

BAB III

PEMBAHASAN

3.1. Tinjauan Perusahaan

Tinjauan terhadap perusahaan secara umum untuk mendukung jalannya suatu sistem organisasi agar sistem dapat berjalan dengan lebih terencana, terarah, dan sesuai dengan peraturan yang harus dipenuhi oleh sistem itu sendiri. Sedangkan secara khusus untuk menjalankan tugasnya dengan lebih terarah sesuai dengan peraturan yang telah ada.

3.1.1. Sejarah Institusi/Perusahaan

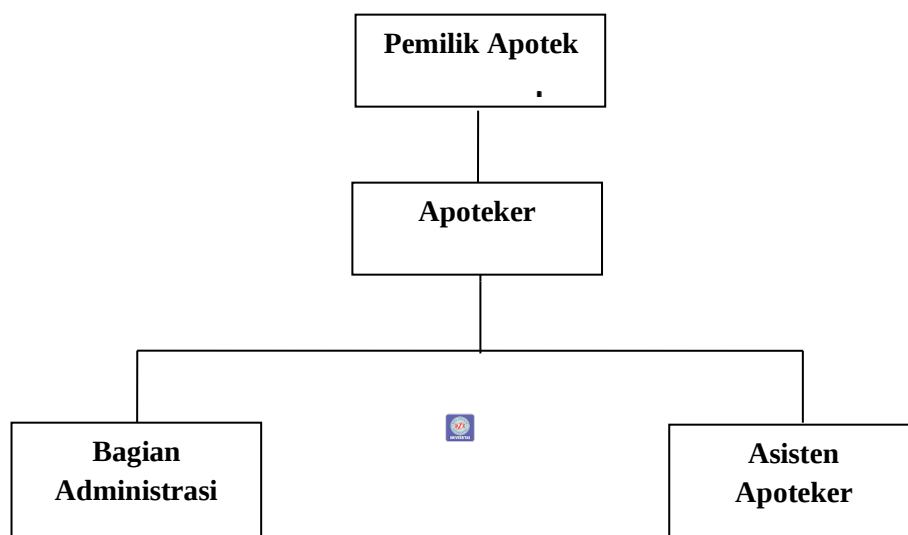
Berdasarkan Nomor SIA : 449/085/SIA/BPMP2T/IX/2016, Apotek Mama merupakan salah satu bidang usaha yang bergerak dalam bidang penjualan obat-obatan dengan mengutamakan pemenuhan kebutuhan dan kepuasan pelanggan. Pada tahun 2012 Bapak Romi Kriska putra dan dua orang temannya mendirikan Apotek Gadi Lamba di daerah Pasar Rebo, karena terjadi ketidakcocokan dalam pengelolaan Apotek maka Bapak Romi Kriska Putra mengundurkan diri dari Apotek Gadi Lamba dan berniat untuk mendirikan usaha sendiri.

Dengan modal yang tidak terlalu banyak Bapak Romi Kriska Putra mendirikan Apotek Mama di daerah Kalimalang Jakarta Timur. Dengan pengelolaan yang baik, Apotek Mama tersebut dapat berjalan dengan lancar. Kemudian Bapak Romi Kriska Putra berniat untuk mendirikan Apotek Mama yang kedua di daerah Depok. Akhirnya di awal tahun yaitu tanggal satu Januari 2013 diresmikannya Apotek Mama yang kedua yang beralamat di Jalan Proklamasi Nomor 35. Berkat kerja keras dari Bapak Romi Kriska Putra maka Apotek Mama dapat berjalan lancar sampai saat ini.

3.1.2 Struktur Organisasi dan Fungsi

Dengan struktur organisasi yang baik, tugas-tugas digolongkan sedemikian rupa sehingga dapat dilaksanakan secara efektif, terarah, dan terawasi, serta hasilnya dapat terkendali. Di Apotek Mama struktur organisasi yang berjalan masih sangat sederhana.

Bentuk struktur organisasi Apotek Mama adalah sebagai berikut :



Sumber: (Administrasi Apotek Mama, 2019)

Gambar III.1
Struktur Organisasi Pada Apotek Mama

Berikut ini adalah fungsi dari masing-masing bagian, yaitu sebagai berikut :

1. Pemilik Apotek
 - a. Menetapkan kebijaksanaan serta aturan-aturan didalam apotek.
 - b. Menerima, memeriksa, dan menyetujui laporan kegiatan apotek yang dibuat oleh bagian administrasi.
 - c. Memantau, mengarahkan, dan mengevaluasi kegiatan penjualan dan pembelian di apotek.

2. Apoteker

- a. Mengawasi dan memantau jenis obat-obatan yang dibutuhkan oleh apotek.
- b. Membaca resep dokter apabila ada pembeli yang membeli obat menggunakan resep dokter.

3. Asisten Apoteker

- a. Membantu apoteker (membaca resep dokter apabila pada saat terjadi pembelian yang menggunakan resep dokter dan apoteker sedang tidak ada di apotek).
- b. Melakukan penjualan obat.
- c. Memberikan informasi yang berkaitan dengan penggunaan atau pemakaian obat yang akan di serahkan pada pasien dan juga memberikan informasi mengenai penggunaan secara tepat, benar, rasional, serta mudah dimengerti pasien atau masyarakat.

4. Bagian Administrasi

- a. Melakukan arsip transaksi keuangan apotek baik dalam kegiatan pembelian dan penjualan obat.
- b. Membuat Laporan-laporan yang ada di Apotek.
- c. Pengarsipan catatan pengobatan pasien.
- d. Pengarsipan Resep
- e. Membuat perencanaan keuangan, mulai dari perencanaan peminjaman, penerimaan, pengeluaran, dan pembayaran.

3.2 Tinjauan Kasus

Dalam tinjauan kasus dapat dijelaskan sebagai berikut:

3.2.1. Proses Bisnis Sistem Berjalan

1. Proses Pemesanan Obat

Pemesanan obat bisa dilakukan dengan dua cara. Pertama pemesanan obat secara langsung, kedua melalui resep dokter. Jika konsumen melakukan pemesanan secara langsung maka konsumen akan memberikan pesanan obat lalu asisten apoteker akan mengecek ketersediaan obat, jika obat tersedia maka asisten apoteker akan membuatkan nota rangkap 2. Jika obat tidak tersedia, konsumen bisa membatalkan proses pemesanan obat.



2. Proses Pembayaran

Jika konsumen melakukan pemesanan obat melalui resep, konsumen akan memberikan resep tersebut ke asisten apoteker. Jika konsumen setuju maka resep akan diberikan ke apoteker untuk dianalisis. Lalu apoteker akan meracik dan menyiapkan obat sesuai resep. Apoteker akan memberikan obat dan etiket kepada asisten apoteker. Asisten apoteker akan membuatkan nota rangkap 2.

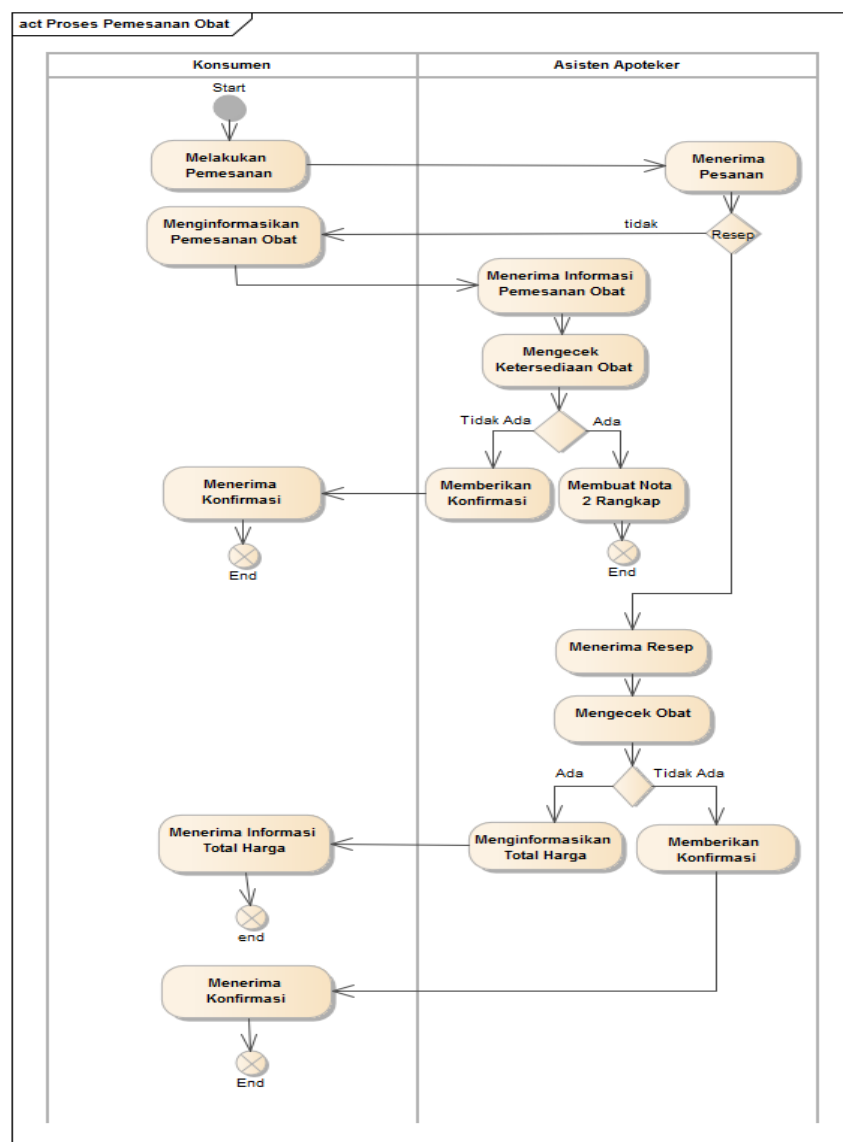
Konsumen akan diinformasikan total harga dan melakukan pembayaran kepada asisten apoteker. Asisten apoteker akan memberikan nota putih dan obat serta menjelaskan etiket yakni aturan pakai kepada konsumen, sedangkan nota berwarna merah akan disimpan sebagai arsip. Lalu asisten apoteker akan mencatat setiap transaksi penjualan harian ke dalam buku kas.

3. Proses Pembuatan Laporan

Bagian Administrasi akan membuat laporan penjualan bulanan dengan merekap transaksi penjualan harian di buku kas. Berdasarkan rekap data tersebut pembuatan laporan terbuat untuk diberikan kepada pemilik.

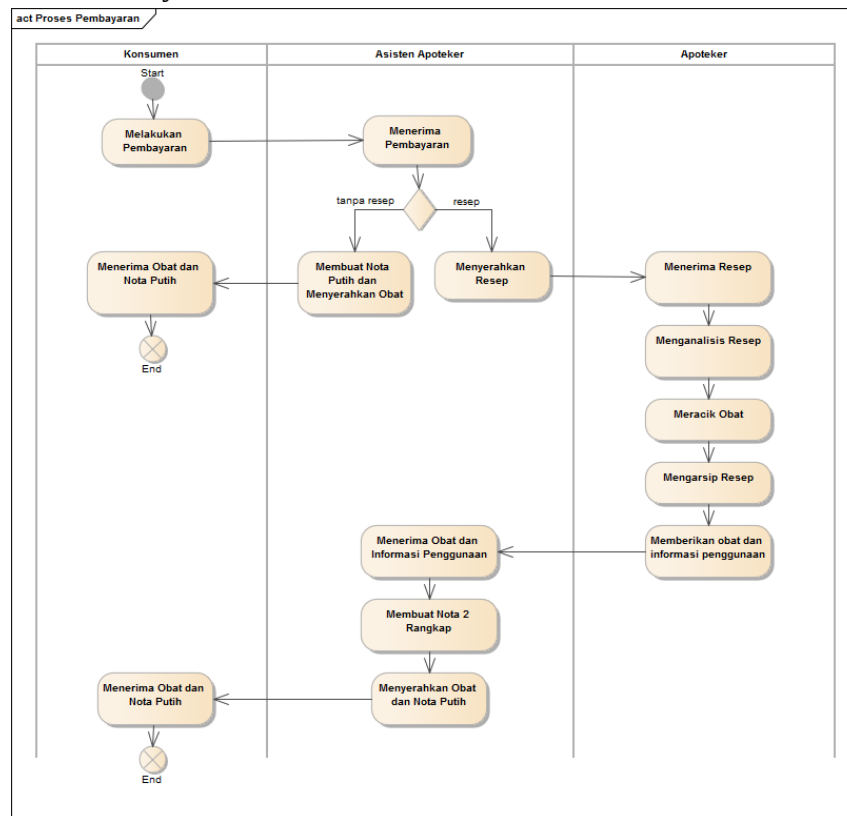
3.2.2. Activity Diagram

1. Proses Pemesanan Obat



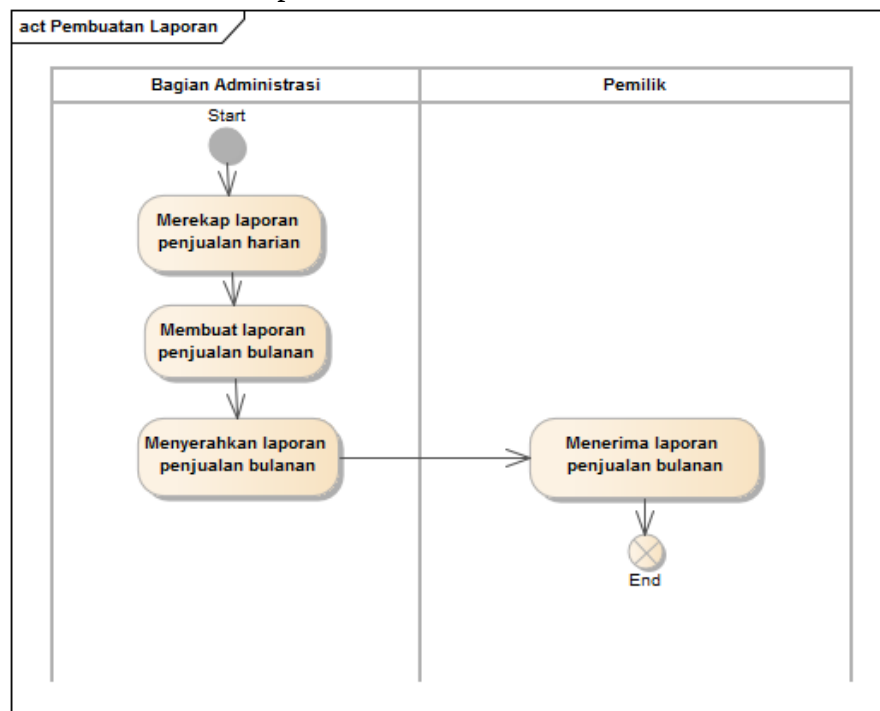
Gambar III.2
Activity Diagram Pemesanan Obat

2. Proses Pembayaran



Gambar III.3
Activity Diagram Proses Pembayaran

3. Proses Pembuatan Laporan



Gambar III.4
Activity Diagram Pembuatan Laporan

3.2.3. Dokumen Masukan

Dokumen masukan merupakan proses awal dari kegiatan Penjualan obat pada Apotek Mama Depok. Adapun dokumen masukan tersebut tersebut adalah sebagai berikut :

1. Nama Dokumen : Resep
 - Fungsi : Untuk mengetahui obat yang dibutuhkan konsumen
 - Sumber : Konsumen
 - Tujuan : Asisten Apoteker
 - Media : Kertas
 - Jumlah : 1 Lembar
 - Frekuensi : Setiap ada pemesanan obat
 - Format : Lampiran A.1
2. Nama Dokumen : Daftar Obat
 - Fungsi : Untuk mengetahui data obat dan harga obat
 - Sumber : Asisten Apoteker
 - Tujuan : Asisten Apoteker
 - Media : Kertas
 - Jumlah : 1 lembar
 - Frekuensi : Setiap pembelian obat dari *Supplier*
 - Format : Lampiran A.2

3.2.4. Dokumen Keluaran

Dokumen Keluaran merupakan proses akhir dari kegiatan Pembelian obat pada Apotek Mama Depok. Adapun bentuk dokumen keluaran tersebut adalah sebagai berikut :

1. Nama Dokumen : Laporan Penjualan
 Fungsi : Melihat Pemasukan penjualan obat
 Sumber : Asisten Apoteker
 Tujuan : Pemilik
 Media : Kertas
 Jumlah : 1 Lembar
 Frekuensi : Perhari
 Format : Lampiran B.1
2. Nama Dokumen : Nota Penjualan
 Fungsi : Bukti lunas pembayaran obat *Customer*
 Sumber : Asisten Apoteker
 Tujuan : *Customer*
 Media : Kertas
 Jumlah : 1 Lembar
 Frekuensi : Setiap membeli obat
 Format : Lampiran B.2

3.2.5. Permasalahan Pokok

Setelah mempelajari sistem yang berjalan pada Apotek Mama, penulis mencoba memberikan suatu analisa tentang sistem tersebut. Dalam hal ini penulis menemukan beberapa permasalahan yang timbul, dimana permasalahan itu harus dilakukan pemecahan agar tidak menghambat kegiatan-kegiatan dalam menyediakan informasi yang lebih akurat khususnya dalam penanganan sistem penjualan. Adapun beberapa permasalahan yang terjadi pada Apotek Mama yang dapat penulis kemukakan adalah sebagai berikut:

1. Pada proses pencatatan penjualan sering terjadi kesalahan dalam perhitungan total transaksi sehingga menyebabkan data atau laporan yang dihasilkan tidak akurat.
2. Dokumen yang digunakan Apotek Mama berupa media kertas sehingga data yang laporan penjualan dan nota pembayaran mudah hilang dan dapat menghambat proses pembuatan laporan sehingga membutuhkan waktu yang lebih lama..

3.2.6. Pemecahan Masalah

Dari permasalahan yang timbul, maka hal tersebut perlu mendapatkan pemecahan agar proses penjualan obat dapat berjalan lebih efektif, Adapun cara pemecahan masalahnya adalah sebagai berikut :

1. Penulis merancang sistem aplikasi untuk proses pencatatan dan pembuatan laporan dengan menggunakan Java sehingga dapat meminimalisir kesalahan dan mempercepat proses pencatatan dan perhitungan laporan penjualan di Apotek Mama.
2. Dengan digunakannya *database* berupa MYSQL, penyimpanan data akan lebih terstruktur, jika diperlukan informasi data dapat ditemukan dengan cepat dan lebih efisien.

3.3 Analisis Kebutuhan Software

Analisis kebutuhan perangkat lunak (*Software Requirements Analysis*) merupakan aktivitas awal dari siklus hidup pengembangan perangkat lunak.

3.3.1. Analisis Kebutuhan

Berdasarkan proses penjualan sistem berjalan pada Apotek Mama, maka tahapan berikutnya adalah analisis kebutuhan. Berikut ini spesifikasi kebutuhan (*System Requirement*) dari sistem Penjualan obat pada Apotek Mama.

Halaman Administrasi

A.1. Asisten Apoteker Login

A.2. Asisten Apoteker Mengakses Menu Data Master

A.3. Asisten Apoteker Mengelola Data Obat

A.4. Asisten Apoteker Mengelola Data User

A.5. Asisten Apoteker Mengakses Menu Transaksi

A.6. Asisten Apoteker Mengelola Transaksi Penjualan

A.7. Asisten Apoteker Mengelola Master Akun

A.8. Asisten Apoteker Mengelola Jurnal Umum

A.9. Asisten Apoteker Mengakses Menu Laporan

A.10. Asisten Apoteker Mengakses Laporan Penjualan

A.11. Asisten Apoteker Mengakses Laporan Jurnal Umum

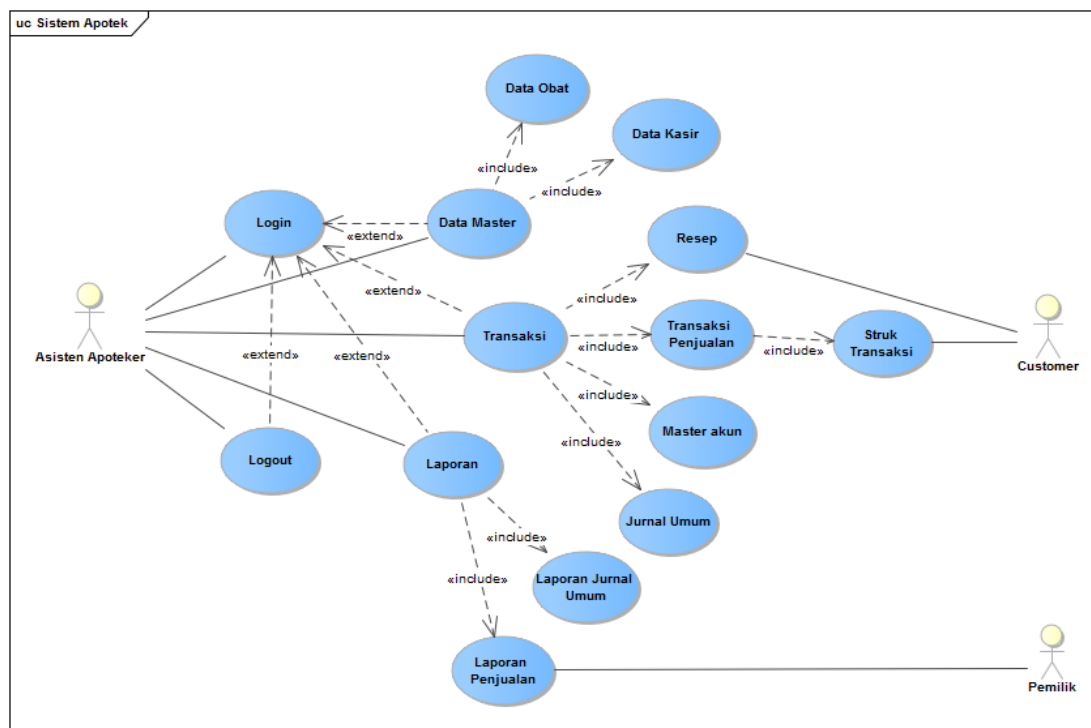
Halaman Pemilik

B.1. Pemilik melakukan Login

B.2. Pemilik Mengakses Laporan Penjualan

3.3.2. Use Case Diagram

1. Use Case Sistem Apotek



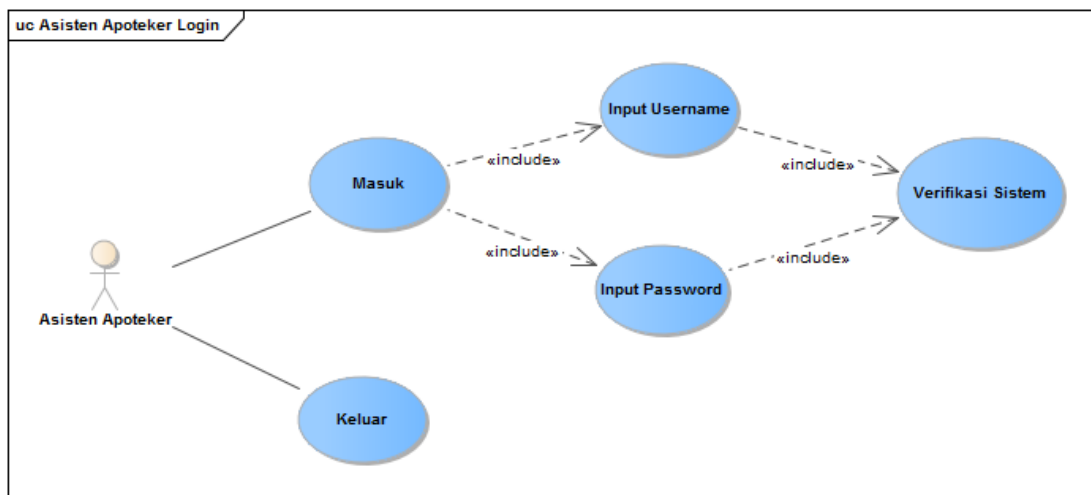
Gambar III.5
Use Case Diagram Sistem Apotek Mama Depok

Tabel III.1
Deskripsi Use Case Halaman Utama sistem Mama Apotek

Use Case Narrative Bagian Asisten Apoteker	
Tujuan	Untuk mengolah data dan melakukan transaksi penjualan
Deskripsi	Sistem ini digunakan untuk mengolah data dan melakukan transaksi bersangkutan
Skenario Utama	
Aktor	Asisten Apoteker, customer, dan pemilik
Kondisi Awal	Aktor Asisten Apoteker membuka halaman utama aplikasi penjualan
Aksi Aktor : Asiten Apoteker	Reaksi Sistem
1. Aktor Memilih Data master	Akan menampilkan form data master
2. Aktor memilih Transaksi	Akan menampilkan form data Transaksi
3. Aktor memilih Laporan	Akan menampilkan form Laporan
Aksi Aktor : Customer	Reaksi Sistem
1. Aktor memberikan data	Akan memproses transaksi pemesanan

pemesanan obat	obat oleh aktor Asiten Apoteker.
Aksi Aktor : Pemilik	Reaksi Sistem
1. Aktor menerima laporan penjualan dari sistem	Akan menampilkan laporan transaksi penjualan obat.
Kondisi Akhir	Jika semuanya benar, maka sistem akan melakukan kehendak aktor.

2. Use Case Login Asiten Apoteker

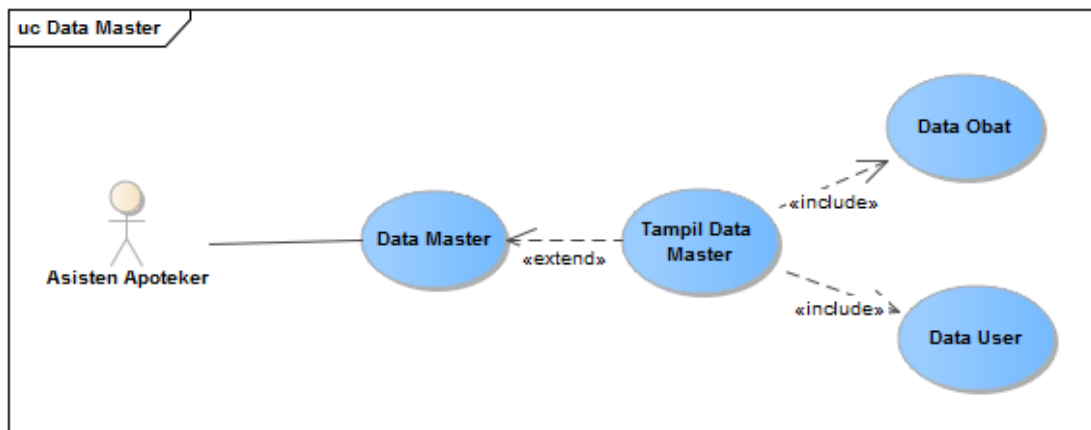


Gambar III.6
Use Case Diagram Bagian Login Asiten Apoteker

Tabel III.2
Deskripsi Use Case Halaman Utama Bagian Login Asiten Apoteker

Use Case Narrative Login Asiten Apoteker	
Tujuan	Melakukan <i>Login</i>
Deskripsi	Sistem ini digunakan untuk mengakses sistem penjualan
Skenario Utama	
Aktor	Asiten Apoteker
Kondisi Awal	Aktor menggunakan sistem penjualan
Aksi Aktor	Reaksi Sistem
1. Aktor Membuka aplikasi sistem penjualan <i>LOGIN</i>	Akan menampilkan form data master
Kondisi Akhir	Jika semuanya benar, maka sistem akan melakukan kehendak aktor.

3. Use Case Mengakses Menu Data Master



Gambar III.7

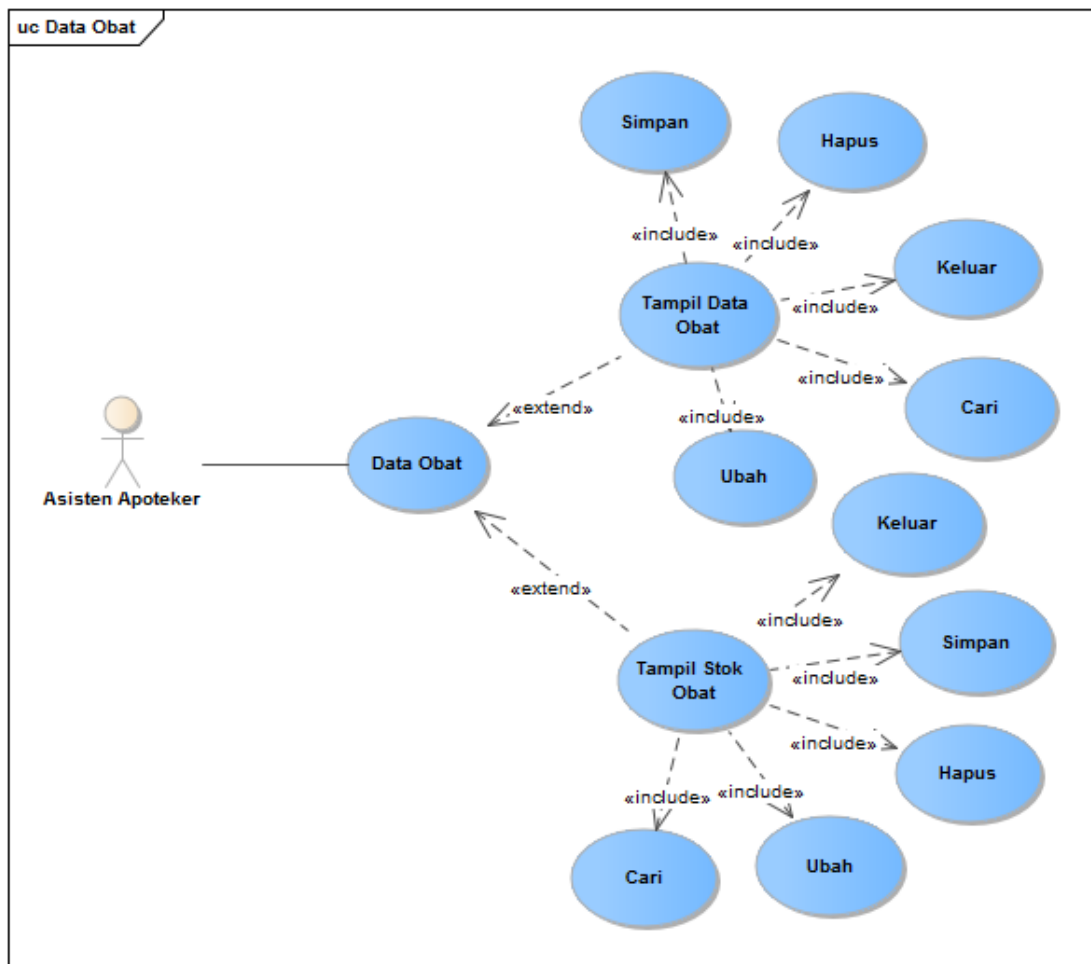
Use Case Diagram Bagian Mengakses Menu Data Master

Tabel III.3

Deskripsi Use Case Halaman Utama Bagian Mengakses Menu Data Master

Use Case Narrative Menu Data Master	
Tujuan	Untuk mengakses dan mengelolah submenu yang ada di menu master.
Deskripsi	Proses untuk mengakses dan mengelola menu data obat, data user.
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Menu data master sudah dapat di akses
1. Aktor memilih Data Obat	Sistem akan menampilkan form menu data obat.
2. Aktor memilih Data Kasir	Sistem akan menampilkan form data kasir
Kondisi Akhir	Sistem dapat mengakses dan mengelola sub menu master.

4. Use Case Mengelola Data Obat



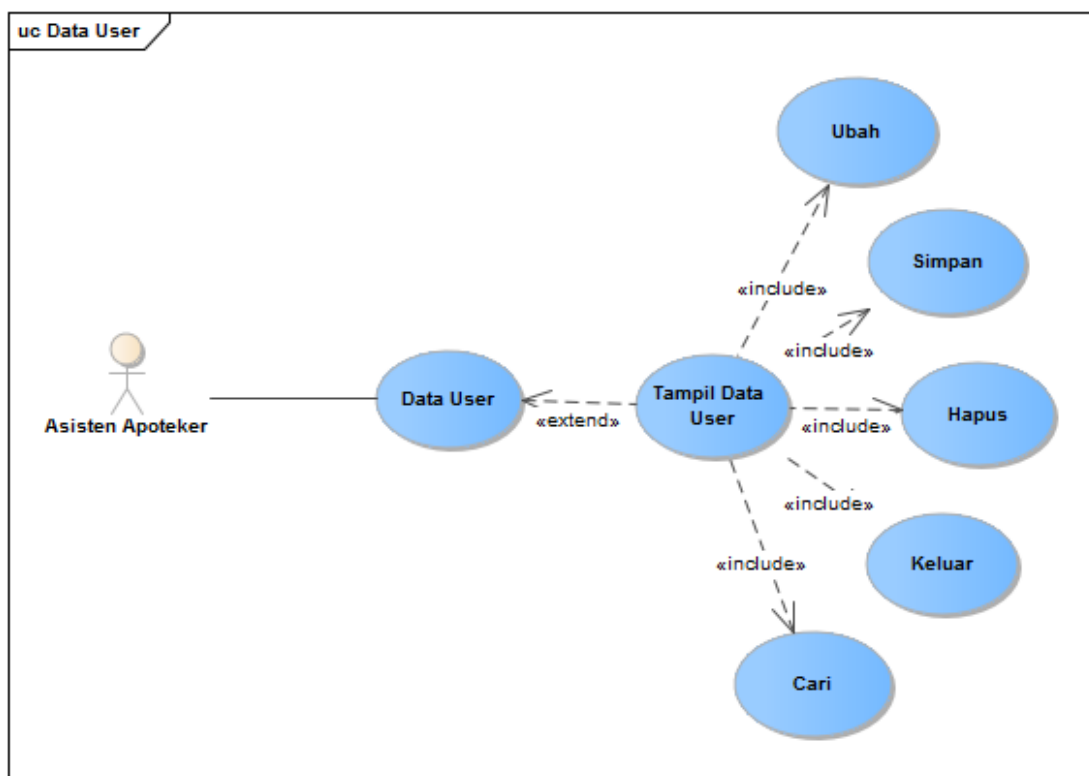
Gambar III.8
Use Case Diagram Bagian Mengelola Data Obat

Tabel III.4
Deskripsi Use Case Halaman Utama Bagian Mengelola Data Obat

<i>Use Case Narrative</i> Data Obat	
Tujuan	Untuk mengelola data obat
Deskripsi	Sistem ini digunakan untuk mengola data dan stok obat
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik submenu master dan masuk ke bagian Data Obat
Aktor	Reaksi Sistem
1. Aktor memilih data obat	Akan menampilkan data obat
2. Aktor memilih stok obat	Akan menampilkan stok obat
3. Aktor memilih tombol Simpan	Data Obat atau stok obat akan tersimpan
4. Aktor memilih tombol Hapus	Data Obat atau stok obat akan terhapus

5. Aktor memilih tombol Keluar	Data Obat atau Stok obat akan keluar kembali ke menu utama
6. Aktor memilih tombol Cari	Sistem akan mencari Data Obat atau stok obat yang telah tersimpan
7. Aktor memilih tombol Ubah	Stok obat akan berubah
Kondisi Akhir	Sistem akan menampilkan menu data obat untuk di akses dan di kelola oleh kasir.

5. Use Case Mengelola Data User



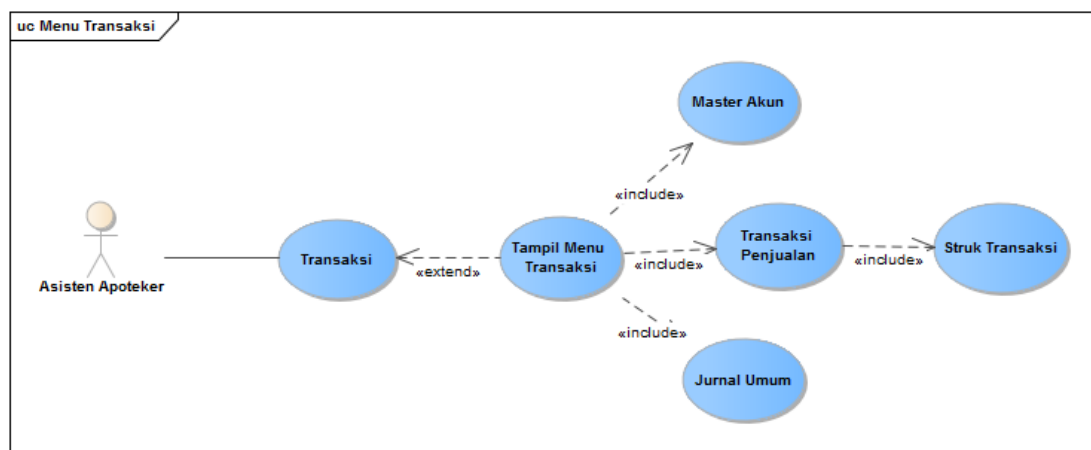
Gambar III.9
Use Case Diagram Bagian Mengelola Data User

Tabel III.5
Deskripsi Use Case Halaman Utama Bagian Mengelola Data User

<i>Use Case Narrative Data User</i>	
Tujuan	Untuk mengelola data User
Deskripsi	Sistem ini digunakan untuk mengelola data_data bersangkutan
Skenario Utama	
Aktor	Asisten Apoteker

Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik submenu master dan masuk ke bagian Data User
Aktor	Reaksi Sistem
1. Aktor memilih data user	Akan menampilkan data user
2. Aktor memilih tombol Simpan	Data user akan tersimpan
3. Aktor memilih tombol Hapus	Data user akan terhapus
4. Aktor memilih tombol Keluar	Data user akan keluar kembali ke menu utama
5. Aktor memilih tombol Cari	Sistem akan mencari Data user yang telah tersimpan
Kondisi Akhir	Sistem akan menampilkan menu data user untuk di akses dan di kelola oleh Asisten apoteker.

6. Use Case Mengakses Menu Transaksi



Gambar III.10

Use Case Diagram Bagian Mengakses Menu Transaksi

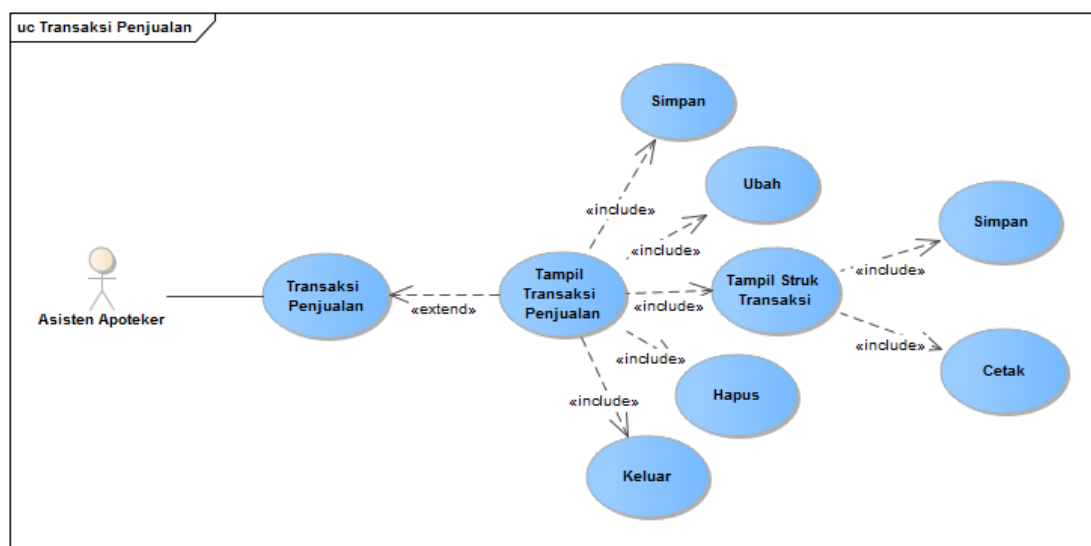
Tabel III.6

Deskripsi Use Case Halaman Utama Bagian Mengakses Menu Transaksi

Use Case Narrative Menu Transaksi	
Tujuan	Untuk mengakses dan mengelola submenu yang ada di menu Transaksi.
Deskripsi	Proses untuk mengakses dan mengelola resep, transaksi penjualan, jurnal umum, master akun
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Menu Transaksi sudah dapat di akses

1. Aktor memilih menu Resep	Sistem akan menampilkan form menu resep
2. Aktor memilih menu transaksi penjualan	Sistem akan menampilkan form menu Transaksi penjualan.
3. Aktor memilih menu Jurnal Umum	Sistem akan menampilkan form Jurnal Umum
4. Aktor memilih menu jurnal master akun	Sistem akan menampilkan menu jurnal master akun
Kondisi Akhir.	Sistem dapat mengakses dan mengelola submenu Transaksi

7. Use Case Mengelola Transaksi Penjualan



Gambar III.11

Use Case Diagram Bagian Mengelola Transaksi Penjualan

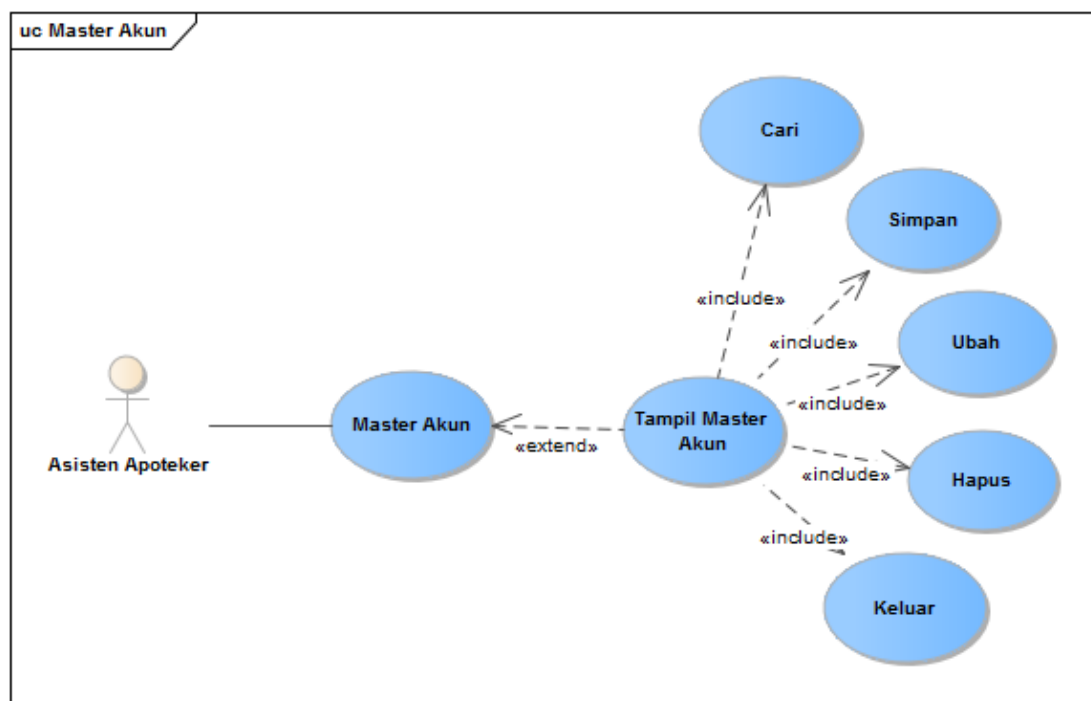
Tabel III.7

Deskripsi Use Case Halaman Utama Bagian Mengelola Transaksi Penjualan

Use Case Narrative Penjualan Obat	
Tujuan	Untuk Menginput Penjualan Obat
Deskripsi	Sistem ini digunakan untuk mengolah data_data yang bersangkutan
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik submenu Transaksi dan masuk ke bagian Penjualan Obat
Aksi Aktor	Reaksi Sistem
1. Aktor memilih Penjualan Obat	Akan menampilkan Penjualan

2. Aktor memilih tombol Simpan	Penjualan Obat akan tersimpan
3. Aktor akan memilih tombol Ubah	Penjualan Obat akan berubah
4. Aktor akan memilih tombol Hapus	Penjualan Obat akan terhapus
5. Aktor akan memilih tombol Keluar	Penjualan Obat akan Keluar dari penjualan obat kembali kemenu utama
6. Aktor akan memilih tombol Cari	Sistem akan mencari Penjualan Obat yang telah tersimpan
7. Aktor memilih tombol Tampil Struk transaksi	Tampil Struk transaksi
8. Aktor memilih tombol Cetak	Struk transaksi akan tercetak
9. Aktor memilih tombol simpan	Struk transaksi akan tersimpan dalam komputer
Kondisi Akhir	Sistem akan menampilkan menu Penjualan Obat untuk di akses dan di kelola Asisten Apoteker

8. Use Case Mengelola Master Akun



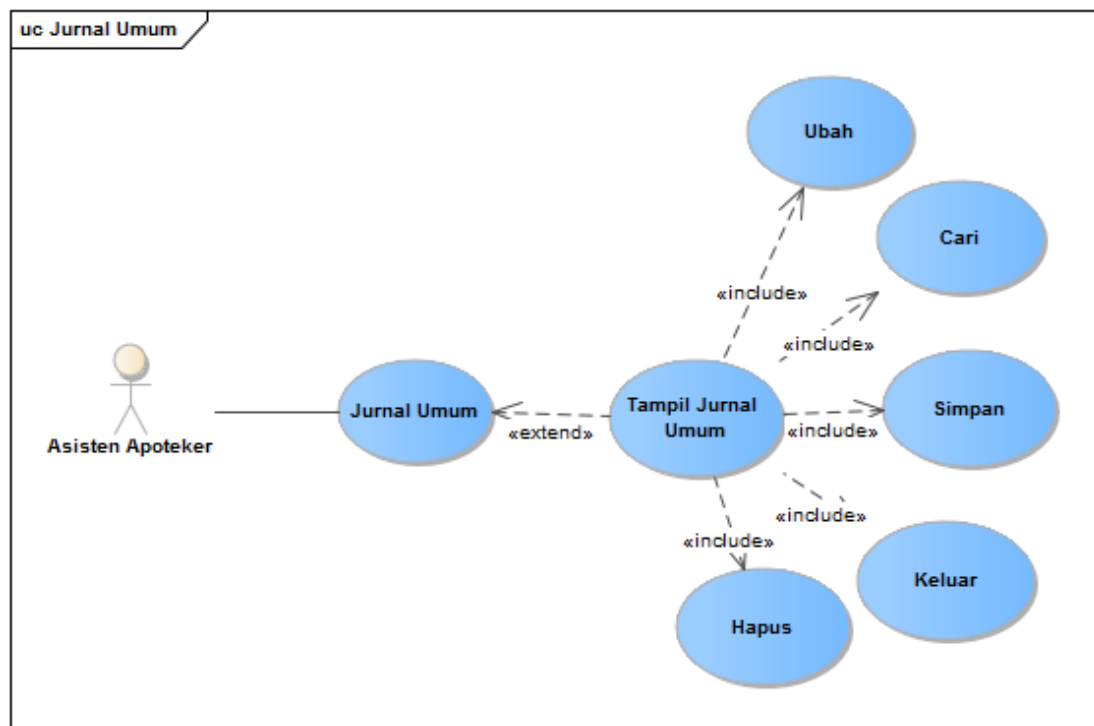
Gambar III.12
Use Case Diagram Bagian Mengelola Master Akun

Tabel III.8
Deskripsi Use Case Halaman Utama Bagian Mengelola Master Akun

Use Case Narrative Master Akun	
Tujuan	Untuk mengelola Master Akun
Deskripsi	Sistem ini digunakan untuk mengolah data_data bersangkutan
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik sub menu Transaksi dan masuk ke bagian Master Akun
Aktor	Reaksi Sistem
1. Aktor memilih Master Akun	Akan menampilkan Master Akun
2. Aktor memilih tombol Simpan	Akun Perkiraan akan tersimpan
3. Aktor memilih tombol Keluar	Akun Perkiraan akan keluar dari master akun
4. Aktor memilih tombol Ubah	Sistem akan mengubah Master Akun yang telah tersimpan
Kondisi Akhir	Sistem akan menampilkan menu Master Akun untuk di akses dan di kelolah oleh Asisten Apoteker.



9. Use Case Mengelola Jurnal Umum



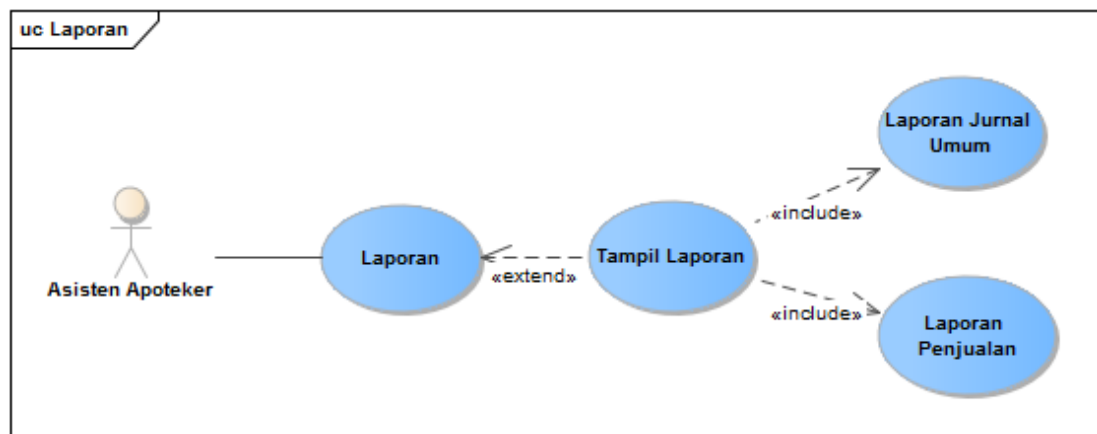
Gambar III.13
Use Case Diagram Bagian Mengelola Jurnal Umum

Tabel III.9
Deskripsi Use Case Halaman Utama Bagian Mengelola Jurnal umum

<i>Use Case Narrative Jurnal Umum</i>	
Tujuan	Untuk mengelolah Jurnal Umum
Deskripsi	Sistem ini digunakan untuk mengolah data_data bersangkutan
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik submenu Transaksi dan masuk ke bagian Jurnal Umum
Aktor	Reaksi Sistem
1. Aktor memilih Jurnal Umum	Akan menampilkan Jurnal Umum
2. Aktor memilih tombol Simpan	Jurnal akan tersimpan
3. Aktor memilih tombol Keluar	Jurnal Umum akan Keluar dan kembali kemenu utama
4. Aktor memilih tombol Cari	Sistem akan mencari Jurnal Umum yang telah tersimpan
Kondisi Akhir	Sistem akan menampilkan menu Jurnal Umum untuk di akses dan di kelola oleh Asisten Apoteker.



10. Use Case Mengakses Laporan



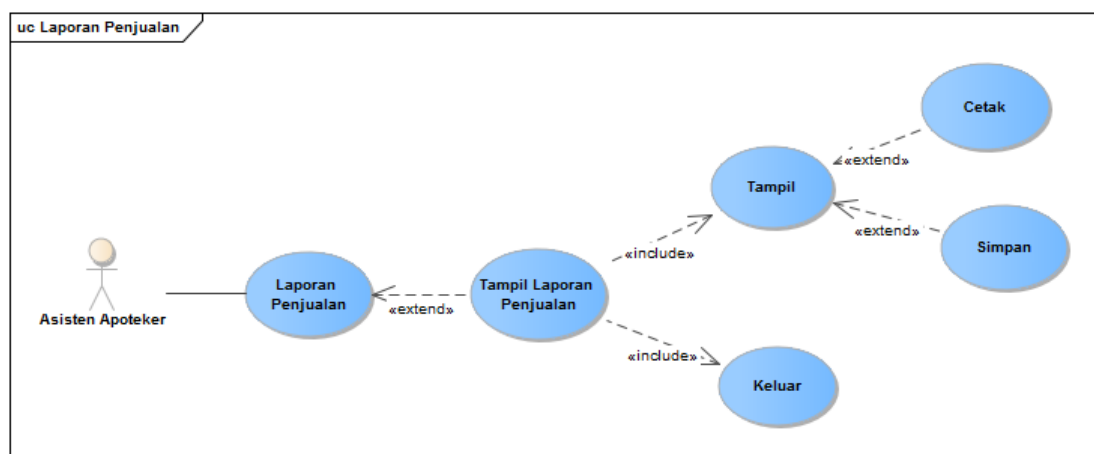
Gambar III.14
Use Case Diagram Bagian Mengakses Laporan

Tabel III.10
Deskripsi Use Case Halaman Utama Bagian Mengakses Laporan

<i>Use Case Narrative Menu Laporan</i>	
Tujuan	Untuk mengakses dan mengelolah submenu yang ada di menu Laporan.

Deskripsi	Proses untuk mengakses dan mengelola Jurnal Umum, Laporan Penjualan.
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Menu Transaksi sudah dapat di akses
1. Aktor memilih Jurnal Umum	Sistem akan menampilkan form Jurnal Umum
2. Aktor Memilih Laporan Penjualan	Sistem akan menampilkan Laporan Penerimaan Kas
Kondisi Akhir	Sistem dapat mengakses dan mengelola sub menu Laporan

11. Use Case Mengakses Laporan Penjualan



Gambar III.15

Use Case Diagram Bagian Mengakses Laporan Penjualan

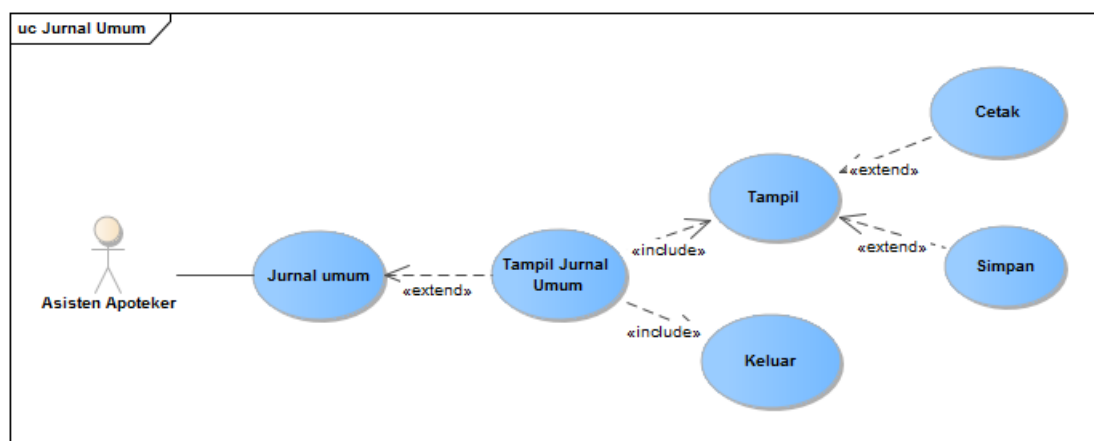
Tabel III.11

Deskripsi Use Case Halaman Utama Bagian Mengakses Laporan Penjualan

Use Case Narative Laporan Penjualan	
Tujuan	Untuk mengelola Laporan penjualan
Deskripsi	Sistem ini digunakan untuk mengola data - data bersangkutan
Skenario Utama	
Aktor	Asisten apoteker
Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik submenu Laporan dan masuk ke bagian Laporan Penjualan
Aktor	Reaksi Sistem

1. Aktor memilih Laporan Penjualan	Akan menampilkan Laporan penjualan
2. Aktor memilih tombol Simpan	Laporan penjualan akan tersimpan
3. Aktor memilih tombol Cetak	Laporan penjualan akan tercetak
4. Aktor memilih tombol Keluar	Laporan penjualan akan Keluar dan kembali ke menu utama
Kondisi Akhir	Sistem akan menampilkan menu Jurnal Umum untuk di akses dan di kelolah oleh Asisten Apoteker.

12. Use Case Mengakses Laporan Jurnal Umum



Gambar III.16

Use Case Diagram Bagian Mengakses Laporan Jurnal Umum

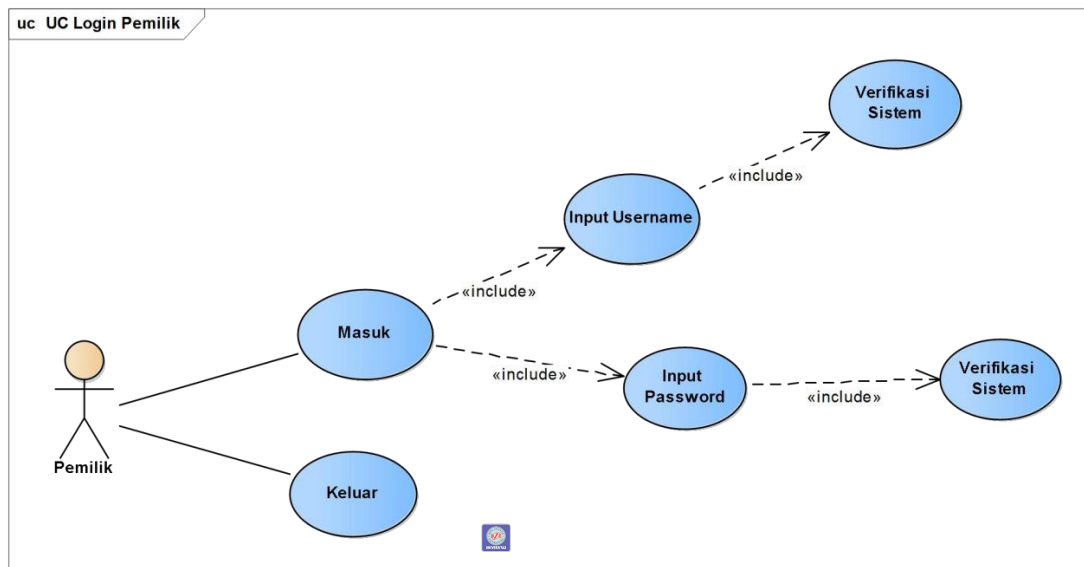
Tabel III.12

Deskripsi Use Case Halaman Utama Bagian Mengakses Laporan Jurnal Umum

<i>Use Case Narrative Jurnal Umum</i>	
Tujuan	Untuk mengelola Jurnal Umum
Deskripsi	Sistem ini digunakan untuk mengolah data_data bersangkutan
Skenario Utama	
Aktor	Asisten Apoteker
Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik submenu Laporan dan masuk ke bagian Jurnal Umum
Aktor	Reaksi Sistem
1. Aktor memilih Jurnal Umum	Akan menampilkan Jurnal Umum
2. Aktor memilih tombol Simpan	Jurnal Umum akan tersimpan
3. Aktor memilih tombol Cetak	Jurnal Umum akan tercetak

4. Aktor memilih tombol Keluar	Jurnal Umum akan Keluar dan kembali ke menu utama
Kondisi Akhir	Sistem akan menampilkan menu Jurnal Umum untuk di akses dan di kelola oleh Asisten Apoteker.

13. Use Case Login Pemilik



Gambar III.17

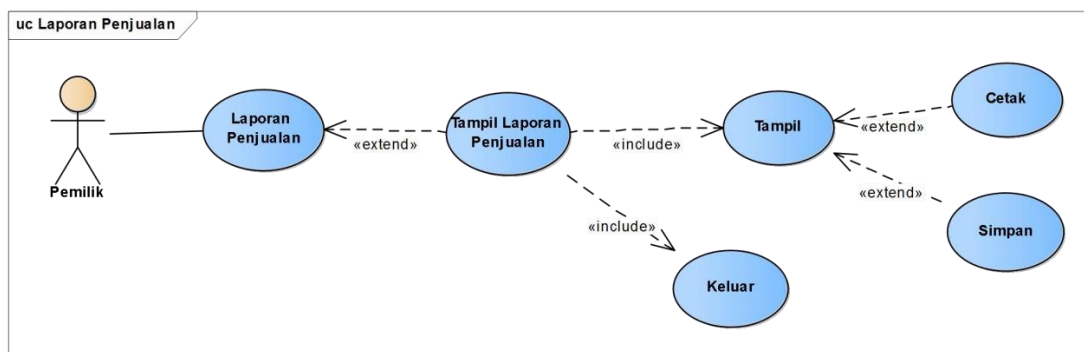
Use Case Diagram Bagian Login Pemilik

Tabel III.13

Deskripsi Use Case Halaman Utama Bagian Login Pemilik

Use Case Narrative Login Pemilik	
Tujuan	Melakukan <i>Login</i>
Deskripsi	Sistem ini digunakan untuk mengakses sistem penjualan
Skenario Utama	
Aktor	Bagian Pemilik
Kondisi Awal	Aktor menggunakan sistem penjualan
Aksti Aktor	Reaksi Sistem
1. Aktor Membuka aplikasi sistem penjualan <i>LOGIN</i>	Akan menampilkan form data master
Kondisi Akhir	Jika semuanya benar, maka sistem akan melakukan kehendak aktor.

14. Use Case Pemilik Mengakses Laporan Penjualan



Gambar III.18

Use Case Diagram Bagian Pemilik Mengakses Laporan Penjualan

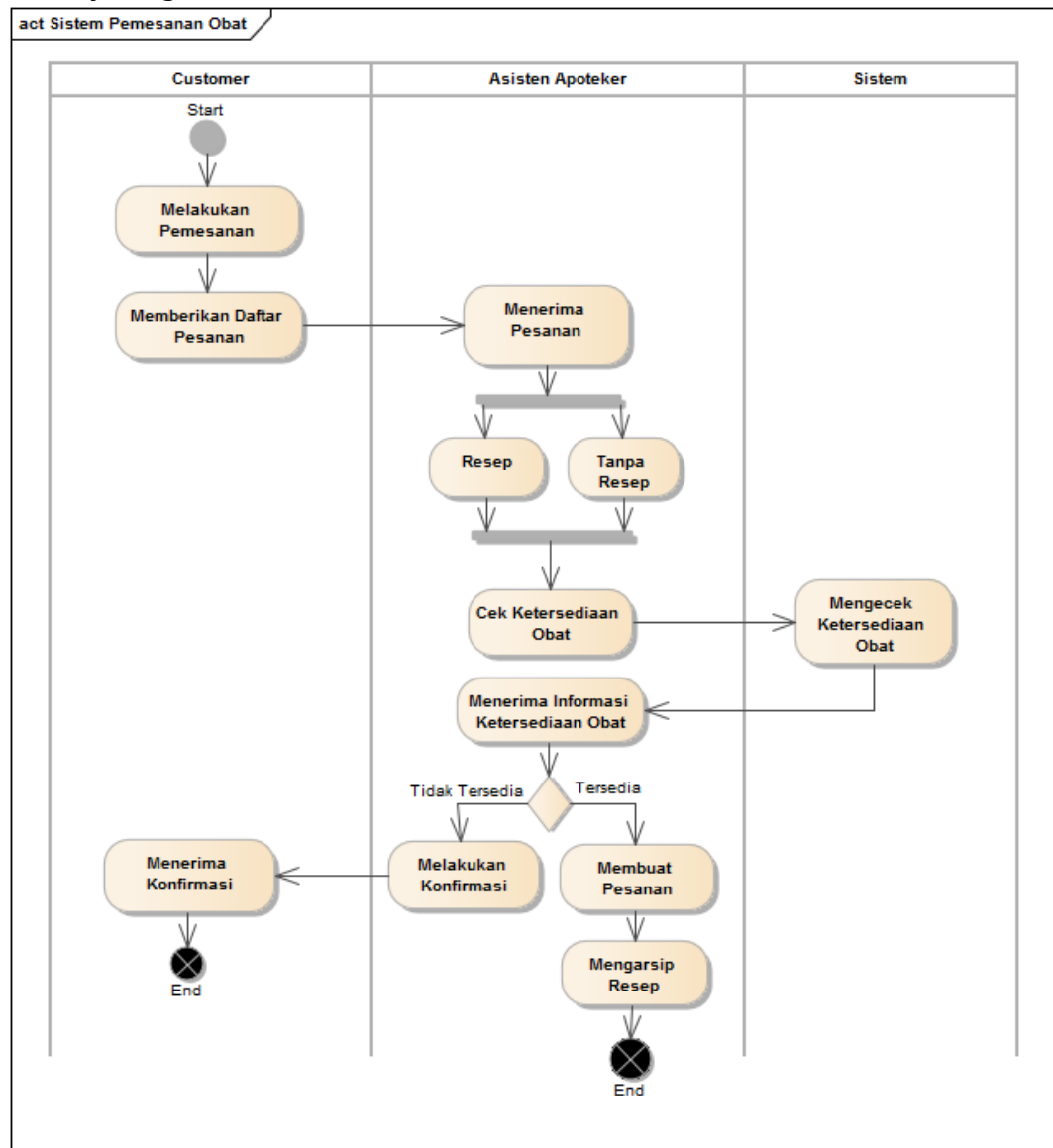
Tabel III.14

Deskripsi Use Case Halaman Utama Bagian Pemilik Mengakses Laporan Penjualan

Use Case Narrative Laporan Penjualan	
Tujuan	Untuk mengelolah Laporan penjualanan
Deskripsi	Sistem ini digunakan untuk mengolah data_data bersangkutan
Skenario Utama	
Aktor	Pemilik
Kondisi Awal	Aktor membuka halaman utama aplikasi penjualan kemudian klik masuk ke bagian Laporan Penjualan
Aktor	Reaksi Sistem
1. Aktor memilih Laporan Penjualan	Akan menampilkan Laporan penjualan
2. Aktor memilih tombol Simpan	Laporan penjualanan kan tersimpan
3. Aktor memilih tombol Cetak	Laporan penjualanan akan tercetak
4. Aktor memilih tombol Keluar	Laporan penjualanan akan Keluar dan kembali ke menu utama
Kondisi Akhir	Sistem akan menampilkan menu Laporan Penjualan untuk di akses oleh pemilik .

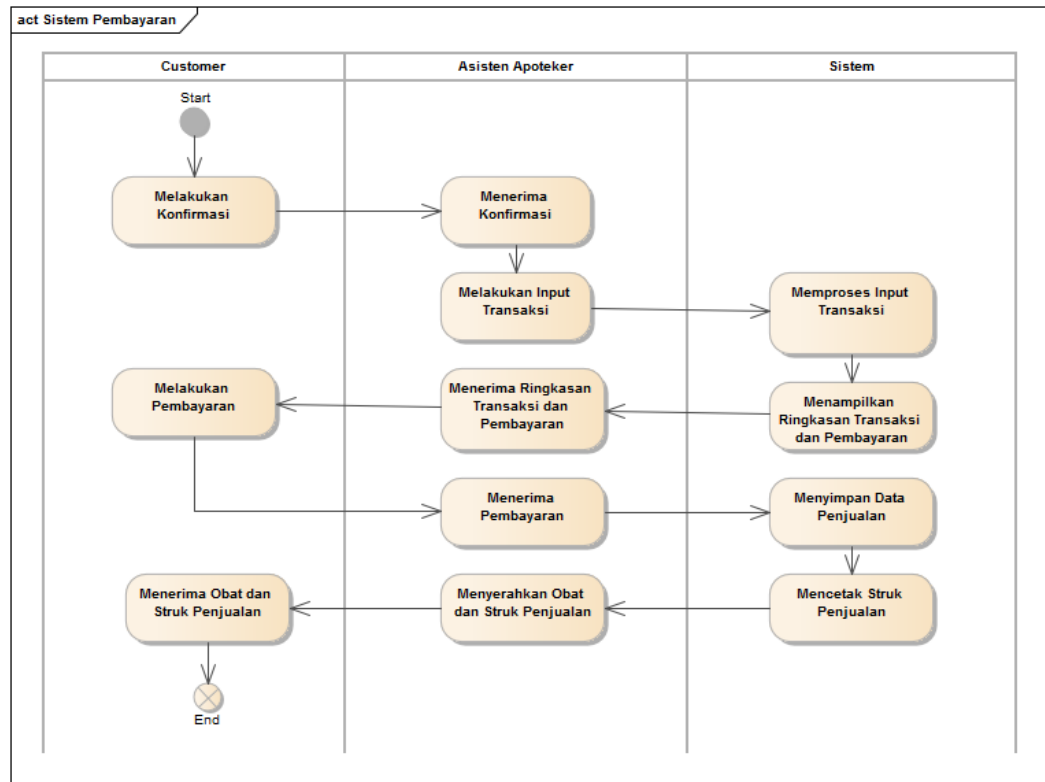
3.3.3. Activity Diagram

1. Activity Diagram Sistem Pemesanan Obat



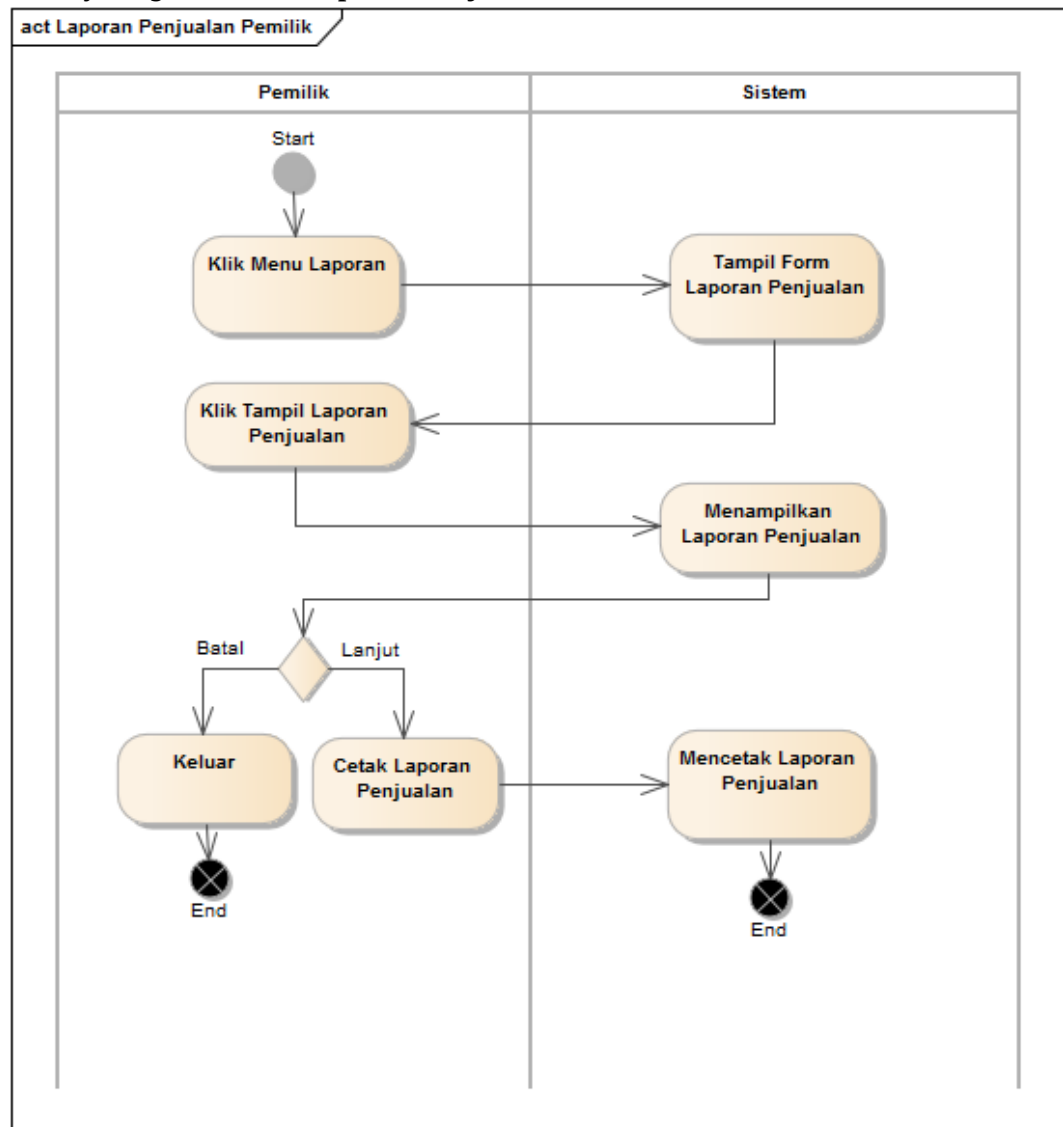
Gambar III.19
Activity Diagram Sistem Pemesanan Obat

2. Activity Diagram Sistem Pembayaran



Gambar III.20
Activity Diagram Sistem Pembayaran

3. Activity Diagram Akses Laporan Penjualan



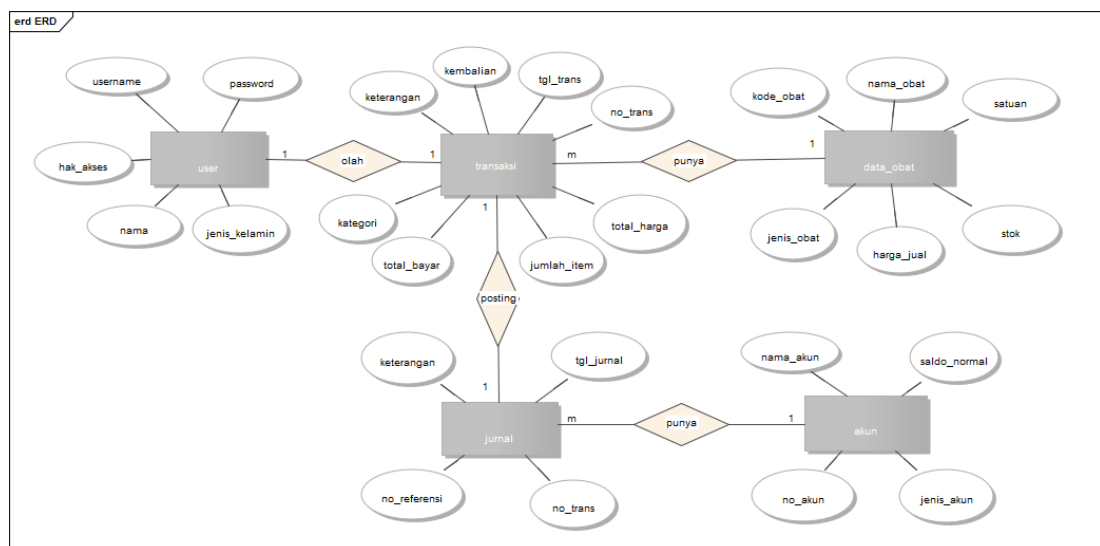
Gambar III.21

Activity Diagram Akses Laporan Penjualan

3.4. Desain

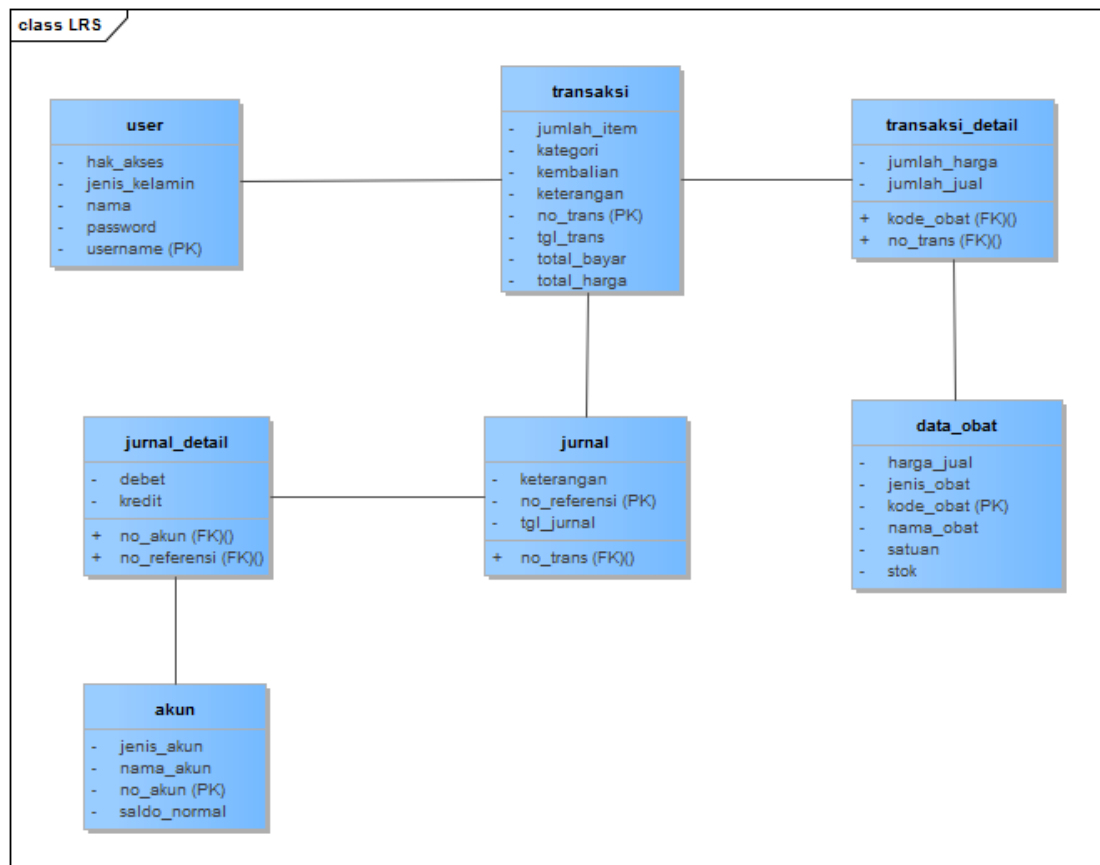
Desain adalah suatu sistem yang berlaku untuk segala jenis perancangan yang mana titik beratnya dilakukan dengan melihat segala sesuatu persoalan tidak secara terpisah atau tersendiri, namun sebagai suatu kesatuan dimana satu masalah dengan lainnya saling terkait.

3.4.1. Entity Relationship Diagram (ERD)



Gambar III.22
Entity Relationship Diagram (ERD)

3.4.2. Logical Record Structure (LRS)



Gambar III.23
Logical Record Structure (LRS)

3.4.3. Spesifikasi File

1. Spesifikasi File User

Nama Database : penjualan_apotek

Nama File : user

Akronim : user.myd

Tipe File : File User

Akses File : Random

Panjang Record : 33 Karakter

Field Key : username

Tabel III. 15
Spesifikasi File User

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	hak_akses	hak_akses	varchar	'kasir','pemilik'	
2	jenis_kelamin	jenis_kelamin	varchar	2	
3	nama	Nama	varchar	15	
4	password	Password	varchar	8	
5	username	username	varchar	8	<i>Primary Key</i>

2. Spesifikasi File Transaksi

Nama Database : penjualan_apotek

Nama *File* : transaksi

Akronim : transaksi.myd

Tipe *File* : File Transaksi

Akses *File* : *Random*

Panjang *Record* : 118 Karakter 

Field Key : no_trans

Tabel III. 16
Spesifikasi File Transaksi

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	jumlah_item	jumlah_item	int	3	
2	kategori	Kategori	varchar	20	
3	kembalian	Kembalian	int	10	
4	keterangan	keterangan	varchar	25	
5	no_trans	no_trans	varchar	25	<i>Primary Key</i>
6.	tgl_trans	tgl_trans	varchar	15	
7	total_bayar	total_bayar	int	10	
8.	total_harga	total_harga	int	10	

3. Spesifikasi File Transaksi Detail

Nama Database : penjualan_apotek

Nama *File* : transaksi detail

Akronim : transaksi_detail.myd

Tipe File : File Transaksi

Akses File : *Random*

Panjang Record : 53 Karakter

Field Key : kode-obat, no_trans

Tabel III. 17
Spesifikasi File Transaksi Detail

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	jumlah_harga	jumlah_harga	int	10	
2	jumlah_jual	jumlah_jual	int	3	
3	kode_obat	kode_obat	varchar	15	<i>Foreign Key</i>
4	no_trans	no_trans	varchar	25	<i>Foreign Key</i>

4. Spesifikasi File Data Obat

Nama Database : penjualan_apotek



Nama File : data obat

Akronim : data_obat.myd

Tipe File : File Master

Akses File : *Random*

Panjang Record : 96 Karakter

Field Key : kode_obat

Tabel III. 18
Spesifikasi File Data Obat

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	nama_obat	nama_obat	varchar	50	
2	jenis_obat	jenis_obat	varchar	15	
3	harga_jual	harga_jual	varchar	10	
4	satuan	Satuan	int	3	
5	stok	Stok	int	3	
6	kode_obat	kode_obat	int	15	<i>Primary Key</i>

5. Spesifikasi File Jurnal

Nama Database : penjualan_apotek

Nama *File* : jurnal

Akronim : jurnal.myd

Tipe *File* : File Transaksi

Akses *File* : *Random*

Panjang *Record* : 68 Karakter

Field Key : no_referensi, no_trans

Tabel III. 19
Spesifikasi File Jurnal

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	keterangan	Keterangan	varchar	25	
2	no_referensi	no_referensi	int	10	<i>Primary Key</i>
3	tgl_jurnal	tgl_jurnal	datetime	8	
4	no_trans	no_trans	varchar	25	<i>Foreign Key</i>

6. Spesifikasi File Jurnal Detail

Nama Database : penjualan_apotek

Nama *File* : jurnal detail

Akronim : jurnal_detail.myd

Tipe *File* : File transaksi

Akses *File* : *Random*

Panjang *Record* : 70 Karakter

Field Key : no_akun, no_referensi

Tabel III. 20
Spesifikasi File Jurnal Detail

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	Debet	Debet	int	15	
2	Kredit	Kredit	int	15	
3	no_akun	no_trans	varchar	25	<i>Foreign Key</i>
4	no_referensi	no_referensi	int	15	<i>Foreign Key</i>

7. Spesifikasi File Akun

Nama Database : penjualan_apotek

Nama *File* : akun

Akronim : akun.myd

Tipe *File* : File Transaksi

Akses *File* : *Random*

Panjang *Record* : 75 Karakter

Field Key : no_akun

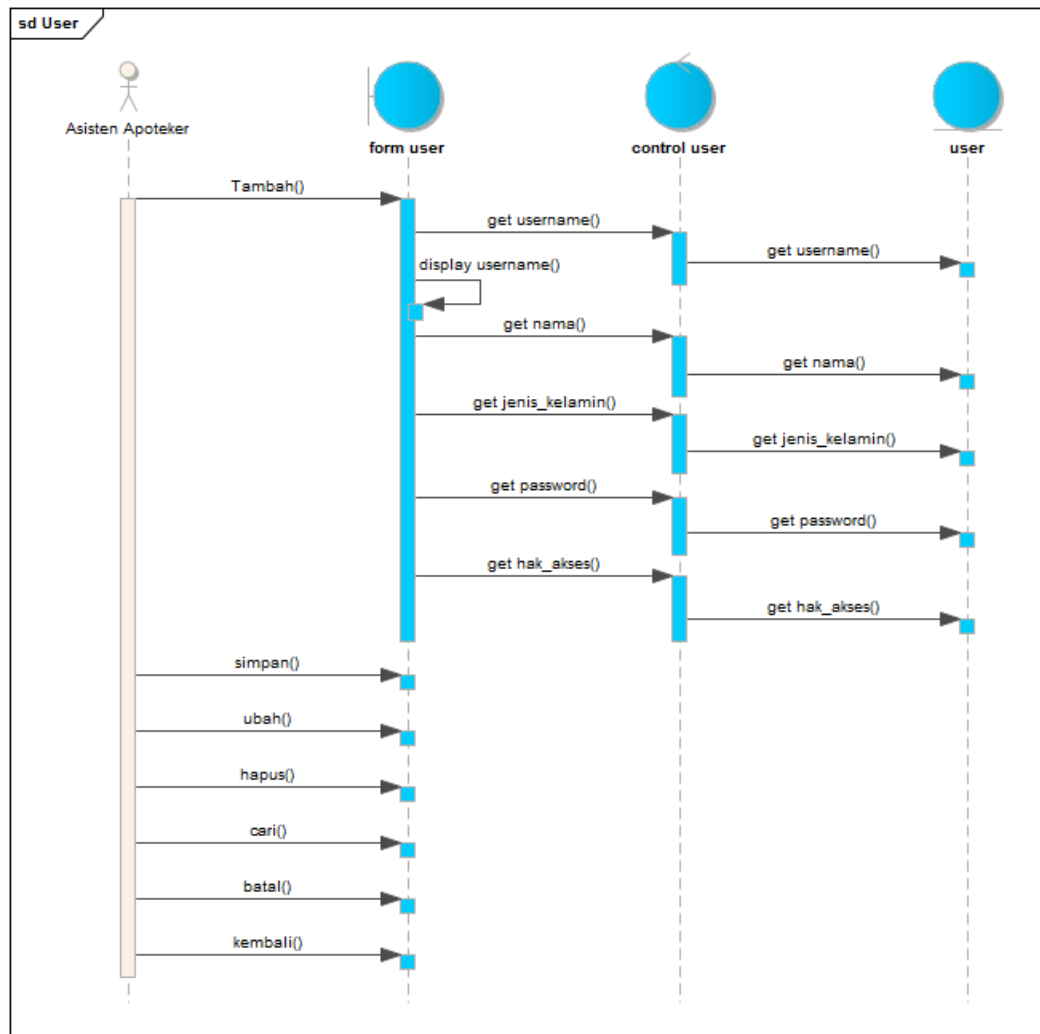


Tabel III. 21
Spesifikasi File Akun

No	Elemen Data	Nama Field	Type	Size	Keterangan
1	jenis_akun	jenis_akun	varchar	25	
2	nama_akun	nama_akun	varchar	25	
3	no_akun	no_akun	varchar	10	<i>Primary Key</i>
4	saldo_normal	saldo_normal	int	15	

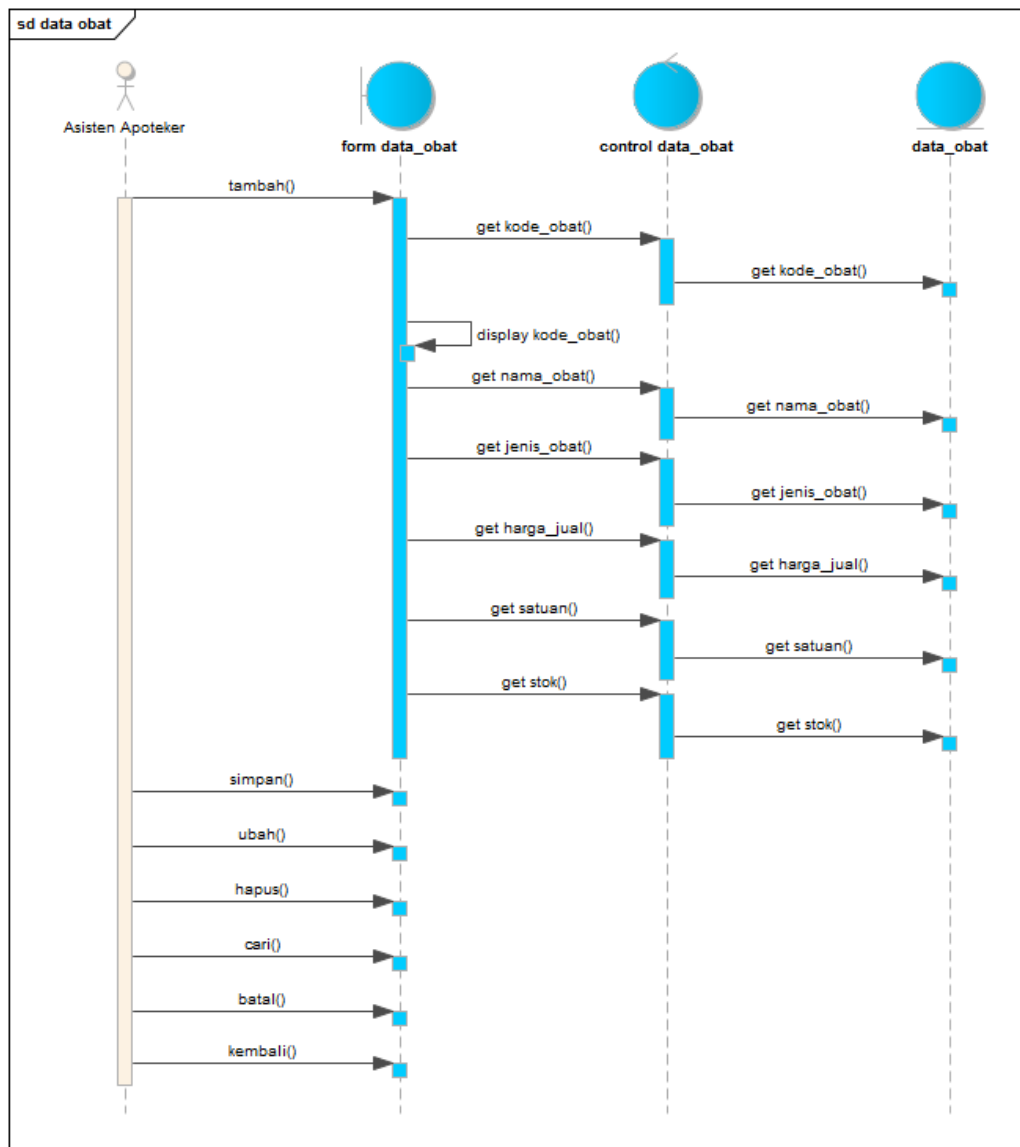
3.4.4. Sequence Diagram

1. Sequence Diagram Data User



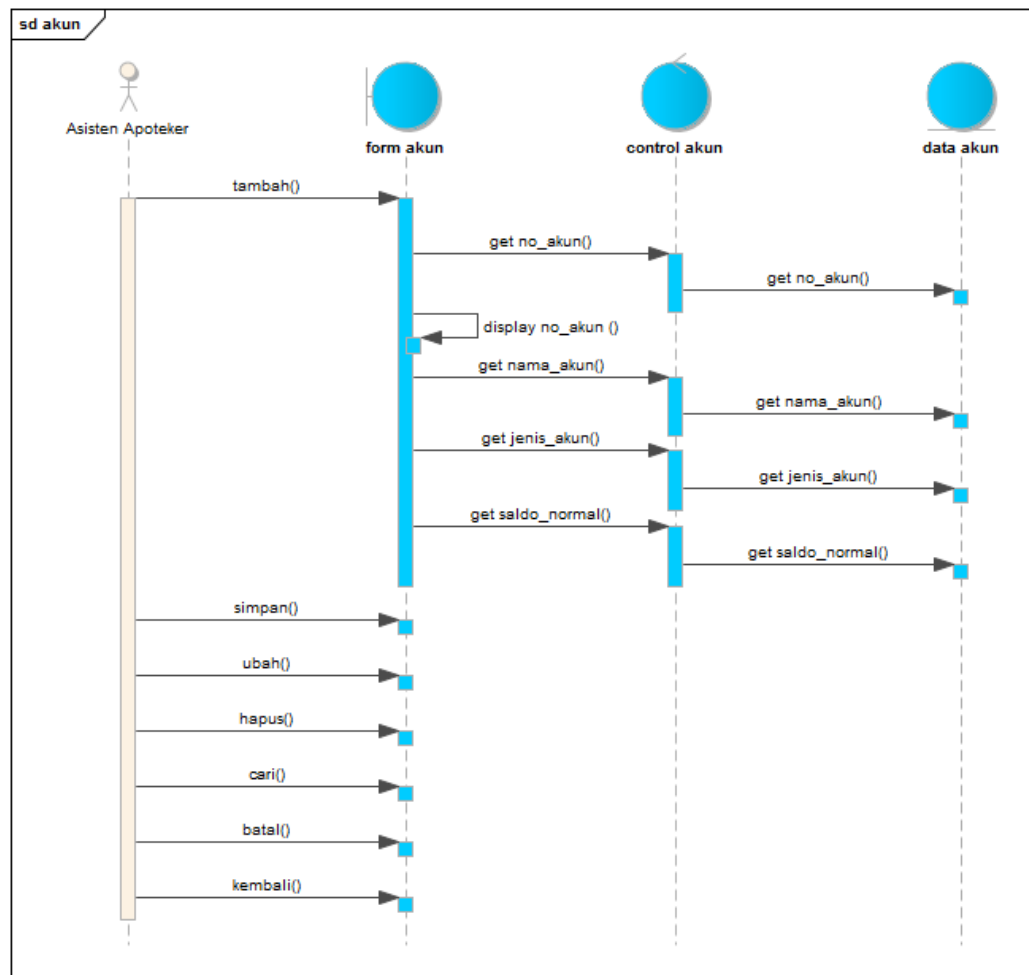
Gambar III.24
Sequence Diagram Data User

2. Sequence Diagram Data Obat



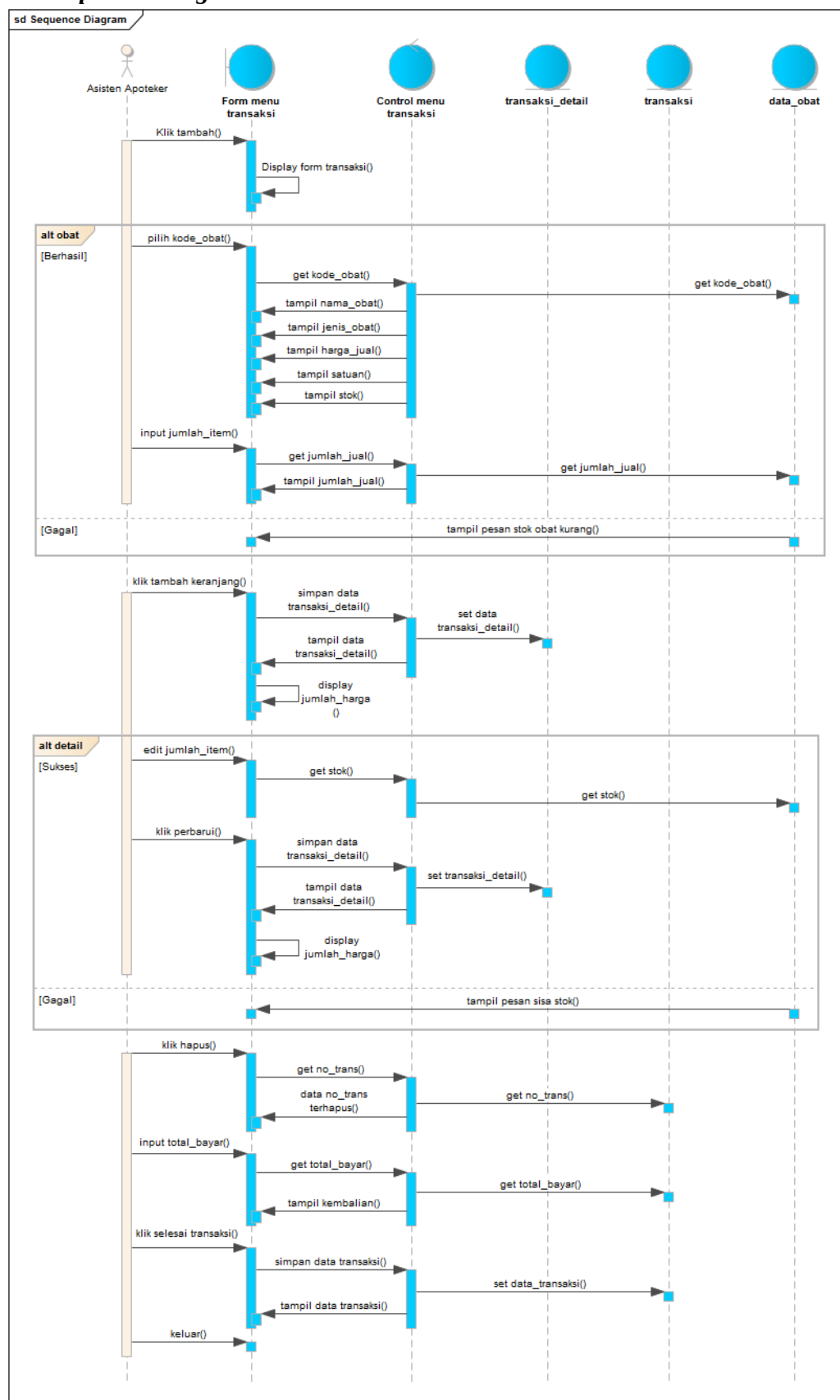
Gambar III.25
Sequence Diagram Data Obat

3. Sequence Diagram Data Akun



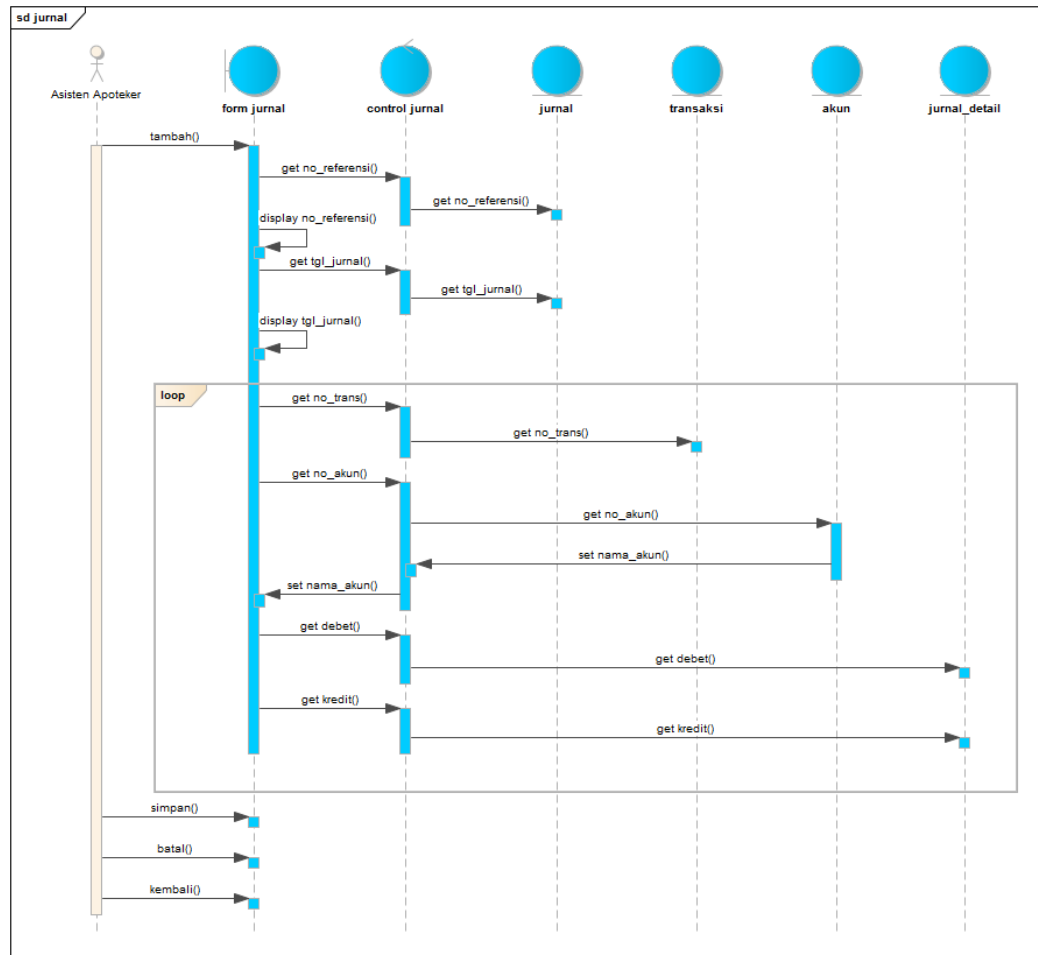
Gambar III.26
Sequence Diagram Data Akun

4. Sequence Diagram Transaksi



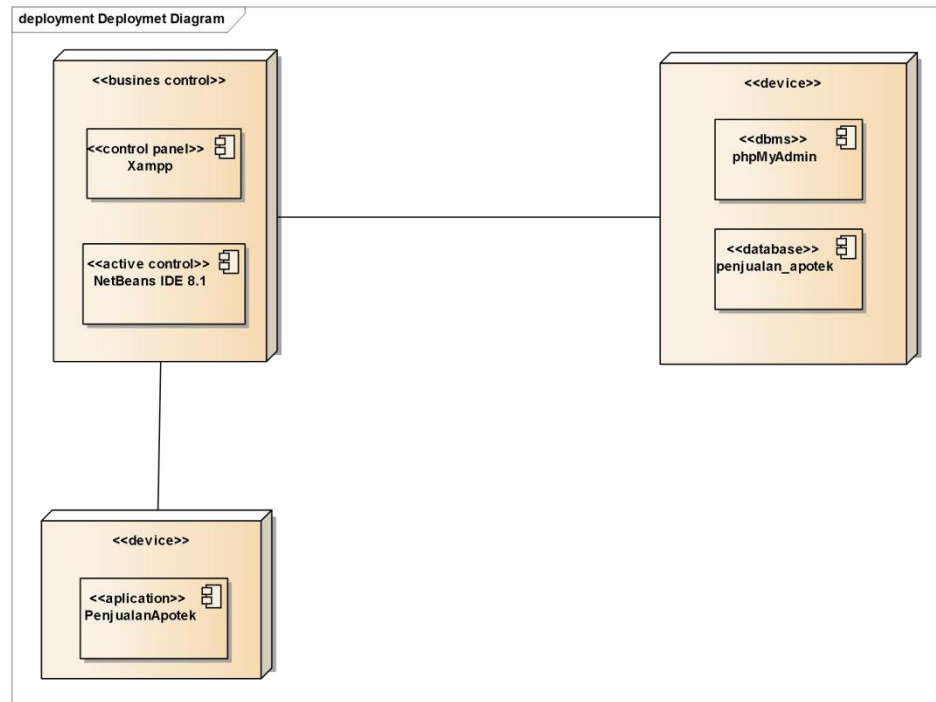
Gambar III.27
Sequence Diagram Transaksi

5. Sequence Diagram Jurnal



Gambar III.28
Sequence Diagram Jurnal

3.4.5. Deployment Diagram



Gambar III.29
Deployment Diagram

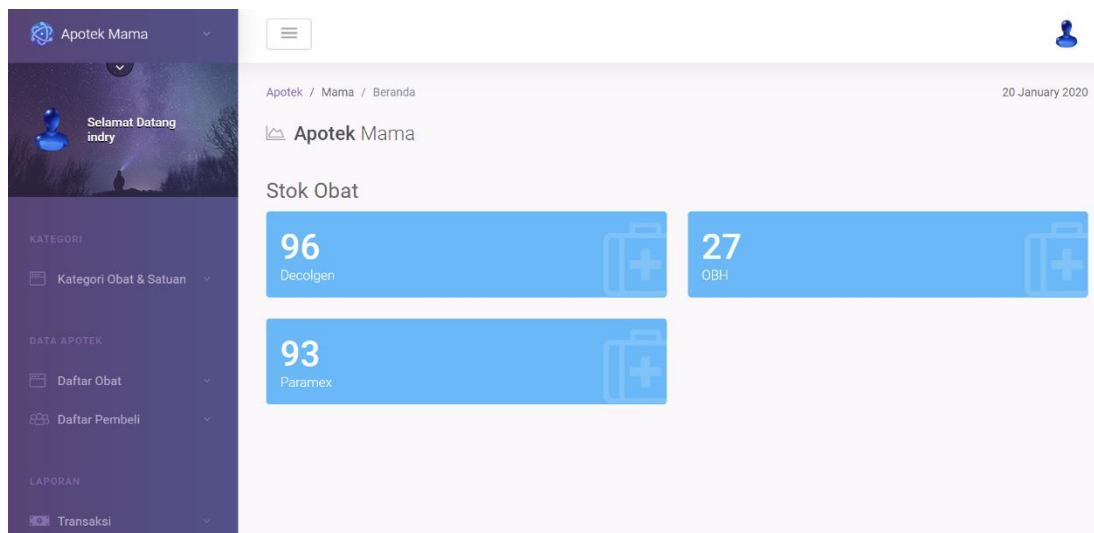


3.4.6. User Interface

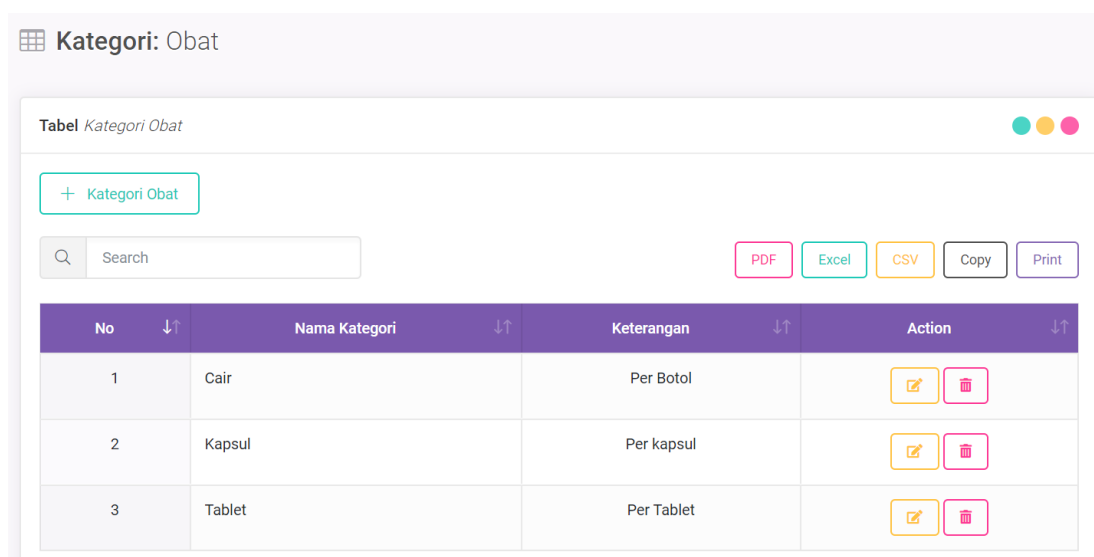
Pengguna diwajibkan untuk memasukkan username dan password kemudian menekan tombol login. Sistem akan melakukan autentikasi terhadap data yang dimasukkan dan menampilkan halaman selanjutnya.

Gambar III.30
User Interface Tampilan Form Login

Pada halaman form ini admin dapat memasukan kode admin, nama admin, password dan level ke dalam database. Data yang telah tersimpan akan otomatis ditampilkan pada tabel list di bawahnya. Hasil yang ditampilkan pada tabel dapat diolah.



Gambar III.31
User Interface Tampilan Form Menu Utama Halaman Asisten Apoteker



Gambar III.32
User Interface Tampilan Form Menu Kategori Obat

 **Kategori: Satuan**

Tabel *Kategori Satuan*

[+ Kategori Satuan](#)

[PDF](#) [Excel](#) [CSV](#) [Copy](#) [Print](#)

No ↓↑	Nama Kategori ↓↑	katsatuan ↓↑	Action ↓↑
1	Botol	Per Botol	Edit Delete
2	Pak	Per pak	Edit Delete

Showing 1 to 2 of 2 entries

[<](#) [1](#) [>](#)

Gambar III.33
User Interface Tampilan Form Kategori Satuan

 **Daftar: Obat**

Tabel *Daftar Obat*

[+ Daftar Obat](#)

[PDF](#) [Excel](#) [CSV](#) [Copy](#) [Print](#)

No ↓	Kode Obat ↓↑	Nama Obat ↓↑	Kategori Obat ↓↑	Kategori Satuan ↓↑	Harga Jual ↓↑	Harga Beli ↓↑	Stok ↓↑	Total Pembelian ↓↑	Tgl Kadaluarsa ↓↑
+ 1	Y9NOc	Paramex	Tablet	Pak	Rp. 3,000,00	Rp. 1,000,00	93	Rp. 93,000,00	18-01-2020
+ 2	cgUjr	Decolgen	Tablet	Pak	Rp. 8,000,00	Rp. 700,00	96	Rp. 67,200,00	18-01-2020
+ 3	CVSxk	OBH	Tablet	Pak	Rp. 5,000,00	Rp. 450,00	27	Rp. 12,150,00	31-01-2020

Gambar III.34
User Interface Tampilan Form Daftar Obat

[+ Daftar Pembeli](#)

Search
 [PDF](#)
[Excel](#)
[CSV](#)
[Copy](#)
[Print](#)

No ↓↑	Nama Pembeli ↓↑	Umur ↓↑	No Telp ↓↑	Jenis Kelamin ↓↑	Alamat ↓↑	Action ↓↑
1	Usep	50	085612341313	Laki-laki	Kelapa Dua	Edit Delete
2	Mia	20	085678905432	Perempuan	Citayam	Edit Delete
3	Sri	35	081254367851	Perempuan	GDC	Edit Delete
4	Sarah	22	082154346709	Perempuan	Cimanggis	Edit Delete
5	Arifin	45	085810243421	Laki-laki	Margonda	Edit Delete

Gambar III.35
User Interface Tampilan Form Daftar Pembeli

[+ Daftar Transaksi](#)

Search
 [PDF](#)
[Excel](#)
[CSV](#)
[Copy](#)
[Print](#)

No ↓↑	Nama Pembeli ↓↑	Nama Obat ↓↑	Qty ↓↑	Total Harga ↓↑	Tanggal Transaksi ↓↑	Action ↓↑
1	Sarah	OBH	4	Rp 20,000,00	19 January 2020	Delete
2	Sarah	Paramex	3	Rp 9,000,00	18 January 2020	Delete
3	Sarah	OBH	11	Rp 55,000,00	19 January 2020	Delete
4	Arifin	Paramex	1	Rp 3,000,00	18 January 2020	Delete
5	Arifin	OBH	1	Rp 5,000,00	14 January 2020	Delete
6	Arifin	OBH	2	Rp 10,000,00	01 January 2020	Delete
7	Arifin	Paramex	3	Rp 9,000,00	18 January 2020	Delete

Gambar III.36
User Interface Tampilan Form Daftar Transaksi

+ Struk Pembelian

PDF Excel CSV Copy Print

No ↓↑	Nama Pembeli ↓↑	Total Pembayaran ↓↑	Jumlah Pembayaran ↓↑	Kembalian Pembayaran ↓↑	Tgl Pembayaran ↓↑	Tgl transaksi ↓↑	Action ↓↑
1	Sarah	Rp 20,000,00	Rp 50,000,00	Rp 30,000,00	21 January 2020	21 January 2020	  
2	Usep	Rp 18,000,00	Rp 50,000,00	Rp 32,000,00	21 January 2020	21 January 2020	  
3	Mia	Rp 40,000,00	Rp 100,000,00	Rp 60,000,00	21 January 2020	21 January 2020	  
4	Sri	Rp 40,000,00	Rp 50,000,00	Rp 10,000,00	21 January 2020	21 January 2020	  
5	Arifin	Rp 50,000,00	Rp 100,000,00	Rp 50,000,00	21 January 2020	21 January 2020	  

Gambar III.37
User Interface Tampilan Form Pembayaran



Gambar III.38
User Interface Tampilan Laporan Jurnal Umum

+ Jurnal

PDF Excel CSV Copy Print

No ↓↑	Kode Akun ↓↑	Keterangan ↓↑	Debet ↓↑	Kredit ↓↑	Action ↓↑
1	11100	Kas	Rp. 4,000, 00	Rp. 0, 00	
2	40000	Penjualan	Rp. 0, 00	Rp. 4,000, 00	
3	11100	Kas	Rp. 4,000, 00	Rp. 0, 00	
4	40000	Penjualan	Rp. 0, 00	Rp. 62,000, 00	
5	40000	Penjualan	Rp. 0, 00	Rp. 4,000, 00	

Gambar III.39
User Interface Tampilan Laporan Penjualan

+ Laporan Penjualan

Q

PDF
Excel
CSV
Copy
Print

No ↓↑	Dari Tanggal ↓↑	Sampai Tanggal ↓↑	Total Penjualan ↓↑	Tanggal Rekap ↓↑	Action ↓↑
1	2020-01-19	2020-02-23	Rp. 118,000, 00	23 February 2020	🖨 👁 🗑
2	2020-02-21	2020-02-23	Rp. 118,000, 00	23 February 2020	🖨 👁 🗑
3	2020-02-01	2020-02-29	Rp. 4,000, 00	23 February 2020	🖨 👁 🗑

3.5. Implementasi

Implementasi dalam Rancang Bangun Sistem Informasi Penjualan Tunai Pada Apotek Mama Depok ini tentunya memerlukan beberapa spesifikasi komputer agar sistem yang telah dibuat mampu berjalan dengan baik. Spesifikasi komputer yang mendukung dalam aplikasi berbasis dekstop ini adalah :



3.5.1. Code Generation

Form Transaksi

```
@extends('admin.template.app')
```

```
@section('title')
```

```
    Daftar Transaksi
```

```
@endsection
```

```
@section('content')
```

```
<script>
```

```
    function hapus(id) {
```

```
        var swalWithBootstrapButtons = Swal.mixin({
```

```
            {
```

```

customClass:

{
    confirmButton: "btn btn-outline-success",
    cancelButton: "btn btn-outline-danger mr-2"
},
buttonsStyling: false
});

swalWithBootstrapButtons

.fire(
{
    title: "Apakah Kamu Yakin Mau delete?",
    text: "",
    type: "warning", 
    showCancelButton: true,
    confirmButtonText: "Yes, delete it!",
    cancelButtonText: "No, cancel!",
    reverseButtons: true
})

.then(function(result)
{
    if (result.value)
    {
        $('#delete_data_'+id).submit();

        swalWithBootstrapButtons.fire(
            "Deleted!",

```

```

        "Your data has been deleted.",
        "success"
    );
}
else if (
    // Read more about handling dismissals
    result.dismiss === Swal.DismissReason.cancel
)
{
    swalWithBootstrapButtons.fire(
        "Cancelled",
        " :)",
        "error" 
    );
}
});
}

function showUser(str) {

    console.log(str);

    $.ajax({
        url : "{{ route('harga') }}"?q="+str,
        type : "GET",

```

```

// data : {'_token' : csrf_token},
success : function(data) {

    console.log(data);

    // if (data[1] == '0') {

    //     // alert('Data Tidak Ditemukan');

    //     // Swal.fire("Ooops!", "Data tidak ditemukan.",
"success");

    // }else{

    //     // Swal.fire("Berhasil!", "Data Berhasil Ditemukan!",
"success");

    //     

document.getElementById("txtHint").innerHTML=data[0];

    // }

    document.getElementById("txtHint").innerHTML=data;

}

});

}

```

```

</script>

```

```

<script>

```

```

    function startCalc(){

        interval = setInterval("calc()",1);}

```

```

function calc(){

y = document.autoSumForm.harga.value;

z = document.autoSumForm.jml_item.value;


document.autoSumForm.total_harga.value = ( y * z );}

function stopCalc(){

clearInterval(interval);}

</script>

<main id="js-page-content" role="main" class="page-content">

<ol class="breadcrumb page-breadcrumb">

<li class="breadcrumb-item"><a

href="javascript:void(0);">Apotek Mama </a></li>

<li class="breadcrumb-item">Daftar</li>

<li class="position-absolute pos-top pos-right d-none d-sm-
block"><span>{{ date('d F Y')}}</span></li>

</ol>

<div class="subheader">

<h1 class="subheader-title">

<i class='subheader-icon fal fa-table'></i> Daftar: <span

class='fw-300'>Transaksi</span>

</h1>

</div>

<div class="row">

<div class="col-xl-12">

<div id="panel-1" class="panel">

```

```

<div class="panel-hdr">
    <h2>
        Tabel      <span      class="fw-300"><i>Daftar
Transaksi</i></span>
    </h2>
    <div class="panel-toolbar">
        <button class="btn btn-panel" data-action="panel-
collapse" data-toggle="tooltip" data-offset="0,10" data-original-
title="Collapse"></button>
        <button class="btn btn-panel" data-action="panel-
fullscreen" data-toggle="tooltip" data-offset="0,10" data-original-
title="Fullscreen"></button>
        <button class="btn btn-panel" data-action="panel-close"
data-toggle="tooltip" data-offset="0,10" data-original-
title="Close"></button>
    </div>
</div>
<div class="panel-container show">
    <div class="panel-content">
        @include('admin.template.partials._alerts')
        <!-- Button trigger modal -->
        @if(auth()->user()->hakakses == 'Admin' || auth()->user()-
>hakakses == 'Kasir')
            <button type="button" class="btn btn-outline-success
waves-effect waves-themed" data-toggle="modal" data-target="#default-

```

```
example-modal">
```

```
    <span class="fal fa-plus mr-2"></span>
```

```
    Daftar Transaksi
```

```
</button>
```

```
    &nbsp;
```

```
@endif
```

```
<!-- Modal -->
```

```
    <div class="modal fade" id="default-example-modal"
tabindex="-1" role="dialog" aria-hidden="true">
```

```
        <div class="modal-dialog" role="document">
```

```
            <div class="modal-content">
```

```
                <div class="modal-header">
```

```
                    <h4 class="modal-title">
```

```
                        Form Input Daftar Transaksi
```

```
                        {{-- <small class="m-0 text-muted">
```

```
                            Below is a static modal example
```

```
                        </small> --}}
```

```
                    </h4>
```

```
                <button type="button" class="close" data-
dismiss="modal" aria-label="Close">
```

```
                    <span aria-hidden="true"><i class="fal fa-
times"></i></span>
```

```
                </button>
```

```
            </div>
```

```

<form action="{{ route('tsaksi.store') }}"
name='autoSumForm' method="POST" accept-charset="utf-8"
class="was-validated" enctype="multipart/form-data">

    {{ csrf_field() }}

    {{ method_field('POST') }}

<div class="modal-body">

    <div class="form-group">

        <label class="form-label"
for="pembeli_id">Nama Pembeli</label>

        <select name="pembeli_id" id="pembeli_id"
class="form-control is-valid" required>
            <option value>-- Pilih --</option>
            @foreach($pembeli as $beli)
                <option value="{{ $beli->id }}">{{ $beli-
>nama_pembeli}}</option>
            @endforeach
        </select>

        <div class="invalid-feedback">Wajib Di
isi</div>

    </div>

    <div class="form-group">

        <label class="form-label"

```

```

for="obat_id">Nama Obat</label>

        <select    name="obat_id"    id="obat_id"
onchange="showUser(this.value)" class="form-control is-valid" required>

        <option value>-- Pilih --</option>

        @foreach($obat as $obt)

            <option value="{{ $obt->id }}">{{ $obt-
>nama_obat}}</option>

        @endforeach

    </select>

    <div    class="invalid-feedback">Wajib    Di
isi</div>

</div>



    <div class="form-group">

        <label                class="form-label"
for="harga">Harga Satuan</label>

        <div id="txtHint">

            <input    type="number"    class="form-
control is-valid" readonly="">

        </div>

        <div    class="invalid-feedback">Wajib    Di
isi</div>

    </div>

    <div class="form-group">

```

```

<label                                class="form-label"
for="jml_item">Qty</label>

<input    type="number" name="jml_item"
onFocus="startCalc();" onBlur="stopCalc();" id="jml_item" class="form-
control is-valid" required >

<div    class="invalid-feedback">Wajib    Di
isi</div>

</div>

<div class="form-group">

<label                                class="form-label"
for="total_harga">Total Harga</label>

<input type="number" name="total_harga"
id="total_harga" class="form-control is-valid" required readonly=""
onchange='tryNumberFormat(this.form.thirdBox);'>

<div    class="invalid-feedback">Wajib    Di
isi</div>

</div>

</div>

<div class="modal-footer">

<button type="button" class="btn btn-outline-
danger" data-dismiss="modal">

<span class="fal fa-times mr-1"></span>

Close

```

```

        </button>

        <button type="submit" class="btn btn-outline-
success">

            <span class="fal fa-save mr-2"></span> Save

        </button>

    </div>

</form>

</div>

</div>

</div>

<br><br> 

<!-- datatable start -->

<table id="dt-basic-example" class="table table-bordered
table-hover table-striped w-100">

    <thead class="bg-primary-600">

        <tr>

            <th><center>No</center></th>

            <th><center>Nama Pembeli</center></th>

            <th><center>Nama Obat</center></th>

            <th><center>Qty</center></th>

            <th><center>Total Harga</center></th>

```

```

<th><center>Tanggal Transaksi</center></th>

@if(auth()->user()->hakakses == 'Admin' || auth()-
>user()->hakakses == 'Kasir')

<th><center>Action</center></th>

@endif

</tr>

</thead>

<tbody>

@foreach($transaksi as $tsaksi)

<tr>

<td align="center">{{ $no++ }}</td>

<td align="center">{{ $tsaksi->pembeli-
>nama_pembeli }}</td> 

<td align="center">{{ $tsaksi->obat->nama_obat
}}</td>

<td align="center">{{ $tsaksi->jml_item }}</td>

<td align="center">Rp {{
number_format($tsaksi->total_harga) }},00</td>

<td align="center">{{ date('d F Y',
strtotime($tsaksi->tgl_transaksi)) }}</td>

@if(auth()->user()->hakakses == 'Admin' ||
auth()->user()->hakakses == 'Kasir')

<td align="center">

<form action="{{route('tsaksi.destroy',
$tsaksi)}}" method="post" id="delete_data_{{ $tsaksi->id }}">

```

```

        {{ csrf_field() }}

        {{ method_field('DELETE') }}

        {{-- <a href="{{ route('tsaksi.edit', $tsaksi)

    }}">

            <button type="button" class="btn btn-sm
            btn-outline-warning" data-toggle="tooltip" data-placement="top"
            title="" data-original-title="Edit">

                <i class="fa fa-edit"></i>

            </button>

        </a> --}}

        <button type="button" class="btn btn-sm
        btn-outline-danger" data-toggle="tooltip" data-placement="top" title=""
        data-original-title="Delete"  onclick="hapus('{{ $tsaksi->id }}')">

            <i class="fas fa-trash-alt"></i>

        </button>

    </form>

</td>

@endif

</tr>

@endforeach

</tbody>

</table>

<!-- datatable end -->

</div>

</div>

```

</div>

</div>

</div>

</main>

@endsection

3.5.2. Blackbox Testing

Tabel.22
Hasil Pengujian Black Box Testing Form Login

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1.	Nama user dan Password tidak diisi kemudian klik tombol <i>login</i>	Nama user: (kosong)  Password: (kosong)	Sistem akan menolak akses dan menampilkan pesan “ <i>LOGIN GAGAL</i> ”	Sesuai harapan	Valid
2.	Mengetikkan Nama user dan Password tidak diisi atau kosong kemudian klik tombol <i>login</i>	Nama user: feby Password : (kosong)	Sistem akan menolak akses dan menampilkan pesan “ <i>LOGIN GAGAL</i> ”	Sesuai harapan	Valid
3.	Nama user tidak di isi (kosong) dan Password diisi kemudian klik tombol <i>login</i>	Nama user: (kosong) Password: admin	Sistem akan menolak Akses dan menampilkan pesan “ <i>LOGIN GAGAL</i> ”	Sesuai harapan	Valid

4.	Mengetikkan salah satu kondisi salah pada Nama user atau Password kemudian klik tombol <i>login</i>	Nama user: febi (benar) Password: 123456 (salah)	Sistem akan menolak akses akan kembali diarahkan ke halaman login dan menampilkan pesan “ LOGIN GAGAL ”	Sesuai harapan	Valid
5.	Mengetikkan Nama user dan Password dengan data yang benar kemudian klik tombol <i>login</i>	Nama user : febi (benar) Password: 12345 (benar)	Sistem akan menerima akses <i>login</i> dan akan menampilkan pesan “ <i>Login</i> Berhasil! Selamat Datang di Halaman Admin, Sri”.	Sesuai harapan	Valid



Tabel.23
Hasil Pengujian Black Box Testing Form Penjualan

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	Tambah data penjualan dengan mengisi data yang tidak lengkap atau masih ada yang kosong	No Faktur Penjualan : (Kosong) Field Lain : (terisi semua)	Sistem tidak akan menyimpan ketika kolom ada yang kosong, dan hanya menyimpan data yang terisi	Sesuai Harapan	Valid
2	Tambah data penjualan dengan mengisi semua yang ada dengan lengkap	Field : (terisi semua)	Sistem akan menyimpan data dan menampilkan” DATA BERHASIL DISIMPAN ”	Sesuai Harapan	Valid

3	Ubah data penjualan dengan mengubah salah satu data	Memilih salah satu Field diubah	Sistem akan mengubah data dan menampilkan” DATA BERHASIL DIUBAH”	Sesuai Harapan	Valid
4	Hapus data penjualan dengan mengklik salah satu field	Memilih satu transaksi penjualan	Sistem akan menghapus data dan menampilkan” DATA BERHASIL DIHAPUS”	Sesuai Harapan	Valid

Tabel.24
Hasil Pengujian Black Box Testing Form Jurnal

No.	Skenario pengujian	Test case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	Tambah jurnal dengan mengisi data yang tidak lengkap atau masih ada yang kosong	Kode Jurnal: (Kosong) Field Lain : (terisi semua) 	Sistem tidak akan menyimpan ketika kolom ada yang kosong, dan hanya menyimpan data yang terisi	Sesuai Harapan	Valid
2	Tambah data jurnal dengan mengisi semua yang ada dengan lengkap	Field : (terisi semua)	Sistem akan menyimpan data dan menampilkan” DATA BERHASIL DISIMPAN”	Sesuai Harapan	Valid
3	Ubah data jurnal dengan mengubah salah satu data	Memilih salah satu Field diubah	Sistem akan mengubah data dan menampilkan” DATA BERHASIL DIUBAH”	Sesuai Harapan	Valid
4	Hapus data jurnal dengan mengklik	Memilih satu transaksi pembelian	Sistem akan menghapus data dan menampilkan” DATA	Sesuai Harapan	Valid

	salah satu field		BERHASIL DIHAPUS”		
--	------------------	--	-------------------	--	--

3.5.3. *Spesifikasi Hardware dan Software*

Tabel III. 25 Spesifikasi *Hardware dan Software*

Kebutuhan	Keterangan
Sistem	Windows 10 Pro 64-bit
Processor	Intel (R) Core (TM) i3-5005U @ 2.0GHz
RAM	2 GB
Monitor	14”
Software	MS Windows 10 Professional 64-Bit (10.0, Build 16299) Web Browser (Mozilla Firefox), NetBeans IDE 8.1, XAMPP V.1.7.7 dan phpmyadmin

