

# KONSTRUKSI DAN PENGUJIAN PERANGKAT LUNAK

# ANALISA SISTEM APOTEK BERBASIS WEB

Oleh:

RAMA RAHMANDA (5214100119)

NAUFAL RAIHAN NOLY (5214100062)

RIZANDI CHOIRUL ANAM (5214100163)

Dosen Pengampu

##### ***Hatma Suryotrisongko S.Kom, M.Eng***

Jurusan Sistem Informasi

Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember

Surabaya 2016

**Daftar Isi**

Latar Belakang...................................................................................................................2

Daftar Pustaka....................................................................................................................2

Activity Diagram................................................................................................................3

Sequence Diagram..............................................................................................................3

Use Case Diagram.............................................................................................................27

Use Case Description........................................................................................................32

State Chart Diagram..........................................................................................................40

Class Diagram...................................................................................................................49

Robustness........................................................................................................................50

Desain GUI.......................................................................................................................65

**Latar Belakang**

Apotek merupakan tempat yang menyediakan barang berupa obat-obatan yang dibutuhkan oleh pelanggan. Pada saat ini terdapat apotek-apotek banyak jenisnya, terdapat apotek modern dan konvensional. Apotek modern sudah banyak dilengkapi teknologi didalamnya mulai dari pembayaran, pencatatan keuangan hingga ke pembuatan resep obat. Disini kami akan membuat analisis desain untuk apotek modern yang berhubungan langsung dengan IT. Supaya dapat membantu proses bisnis juga proses pembuatan software untuk apotek ini.

**Daftar Pustaka**

UML FOR THE IT BUSINESS ANALYST, 2nd EDITION

**REQUIREMENT REVIEW**

# Functional Requirement

1. Sistem dapat menampilkan Jumlah Obat berdasarkan Stock obat
2. Sistem dapat menghitung jumlah uang yang harus dibayarkan untuk membeli obat
3. Sistem dapat menampilkan User yang terdaftar dalam halaman admin
4. Sistem dapat mengupdate data stock baru
5. Sistem dapat mengurangi dan menambahkan user
6. Sistem dapat memberikan laporan kepada pemilik
7. Sistem dapat memberikan laporan penambahan atau pengurangan stock obat

# Non-Functional Requirements

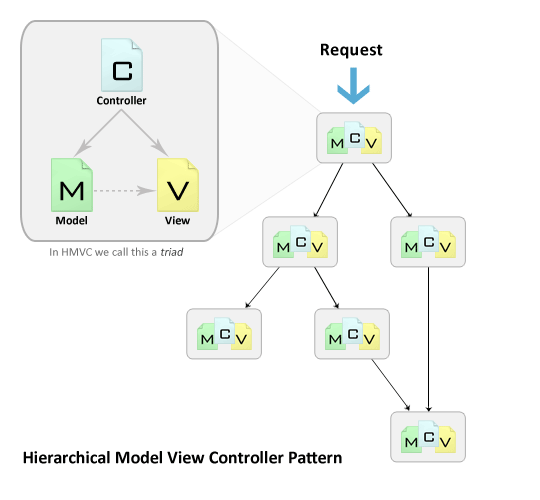
1. Performance
   1. Sistem dapat di buka kurang dari 2 menit
   2. Sistem menampilkan jumlah stock Obat dan nama Obat dengan tepat
   3. Laporan penambahan dan pengurangan obat dibuat laporan dengan tepat
2. Security
   1. Data dalam System tersimpan secara aman dan tidak dapat digunakan tanpa authorization atau oleh pihak yang tidak berwenang.

# Maintainability

1. Kode dalam sistem menggunakan desain pattern
2. Kode dalam sistem memuat penjelasan singkat dalam bentuk comment

# DESIGN PATTERN

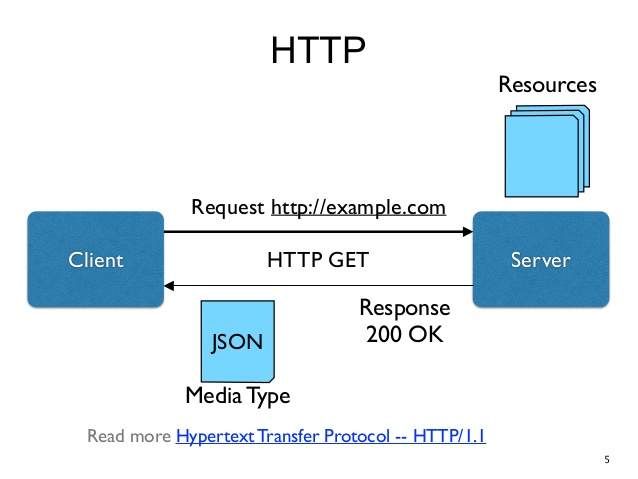
Desain pattern yang digunakan adalah HMVC (Hierarchical model view controller) yang merupakan pengembangan dari MVC, dengan dilakukannya pengelompokan MVC berdasarkan fungsi yang akan digunakan atau dalam bentuk hierarki (parent – child )



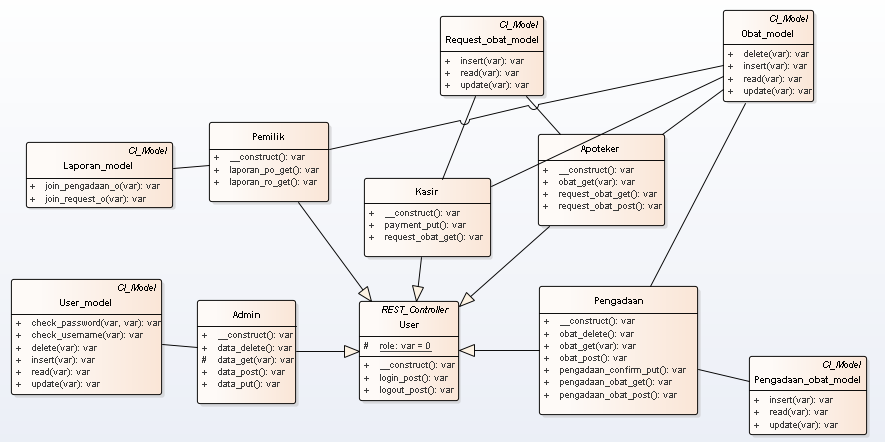
# ARCHITECTURE

Sistem menggunakan 2 layer arsitektur dimana ada bagian aplikasi dan presentasi. Hal tersebut menyebabkan kami menggunakan REST(representational state transfer) dengan memisahkan client dan server. Kedua node tersebut akan saling berhubungan melalui http request melalui url dengan http method get,post,update,delete. Client akan mengakses url dari server dan server akan mengembalikan data dalam bentuk json yang nantinya data tersebut akan diolah lagi oleh klien untuk ditampilkan dalam layer presentasi.

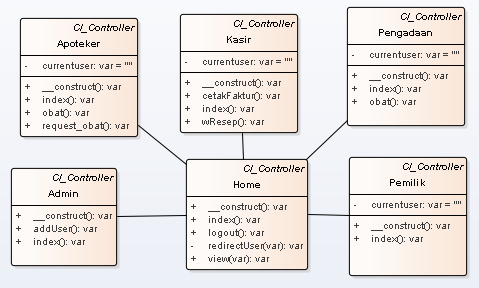
Ilustrasi gambar :



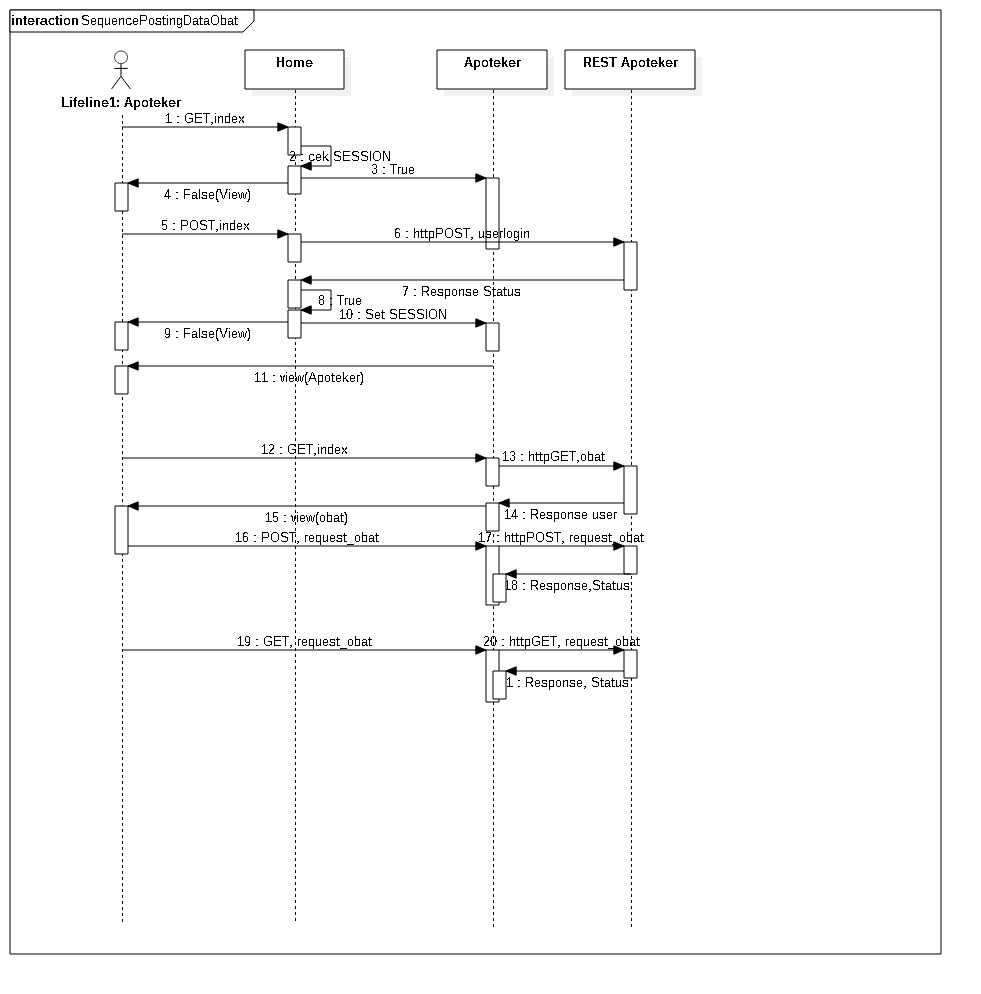
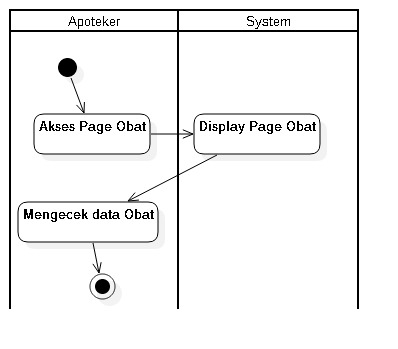
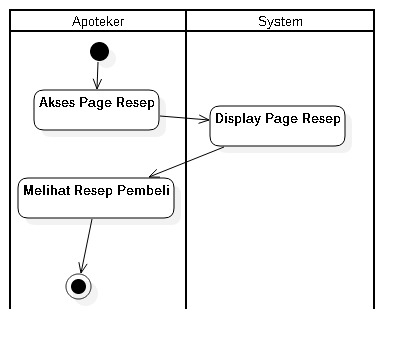
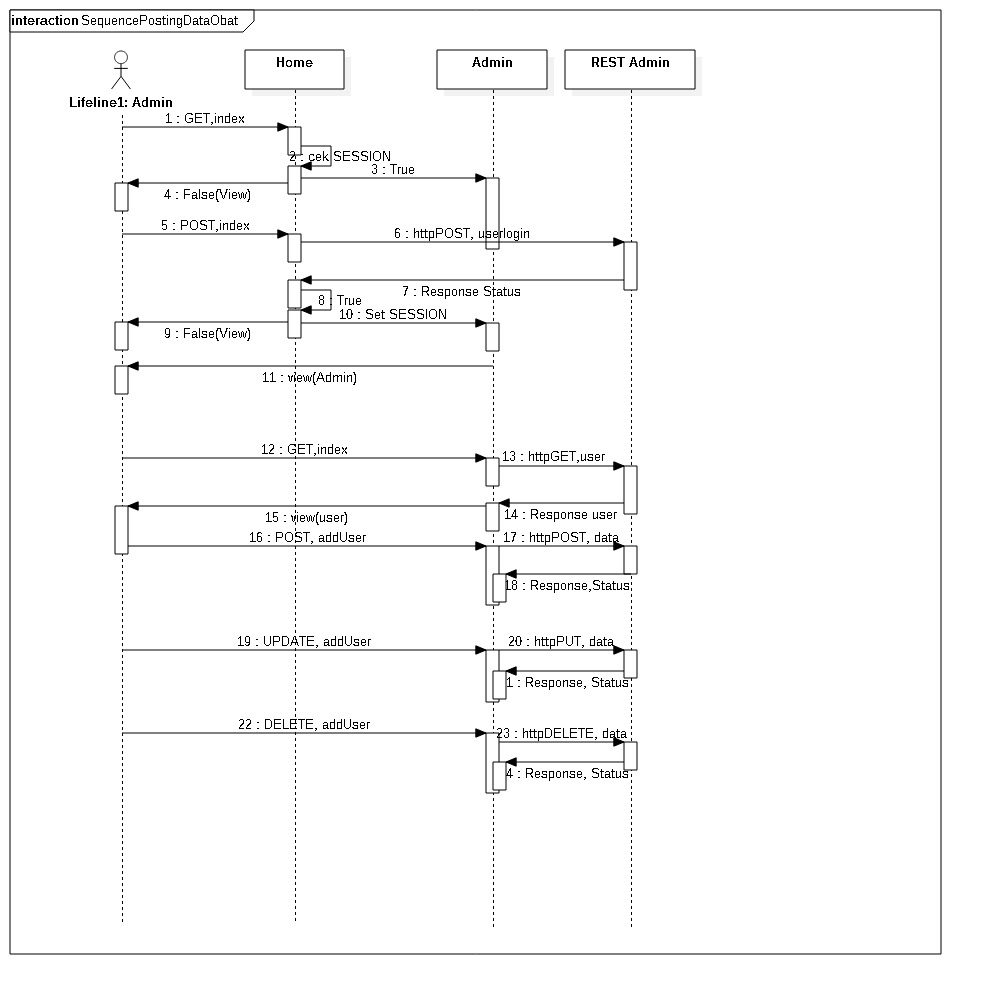
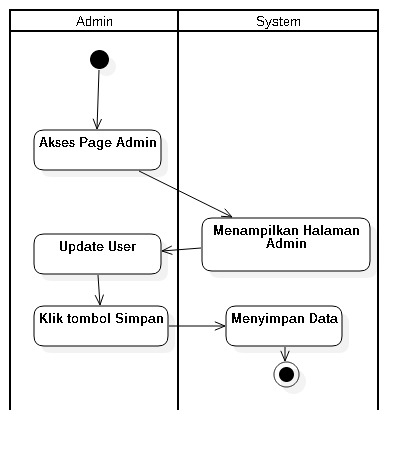
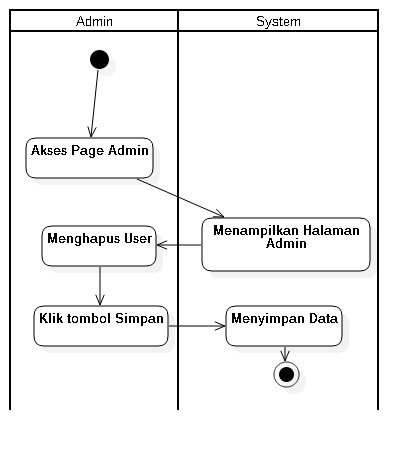
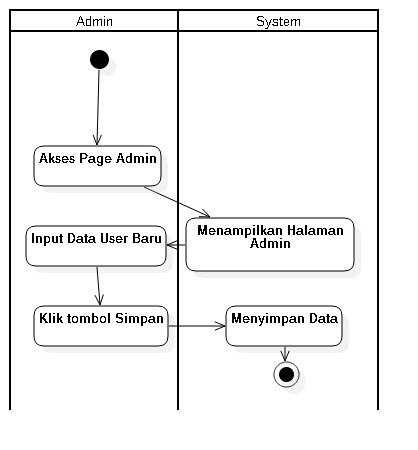
# Class Diagram Server

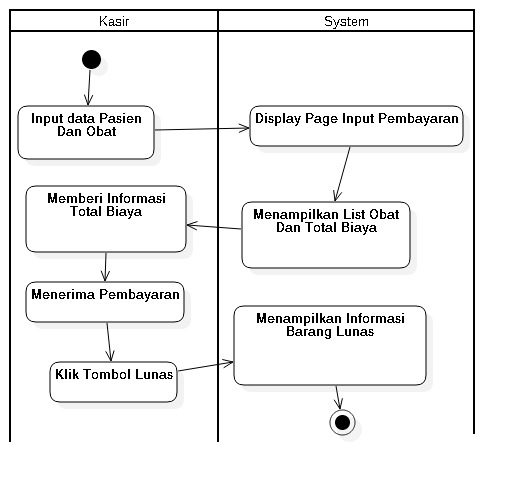


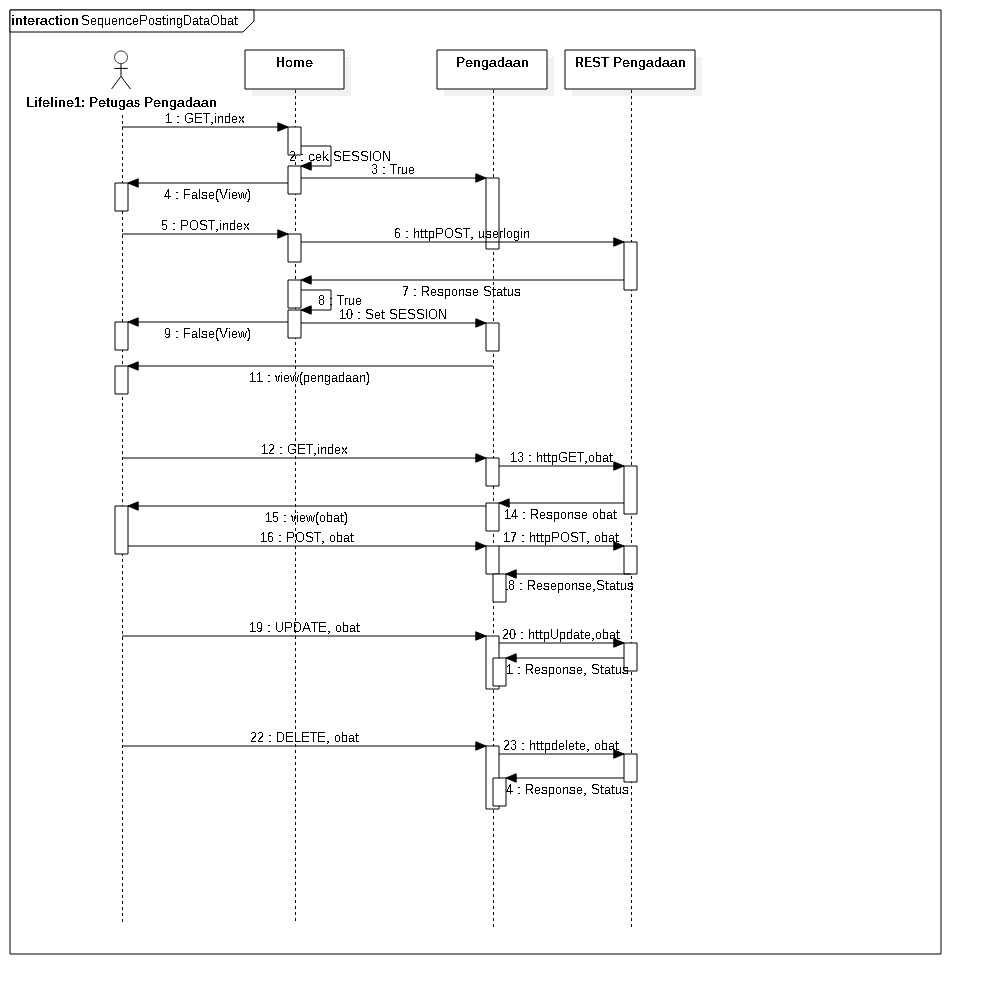
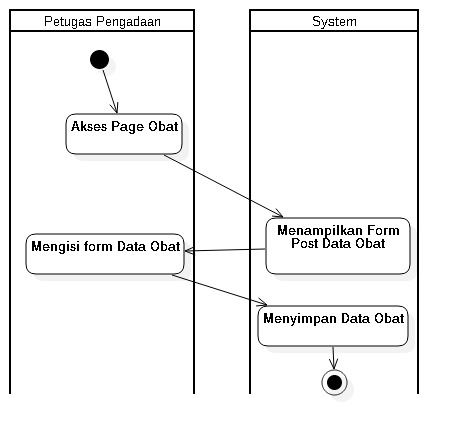
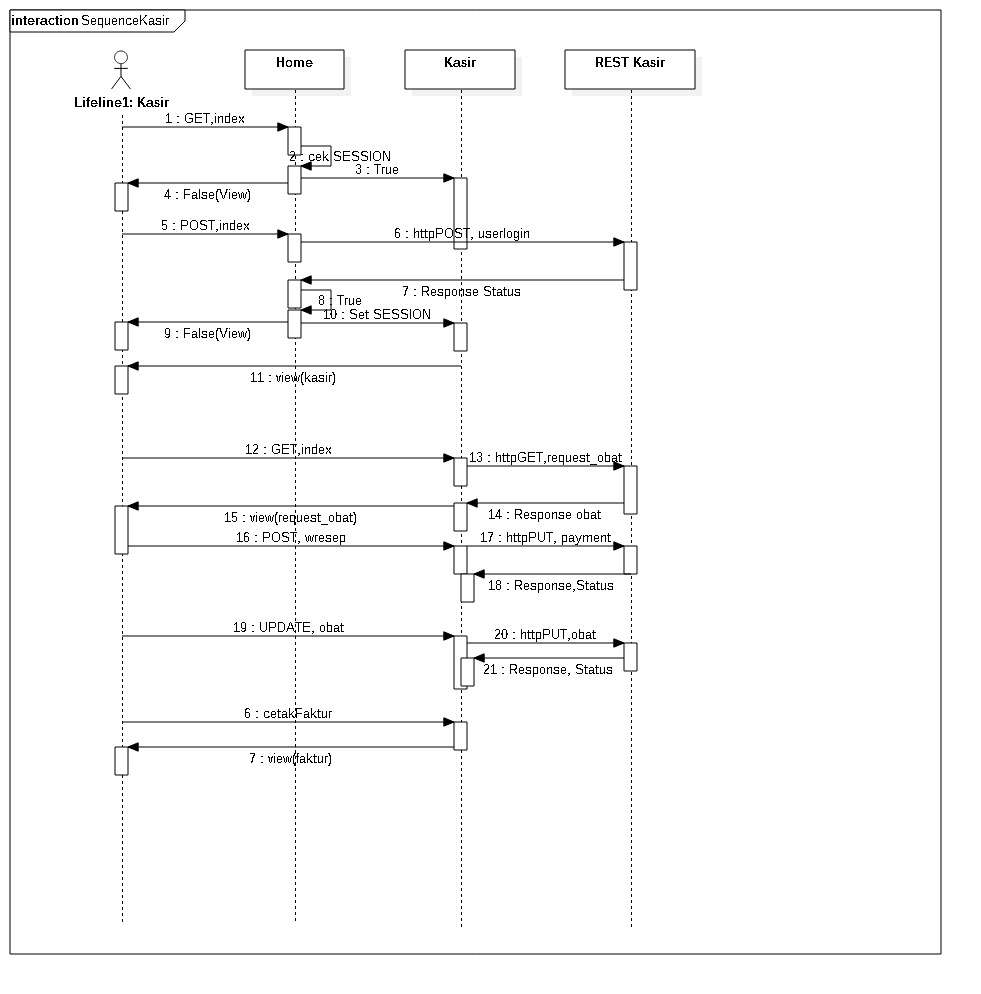
# Class Diagram Client

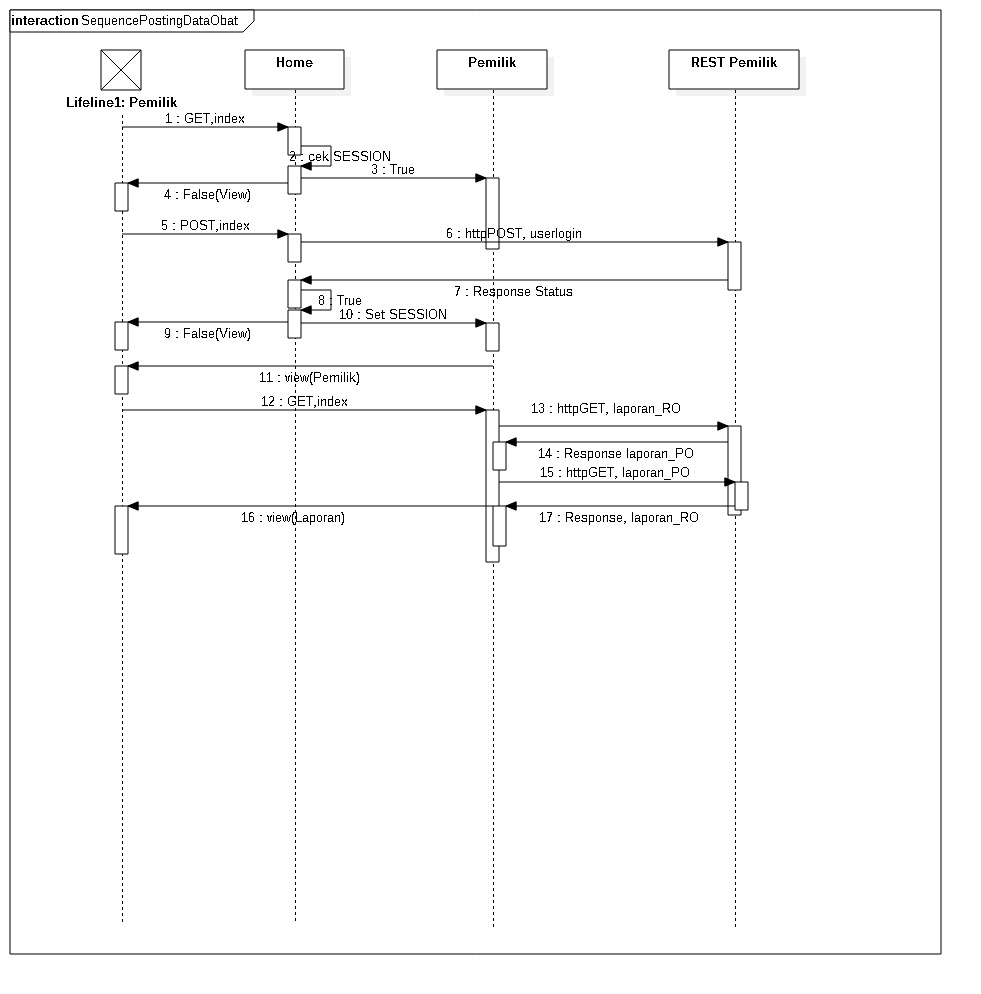


**Activity and Sequence Diagram**



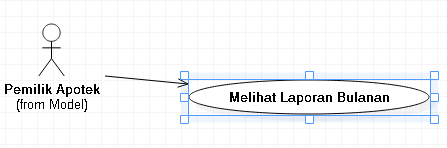




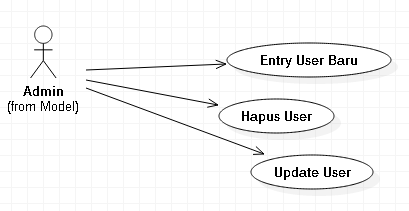


**Use Case Diagram**

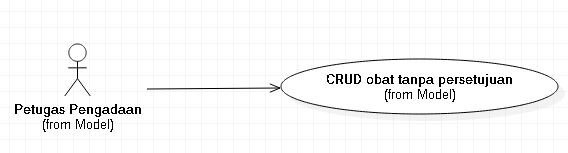
**Usecase Pemilik Apotek**

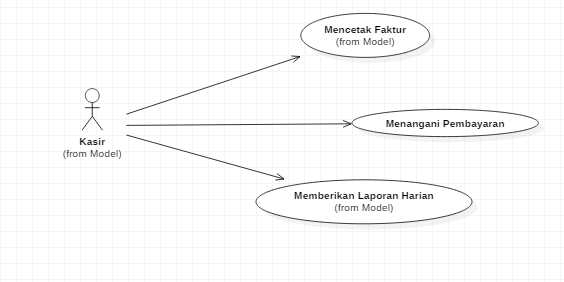


**Usecase Admin**

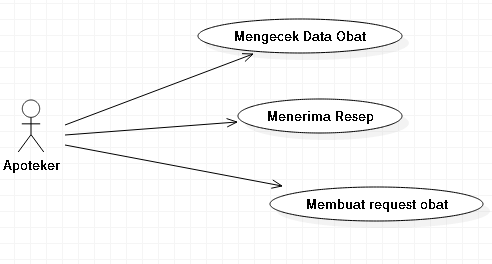


**Usecase Petugas Pengadaan**



**Usecase Kasir**

**Usecase Apoteker**

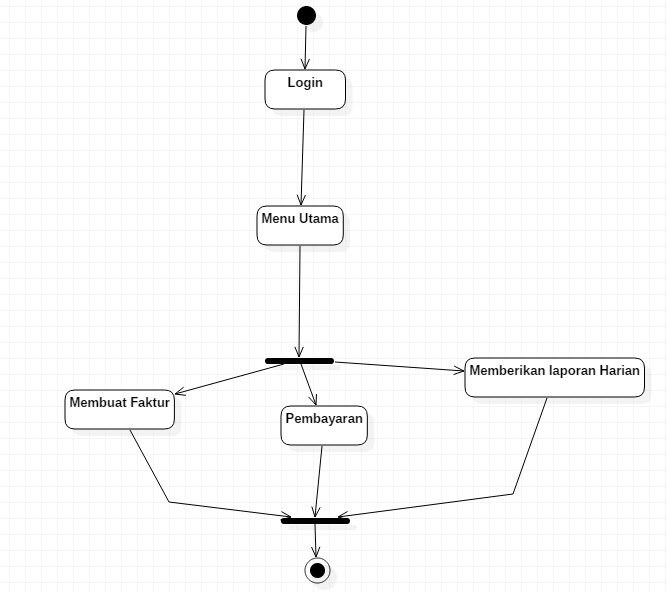


**Use Case Description**

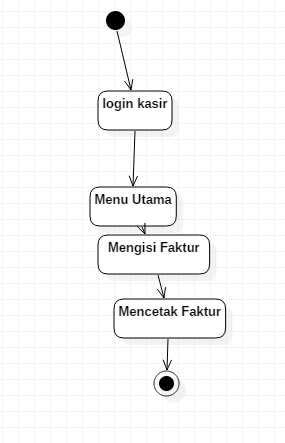
| No | Use Case | Deskripsi |
| --- | --- | --- |
| **1** | Login | **Deskripsi singkat** : user apotek memasuki aplikasi agar dapat menggunakan fungsi - fungsi aplikasi  **Primary Actor** : User Apotek  **Pre-Condition** : Sistem menyediakan laman login yang dapat diisi  **Post-Condition** : Sistem mengarahkan ke laman administrasi setelah login terotorisasi  **Alternate-Condition** : Sistem memberitahu bahwa login gagal karena informasi yang di input salah  **Primary Flow** :   * User Memilih fitur login * System Menampilkan layar login * User Mengisi username dan password * System Memeriksa username dan password * System Menampilkan laman user   **Alternate Flow** :   * User Memilih fitur login * System Menampilkan layar login * User Mengisi username dan password * System Memeriksa username dan password * User Mengisi username password * (Diulangi terus sampai login benar) |
|  |  |
|  |  |
|  |  |
| **2** | Melihat Laporan Bulanan | **Deskripsi singkat** : Pemilik Apotek melihat laporan bulanan hasil rekapan kasir per bulan  **Primary Actor** : Pemilik Apotek  **Pre-Condition** : Pemilik Apotek sudah login  **Post-Condition** : Laporan telah di cek dan keluar dari laman laporan  **Primary Flow** : - User Akses Page Laporan Bulanan   * System Diplay Laporan Bulanan * User mengecek laporan   **Alternate Flow** : - |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **4** | Update User | **Deskripsi Singkat** : Admin update status data user baru yang ada pada database  **Primary Actor** : Admin  **Pre-Condition** :Admin sudah Login  **Post-Condition** : Data User pada database terupdate  **Primary Flow** :   * User Akses Page Admin * System Menampilkan Halaman Admin * System Update user * User Klik tombol Simpan * System Menyimpan Data |
|  |  |
|  |  |
|  |  |
|  |  |
| 5 | Entry User Baru | **Deskripsi Singkat** : Admin memasukan data user baru yang ada pada database  **Primary Actor** : Admin  **Pre-Condition** :Admin sudah Login  **Post-Condition** : Data User pada database terupdate  **Primary Flow** :   * User Akses Page Admin * System Menampilkan Halaman Admin * User Input Data User Baru * User Klik tombol Simpan * System Menyimpan Data |
|  |  |
|  |  |
|  |  |
| 6 | Hapus User | **Deskripsi Singkat** : Admin menghapus data User yang ada pada database  **Primary Actor** : Admin  **Pre-Condition** :Admin sudah Login  **Post-Condition** : Data User pada database terupdate  **Primary Flow** :   * User Akses Page Admin * System Menampilkan Halaman Admin * User Menghapus User * User Klik tombol Simpan * System Menyimpan Data |
|  |  |
|  |  |
|  |  |
|  |  |
| 7 | Update Obat | **Deskripsi Singkat** : Apoteker update data Obat yang ada pada database  **Primary Actor** : Apoteker  **Pre-Condition** :Apoteker sudah Login  **Post-Condition** : Data Obat pada database terupdate  **Primary Flow** :   * User Akses Page Apoteker * System Menampilkan Halaman Apoteker * Apoteker Update data Obat * Apoteker Klik tombol Simpan   - System Menyimpan Data |
|  |  |
|  |  |
|  |  |
|  |  |
| 8 | Entry Obat Baru | **Deskripsi Singkat** : Apoteker memasukkan data Obat baru yang ada pada database  **Primary Actor** : Apoteker  **Pre-Condition** :Apoteker sudah Login  **Post-Condition** : Terdapat Data obat baru pada Database  **Primary Flow** :   * User Akses Page Apoteker * System Menampilkan Halaman Apoteker * Apoteker Update Obat * Apoteker Klik tombol Simpan * System Menyimpan Data Obat |
|  |  |
|  |  |
|  |  |
|  |  |
| 9 | Menangani Pembayaran | **Deskripsi Singkat** : Kasir menangani pembayaran pembeli obat  **Primary Actor** : Kasir  **Pre-Condition** : Kasir sudah login  **Post-Condition** : Uang diterima  **Primary Flow** :   * User Input data Pasien Dan Obat * System Display Page Input * Menampilkan List Obat Dan Total Biaya * User Memberi Informasi Total Biaya * User Menerima Pembayaran * User Klik Tombol Lunas * System Menampilkan Informasi Barang Lunas |
|  |  |
|  |  |
|  |  |
|  |  |
| 10 | Memproses pembelian Obat dengan Resep | **Deskripsi Singkat** : Apoteker mengatasi pelanggan yang membeli obat dengan resep  **Primary Actor** : Apoteker  **Pre-Condition** Apoteker sudah Login  **Post-Condition** : Obat diberikan yang sesuai  **Alternate-Condition** : Obat tidak tersedia  **Primary Flow** :   * Apoteker Meminta Nomor Resep * Apoteker Mengecek data obat * Apoteker Mencari obat * Verifikasi Obat * Memberikan Obat   **Alternate Flow:**   * Apoteker Meminta Nomor Resep * Apoteker Mengecek data obat * Apoteker Mencari obat * Verifikasi Obat * Apoteker Memberikan obat seadanya |
|  |  |
|  |  |
|  |  |
| 11 | Memproses pembelian Obat Tanpa Resep (kemudian mendapatkan resep dari apoteker) | **Deskripsi Singkat** : Apoteker mengatasi pelanggan yang membeli obat tanpa  **Primary Actor** : Apoteker  **Pre-Condition** : Apoteker sudah Login  **Post-Condition** : Obat diberikan yang sesuai  **Alternate-Condition** : Obat tidak tersedia  **Primary Flow** :   * Apoteker Memberikan No antrian * Apoteker Input Keluhan * Apoteker Memberikan Informasi Pada Apoteker * Apoteker Mengecek data obat * Apoteker Mencari obat * Verifikasi Obat * Apoteker Memberikan Obat   **Alternate Flow:**   * Apoteker Mengecek data obat * Apoteker Mencari obat * Verifikasi Obat * Apoteker Memberikan Obat * Apoteker Memberikan Obat yang setipe |
|  |  |
|  |  |
|  |  |
|  |  |
| 12 | Mengecek Data Obat | **Deskripsi Singkat** : Apoteker Mengecek data obat untuk pemberian obat bagi Petugas penjualan yang akan diberikan ke pembeli obat  **Primary Actor** : Apoteker  **Pre-Condition** :Apoteker sudah Login  **Post-Condition** : Apoteker berhasil mengecek data obat yang tersedia  **Primary Flow** :   * Apoteker Akses Page Obat * System Display Page Obat * Apoteker Mengecek data Obat |
|  |  |
|  |  |
|  |  |
|  |  |
| 16 | Menerima Resep | **Deskripsi Singkat** : Apoteker menerima resep dokter dari Petugas Penjualan  **Primary Actor** : Apoteker  **Pre-Condition** :Apoteker sudah Login  **Post-Condition** : Apoteker menerima resep dokter  **Primary Flow** :   * Apoteker Akses Page Resep * System Display Page Resep * Apoteker Melihat Resep Pembeli |
|  |  |
|  |  |
|  |  |
|  |  |
| 17 | Menerima Informasi Keluhan | **Deskripsi Singkat** : Apoteker menerima informasi keluhan dari form keluhan yang diberikan pembeli obat melalui Petugas Penjualan  **Primary Actor** : Apoteker  **Pre-Condition** :Apoteker sudah Login  **Post-Condition** : Apoteker menerima informasi keluhan   * Apoteker Akses page Keluhan * System Display page Keluhan * Apoteker Menerima Keluhan Pembeli |

# State Chart Diagram

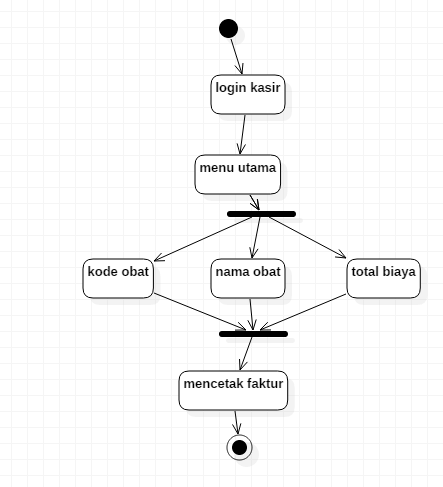
## Kasir



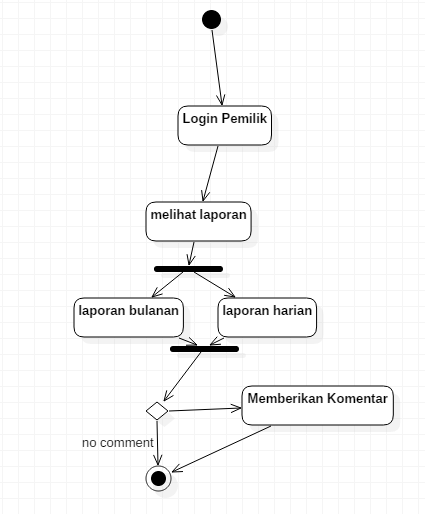
## Kasir Membuat Faktur



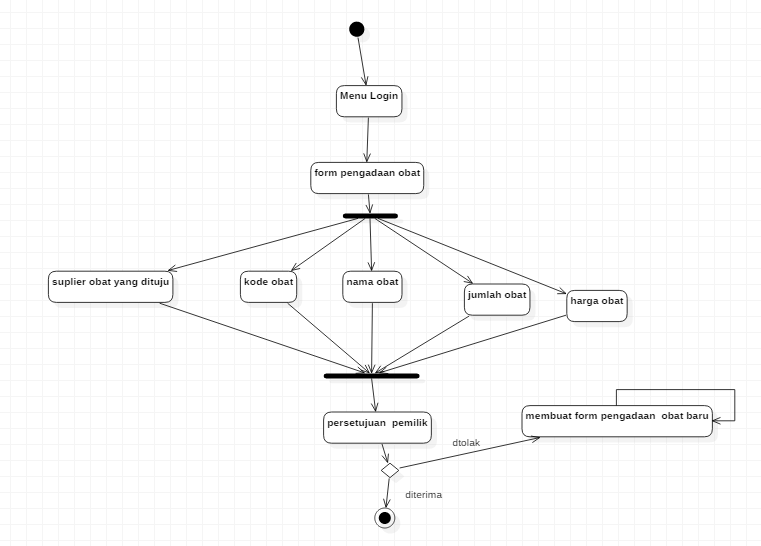
## Kasir pembayaran



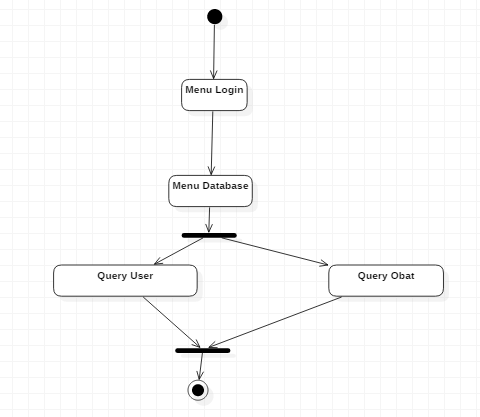
## Pemilik Apotek



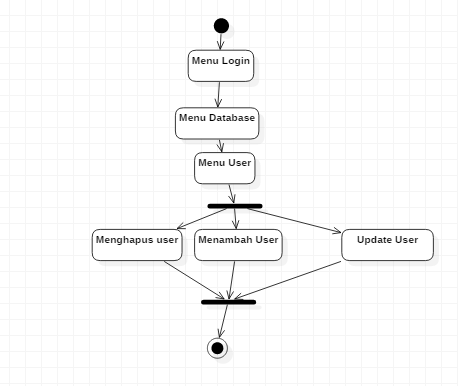
## Pengadaan Obat



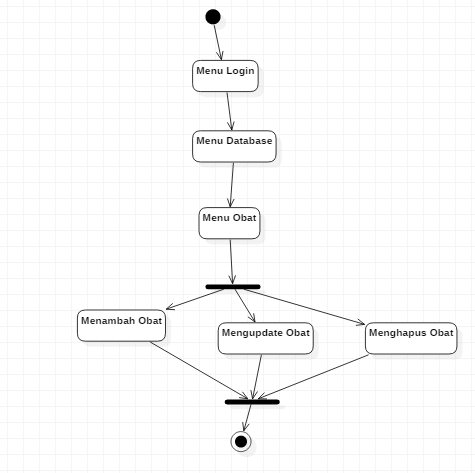
## Admin Database



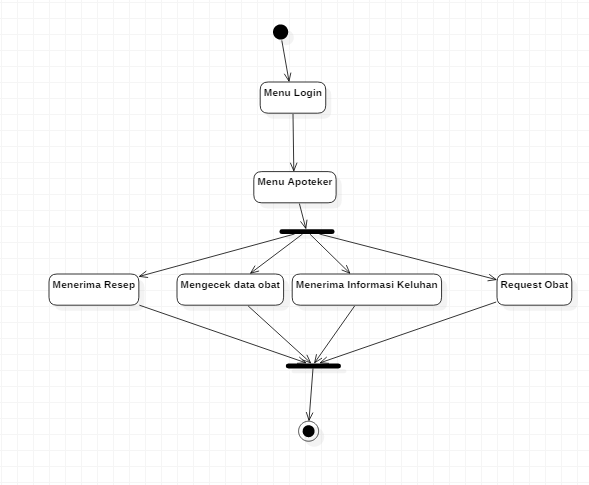
## Admin User



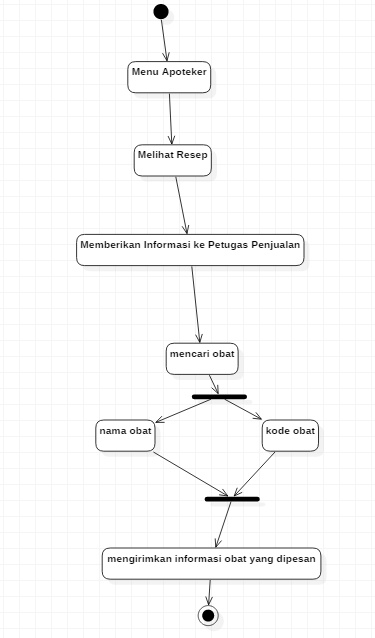
## Admin Apoteker Obat



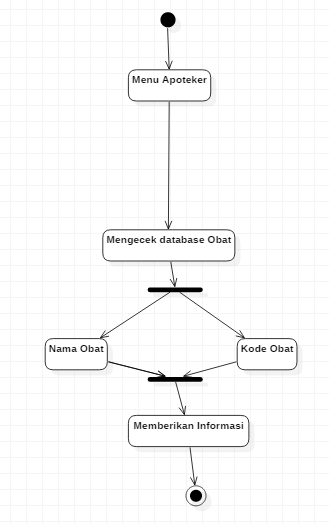
## Apoteker



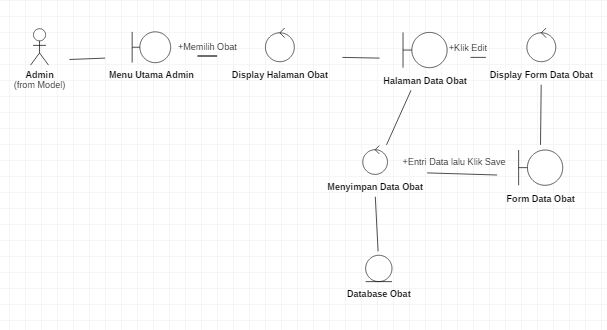
## Apoteker Resep

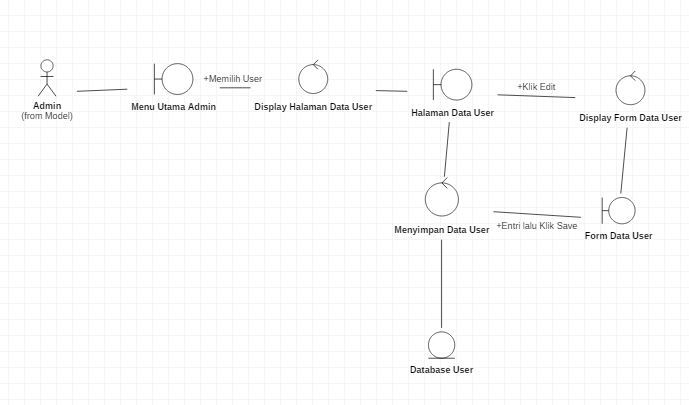


## Apoteker Mengecek Data Obat

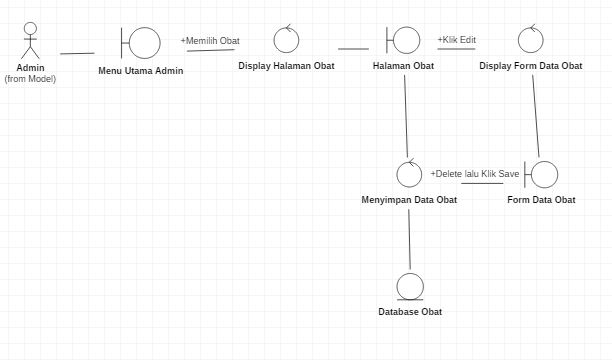


# Robustness

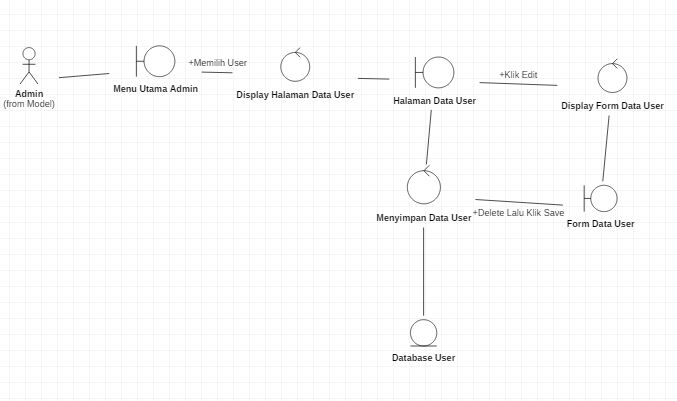
****Robustness (Admin Atau Apoteker - Entri Data Obat)



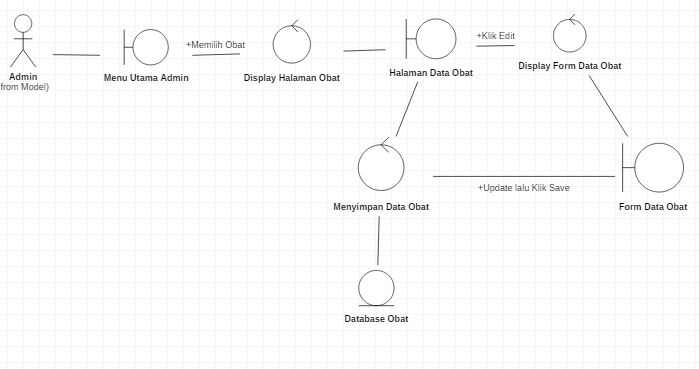
## Robustness (Admin atau Apoteker - Menghapus Data Obat)



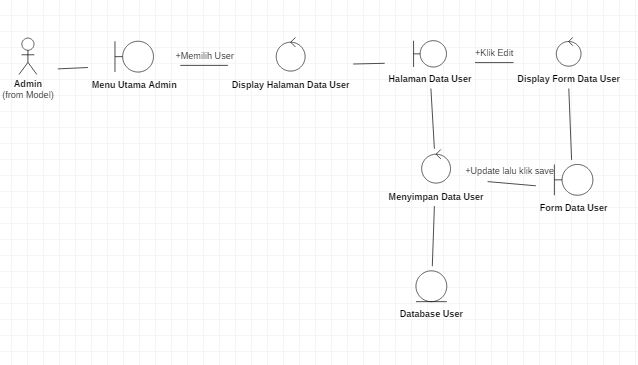
## Robustness (Admin - Menghapus User)



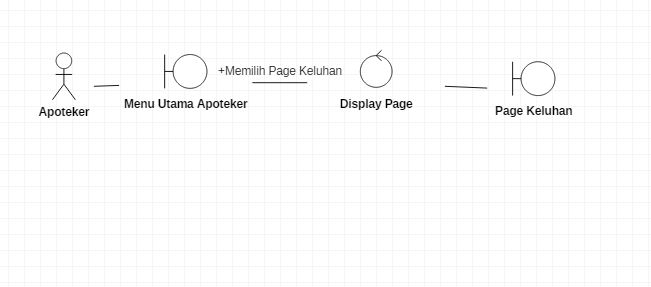
## Robustness (Admin atau Apoteker - Update Data Obat)



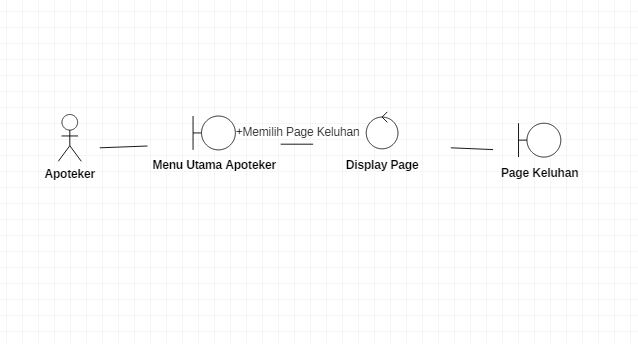
## Robustness (Admin - Update User Baru)



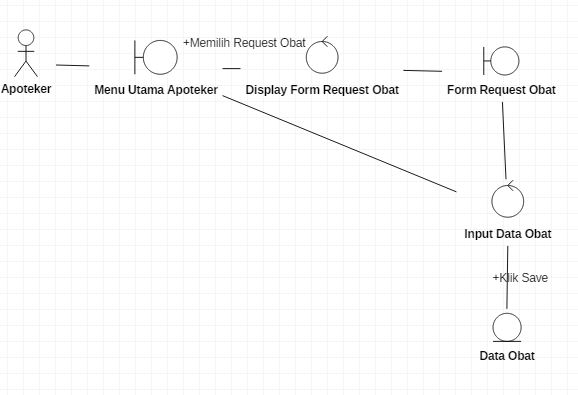
## Robustness (Apoteker - Menerima Informasi Keluhan)

****Robustness (Apoteker - Menerima Resep)

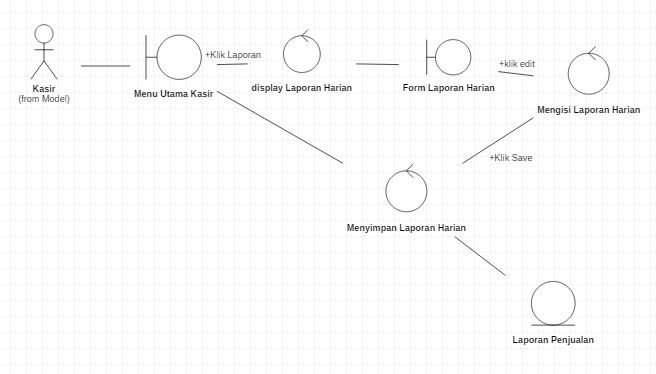
## Robustness (Apoteker - Mengecek Data Obat)



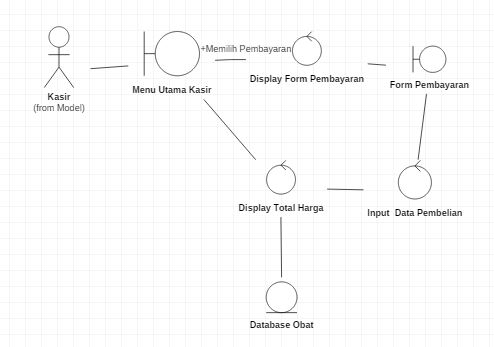
## Robustness (Apoteker - Request Obat)



## Robustness (Kasir - Memberikan Laporan Harian)

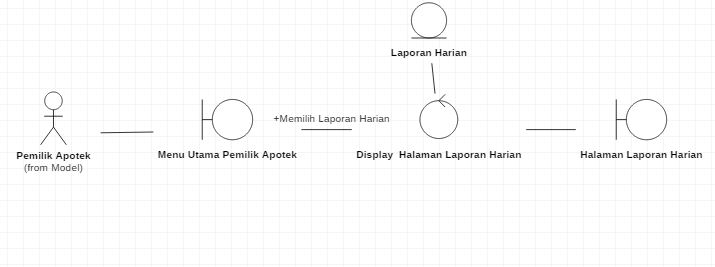


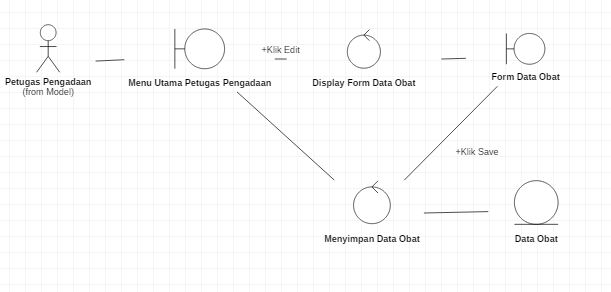
## Robustness (Kasir – Menangani Pembayaran)

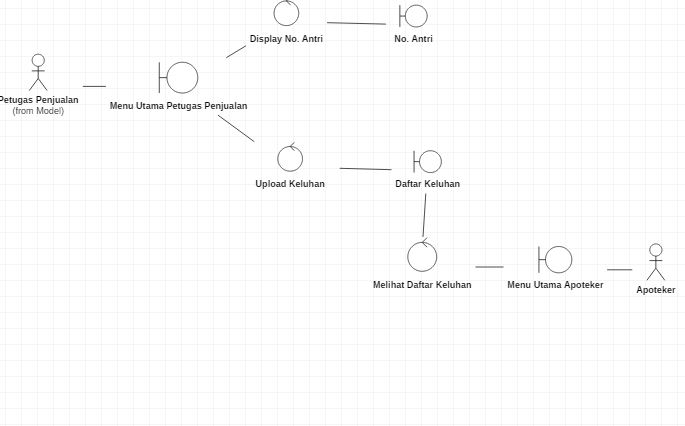


## Robustness (Pemilik Apotik – Melihat Laporan Bulanan)

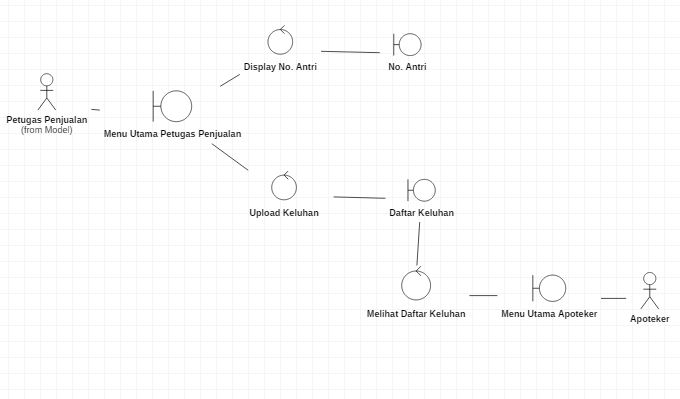
## Robustness (Pemilik Apotik – Melihat Laporan Harian)



****Robustness (Petugas Pengadaan – Posting Data Obat)

Robustness (Apoteker – Membeli dengan Resep)

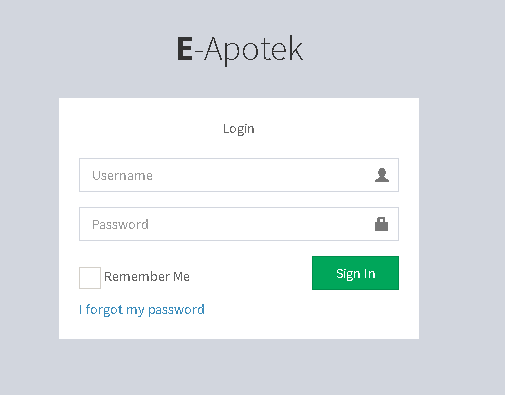
Apoteker

Robustness (Petugas Penjualan – Membeli tanpa Resep)

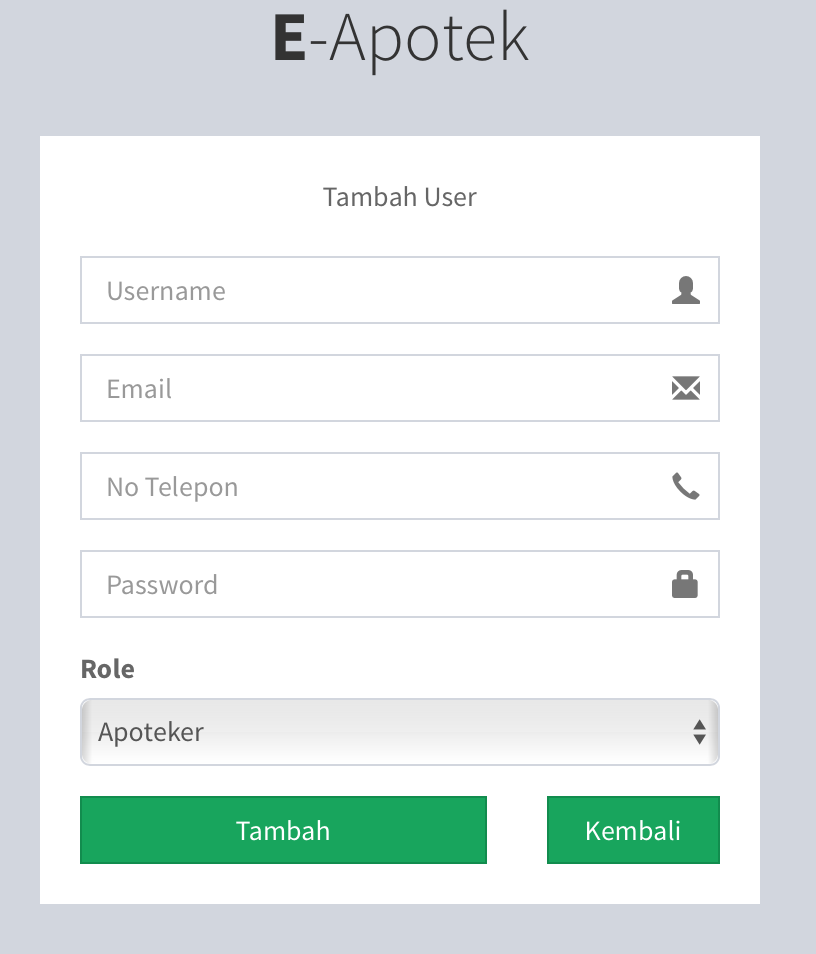
Apoteker

**Design GUI**

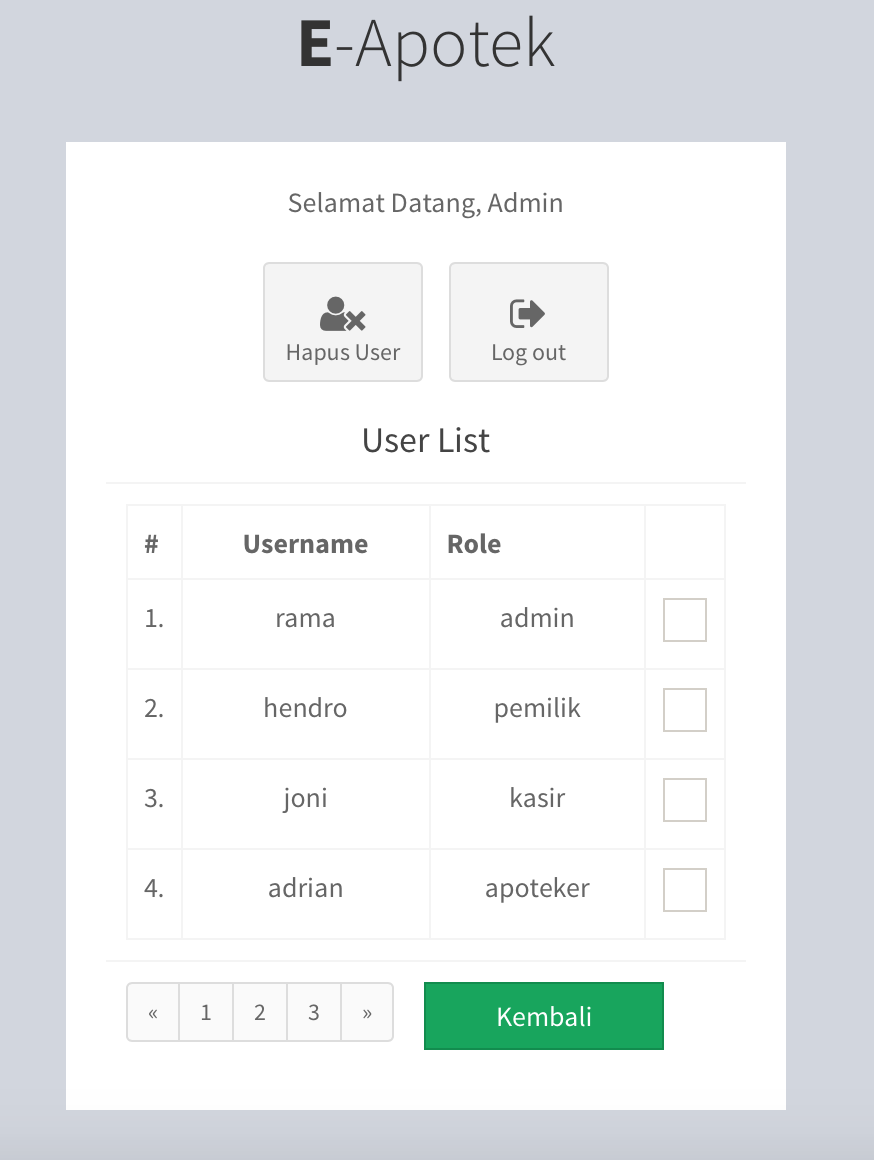
GUI Login



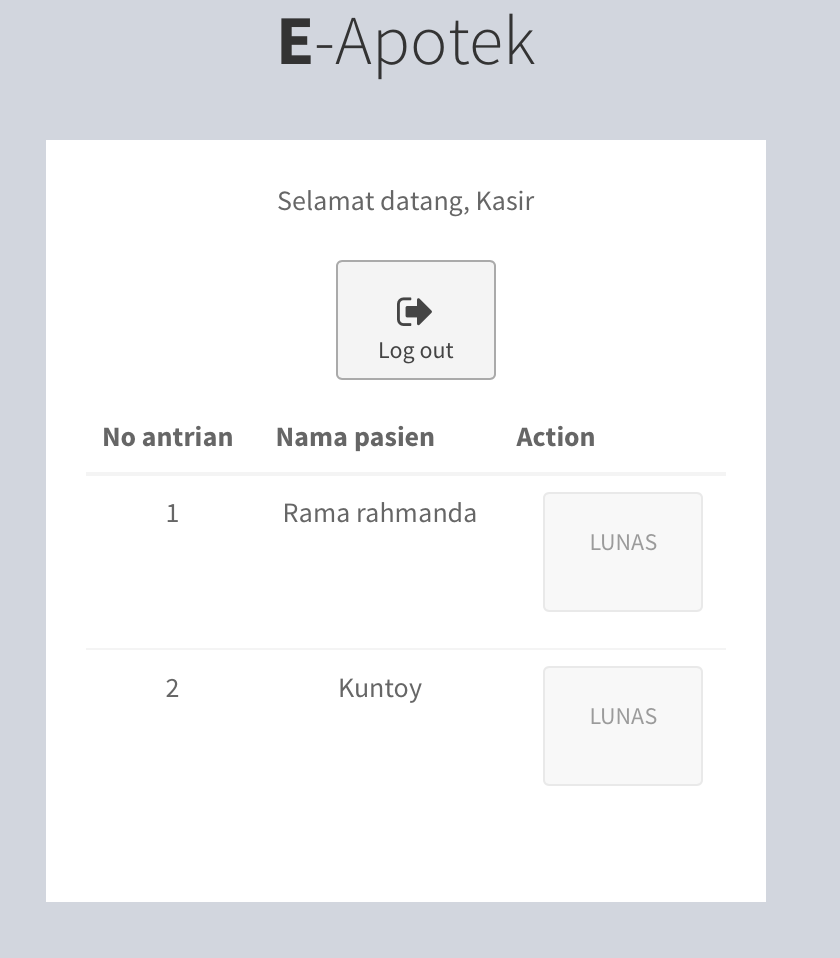
GUI Tambah User



GUI Login Admin



GUI Login Kasir



GUI Login Pengadaan Obat

