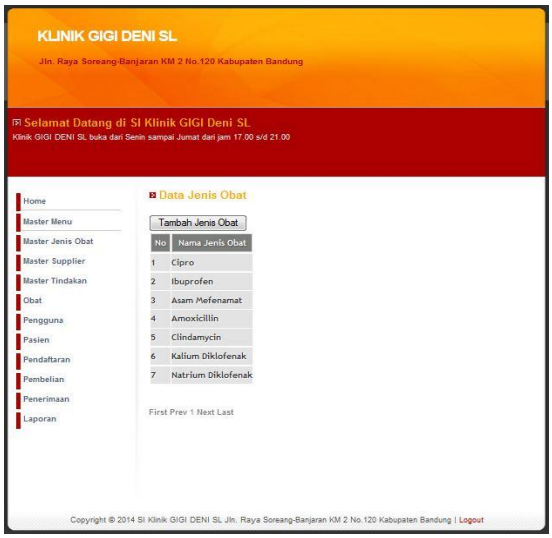
4.4.7.18. Tampilan Input Jenis Obat



Gambar 4.69 Tampilan Input Jenis Obat

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Nama Jenis Obat, Simpan, Batal dan Logout.

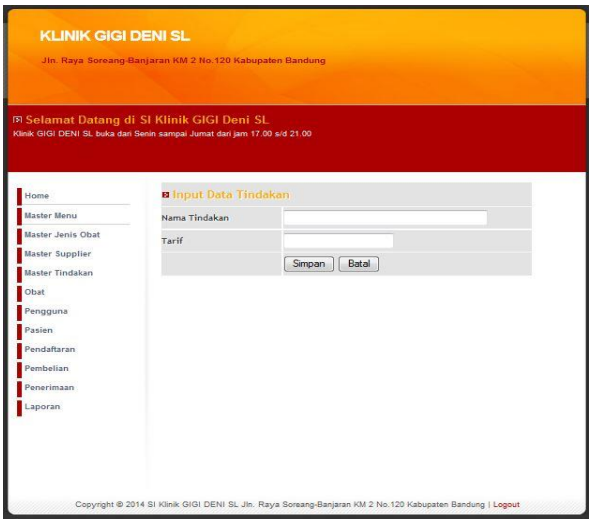
4.4.7.19. Tampilan List Daftar Data Jenis Obat



Gambar 4.70 Tampilan List Daftar Jenis Obat

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Tambah Jenis Obat, No, Nama Jenis Obat, Frist, Prev, Next, Last dan Logout.

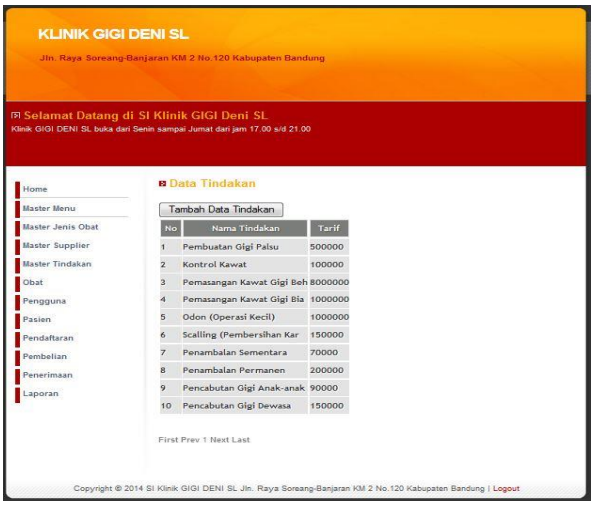
4.4.7.20. Tampilan Input Data Tindakan



Gambar 4.71 Tampilan Input Data Tindakan

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Nama Tindakan, Tarif, Simpan, Batal dan Logout.

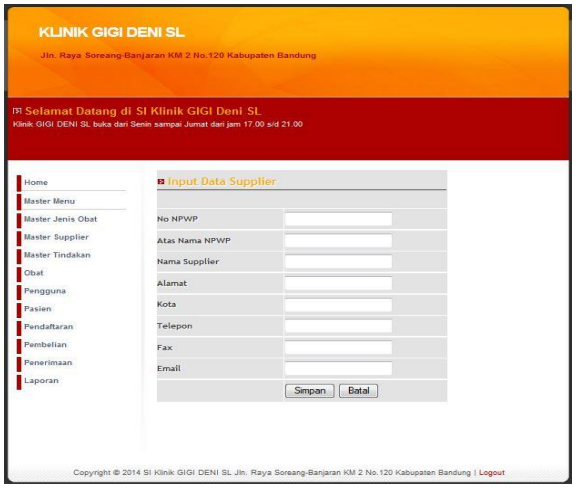
4.4.7.21. Tampilan List Daftar Data Tindakan



Gambar 4.72 Tampilan List Daftar Tindakan

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Tambah Data Tindakan, No, Nama Tindakan, Tarif, Frist, Prev, Next, Last dan Logout.

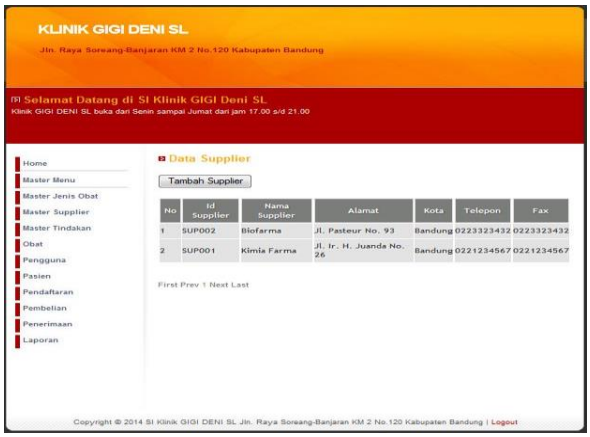
4.4.7.22. Tampilan Input Data Supplier



Gambar 4.73 Tampilan Input Data Supplier

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu No NPWP, Atas Nama NPWP, Nama Supplier, Alamat, Kota, Telepon, Fax, Email, Simpan, Batal dan Logout.

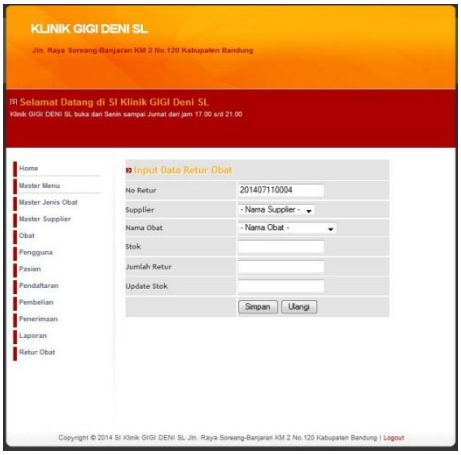
4.4.7.23. Tampilan List Daftar Data Supplier



Gambar 4.74 Tampilan List Daftar Supplier

Pada tampilan di atas bagian sebelah kanan pada layar terdapat menu Tambah Supplier, No, Id Supplier, Nama Supplier, Alamat, Frist, Kota, Telepon, Fax, Prev, Next, Last dan Logout.

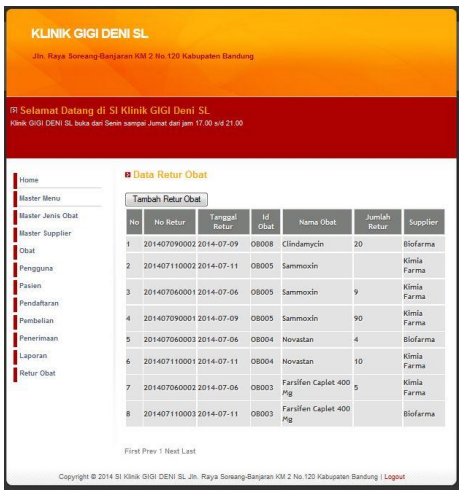
4.4.7.24. Tampilan Input Data Retur Obat



Gambar 4.75 Tampilan Input Data Retur Obat

Pada tampilan di atas bagian sebelah kanan pada layar terdapat field No. Retur, Supplier, Nama Obat, Stok, Jumlah Retur, Update Stok, Simpan, Ulangi dan Logout.

4.4.7.25. Tampilan List Daftar Data Retur Obat



Gambar. 4.76 Tampilan List Daftar Data Retur Obat

4.5 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak tersebut. Tujuan dari pengujian ini adalah untuk mengetahui kemampuan perangkat lunak dan menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu memprepresentasikan kajian pokok dari spesifikasi, analisis perancangan dan pengkodean dari perangkat lunak itu sendiri.

Dalam tahap ini penulis menggunakan pengujian black box, karena dengan pengujian black box kita dapat mengetahui apakah perangkat lunak yang dibuat dapat berfungsi dengan benar dan telah sesuai dengan yang diharapkan. Pengujian black box adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika intern perangkat lunak. Pengujian black box merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dibangkitkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak dicek apakah telah sesuai dengan yang diharapkan dan berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

4.5.1 Rencana Pengujian

Rencana pengujian Sistem Informasi Klinik Drg Deni S.L ini menggunakan data uji berupa sebuah data masukan dari admin pada Aplikasi yang telah dibuat sehingga akan menghasilkan keluaran yang diharapkan. Adapun pengujian Aplikasi ini penulis mempercayakan pada para tim penguji demi keakuratan hasil pengujian.

4.5.2 Kasus dan Hasil Pengujian

1. Pengujian Login Petugas

Berikut ini adalah tabel login petugas telah terdaftar

Tabel 4.24 Pengujian login Petugas

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data login petugas Username : irwan Password : irwan	Tercantum pada Textbox username & password	Dapat mengisi Login petugas sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Klik Login	Dapat masuk ke halaman selanjutnya untuk petugas yang telah terdaftar	Tombol login dapat berfungsi sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data login petugas Username : irwan Password : qwerty	Tidak dapat login dan menyampailkan pesan untuk mengulangi kembali proses login	Petugas gagal login dan menampilkan pesan"Userid atau Password Anda tidak benar. Ulangi Lagi”	[√] diterima [] ditolak

2. Pengujian Login Asisten Dokter

Berikut ini adalah tabel login dokter telah terdaftar

Tabel 4.25 Pengujian login Asisten Dokter

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data Login Asisten dokter Username : ineu Password :ineu	Tercantum pada Textbox username & password	Dapat mengisi Login dokter sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Klik Login	Dapat masuk ke halaman selanjutnya untuk dokter yang telah terdaftar	Tombol login dapat berfungsi sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data login Asisten dokter Username : ineu Password : asdfgh	Tidak dapat login dan menyampailkan pesan untuk mengulangi kembali proses login	Dokter gagal login dan menampilkan pesan"Userid atau Password Anda tidak benar. Ulangi Lagi”	[√] diterima [] ditolak

3. Pengujian Pendaftarana Pasien

Berikut ini adalah tabel pengujian pendaftaran pasien yang dilakukan oleh petugas

Tabel 4.26 Pengujian Pendaftaran pasien

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Petugas mengisi form pendaftaran pasien secara lengkap	Tercantum pada textbox form pendaftaran pasien	Dapat melakukan pengisian form pendaftaran sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Klik simpan	Menampilkan cetak kartu pasien dan daftar tabel pasien	Data pengisian daftar pasien tersimpan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pengisisan data Pasien tidak lengkap	Data yang disikan apabila tidak lengkap, tidak tersimpan kedalam data base.	Kembali pada halaman data pasien.	[√] diterima [] ditolak

4. Pengujian Pendaftaran Berobat.

Berikut ini adalah tabel pengujian pendaftaran berobat yang dilakukan oleh petugas

Tabel 4.27 Pengujian Pendaftaran Berobat

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Petugas mengisi form pendaftaran berobat secara lengkap	Tercantum pada textbox form pendaftaran berobat	Dapat melakukan pengisian form pendaftaran berobat sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Klik simpan	Menampilkan data pasien berobat	Data pengisian pendaftar bereobat tersimpan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pengisisan data id pasien tidak benar	Tidak dapat menampilkan data pasien untuk pendaftaran	Data pasien tidak muncul pada tabel pendaftaran berobat	[√] diterima [] ditolak

5. Pengujian Rekamedis

Berikut ini adalah tabel pengujian rekamedis yang dilakukan oleh asisten dokter

Tabel 4.28 Pengujian Rekamedis

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Asisten dokter mengisi form input data rekamedis	Tercantum pada textbox form input data rekamedis	Dapat melakukan pengisian form input data rekamedis sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Klik simpan	Menampilkan daftar pasien berobat selanjutnya	Data pengisian rekamedis tersimpan sesuai dengan yang diharapkan	[√] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pengisisan data rekamedis tidak lengkap	Menampilkan daftar berobat selanjutnya	Data history medis tidak lengkap	[√] diterima [] ditolak

6. Pengujian Resep dan Pembayaran.

Berikut ini adalah tabel pengujian resep dan pembayaran yang dilakukan oleh kasir.

Tabel 4.29 Pengujian Resep dan Pembayaran.

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
kasir mengisi form input data resep	Tercantum pada textbox form input data resep	Menampilkan data resep	[√] diterima [ ] ditolak
- Klik Tambah Ke List - Klik Cetak Kwitansi Pembayaran	- menampilkan data resep pada list obat - Menampilkan cetak kwitansi pembayaran	- Datar resep tampil ditabel list obat dan rincian pembayaran secara otomatis terisi - data kwitansi pembayaran terisi dan siap dicetak.	[√] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pengisian data resep tidak lengkap	- menampilkan data resep pada list obat - Menampilkan cetak kwitansi pembayaran	- Datar resep tidak tampil ditabel list obat dan rincian pembayaran tidak terisi - data kwitansi pembayaran tidak terisi.	[√] diterima [] ditolak

4.5.3. Kesimpulan Hasil Pengujian

Sesuai dengan hasil pengujian dengan menggunakan metode *black box* seluruh program dapat berjalan dengan baik. Tetapi apabila dikemudian hari ditemukan kesalahan-kesalahan yang baru, maka penulis akan melakukan perbaikan.