BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem yang Sedang Berjalan

Analisis yang berjalan pada Apotek Nusa Indah ini bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah yang dihadapi sistem untuk dijadikan landasan usulan perancangan analisis sistem yang sedang berjalan yang dilakukan berdasarkan urutan kejadian yang ada pada Apotek Nusa Indah.

4.1.1 Analisis Dokumen

Analisis dokumen merupakan kegiatan pengumpuan informasi mengenai dokumen-dokumen yang digunakan dalam suatu sistem. Tujuan dari analisis dokumen adalah mengetahui dan memahami dokumen-dokumen apa saja yang terlibat dan mengalir dalam sistem yang sedang berjalan. Dokumen yang digunakan pada sistem Pengelolaan data obat di Apotek Nusa Indah adalah sebagai berikut :

1. Nama Dokumen : Faktur
   Sumber : Apotek Nusa Indah
   Rangkap : -

Fungsi : Tanda bukti data obat yang telah dipesan oleh apotek kepada supplier terkait.

Item Data : No_Faktur, Tanggal, Nama_Obat, No_Batch, Satuan_Unit, Jumlah_Obat, Harga_Satuan, Sub_Total.

2. Nama Dokumen : Laporan Obat Masuk

   Sumber : Apotek Nusa Indah

   Rangkap : -

   Fungsi : Menyimpan data penerimaan obat harian dari supplier ke apotek.

   Item Data : Tanggal, Nama_Barang, PBF, No_Batch, Expire_Date, Jumlah, Harga.

4.1.2 Analisis Prosedur yang Sedang Berjalan

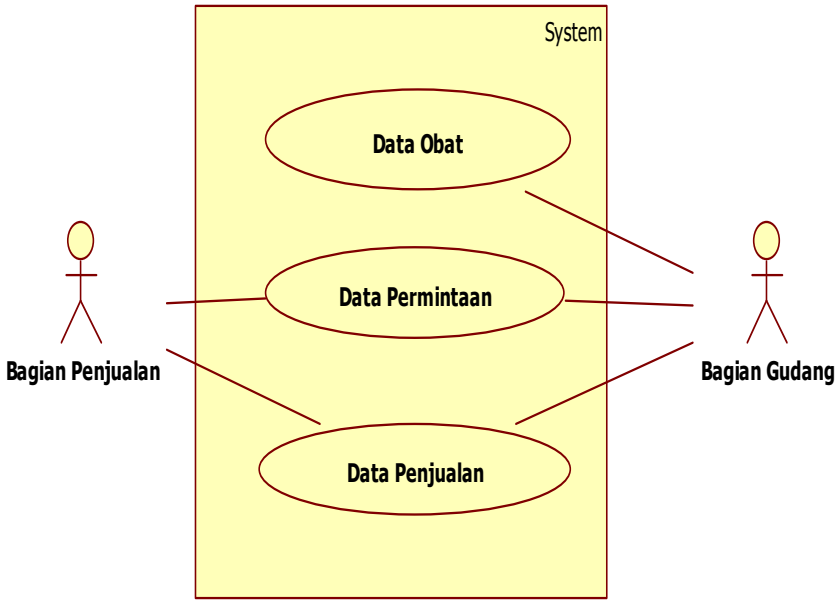
Analisis prosedur yang sedang berjalan menguraikan secara sistematis aktifitas - aktifitas yang terjadi dalam proses pengelolaan data obat. Untuk menentukan kebutuhan dari pengguna yaitu dengan cara menganalisis sistem yang sedang berjalan, kemudian mengevaluasinya. Adapun hasil dari proses analisis sistem informasi pengelolaan data obat yang sedang berjalan adalah sebagai berikut :

1. Proses penjualan obat berlangsung ketika pasien datang membawa resep dokter. Kemudian apoteker akan mengecek ketersediaan obat ke bagian gudang. Jika ada obat akan langsung diperhitungkan total harga yang harus dibayar oleh pasien tersebut.

2. Pasien membayar total harga yang sudah diberitahukan oleh apoteker, dan apoteker menyerahkan obat ke pada pasien tersebut.
3. Pada bagian gudang, setiap harinya akan mengecek ketersedian obat dan mengecek masa kadaluarsa obat di gudang. Jika obat sudah minimum dan terdapat obat yang kadaluarsa maka bagian gudang akan langsung membuat PO ke supplier untuk memesan obat yang dibutuhkan.
4. Supplier akan mengantarkan obat beserta dengan faktur. Bagian gudang akan mengecek obat yang datang dengan obat yang dipesan. Jika sudah sesuai maka bagian obat akan langsung mencatatnya di agenda barang masuk.
5. Pemerikasaan obat akan dilakukan setiap bulannya guna menghindari obat yang kadaluarsa. Jika terdapat obat yang kadaluarsa maka petugas apotek akan langsung membuangnya dengan mencatatnya terlebih dahulu di buku laporan barang keluar.

4.1.2.1 Usecase Diagram

Berikut ini adalah *diagram use case* yang menggambarkan proses utama dari sistem yang sedang berjalan :



Gambar 4.1 *Use Case Diagram* Sistem yang sedang Berjalan

Skenario Usecase yang sedang berjalan

Nama Use Case : Data Obat

Aktor : Bagian Gudang

Tabel 4.1 Skenario *Use Case* Data Obat yang Sedang Berjalan

Aksi Aktor	Reaksi Sistem
1. Mengecek ketersediaan Obat	
2. Memasukan Data Obat baru	
	3. Merekap Data Obat

Nama Use Case : Data Permintaan

Aktor : Bagian Gudang

Tabel 4.2 Skenario *Use Case* Data Permintaan yang Sedang Berjalan

Aksi Aktor	Reaksi Sistem
1. Membuat surat permintaan barang	
	2. Menerima surat permintaan barang
	3. Mengirim barang pesanan
4. Mengecek kondisi barang	
5. Menerima barang pesanan	
	7. Memberikan faktur dan kwitansi

Nama Use Case : Data Penjualan

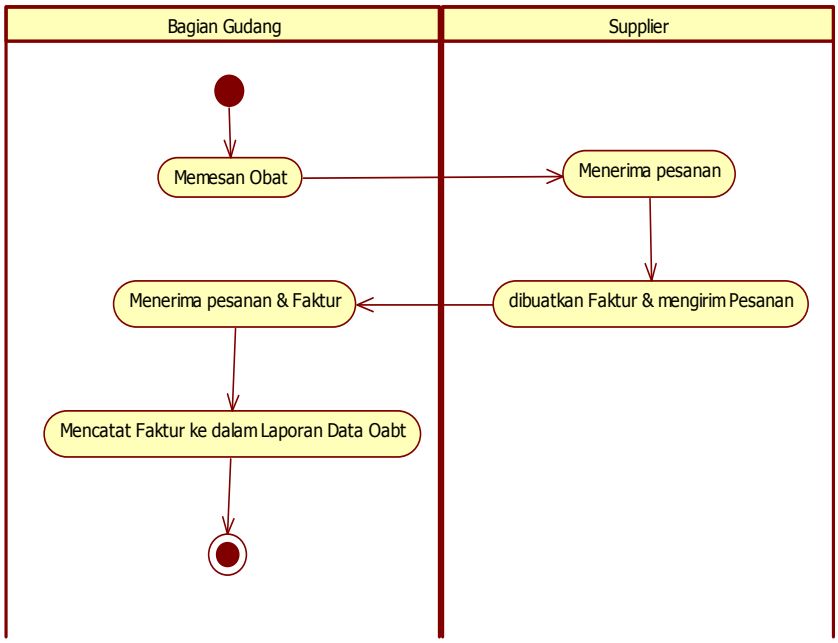
Aktor : Bagian Penjualan

Tabel 4.3 Skenario *Use Case* Data Penjualan yang Sedang Berjalan

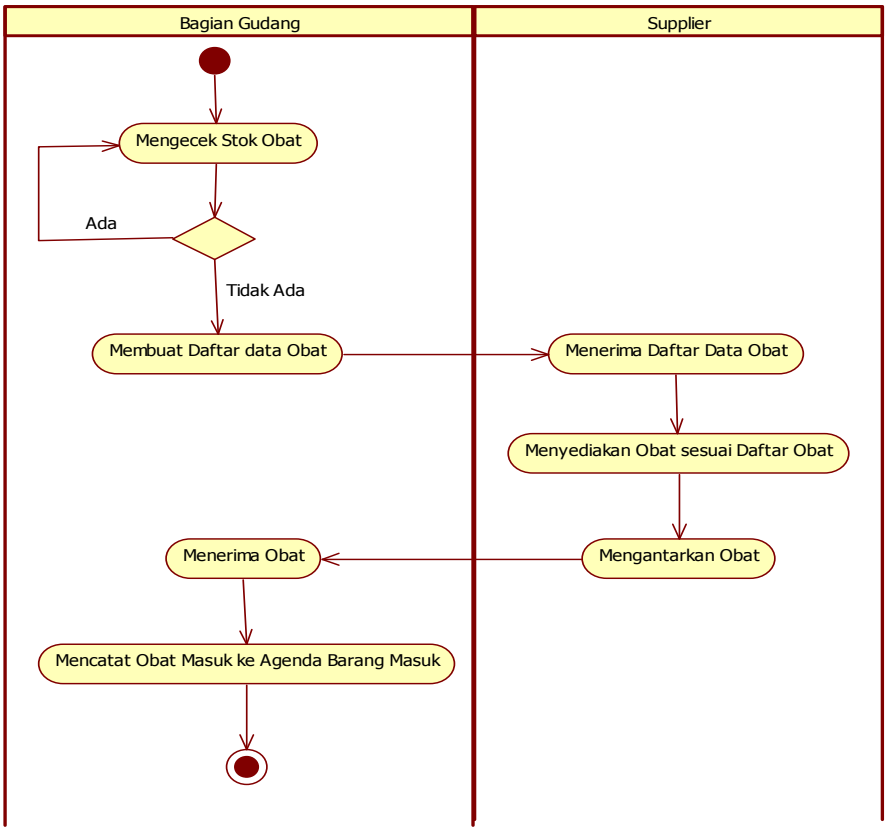
Aksi Aktor	Reaksi Sistem
1. Menerima Resep	
	2. Mengecek Ketersediaan Obat
3. Menghitung Total Bayar	
	Membayar Total Bayar
4. Menyerahkan Obat	

4.1.2.2 Activity Diagram

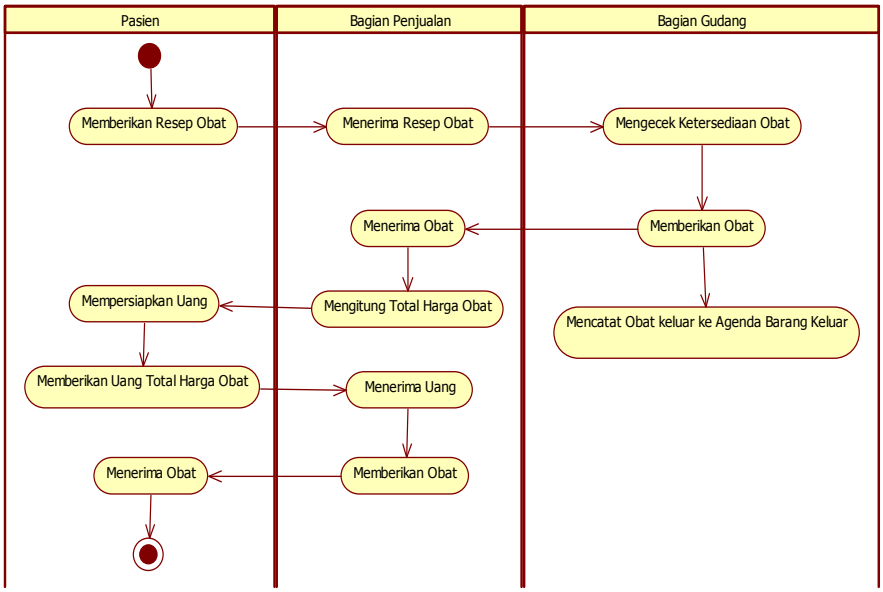
Berikut ini adalah *diagram activity* yang menggambarkan aktivitas yang terjadi dalam sistem yang berjalan, diagram ini menjelaskan detail dari proses use case sebelumnya.



Gambar 4.2 *Activity Diagram* Sistem Data Obat yang Sedang Berjalan



Gambar 4.3 *Activity Diagram* Sistem Permintaan Obat yang sedang Berjalan



Gambar 4.4 *Activity Diagram* Sistem Penjualan Obat yang sedang Berjalan

4.1.3 Evaluasi Sistem yang Berjalan

Melihat dari sistem yang sedang berjalan, bahwa kegiatan dari proses pengeluaran dan pemasukan barang masih memiliki kekurangan dan kelemahan, antara lain sebagai berikut :

Tabel 4.4 Evaluasi Sistem yang Sedang Berjalan

No.	Permasalahan	No.	Rencana Perancangan / Solusi
1.	Dalam proses pencatatan data obat masuk dan keluar masih manual dengan mencatat ke agenda barang keluar dan agenda barang masuk, sehingga kesulitan mencari data sebelumya.	1.	Dengan adanya Sistem Informasi Pengelolaan Data Obat yang baru dapat mempermudah penginputan data obat.
2.	Sulitnya penentuan stok obat dengan mencatat setiap penerimaan dan pengeluaran obat dengan menjumlahkan atau mengurangi stok obat yang ada di agenda stok obat.	2.	Dengan adanya Sistem Informasi Pengelolaan Data Obat yang baru dapat mempermudah menentukan stok obat yang otomatis bertambah dan berkurang.
3.	Dalam pencarian obat yang kadaluarsa masih kesulitan	3.	Dengan adanya Sistem Informasi Pengelolaan Data Obat dapat mempercepat pencarian data obat yang sudah kadaluarsa begitu juga dengan stok obat yang sudah minimum dengan menggunakan sistem alert. (penandaan warna pada data)

Evaluasi sistem ini dibuat untuk membantu meminimalisasi kekurangan dan kelemahan pada sistem yang sedang berjalan sehingga diharapkan dengan adanya perancangan sistem informasi persediaan barang ini dapat memenuhi kebutuhan di Apotek Nusa Indah.

4.2 Perancangan Sistem

Perancangan Sistem dibuat sebagai tahapan untuk mempersiapkan proses implementasi sistem yang diinginkan, dan untuk menggambarkan secara jelas proses-proses yang diinginkan oleh *user.* Sesuai dengan metode pendekatan yang digunakan adalam pendekatan *objeck oriented*, maka metode yang digunakan untuk menggambarkan seluruh proses dan objek adalah *Unified Model Language.*

Adapun perancangan ini meliputi :

1. Perancangan *Use Case Diagram* dan skenarionya
2. Perancangan *Activity Diagram*
3. Perancangan *Sequence Diagram*
4. Perancangan *Class Diagram*
5. Perancangan *Object Diagram*
6. Perancangan *Component Diagram*
7. Perancangan *Deployment Diagram*

4.2.1. Tujuan Perancangan Sistem

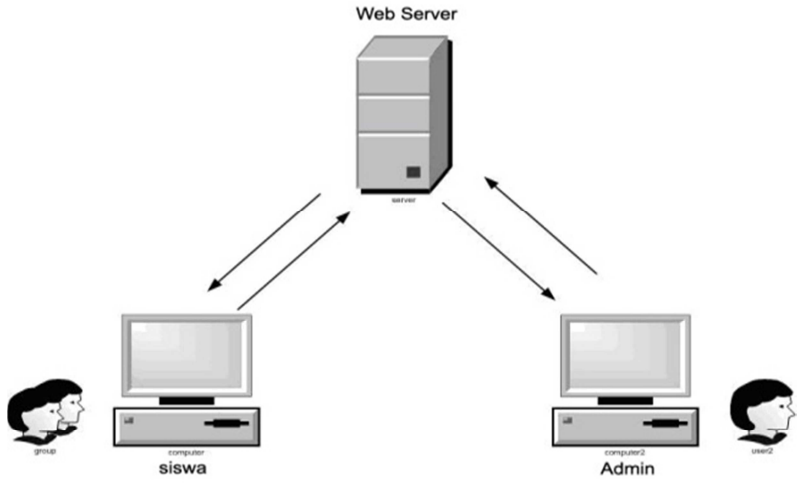
Merancang Aplikasi Sistem Informasi Pengelolaan Data Obat, dengan menggunakan pemograman PHP dan MySQL sebagai media *database*. Berupaya untuk mencoba memberikan sesuatu yang dapat

mempermudah petugas Apotek Nusa Indah dengan dirubahnya sistem pencatatan data obat yang masih manual menjadi terkomputerisasi.

4.2.2. Gambaran Umum Sistem yang Diusulkan

Website yang akan dibangun memilik 2 tipe user dimana terdapat bagian gudang dan Admin. Registered user artinya bagian gudang yang telah melakukan registrasi sehingga memiliki username dan password khusus untuk dapat mengakses layanan yang lebih lanjut. Sedangkan admin adalah *administrator* yang memegang akses penuh. Admin dapat menambah, merubah, menghapus data admin itu sendiri ,user, data obat, data penjualan, dan data permintaan dan sebagainya.

Berikut ini adalah skema gambaran umum dari aplikasi ini :



Gambar 4.5 Gambaran Umum Sistem Informasi pengelolaan data obat

(Sumber : Budi Irawan, 2005. *Jaringan Komputer*. GRAHA ILMU, Yogyakarta)

4.2.3 Perancangan Prosedur Sistem yang Diusulkan

Perancangan Prosedur Sistem yang diusulkan melakukan beberapa perubahan pada proses manual yang dianggap lambat dalam proses kerjanya, prosedur yang diusulkan adalah sebagai berikut :

Prosedur pengeluaran obat ke pasien yang diusulkan :

1. Pasien datang dengan membawa resep yang kemudian diserahkan ke petugas apotek.
2. Bagian Gudang akan melakukan pencarian obat dalam data base sesuai yang terdapat pada resep, apabila obat diminta habis atau tidak ada di Apotek maka petugas akan menyerahkan kembali resep ke pasien tersebut.
3. Bila obat ada di Apotek, apoteker akan mencatat keterangan obat sesuai yang dibutuhkan pada resep. Kemudian bagian penjualan akan menghitung total harga yang harus dibayarkan oleh pasien. Pasien dipanggil untuk membayar resep dan obat tersebut diserahkan.
4. Kemudian bagian gudang akan menginputkan data obat yang keluar kedalam database, maka stok obat yang terpakai akan otomatis berkurang.

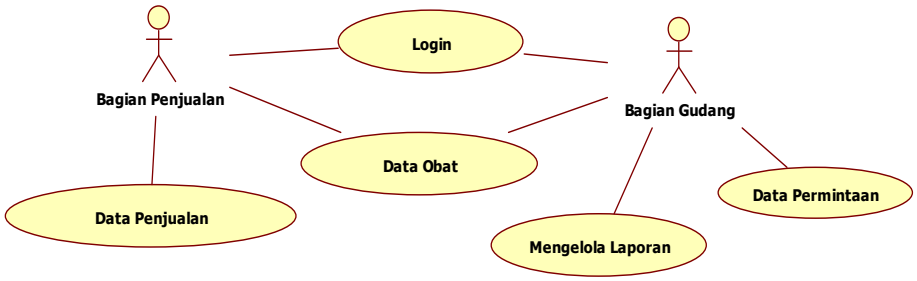
   Prosedur permintaan obat yang diusulkan sebgai berikut :

   1. Untuk melakukan permintaan kebutuhan obat di apotek terlebih dahulu bagian gudang membuat lembar permintaan dengan mengambil data obat di database apotek.
   2. Kemudian petugas apotek memberikannya ke supplier apotek yang biasa mensupply obat di apotek.

3. Supplier akan mengirimkan obat yang sudah dipesan oleh apotek beserta faktur. Kemudian apotek mengecek barang yang datang dimulai dengan jumlah obat sampai pada kualitas obatnya dan kemasannya dengan lembar permintaan yang sudah dibuat sebelumnya.
4. Jika diantara barang yang datang ada kerusakan atau tidak sesuai dengan apa yang dipesan maka petugas tidak akan memasukan data barang masuk ke dalam database. Penginputan dilakukan hanya ketika barang datang sudah sesuai dengan lembar permintaan yang dibuat sebelumnya terkecuali stok obat yang benar-benar sudah habis.

4.2.3.1 *Use case Diagram*

Berikut ini adalah diagram use case yang menggambarkan proses utama dari sistem yang diusulkan :



Gambar 4.6 *Use Case Diagram* Sistem yang Diusulkan

Skenario Use Case Sistem yang Diusulkan

Nama Use Case : Login

Aktor : Bagian Gudang, Bagian Penjualan

Tabel 4.5 Skenario *Use Case* Login

Aktor	Reaksi Sistem
1. User memasukkan username dan password	
	2. Sistem mengecek valid atau tidaknya masukan
	3. Jika data valid maka sistem akan menampilkan menu utama, jika tidak sistem akan menampilkan pesan error.

Nama Use Case : Data Obat

Aktor : Bagian Gudang

Tabel 4.6 Skenario *Use Case* Data Obat yang Diusulkan

Aktor	Reaksi Sistem
1. User dapat menambah, mengubah, menghapus serta mencari data barang	
	2. Menyimpan data obat yang di input
	3. Menampilkan pesan input berhasil

Nama Use Case : Data Penjualan

Aktor : Bagian Penjualan

Tabel 4.7 Skenario *Use Case* Data Penjualan yang diusulkan

Aktor	Reaksi Sistem
1. Menerima kumpulan resep	
2. Menginput data obat yang keluar	

Aktor	Reaksi Sistem
	3. Menyimpan inputan/proses yang dilakukan oleh user
	4. Menampilkan pesan input berhasil

Nama Use Case : Data Permintaan

Aktor : Bagian Gudang

Tabel 4.8 Skenario *Use Case* Data Permintaan yang Diusulkan

Aktor	Reaksi Sistem
1. Menginput data barang yang akan dipesan	
	2. Menyimpan inputan/proses yang dilakukan oleh user
	3. Menampilkan pesan input berhasil

Nama Use Case : Mengelola Laporan

Aktor : Bagian Gudang

Tabel 4.9 Skenario *Use Case* Mengelola Laporan yang Diusulkan

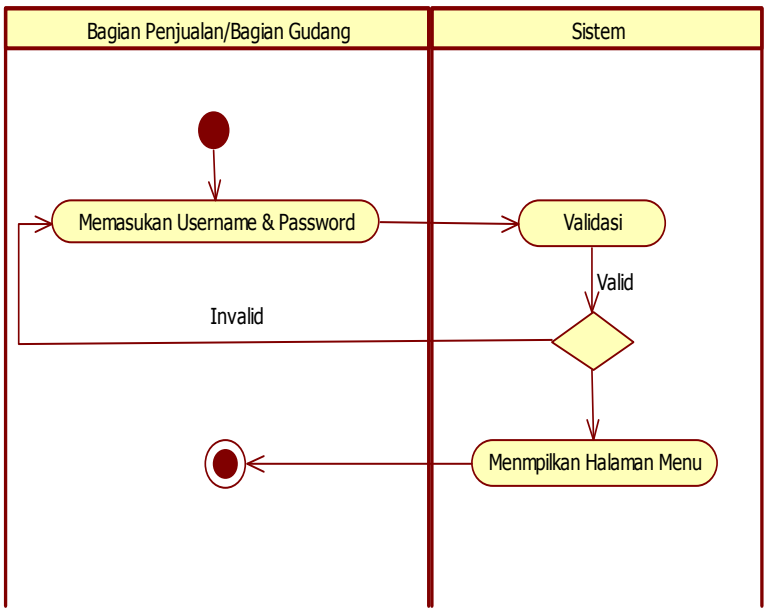
Aktor	Reaksi Sistem
1. Pilih laporan yang akan dicetak	
	2. Menampilkan laporan yang akan dicetak
3. Mencetak laporan	

4.2.3.2 *Activity Diagram*

Activity diagram memodelkan alur kerja (*workflow*) sebuah proses bisnis dan urutan aktifitas dalam suatu proses.

a. *Activity Diagram* Login

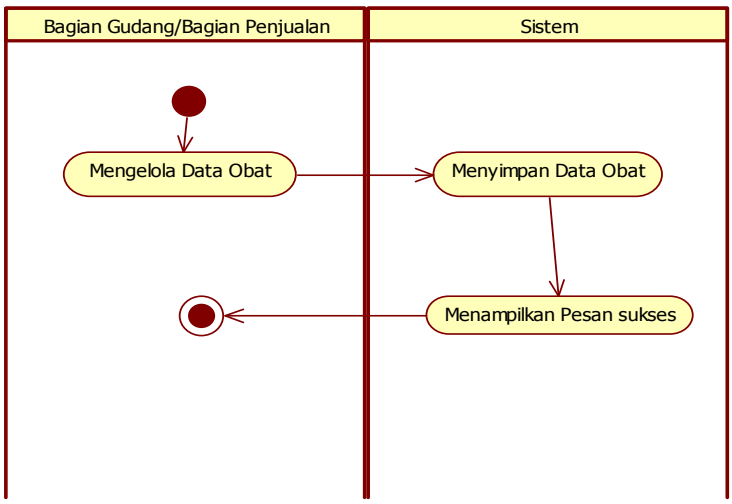
Activity Diagram Login memperlihatkan aktifivitas login. Aktor yang berperan dalam proses ini adalah Bagian Penjualan dan Bagian Gudang. Adapun diagram aktivitas login adalah sebagai berikut :



Gambar 4.7 *Activity Diagram* Login

b. *Activity Diagram* Data Obat

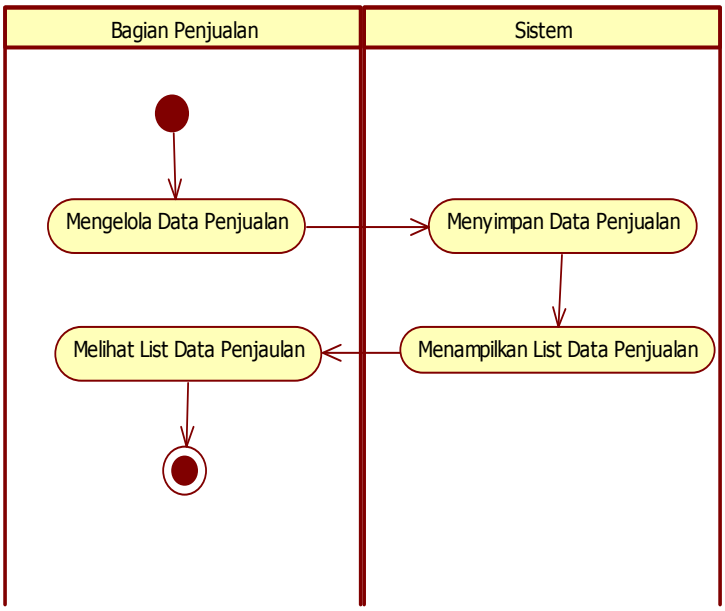
Activity Diagram pengolahan data obat memperlihatkan segala aktifivitas penginputan data obat. Aktor yang berperan dalam proses ini adalah bagian gudang. Adapun diagram aktivitas data obat adalah sebagai berikut :



Gambar 4.8 *Activity Diagram* Data Obat yang Diusulkan

c. *Activity diagram* Data Penjualan

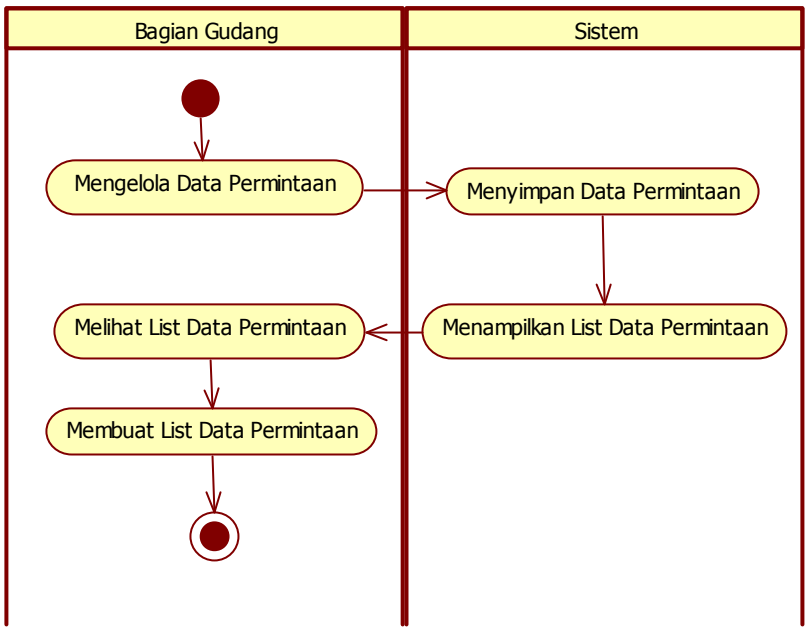
Activity Diagram data penjualan memperlihatkan segala aktifivitas penginputan data penjualan. Aktor yang berperan dalam proses ini adalah bagian penjualan. Adapun diagram aktivitas Data Penjulaan adalah sebagai berikut :



Gambar 4.9 *Activity Diagram* Data Penjualan yang Diusulkan

d. *Activity Diagram* Data Permintaan yang Diusulkan

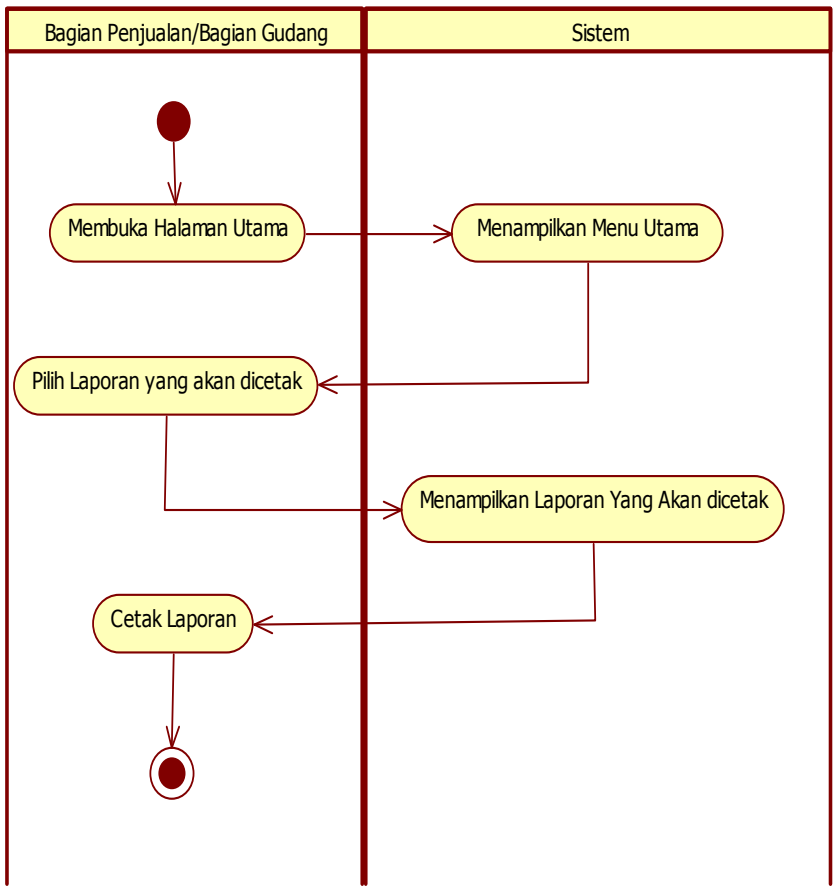
Activity Diagram data permintaan memperlihatkan segala aktifivitas penginputan data permintaan. Aktor yang berperan dalam proses ini adalah bagian gudang. Adapun diagram aktivitas Data Permintaan adalah sebagai berikut :



Gambar 4.10 *Activity Diagram* Data Permintaan yang Diusulkan

e. *Activity Diagram* Mengelola Laporan

Activity Diagram Mengelola Laporan memperlihatkan segala aktifivitas percetakan laporan. Aktor yang berperan dalam proses ini adalah bagian gudang. Adapun diagram aktivitas Mengelola Laporan adalah sebagai berikut:



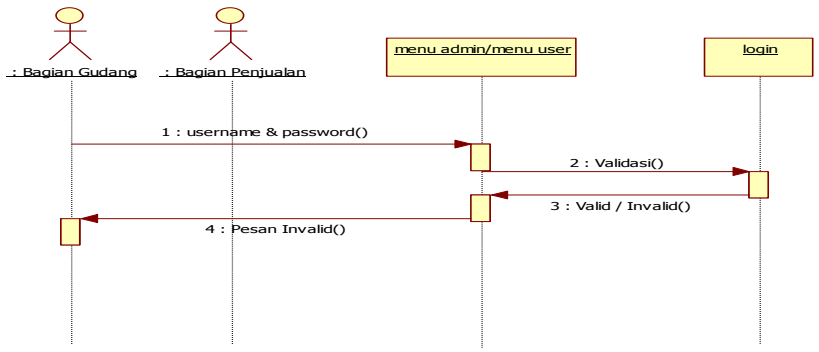
Gambar 4.11 *Activity Diagram* Mengelola Laporan

4.2.3.3 *Sequence Diagram*

Sequence diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan *use case*. *Sequence diagram* juga menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram *sequence* harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

a. *Diagram Sequence* Login

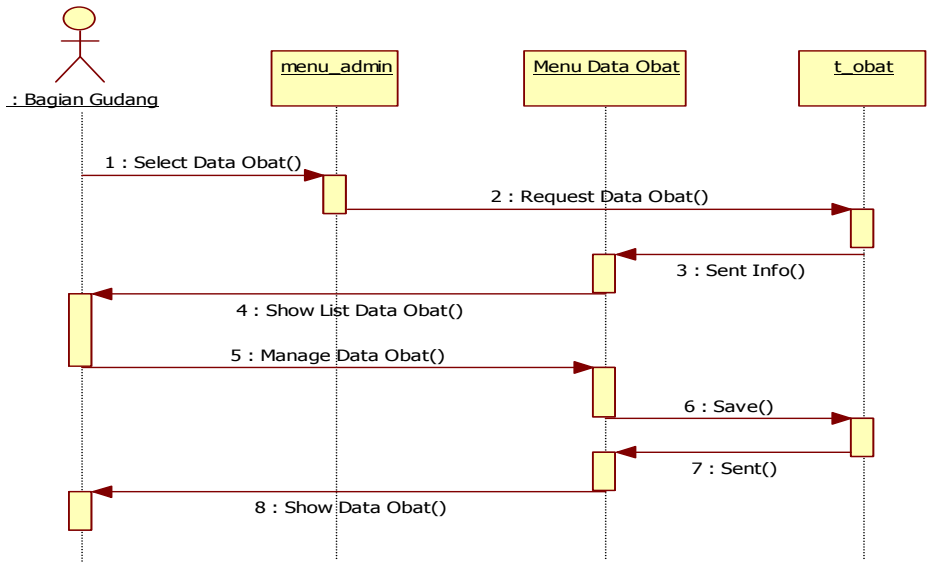
Berikut merupakan *sequence diagram* untuk *Login.* Adapun gambarnya sebagai berikut :



Gambar 4.12 *Sequence diagram* Login

b. *Sequence Diagram* Data Obat

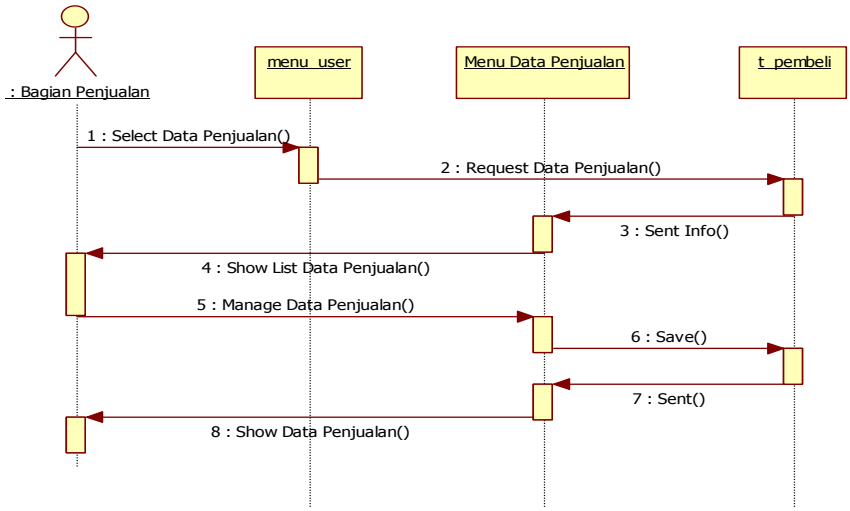
Berikut merupakan *sequence diagram* untuk Data Obat. Adapun gambarnya sebagai berikut :



Gambar 4.13 *Sequence Diagram* Data Obat

c. *Sequence diagram* Mengelola Data Penjualan

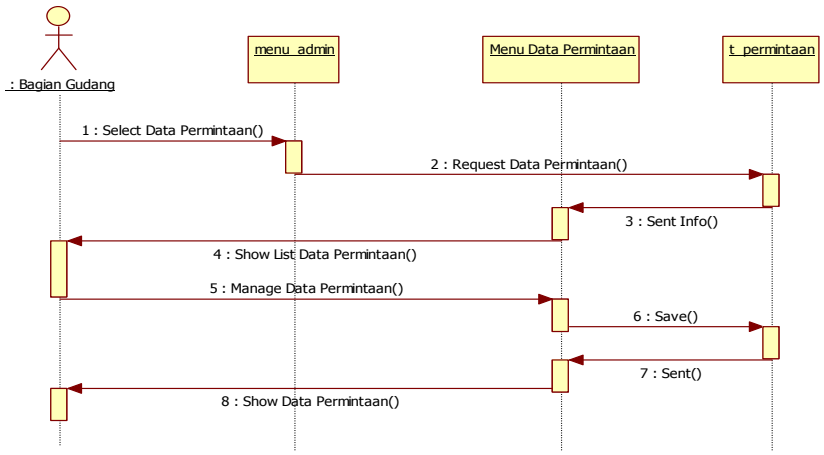
Berikut merupakan *sequence diagram* untuk Data Penjualan. Adapun gambarnya sebagai beikut :



Gambar 4.14 *Sequence Diagram* Data Penjualan

d. *Sequence Diagram* Mengelola Data Permintaan

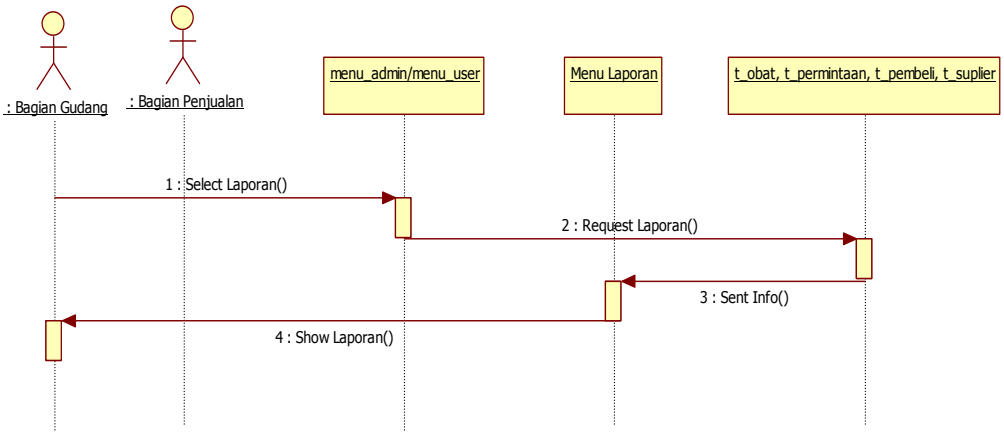
Berikut merupakan *sequence diagram* untuk Data Permintaan. Adapun gambarnya sebagai beikut :



Gambar 4.15 *Sequence Diagram* Data Permintaan

e. *Sequence Mengelola* Data Laporan

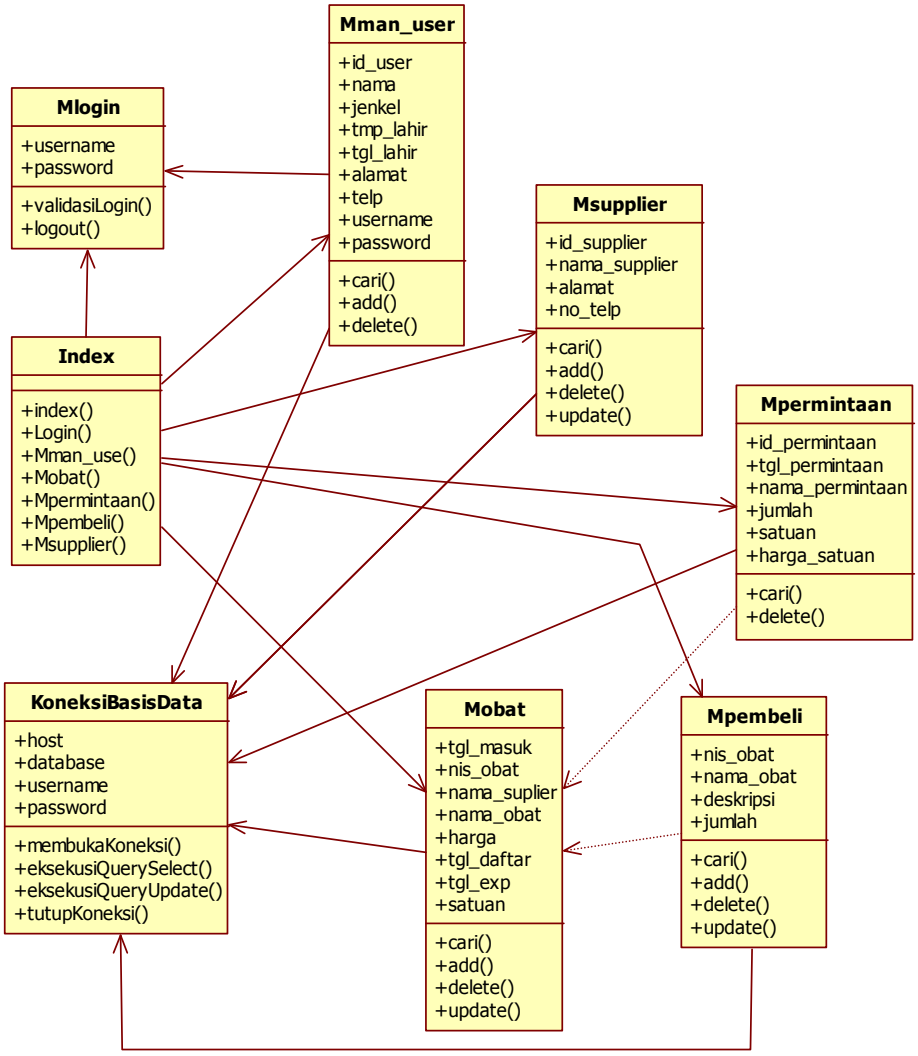
Berikut merupakan *sequence diagram* untuk Data Permintaan. Adapun gambarnya sebagai beikut :



Gambar 4.16 *Sequence Diagram* Data Laporan

4.2.3.4 *Class Diagram*

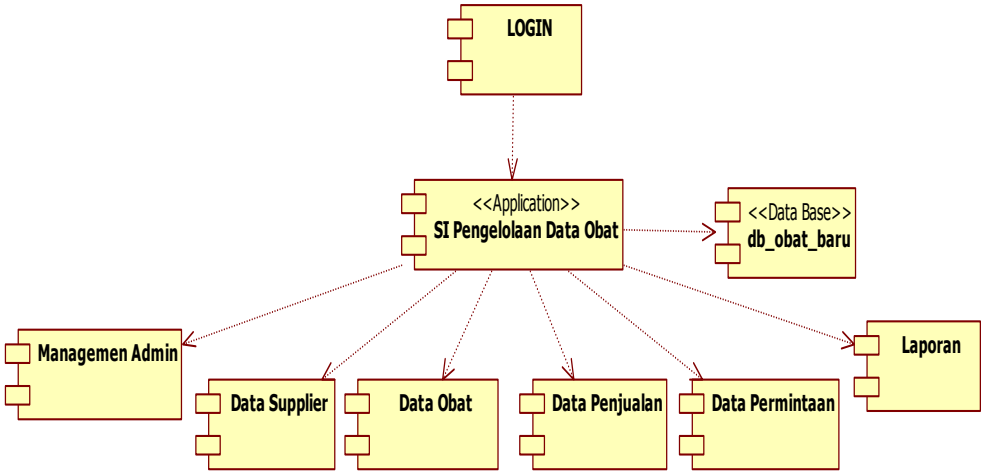
Class Diagram membantu kita dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. *Class Diagram* memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model desain dari suatu sistem.



Gambar 4.17 *Class Diagram* Sistem Informasi Pengelolaan Data Obat

4.2.3.5 *Component Diagram*

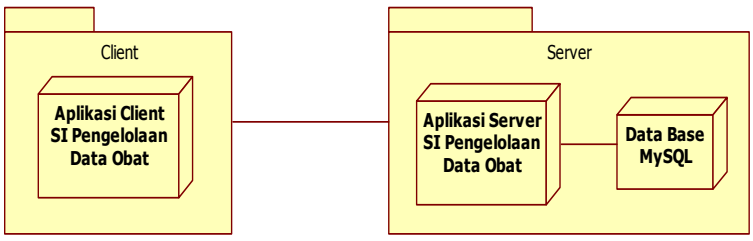
Component adalah unit fisik yang nyata yang menjadi bagian dari *deployment independent*. *Component* ini diimplementasikan meskipun pada sistem yang kecil. Perancangan menu pada sistem informasi yang diusulkan digambarkan oleh *component diagram* berikut :



Gambar 4.18 *Component Diagram* Sistem Informasi Pengelolaan data Obat

4.2.3.6 *Deployment Diagram*

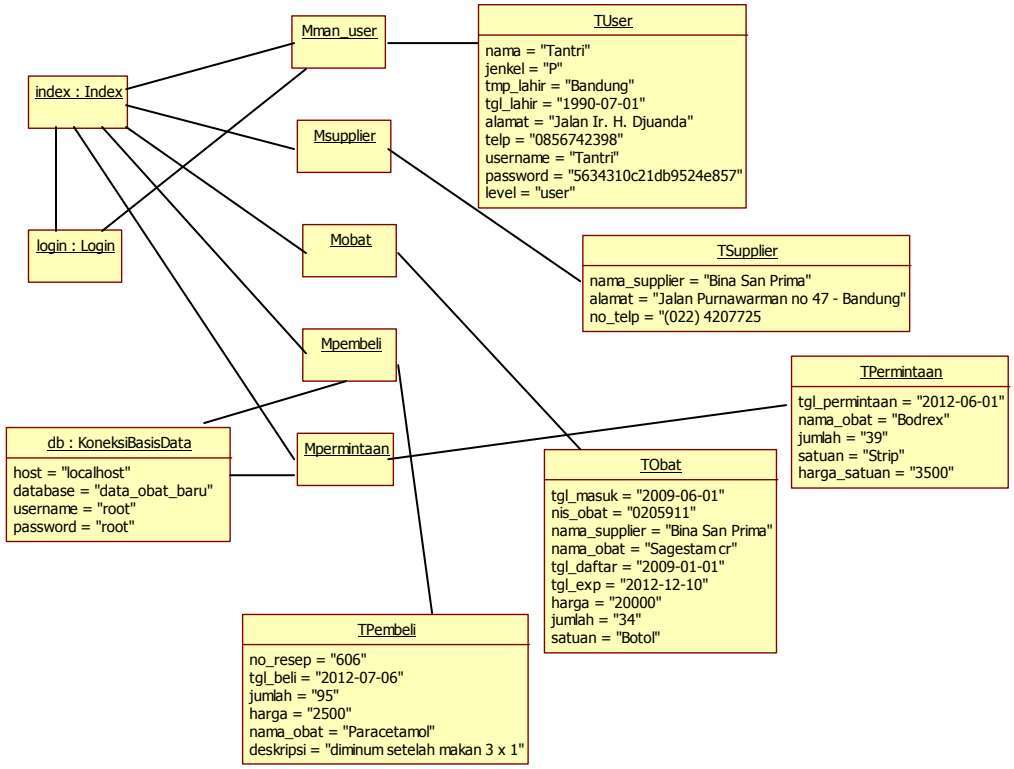
Berikut ini *Deployment* diagram yang menggambarkan susunan fisik dan perangkat lunak dalam aplikasi :



Gambar 4.19 *Deployment Diagram* sistem Informasi Pengelolaan Data Obat

4.2.3.7 *Object Diagram* yang Diusulkan

Berikut ini *object diagram* yang menggambarkan struktur sistem dari segi penamaan objek dan jalannnya objek dalam sistem pengelolaan data obat :



Gambar 4.20 *Object Diagram* Sistem Informasi Pengelolaan Data Obat

4.2.4 Kodefikasi

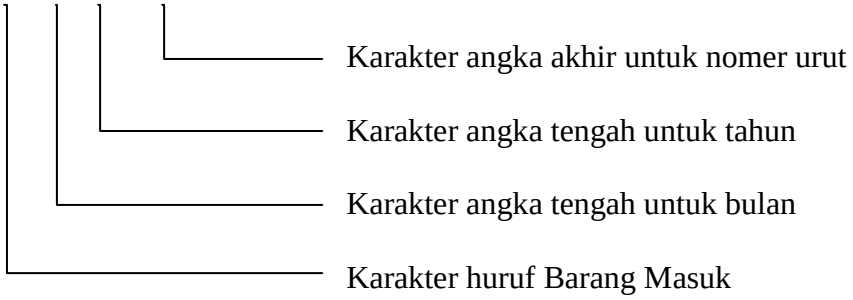
Kodefikasi ini berfungsi untuk mendefinisikan suatu objek secara singkat, mengklasifikasikan data dan memasukan data. Adapun tujuan dari kodefikasi itu sendiri adalah :

a. Memperkecil kemungkinan terjadinya kesalahan dalam penginputan data.
b. Mengefisiensikan penyimpanan data dalam *memory*.
c. Mempermudah dalam mengidentifikasikan suatu proses.
d. Mengurangi terjadinya redudansi.
e. Menghemat media penyimpanan data atau *memory*.

Pada program Sistem Informasi Pesediaan Barang pada Bagian Logistik UTD PMI ini terdapat beberapa kodefikasi yang ada diperusahaan untuk memudahkan dalam pencarian data yaitu :

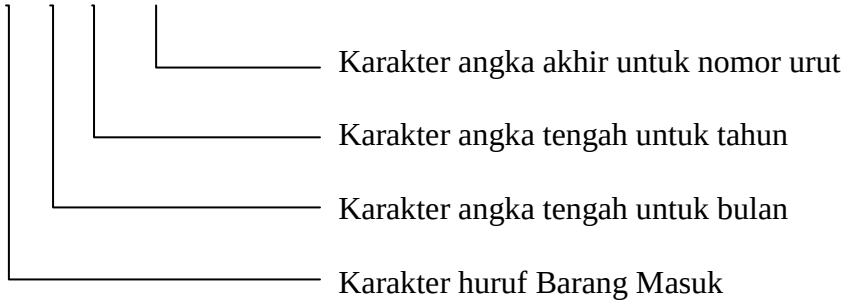
1) Kode Barang Masuk (No. Permintaan)

XX/xx/xx/xxxxxx



Contoh :

BM/06/12/000003

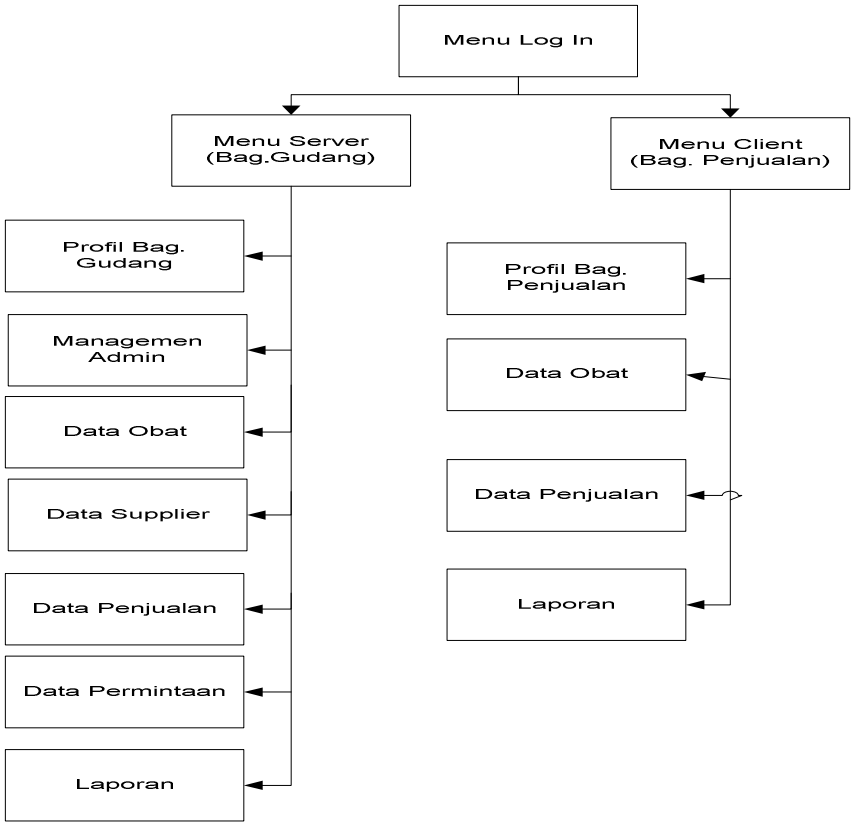


4.2.5 Perancangan Antar Muka

Perancangan antar muka ini menggambarkan rancangan dari keseluruhan tampilan yang ada pada sistem yang telah dibuat. Adapun rancangan tampilan yang dibuat meliputi struktur menu, perancangan input, dan perancangan output.

4.2.5.1 Struktur Menu

Pada perancangan struktur menu ini akan dijelaskan tentang menu-menu yang ada dan tersusun pada sistem yang telah dibuat ini. Berikut gambarnya :



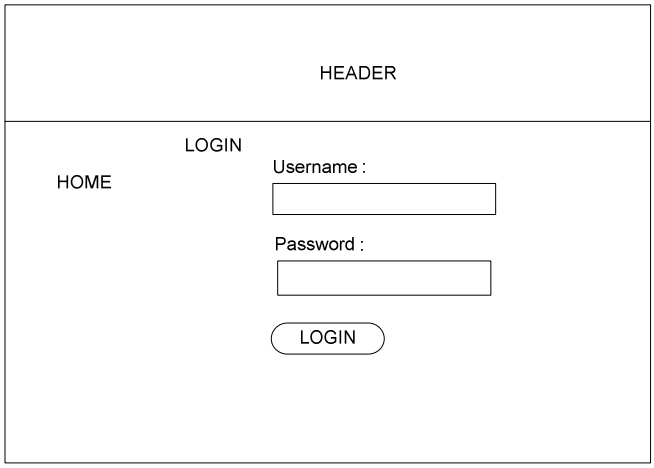
Gambar 4.21 Struktur Menu Sistem Informasi Pengelolaan Data Obat

4.2.5.2 Perancangan Input

Perancangan input yaitu sebuah tampilan yang dirancang sebagai tempat untuk memasukan data-data yang diberikan oleh pengguna sistem. Adapun masukan pada sistem Pengelolaan Data Obat ini adalah sebagai berikut :

1. Desain Form Login

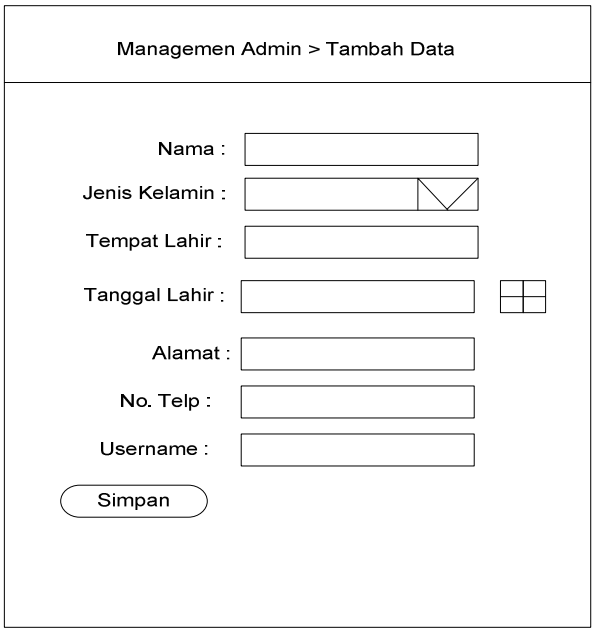
Form login pada sistem ini penggunanya yaitu Bagian Gudang dan Apoteker. Melalui form ini pengguna yang boleh masuk sistem adalah pengguna yang mengetahui username dan password.



Gambar 4.22 Desain Login

2. Desain Form Input Managemen Admin

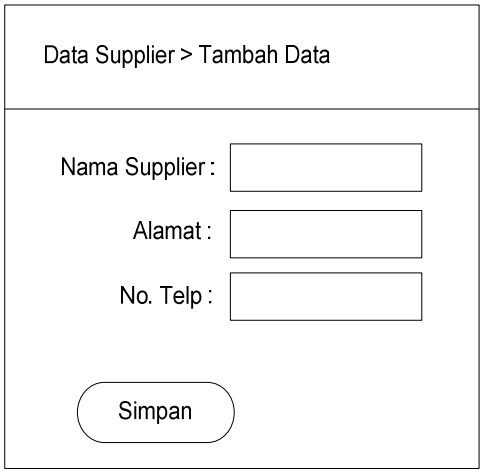
Form input managemen admin di akses oleh Admin yaitu Bagian Gudang. Ketika admin meng-klik tambah data dan kemudian inputannya di simpan ke database yang telah terkoneksi.



Gambar 4.23 Desain Form Input Managemen Admin

3. Desain Form Input Data Supplier

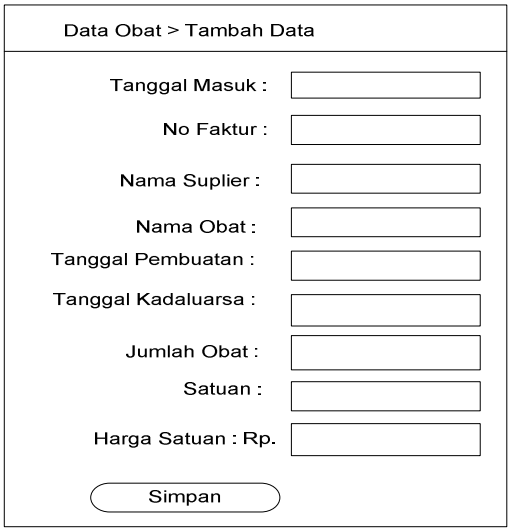
Form input managemen admin di akses oleh Admin yaitu Bagian Gudang. Ketika admin meng-klik tambah data dan kemudian inputannya di simpan ke database yang telah terkoneksi.



Gambar 4.24 Desain Form Input Data Supplier

4. Desain Form Input Data Obat

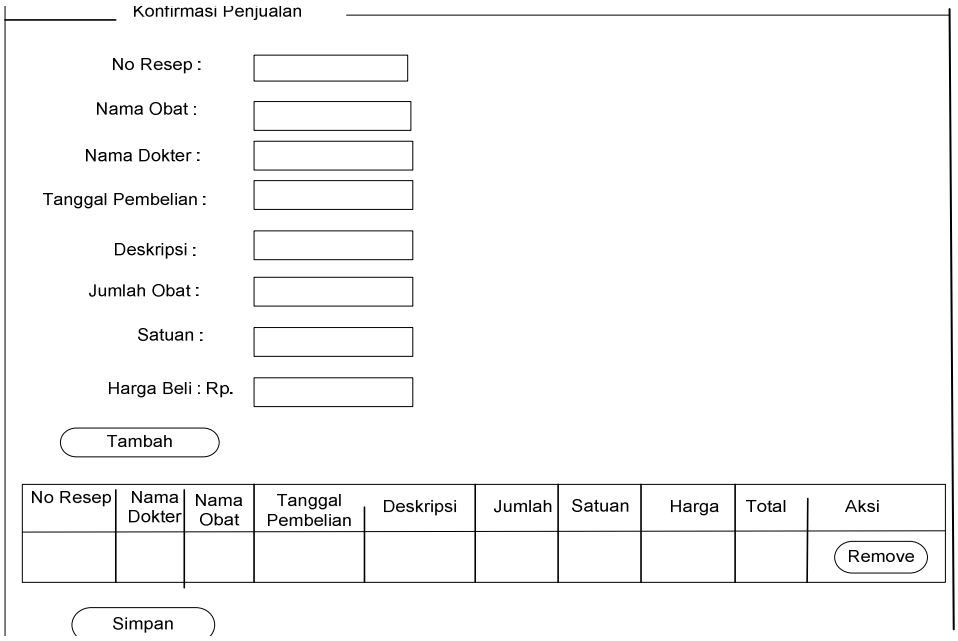
Form input data obat di akses oleh Admin yaitu Bagian Gudang. Ketika admin meng-klik tambah data dan kemudian inputannya di simpan ke database yang telah terkoneksi.



Gambar 4.25 Desain Form Input Data Obat

5. Desain Form Input Data Penjualan

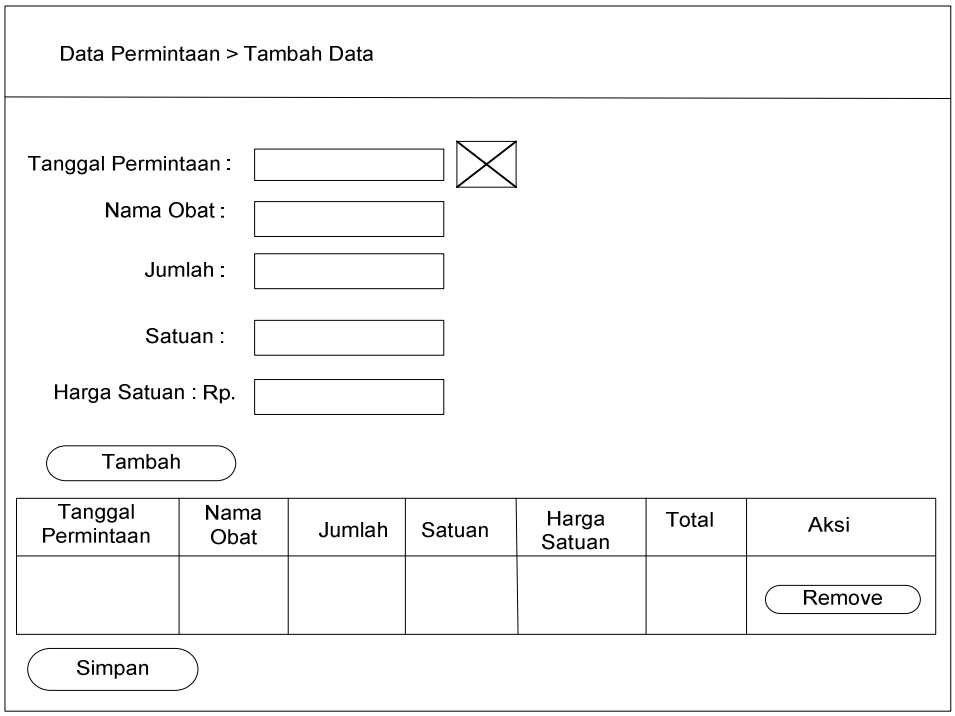
Form input data penjualan di akses oleh user yaitu bagian penjualan. Ketika bagian penjualan meng-klik tambah data dan kemudian inputannya di simpan ke database yang telah terkoneksi.



Gambar 4.26 Desain Form Input Data Penjualan

6. Desain Form Input Data Permintaan

Form input data permintaan di akses oleh admin yaitu bagian gudang. Ketika bagian gudnag meng-klik tambah data dan kemudian inputannya di simpan ke database yang telah terkoneksi.



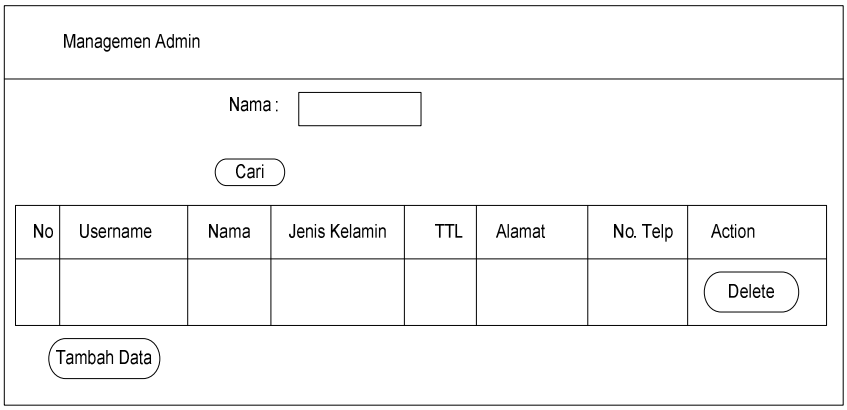
Gambar 4.27 Desain Form *Input* Data Permintaan

4.2.5.3 Perancangan Output

Perancangan Output atau keluaran merupakan hasil dari pengolahan data yang diterima dari proses masukan data berupa informasi yang berguna bagi pengguna atau *user*. Adapun rancangan output dari aplikasi pengelolaan data obat ini adalah sebagai berikut :

a. Desain Tampilan Output Managemen Admin

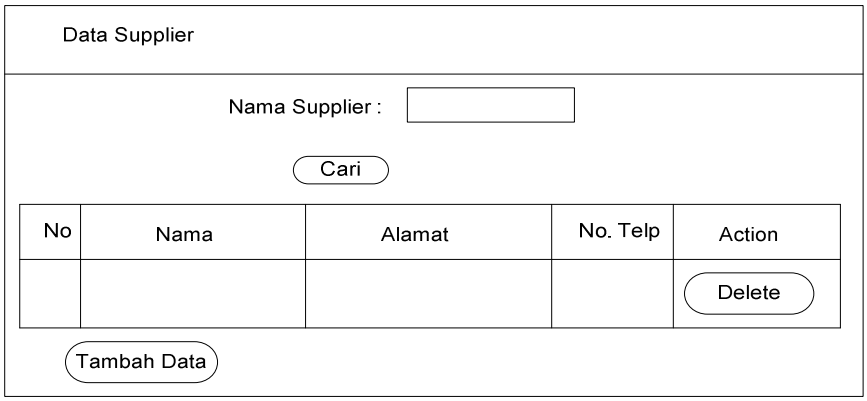
Desain tampilan keluaran ini adalah tampilan berupa tabel yang berisi data admin yang telah tersimpan di database yang telah di inputkan admin.



Gambar 4.28 Desain Tampilan *Output* Managemen Admin

b. Desain Tampilan Output Data Supplier

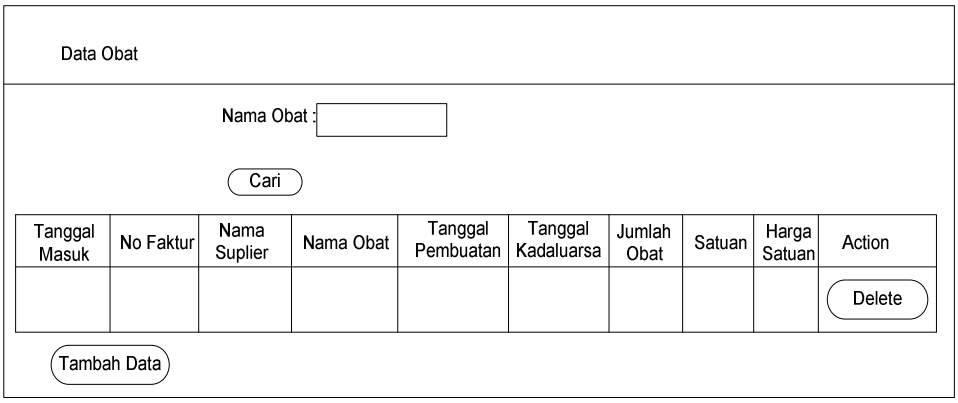
Desain tampilan keluaran ini adalah tampilan berupa tabel yang berisi data supplier yang telah tersimpan di database yang telah di inputkan admin.



Gambar 4.29 Desain Tampilan *Output* Data Supplier

c. Desain Tampilan Output Data Obat

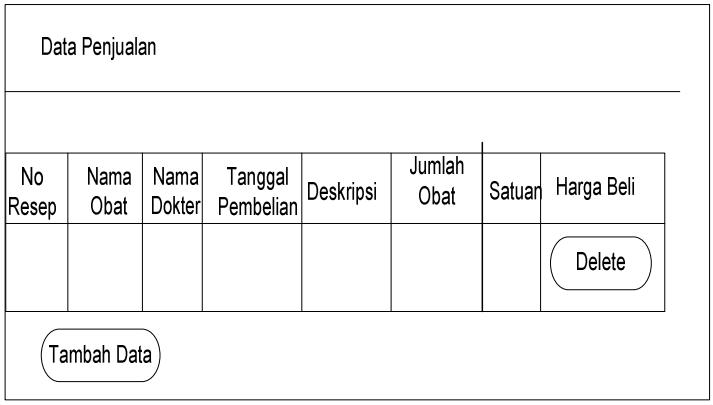
Desain tampilan keluaran ini adalah tampilan berupa tabel yang berisi data obat yang telah tersimpan di database yang telah di inputkan admin.



Gambar 4.30 Desain Tampilan Output Data Obat

d. Desain Tampilan Output Data Penjualan

Desain tampilan keluaran ini adalah tampilan berupa tabel yang berisi data konfirmasi penjualan yang telah tersimpan di database yang telah di inputkan oleh bagian penjualan.



Gambar 4.31 Tampilan *Output* Data Penjualan

e. Desain Tampilan Output Data Permintaan

Desain tampilan keluaran ini adalah tampilan berupa tabel yang berisi data konfirmasi permintaan yang telah tersimpan di database yang telah di inputkan Bagian Gudang.

Data Obat Minimum

Tanggal Masuk	No Faktur	Nama Suplier	Nama Obat	Tanggal Pembuatan	Tanggal Kadaluarsa	Jumlah Obat	Satuan	Harga Satuan	Action
									Pembelian

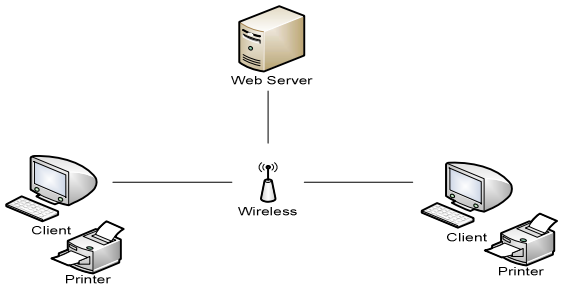
Tambah Data

Gambar 4.32 Desain Tampilan *Output* Data Permintaan

4.2.6 Perancangan Arsitektur Jaringan

Perancangan arsitektur jaringan adalah bentuk umum dari suatu rancangan program unutk memudahkan pemakai dalam menjalankan program komputer. Pada perancangan arsitektur ini disesuaikan dengan fungsinya bagi pengguna agar tidak mengalami kesulitan dalam memilih menu-menu yang diinginkan yaitu perancangan menu *admin* dan perancangan menu *user*.

Berikut adalah gambar perancangan arsitektur yang penulis rancang :



Gambar 4.32 Arsitektur Jaringan yang diusulkan