



## BAB 3 ANALISIS DAN PERANCANGAN

### 3.1 Analisis Sistem yang Berjalan

Untuk membangun sistem yang sesuai dengan permasalahan, maka perlu melakukan sebuah analisis untuk menentukan kebutuhan apa saja yang akan digunakan dalam pengembangan sistem informasi ini. Setelah dilakukan sebuah analisa alur proses pada tempat studi kasus, penulis menemukan sebuah proses manajemen obat yang berjalan, berikut adalah alur proses tersebut.

Alur proses permintaan obat petugas PSC 119 pada gudang farmasi Dinas Kesehatan Jombang :

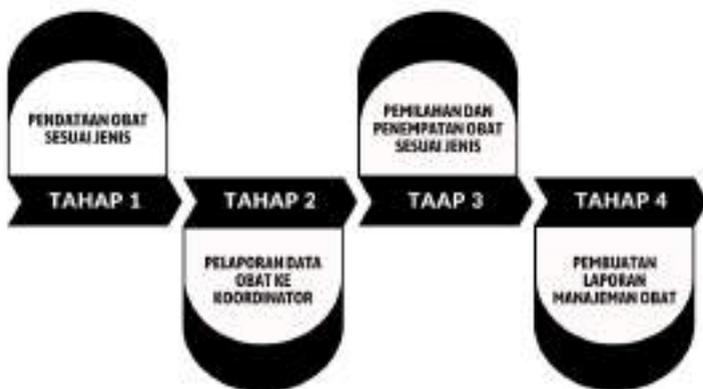


Gambar 3.1 alur permintaan obat (sumber : psc 119 jombang)

- Perawat PSC mengecek stok di ruang penyimpanan obat dan juga di dalam ambulan.
- Perawat PSC mendata obat apa saja yang sudah habis dan memilah obat yang sudah kadaluarsa (ED).

- c. Perawat PSC melaporkan hasil pengecekan ke Koordinator PSC.
- d. Koordinator PSC membuat surat permintaan obat ke gudang farmasi Dinas Kesehatan Kabupaten Jombang dengan pemberitahuan ke Kepala Sub Divisi pelayanan kesehatan (Yankes) Rujukan.
- e. Kepala Sub Divisi pelayanan kesehatan (Yankes) Rujukan menyetujui dan Koordinator PSC menindaklanjuti surat permintaan obat.
- f. Surat permintaan obat diserahkan ke gudang farmasi Dinas Kesehatan Jombang.
- g. Petugas gudang farmasi memverifikasi surat permintaan.
- h. Petugas gudang memberikan stok obat sesuai surat permintaan.

Alur proses perawat PSC 119 menerima stok obat baru :



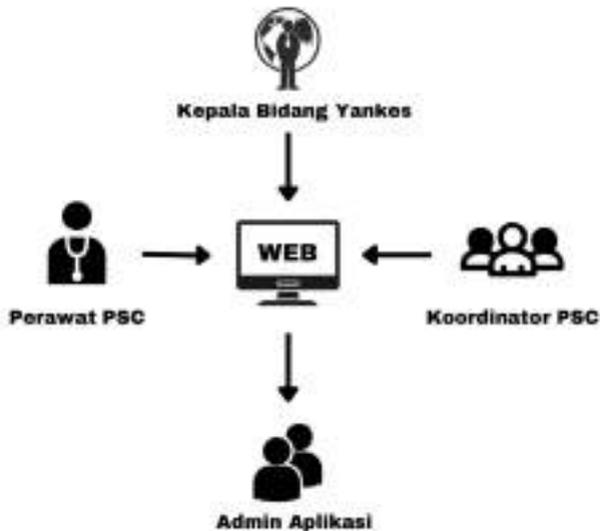
Gambar 3.2 penerimaan stok (sumber : psc 119 jombang)

- a. Perawat PSC mendata obat masuk secara manual dan diinput sesuai dengan struk yang diterima dari gudang farmasi.
- b. Perawat PSC melaporkan hasil pendataan obat pada koordinator PSC agar diketahui.

- c. Perawat PSC mengklasifikasikan obat mana yang disimpan di ruang penyimpanan dan mana yang dimasukkan ke ambulan.
- d. Perawat PSC membuat laporan obat masuk dan keluar serta merekap dalam waktu harian dan bulanan secara manual tertulis.

### 3.2 Analisis Sistem yang Diusulkan

Berdasarkan hasil analisa pada sistem yang berjalan saat ini, maka diusulkan sebuah penelitian tentang rancang bangun aplikasi manajemen obat pada PSC 119 SIBANGJO Kabupaten Jombang yang bertujuan untuk mempercepat penanganan pelayanan PSC serta menerapkan konsep digitalisasi pada alur proses bisnis. Berikut gambaran sistem yang diusulkan.



Gambar 3.3 sistem yang diusulkan (sumber : dokumen pribadi)

Dalam sistem proses yang diusulkan, semua pihak baik dari Perawat PSC, Koordinator PSC dan juga Kepala Bidang dan Kepala Divisi Yankes bisa langsung memantau laporan manajemen stok obat pada PSC 119 SIBANGJO serta dapat mengetahui rekap laporan tiap harian, bulanan bahkan tahunan dari stok obat tersebut tanpa menunggu hasil pendataan pihak lain terlebih dahulu. Data yang diinput oleh Perawat PSC saat menerima obat dari pihak farmasi akan otomatis diolah sistem aplikasi sehingga memudahkan saat pelaporan ke Koordinator.

### 3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berisi tentang proses yang dilakukan oleh suatu sistem. Adapun kebutuhan fungsionalnya adalah sebagai berikut :

Tabel 3.1 kebutuhan fungsional

No	Kebutuhan Fungsional	Proses
1	Sistem dapat mengelola data obat	<ul style="list-style-type: none"> <li>• Admin dapat tambah data, edit data, dan hapus data stok obat.</li> <li>• Admin dapat mengubah status obat menjadi aktif atau tidak.</li> </ul>
2	Sistem bisa merekap daftar pegawai PSC	<ul style="list-style-type: none"> <li>• Admin bisa menambahkan data pegawai menjadi <i>User</i>, serta mengedit dan merubah <i>Password User</i> dan menghapus status <i>User</i>.</li> </ul>
3	Sistem mampu mengklasifikasikan daftar <i>Supplier</i>	<ul style="list-style-type: none"> <li>• Admin mampu menambah daftar <i>supplier</i> obat,</li> </ul>



		mengubah dan menghapus data tersebut.
4	Sistem dapat merekam data pembelian dan pengeluaran stok obat	<ul style="list-style-type: none"> <li>• Admin dapat melihat seluruh pembelian stok obat serta pengeluarannya</li> <li>• <i>User</i> pegawai bisa juga melihat seluruh stok obat serta pengeluarannya dan mencetaknya</li> </ul>
5	Sistem mampu berikan hasil rekap data laporan tiap harian, bulanan dan tahunan	<ul style="list-style-type: none"> <li>• Admin mampu mencetak serta merekap laporan tiap harian, bulanan dan juga tahunan</li> <li>• <i>User</i> pegawai juga dapat melakukan hal yang sama seperti admin dalam hal rekapitulasi stok obat</li> </ul>

### 3.2.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional dalam pengembangan sistem informasi merujuk pada persyaratan yang tidak terkait langsung dengan fungsionalitas sistem, tetapi mempengaruhi performa, keamanan, keandalan, dan faktor lain yang berhubungan dengan pengalaman pengguna dan lingkungan operasional sistem. Beberapa kebutuhan non fungsional yang umum dalam pengembangan sistem informasi meliputi :

1. Kinerja : Sistem harus mampu memberikan respon yang cepat dan waktu pemrosesan yang efisien, serta dapat



menangani jumlah pengguna atau data yang tinggi tanpa penurunan kinerja yang signifikan.

2. **Keamanan** : Sistem harus memiliki tingkat keamanan yang memadai untuk melindungi data dan informasi sensitif dari akses yang tidak sah, peretasan, atau kebocoran.
3. **Scalability** (Skalabilitas) : Sistem harus dapat dengan mudah berkembang dan menyesuaikan diri dengan pertumbuhan pengguna, *volume* data, atau kebutuhan bisnis yang berubah tanpa mengalami gangguan signifikan.
4. **Availability** (Ketersediaan) : Sistem harus tersedia dan dapat diakses oleh pengguna sesuai dengan jadwal yang ditetapkan, dengan sedikit atau tanpa waktu henti (*downtime*) yang tidak terencana.
5. **Usability** (Kemudahan Penggunaan) : Sistem harus mudah digunakan, memiliki antarmuka pengguna yang intuitif, dan memperhatikan kebutuhan dan preferensi pengguna, sehingga pengguna dapat dengan cepat menguasai dan menggunakan sistem dengan efektif.
6. **Reliability** (Keandalan) : *Reliability* (Keandalan): Sistem harus dapat beroperasi secara konsisten dan tanpa kesalahan, dengan tingkat kesalahan atau kerusakan minimal.

### 3.2.3 Desain Sistem yang Diusulkan

Tahap ini menjelaskan mengenai Gambaran umum sistem yang diusulkan yang bertujuan untuk menghasilkan perancangan sistem informasi manajemen obat pada PSC 119 SIBANGJO.

#### A. Pemodelan *Use Case*

Pada pemodelan *use case* menjelaskan tugas dan fungsi dari setiap aktor yang ada pada sistem informasi manajemen obat. Ada 2 aktor yakni Admin dan *User* Pegawai (Koordinator PSC, Perawat PSC dan Kepala Bidang Yankes).

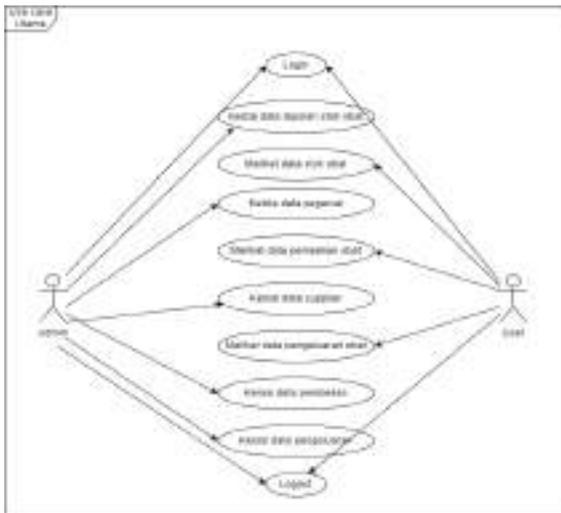


Tabel 3.2 aktor dan user

No	Aktor	Keterangan
1	Admin	Memiliki hak akses sepenuhnya atas sistem yang telah dibangun, mulai dari hak kelola stok obat, data pegawai, data <i>supplier</i> dan juga kelola data laporan pembelian serta pengeluaran obat.
2	User	Memiliki akses untuk melihat serta mencetak data stok obat dan juga data laporan pembelian serta pengeluaran obat.

## B. Use Case

*Use case diagram* berisi kebutuhan bisnis dari sistem dan juga mengilustrasikan interaksi antara sistem dengan lingkungannya. *Use case diagram* adalah ikhtisar dalam bentuk grafis dari aktor-aktor yang terlibat di dalam sistem, berbagai fungsi yang dibutuhkan oleh para aktor tersebut, serta bagaimana fungsi-fungsi tersebut saling berinteraksi.



Gambar 3.4 use case diagram

Gambar diatas merupakan *use case* utama dari sistem yang akan dibuat. Berikut *use case* dari setiap aktor dari tahap pemodelan :

1) *Use Case Admin*

Admin memiliki hak akses penuh atas sistem yang telah dibangun, mulai dari Mengelola data stok obat, data pegawai, data *supplier*, data pembelian serta mengelola data pengeluaran dan juga rekap seluruh proses. Berikut model *use case* admin.

2) *Use Case User*

User terdiri dari tiga akun. *User1* akses untuk Kepala Bidang Yankes atau Kepala Sub divisi, *User2* akses untuk koordinator PSC serta *User3* untuk perawat PSC. *Use case User* memiliki beberapa akses namun terbatas. *Use case User* memiliki akses untuk melihat laporan stok barang, melihat rekap pembelian dan juga melihat rekap pengeluaran obat. Berikut model *use case User*.

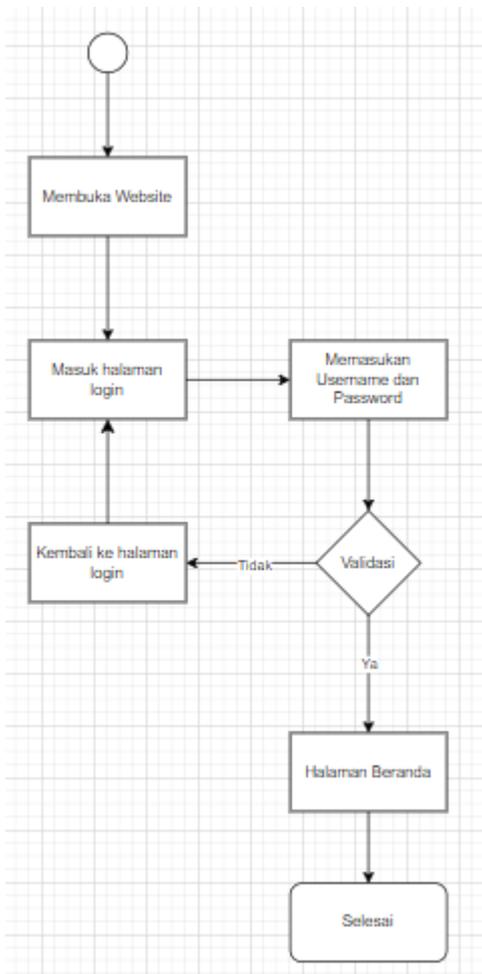
### C. *Activity/Robustness Diagram*

*Activity diagram* merupakan *workflow* secara grafis untuk mengilustrasikan alur bisnis atau *workflow* operasional dari komponen yang terdapat dalam sistem. *Activity* diagram bersifat independen dari *class*, alur aktivitas dalam *use case*, atau desain detail dari *method*.

1) *Activity Diagram Login*

Diagram *login* menunjukkan bahwa ketika *User* melakukan *login* adalah dengan mengisi *Username* dan *Password* kemudian sistem akan melakukan validasi apabila benar akan lanjut ke halaman beranda serta apabila salah *User* akan diminta untuk memasukkan *Username* dan *Password* kembali.



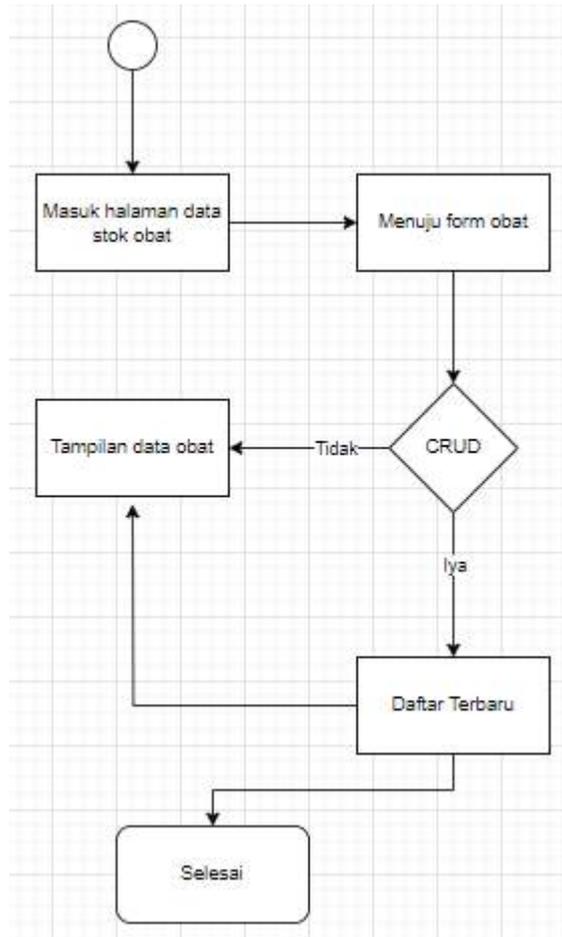


Gambar 3.5 activity diagram login

## 2) Activity Diagram Kelola Data Stok Obat

Diagram kelola stok obat menggambarkan bahwa sistem akan memproses ketika *User* atau admin ingin melihat data maka akan ditampilkan rekap data obat, tetapi jika *User* atau admin ingin menghapus, mengedit bahkan menambahkan data obat, maka sistem akan menampilkan

form edit, dan untuk proses akhirnya data yang ada di *database* secara otomatis juga diperbaharui.

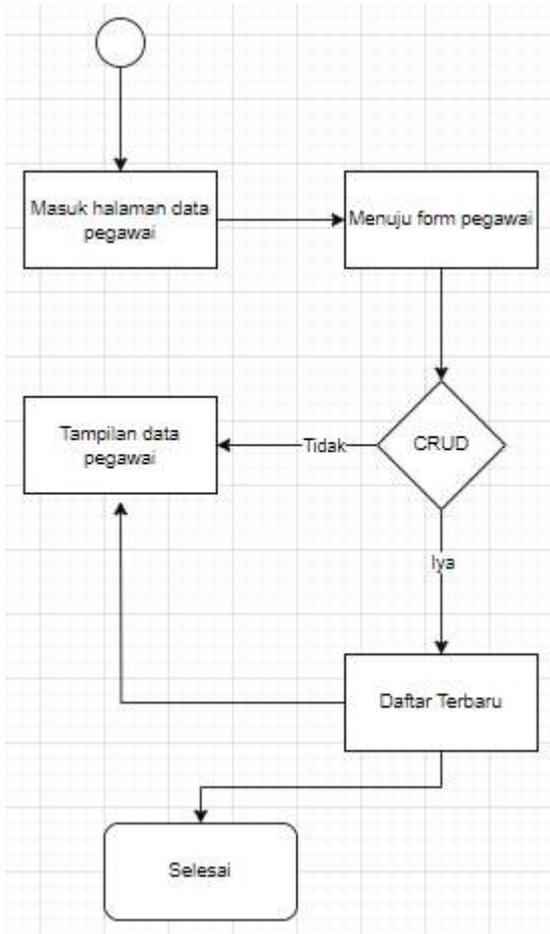


Gambar 3.6 activity diagram kelola stok obat

### 3) Activity Diagram Kelola Data Pegawai

Pada diagram kelola data pegawai, admin bisa menambah, mengedit bahkan bisa menghapus data *User*. Secara otomatis sistem akan menampilkan daftar nama

pegawai yang mempunyai akun pada sistem tersebut. Setelah sistem sudah melakukan verifikasi maka *User* bisa *login* dengan akun yang sudah ada.

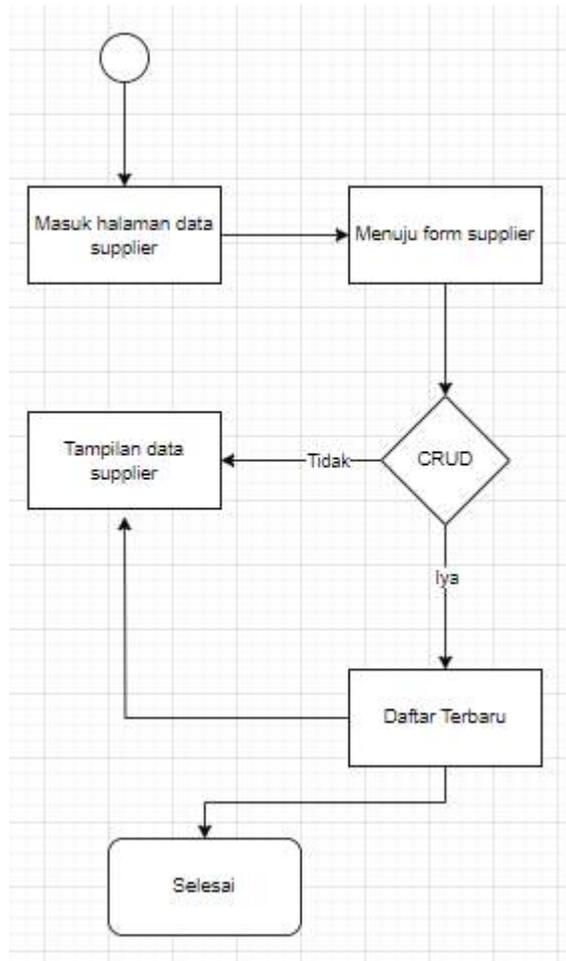


Gambar 3.7 activity diagram kelola data pegawai

#### 4) Activity Diagram Kelola Data Supplier

Diagram kelola data *supplier* menunjukkan bahwa hanya admin saja yang mampu mengedit serta menambahkan

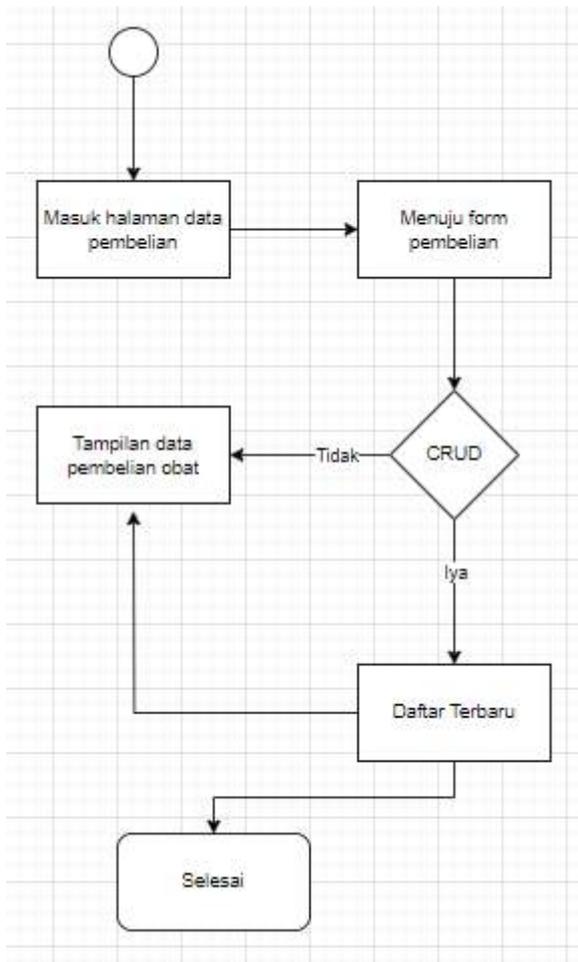
*supplier* kedalam proses dari manajemen obat. Ketika admin mengedit sebuah data *supplier*, maka sistem akan menampilkan *form* edit serta data akan otomatis masuk ke *database*.



Gambar 3.8 activity diagram kelola data supplier

##### 5) Activity Diagram Kelola Data Pembelian Obat

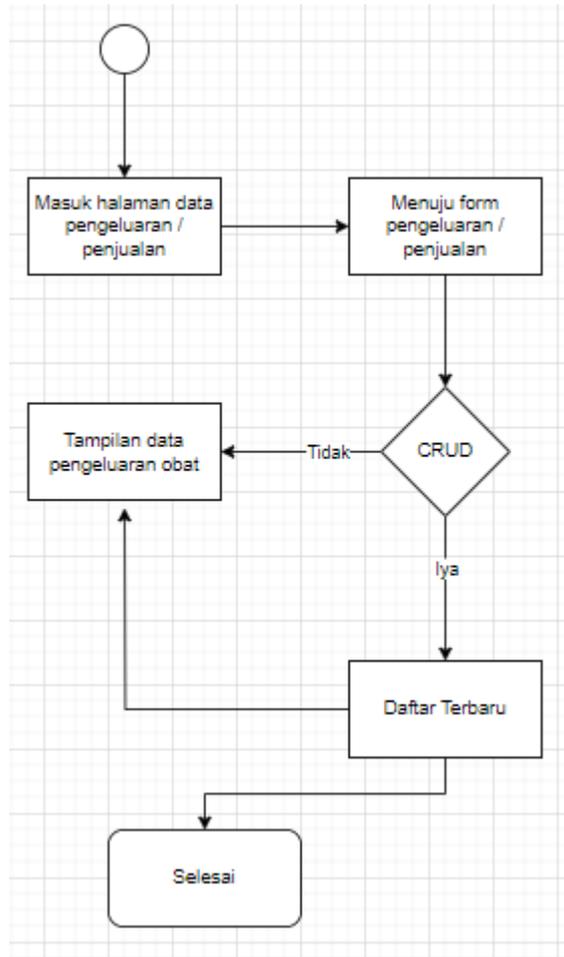
Pada diagram kelola data pembelian obat semua data yang berhubungan dengan data masuk obat akan terekap ke dalam *database* yang akan ditampilkan, selain itu sistem akan memberikan pilihan untuk klasifikasi rekap harian, bulanan, maupun tahunan.



Gambar 3.9 activity diagram pembelian obat

## 6) Activity Diagram Kelola Data Pengeluaran Obat

Seperti halnya diagram kelola data pembelian obat, diagram kelola data pengeluaran obat juga akan menampilkan semua data yang berhubungan dengan data keluar obat yang direkap ke dalam *database*, selain itu sistem akan memberikan pilihan untuk klasifikasi rekap harian, bulanan, maupun tahunan.

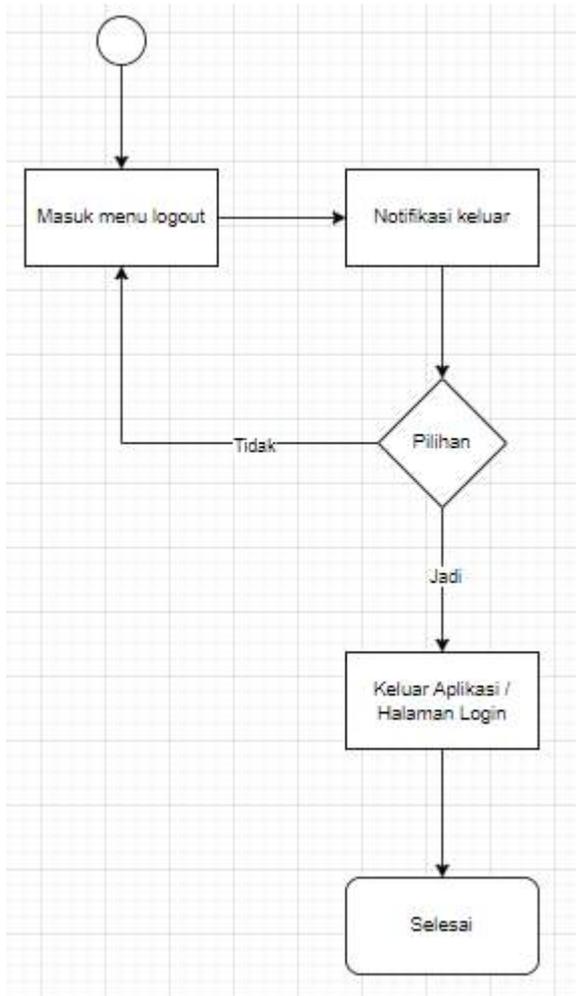


Gambar 3.10 activity diagram pengeluaran obat



7) *Activity Diagram Logout*

*Diagram logout* menunjukkan proses untuk keluar dari sistem aplikasi.



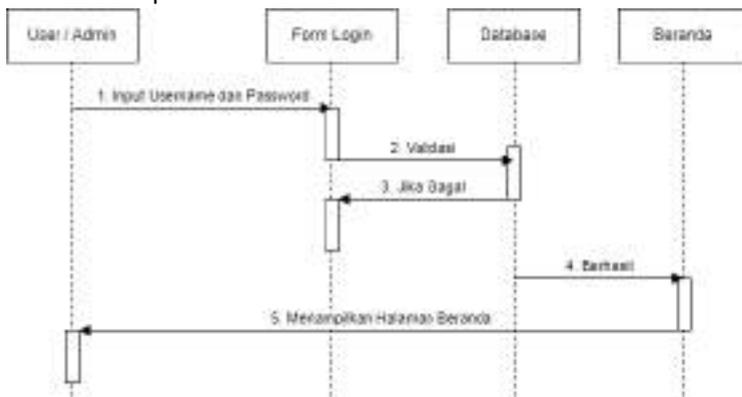
Gambar 3.11 activity diagram logout

#### D. Sequence Diagram

*Sequence Diagram* memperlihatkan bagaimana objek berorientasi satu dengan lainnya disertai urutan terjadinya interaksi tersebut. Penting diperhatikan bahwa *Sequence diagram* menunjukkan interaksi yang berlangsung pada suatu skenario tertentu. Proses direpresentasikan secara vertikal, sedangkan interaksi dinyatakan dengan tanda panah.

##### 1) Sequence Diagram Login

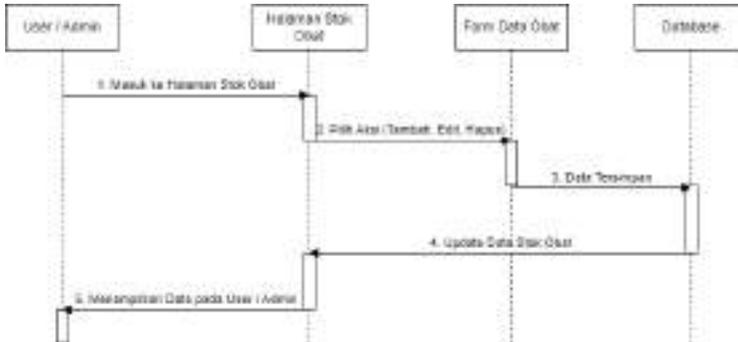
Diagram ini menggambarkan interaksi antar objek pada skenario *Login User* atau admin. Pada Gambar dibawah ini menunjukan interaksi objek (pengguna dan sistem) pada saat skenario *login* dengan akhir sistem akan menampilkan beranda.



Gambar 3.12 sequence diagram login

##### 2) Sequence Diagram Kelola Data Stok Obat

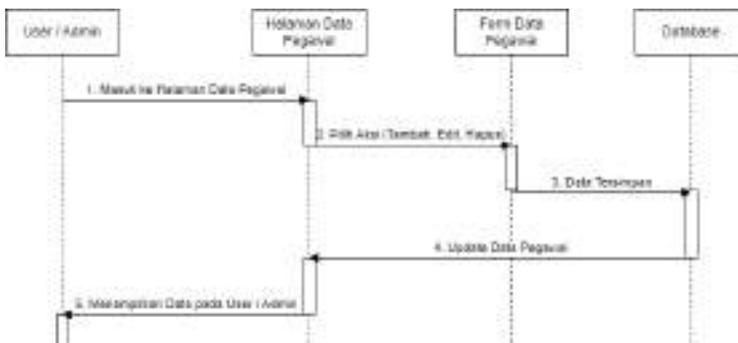
Diagram ini menggambarkan interaksi antar objek pada skenario kelola data stok obat. Pada Gambar dibawah ini menunjukan interaksi objek (pengguna dan sistem) pada saat skenario kelola data stok obat, mulai dari *User* masuk ke halaman stok obat, *User* memilih aksi (tambah, edit dan hapus), lalu data tersimpan ke *database* dan proses terakhir data ditampilkan pada halaman beranda stok obat.



Gambar 3.13 sequence diagram kelola stok obat

### 3) Sequence Diagram Kelola Data Pegawai

Diagram ini menggambarkan interaksi antar objek pada skenario kelola data pegawai. Pada Gambar dibawah ini menunjukkan interaksi objek (pengguna dan sistem) pada saat skenario kelola data pegawai, mulai dari *User* masuk ke halaman data pegawai, *User* memilih aksi (tambah, edit dan hapus), lalu data tersimpan ke *database* dan proses terakhir data ditampilkan pada halaman beranda data pegawai.

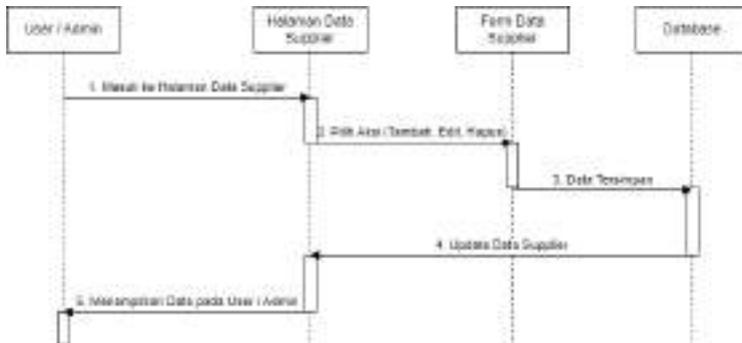


Gambar 3.14 sequence diagram kelola data pegawai

### 4) Sequence Diagram Kelola Data Supplier

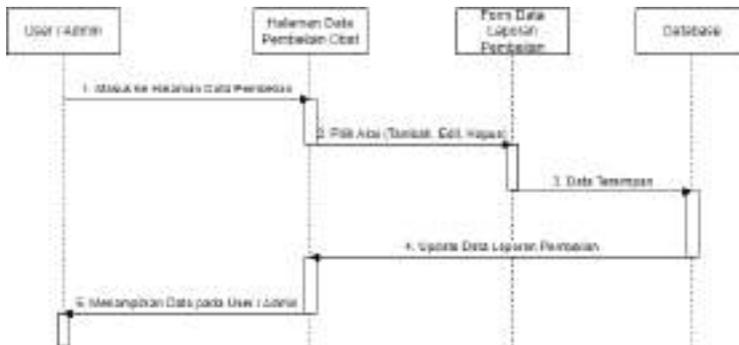
Diagram ini menggambarkan interaksi antar objek pada skenario kelola data *supplier*. Pada Gambar dibawah ini

menunjukkan interaksi objek (pengguna dan sistem) pada saat skenario kelola data *supplier*, mulai dari *User* masuk ke halaman data *supplier*, *User* memilih aksi (tambah, edit dan hapus), lalu data tersimpan ke *database* dan proses terakhir data ditampilkan pada halaman beranda data *supplier*.



Gambar 3.15 sequence diagram kelola data *supplier*

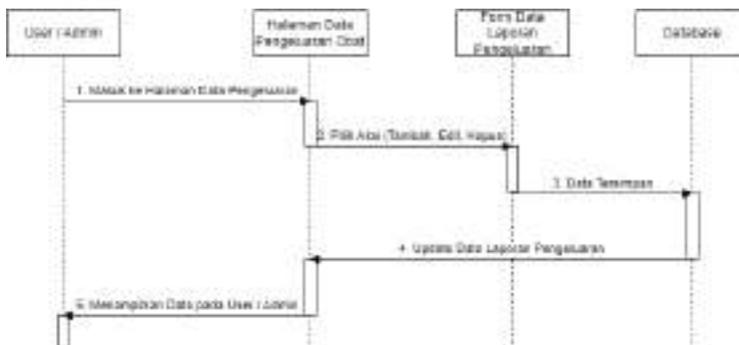
- 5) *Sequence Diagram* Kelola Data Pembelian Obat  
 Diagram ini menggambarkan interaksi antar objek pada skenario kelola data pembelian obat. Pada Gambar dibawah ini menunjukkan interaksi objek (pengguna dan sistem) pada saat skenario kelola data pembelian obat, mulai dari *User* masuk ke halaman data pembelian obat, *User* memilih aksi (tambah, edit dan hapus), lalu data tersimpan ke *database* dan proses terakhir data ditampilkan pada halaman beranda data pembelian obat.



Gambar 3.16 sequence diagram pembelian obat

#### 6) Sequence Diagram Kelola Data Pengeluaran Obat

Diagram ini menggambarkan interaksi antar objek pada skenario kelola data pengeluaran obat. Pada Gambar dibawah ini menunjukkan interaksi objek (pengguna dan sistem) pada saat skenario kelola data pengeluaran obat, mulai dari *User* masuk ke halaman data pembelian obat, *User* memilih aksi (tambah, edit dan hapus), lalu data tersimpan ke *database* dan proses terakhir data ditampilkan pada halaman beranda data pengeluaran obat.

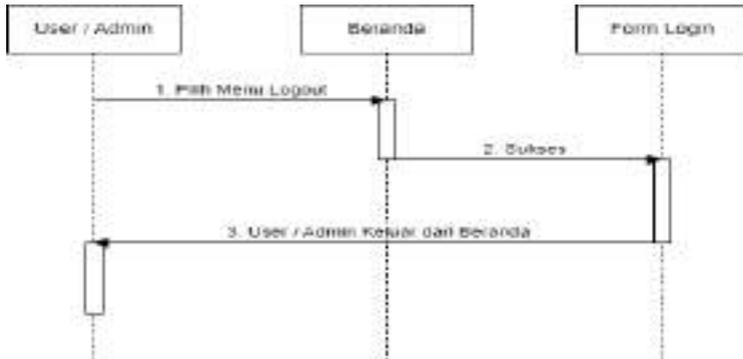


Gambar 3.17 sequence diagram pengeluaran obat

#### 7) Sequence Diagram Logout

Diagram ini menggambarkan interaksi antar objek pada skenario *logout* *User* atau *admin*. Pada Gambar dibawah

ini menunjukkan interaksi objek (pengguna dan sistem) pada saat skenario *logout User*, yang pada proses akhirnya *User* akan dialihkan pada halaman awal *login*.



Gambar 3.18 sequence diagram logout

### E. Class Diagram

*Class Diagram* adalah elemen mendasar dari setiap solusi yang berbasis objek. *Class diagram* memperlihatkan berbagai *class* serta hubungan antar *class* yang terdapat dalam sistem yang disertai dengan atribut dan operasi dari setiap *class* serta dapat dilihat rincian atribut dari setiap *class*. Berikut ini Gambaran dari class diagram sistem informasi manajemen stok obat di PSC 119 SIBANGJO.



Gambar 3.19 class diagram

## F. Desain Basis Data

Desain basis data berikut berisi rancangan basis data yang dibuat pada sistem informasi manajemen obat yang didasarkan pada rancangan class diagram dari bab sebelumnya. Bab ini menjelaskan rincian dari setiap class (Tabel) *database* yang akan digunakan untuk menyimpan data-data pada sistem yang dibuat.

Tabel 3.3 user

### 1) Tabel User

No	Atribut	Tipe Data
1	id_user	int (11)
2	username	varchar (30)
3	fullname	varchar (50)



4	<i>password</i>	varchar (255)
5	alamat	varchar (255)
6	hp	varchar (20)
7	foto	varchar (50)
8	level	enum('admin','pegawai')

Tabel *User* digunakan untuk menyimpan data pengguna yang mengakses sistem aplikasi tersebut.

*Tabel 3.4 supplier*

2) Tabel *Supplier*

No	Atribut	Tipe Data
1	<i>id_supplier</i>	varchar (15)
2	<i>nama_supplier</i>	varchar (225)
3	alamat	varchar (255)
4	telp	varchar (20)

Pada Tabel *supplier* digunakan untuk menyimpan daftar *supplier* distributor pengembang obat

*Tabel 3.5 penjualan*

3) Tabel Penjualan

No	Atribut	Tipe Data
1	<i>id_penjualan</i>	varchar (20)
2	<i>nama_pembeli</i>	varchar (30)
3	<i>tgl_penjualan</i>	date
4	<i>id_user</i>	int (11)

Tabel penjualan digunakan untuk menyimpan data penjualan atau data obat keluar.

*Tabel 3.6 pembelian*

4) Tabel Pembelian

No	Atribut	Tipe Data
1	<i>id_pembelian</i>	varchar (20)
2	<i>tgl_pembelian</i>	date

3	id_supplier	varchar (15)
4	id_user	int (11)

Tabel pembelian digunakan untuk menyimpan data pembelian obat.

*Tabel 3.7 detail pembelian*

5) Tabel Detail Pembelian

No	Atribut	Tipe Data
1	id_pembelian	varchar (20)
2	id_barang	varchar (6)
3	qty	smallint (6)
4	harga	double

Tabel detail pembelian merupakan Tabel yang digunakan untuk merekap data pembelian yang masuk ke database.

*Tabel 3.8 detail penjualan*

6) Tabel Detail Penjualan

No	Atribut	Tipe Data
1	id_penjualan	varchar (20)
2	id_barang	varchar (6)
3	qty	smallint (6)
4	harga	double

Tabel detail pengeluaran / penjualan merupakan Tabel yang digunakan untuk merekap data obat yang sudah keluar stok.

*Tabel 3.9 barang*

7) Tabel Barang

No	Atribut	Tipe Data
1	kode_barang	varchar (6)
2	nama_barang	varchar (255)
3	brand	varchar (255)

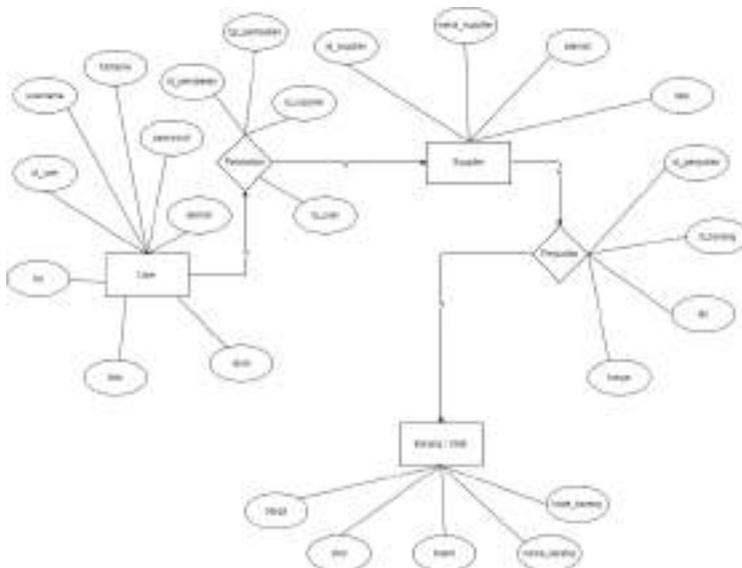


4	stok	int (11)
5	harga	double

Tabel barang merupakan Tabel yang digunakan untuk menyimpan data stok obat.

### G. Desain ER Diagram

Pada bagian desain ER diagram ini akan ditampilkan sebuah Gambar mengenai suatu model data yang menunjukkan entitas beserta atributnya yang bertujuan untuk mengetahui relasi antar objek.



Gambar 3.20 ER diagram

### H. Desain *User Interface*

Pada sub bab ini akan diberikan gambaran mengenai desain antarmuka aplikasi. Dimana antarmuka aplikasi ini digunakan oleh aktor untuk berinteraksi dengan sistem yang akan dibangun. Berikut adalah desain *User interface* aplikasi :

### 1. Halaman *Login*

Pada bagian ini akan dijelaskan sebuah halaman tampilan awal aplikasi, yang terdapat kolom pengisian *Username* dan sebuah kolom yang digunakan untuk memasukan *Password* saat masuk aplikasi. Berikut gambarannya.



Gambar 3.21 ui login

### 2. Halaman *Dashboard*

Pada gambaran halaman *dashboard* aplikasi akan mulai ditampilkan sebuah desain menu yang terdapat pada aplikasi, antara lain Halaman Data Obat, Data Pegawai, Data *Supplier* dan juga Halaman Laporan Rekap Stok Obat. Berikut gambarannya.

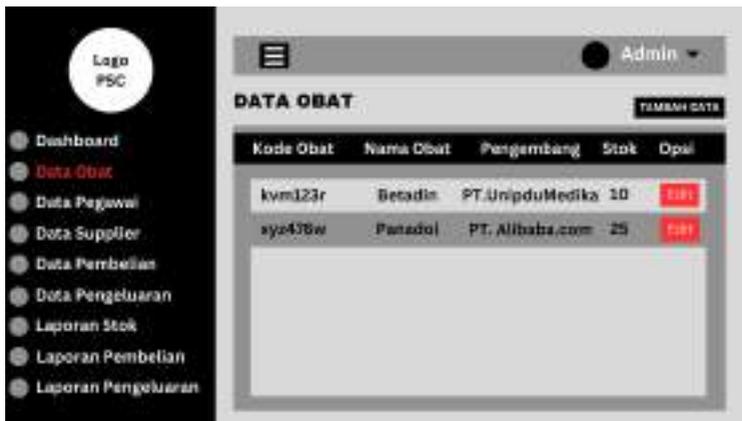




Gambar 3.22 ui dashboard

### 3. Halaman Data Obat

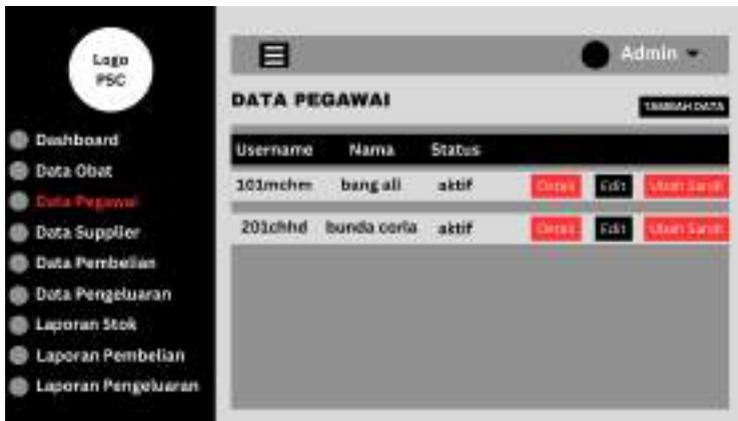
Pada gambaran halaman data obat dapat dijelaskan bahwa semua obat yang sudah masuk atau terinput ke sistem akan muncul serta ditampilkan dengan keterangan detail obat tersebut contohnya Kode Obat, Jumlah Stok, dan Status Obat. Berikut gambarannya.



Gambar 3.23 ui data obat

### 4. Halaman Data Pegawai

Pada gambaran halaman ini dapat dijelaskan bahwa semua *User* yang terdaftar pada sistem akan ditampilkan pada halaman data pegawai. Selain itu pada halaman ini juga akan ditampilkan tiga opsi untuk mengolah data tersebut. Contohnya opsi edit data, lihat detail data dan opsi ubah *Password*. Berikut gambarannya.



Gambar 3.24 ui data pegawai

##### 5. Halaman Laporan Stok

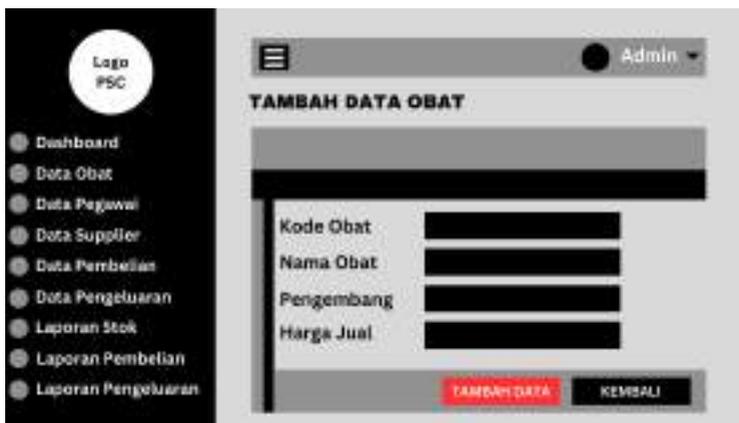
Halaman ini merupakan tampilan untuk mengetahui rekapitulasi tentang stok obat. Ada tiga hal yang ditampilkan yakni laporan stok obat, laporan pembelian serta laporan pengeluaran obat. Pada masing-masing halaman *User* bisa melihat rekap stok obat periode waktu hari, bulan bahkan tahun. Berikut gambarannya.



Gambar 3.25 ui laporan stok

#### 6. Halaman Tambah Data Obat

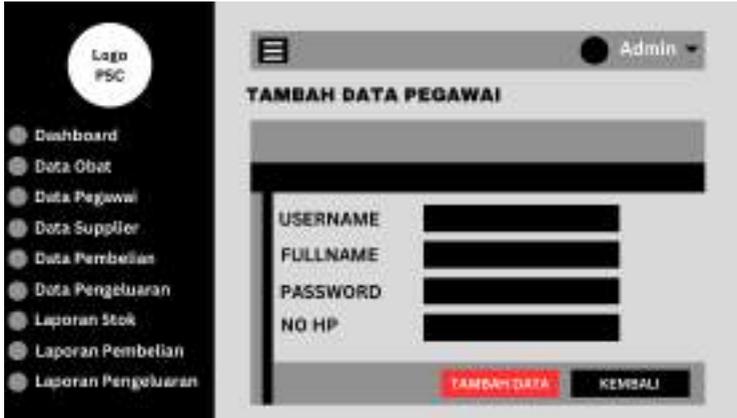
Halaman tambah data obat ini terdapat pada tampilan antarmuka aplikasi sistem, halaman ini digunakan untuk menambahkan data yang otomatis akan tersimpan pada database aplikasi. Berikut gambarannya.



Gambar 3.26 ui tambah obat

#### 7. Halaman Tambah Data Pegawai

Halaman ini juga seperti halaman tambah data obat, dimana data yang terinput pada tambah data pegawai ini akan otomatis tersimpan pada database aplikasi. Berikut gambarannya.



Gambar 3.27 ui tambah pegawai