

BAB III

PEMBAHASAN

3.1. Tinjauan Perusahaan

3.1.1. Sejarah Perusahaan

PD Alfath Diesel adalah sebuah perusahaan dagang yang bergerak di bidang jual beli mesin diesel. Perusahaan ini didirikan oleh bapak H. Rokib pada tahun 1987 yang berlokasi di desa lemah duhur RT/RW 013/004 kecamatan tempuran kabupaten karawang.

Awal mula berdirinya perusahaan ini dimulai dari keinginan pemilik memiliki bisnis sendiri yang dimulai dari menjual mesin diesel tractor bekas dan berlanjut hingga pemilik mendirikan sebuah perusahaan jual beli sepeda motor. Perusahaan PD. Alfath Diesel masih termasuk perusahaan kecil dan berkat ketekunan dan keuletan pemilik perusahaan dalam meningkatkan usahanya, maka perusahaan dapat berkembang dengan baik dengan surat keterangan usaha dari pemerintah desa. PD Alfath Diesel mempunyai Visi dan Misi sebagai berikut:

VISI: Menjadi perusahaan yang terkemuka dan menciptakan nilai bagi pelanggan

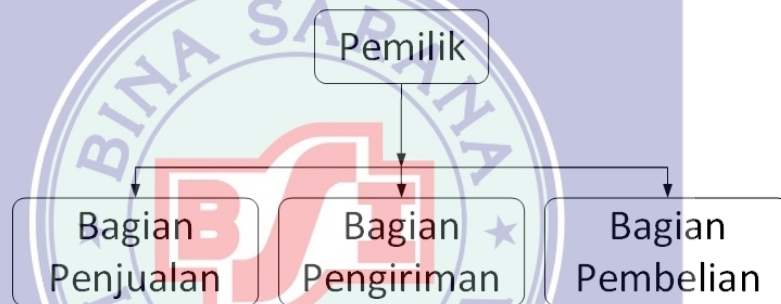
MISI:

1. Memberikan produk yang berkualitas terjamin mutu
2. Menjadi usaha penjualan yang mampu bersaing
3. Menyediakan jasa penyediaan dan pengiriman produk yang cepat dan tepat sesuai dengan kesepakatan

4. Menciptakan lapangan kerja serta dapat mensejahterakan karyawan dan lingkungan sekitar usaha

3.1.2. Struktur Organisasi dan Fungsi

Struktur organisasi merupakan suatu pola hubungan yang diciptakan diantara komponen-komponen atau bagian-bagian yang terdapat didalam outlet. Pola hubungan antara wewenang untuk memerintah dan mengambil keputusan serta tanggung jawab dalam melaksanakan tugas yang dibebankan. Adapun struktur organisasi dan uraian tugas yang terdapat di PD Alfath Diesel adalah sebagai berikut:



Sumber: PD Alfath Diesel

Gambar III.1. Struktur Organisasi

Adapun Fungsi dari masing-masing bagian struktur organisasi sesuai dengan uraian tugasnya adalah sebagai berikut:

1. Pemilik

Menyusun strategi dan visi

- a. Memimpin redaksi
- b. Melihat laporan keuangan
- c. Mengatur segala kebijakan
- d. Memastikan bahwa prinsip tata kelola perusahaan benar-benar diterapkan dengan baik

2. Bagian Keuangan
 - a. Mengontrol aktivitas keuangan atau transaksi keuangan perusahaan
 - b. Membuat laporan mengenai aktivitas keuangan perusahaan
 - c. Melakukan verifikasi terhadap keabsahan dokumen yang diterima
 - d. Melakukan evaluasi keuangan
3. Bagian Penjualan
 - a. Menjaga dan meningkatkan volume penjualan
 - b. Menjalin komunikasi yang baik dengan pelanggan
 - c. Mempertahankan pelanggan yang telah ada
 - d. Memastikan pencapaian target penjualan
 - e. Melaporkan aktivitas penjualan perusahaan kepada pelanggan
4. Bagian Pengiriman
 - a. Mengirimkan barang kepada pelanggan
 - b. Mengirimkan berkas-berkas sebagai penghantar barang
5. Bagian Pembelian
 - a. Mencari dan menganalisa calon penjual yang sesuai dengan barang yang dibutuhkan
 - b. Melakukan negosiasi harga sesuai standar kualitas barang dan memastikan tanggal pengiriman barang
 - c. Melakukan koordinasi dengan pihak penjual mengenai kelengkapan dokumen pendukung barang sesuai standar mutu yang berlaku.

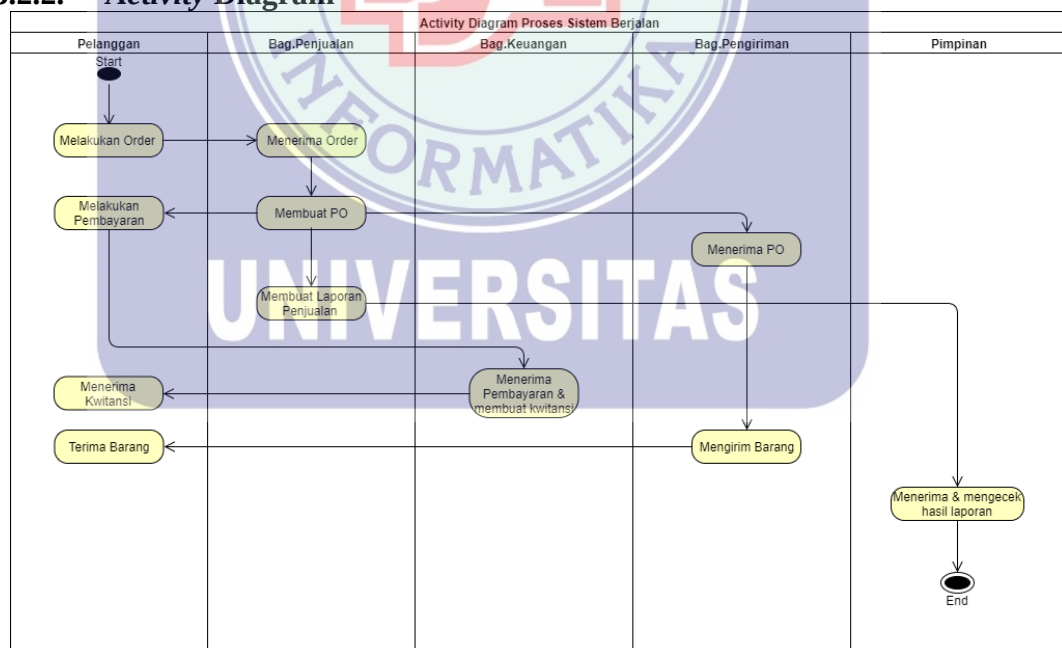
3.2. Tinjauan Kasus

3.2.1. Proses Bisnis Sistem Berjalan

Proses bisnis penjualan kredit PD.Alfath Diesel adalah sebagai berikut:

1. Pelanggan menelpon atau datang langsung ke PD. Alfath Diesel untuk memilih barang yang akan dibeli.
2. Pelanggan memesan diesel yang sudah dipilih spesifikasinya.
3. Pegawai PD Alfath Diesel membuat invoice dan akan di konfirmasi ke pelanggan.
4. Pelanggan melakukan proses pembayaran sesuai invoice yang diterima dari pegawai PD Alfath Diesel.
5. Pegawai PD Alfath Diesel membuat kwitansi sebagai bukti pembayaran.
6. Pegawai PD Alfath Diesel membuat laporan penjualan untuk diserahkan ke atasan.
7. Atasan menerima dan mengecek laporan penjualan diesel.

3.2.2. Activity Diagram



Gambar III.2.
Activity Diagram Proses Bisnis Sistem Berjalan

3.2.3. Dokumen Masukkan

Spesifikasi dokumen masukkan merupakan dokumen yang didapat dari luar PD. Alfath Diesel yang diperlukan dalam sistem penjualan mesin diesel dan selanjutnya dapat diolah sesuai dengan kepentingan sistem penjualan, adapun dokumen yang dibutuhkan dalam dokumen masukan antara lain:

1. Nama Dokumen : Fotokopi KTP
- Fungsi : Identitas Pelanggan dan syarat pemesanan kredit
- Sumber : Pelanggan
- Jumlah : Satu Lembar
- Tujuan : Bagian Penjualan
- Frekuensi : Setiap pengajuan kredit sepeda motor
- Media : Kertas
- Bentuk : Lampiran A.2 (Dok.Sistem Berjalan)

3.2.4. Dokumen Keluaran

Spesifikasi dokumen keluaran merupakan dokumen dari hasil proses masukan sistem pengolahan PD Lajuar Motor Karawang yang diperlukan dalam sistem penjualan kredit sepeda motor dan selanjutnya dapat diolah sesuai dengan kepentingan sistem penjualan kredit, adapun dokumen yang dibutuhkan dalam dokumen keluaran, yaitu:

1. Nama Dokumen : kwitansi
- Fungsi : Sebagai bukti pemesanan mesin diesel
- Sumber : Bagian Penjualan

Jumlah : Dua Lembar

Tujuan : Bagian Pengiriman dan Arsip

Frekuensi : Setiap ada pemesanan penjualan mesin diesel

Media : Kertas

Bentuk : Lampiran B.1 (Dok.Sistem Berjalan)

2. Nama Dokumen : Laporan Penjualan

Fungsi : Untuk merekapitulasi hasil penjualan

Sumber : Bagian Penjualan

Jumlah : Satu Lembar

Tujuan : Untuk mengetahui hasil penjualan

Frekuensi : Setiap ingin membuat laporan rekapan penjualan

Media : Kertas

Bentuk : Lampiran B.2 (Dok.Sistem Berjalan)

3.2.5. Permasalahan Pokok

Permasalahan yang ada pada sistem penjualan mesin diesel PD Alfath Diesel menurut hasil analisa dari penulis, terlihat bahwa permasalahan yang dihadapi adalah pengolahan transaksi penjualan. bukti transaksi yang jarang sekali digunakan semuanya menggunakan manual dan hanya proses negosiasi saja. sehingga mengakibatkan ketidakpuasan pelanggan, ketika mesin diesel tidak bisa berjalan maka pembeli melakukan retur. ketika barang diretur bagian penjualan lupa waktu penjualan dan mesin diesel dianggap tidak bisa diretur atau diperbaiki pihak Alfath Diesel

3.2.6. Pemecahan Masalah

Melihat gambaran umum permasalahan yang dihadapi oleh PD Alfath Diesel, penulis menyarankan bahwa perlu adanya rancang bangun sistem informasi yang dapat membantu mempercepat pengolahan data dan laporan penjualan khususnya bulanan, perlu adanya data pesanan pada surat pesanan untuk mempermudah pembeli mengetahui perincian biaya. sehingga ketika retur barang pihak Alfath bisa segera melakukan tindakan perbaikan.

3.3. Analisis Kebutuhan Software

Berdasarkan proses bisnis pada PD Alfath Diesel maka tahapan berikutnya adalah analisa kebutuhan dimana Bagian Penjualan dapat masuk ke sistem penjualan mesin diesel, didalam penjualan mesin diesel Bagian penjualan dapat menginput semua transaksi penjualan mesin diesel, Bagian Penjualan dapat mencetak laporan penjualan mesin diesel yang nanti akan di berikan kepada pemilik PD Alfath Diesel

3.3.1. Analisis Kebutuhan

- 1 Sales
 - A1. Sales Login
 - A2. Sales Mengakses Menu Utama
 - A3. Sales Megelola Data Pelanggan
 - A4. Sales Megelola Data Penjualan
 - A5. Sales Megelola Kwitansi
 - A6. Sales Megelola Pembayaran
 - A7. Sales Mencetak Pesanan Penjualan
 - A8. Sales Cetak Kwitansi

A9. Sales Cetak Pembayaran

A10. Sales Logout

2. Administrasi

B1. Administrasi Login

B2. Administrasi Membuka Menu Utama

B3. Adminsitrasi Mengelola Data Akun

B4. Administrasi Mengelola Data Pengguna

B5. Administrasi Mengelola Jurnal

B6. Administrasi Logout

3. Pemilik

C1. Pemilik Login

C2. Pemilik Membuka Menu Utama

C3. Pemilik Melihat Laporan Pelanggan

C4. Pemilik Melhat Laporan mesin diesel

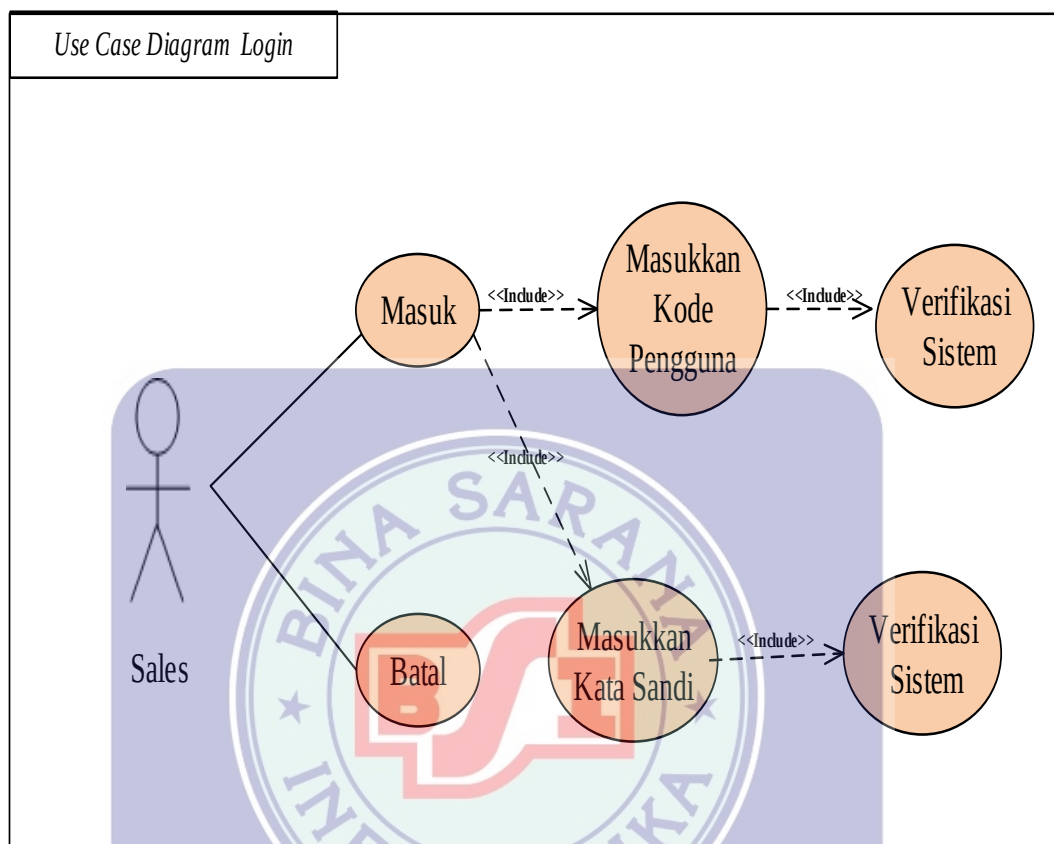
C5. Pemilik Melihat Laporan Penjualan

C6. Pemilik Logout



3.3.2. Usecase Diagram

A1. Use Case Diagram Sales Login



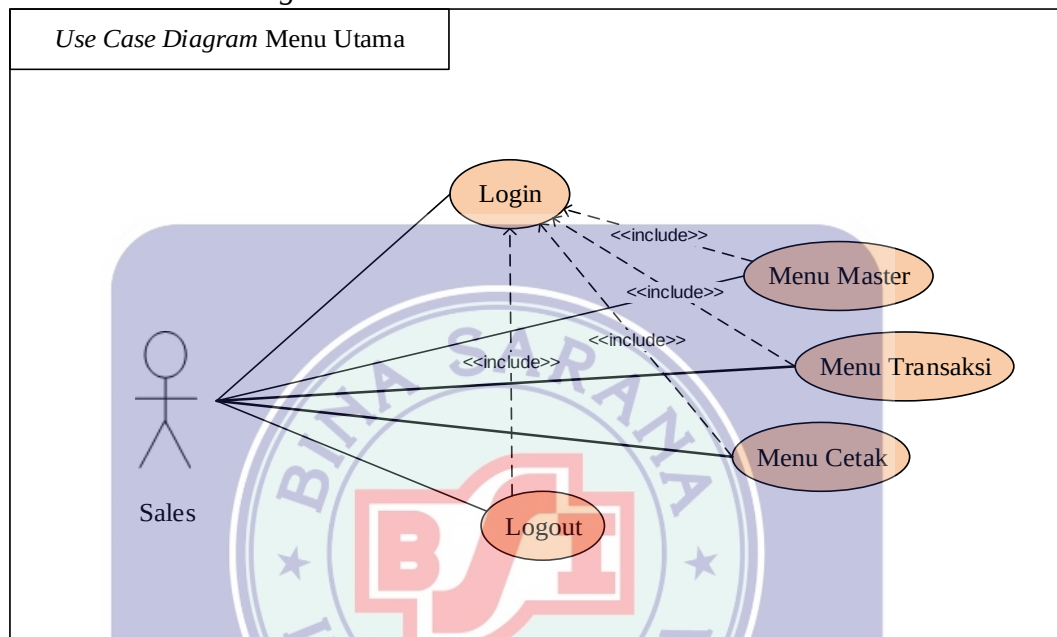
Gambar III.3
Use Case Diagram Sales Login

Tabel III.1
Deskripsi Use Case Diagram Sales Login
Use Case Narrative Login

Tujuan	Melakukan <i>login</i> dan masuk kdalam sistem penjualan mesin diesel
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mengakses sistem penjualan mesin diesel
Skenario Utama	
Aktor	<i>Sales</i>
Kondisi Awal	Aktor membuka aplikasi penjualan mesin diesel
Aksi Aktor	Reaksi Sistem
1. Aktor Memilih tombol <i>Login</i>	Sistem akan menampilkan <i>textbox</i> untuk mengisi kode pengguna dan kata sandi Sistem akan membatalkan proses dan

2. Aktor Memilih tombol Keluar	keluar dari aktivitas
Kondisi Akhir	Jika perintah sesuai maka sistem akan masuk kedalam aplikasi tersebut dan <i>sales</i> dapat melakukan aktivitas didalam sistem.

A2. Use Case Diagram Sales Membuka Menu Utama



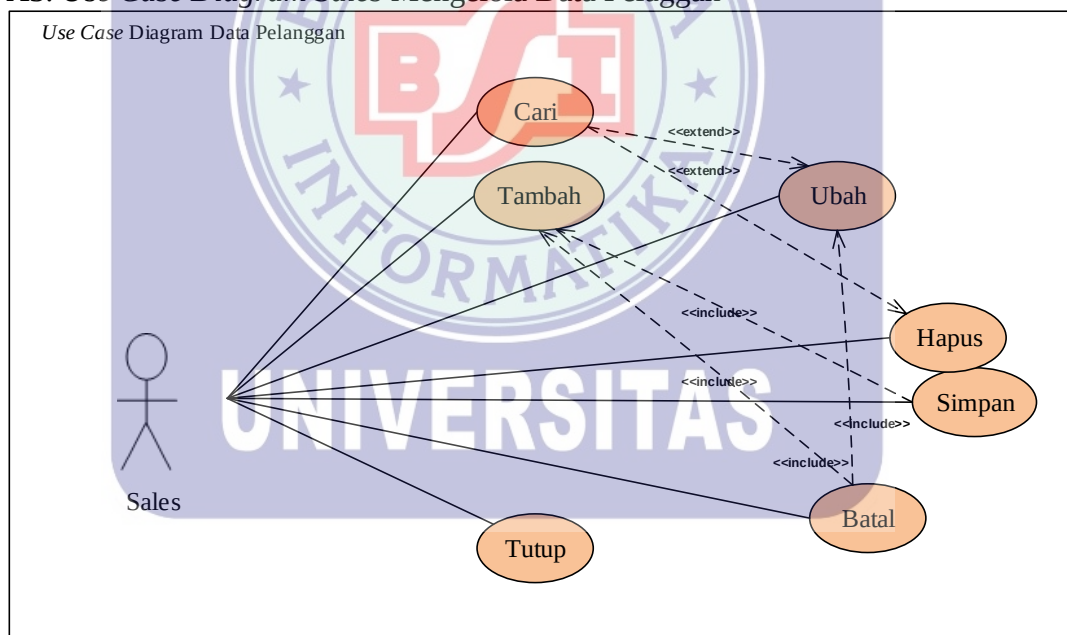
Gambar III.4
Use Case Diagram Sales Membuka Menu Utama

Tabel III.2
Deskripsi Use Case Diagram Sales Membuka Menu Utama

Use Case Narrative Menu Utama	
Tujuan	<i>Sales</i> dapat melakukan pengolahan data yang ada dimenu master, menu transaksi, menu cetak dan <i>logout</i> dari aplikasi
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mengelola sistem penjualan mulai dari <i>input</i> data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	<i>Sales</i>
Kondisi Awal	Aktor membuka aplikasi sistem penjualan

Aksi Aktor	Reaksi Sistem
1. Aktor Memilih tombol Menu Master	Sistem akan menampilkan submenu menu master seperti data mesin diesel dan data pelanggan
2. Aktor Memilih tombol Menu Transaksi	Sistem akan menampilkan submenu menu transaksi seperti pesanan penjualan, kwitansi dan pembayaran
3. Aktor Memilih tombol Menu Cetak	Sistem akan menampilkan submenu cetak seperti cetak pesanan penjualan, cetak kwitansi, cetak surat jalan, dan cetak pembayaran
4. Aktor memilih tombol <i>Logout</i>	Sistem akan keluar dari aplikasi penjualan
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases semua menu utama.

A3. Use Case Diagram Sales Mengelola Data Pelanggan



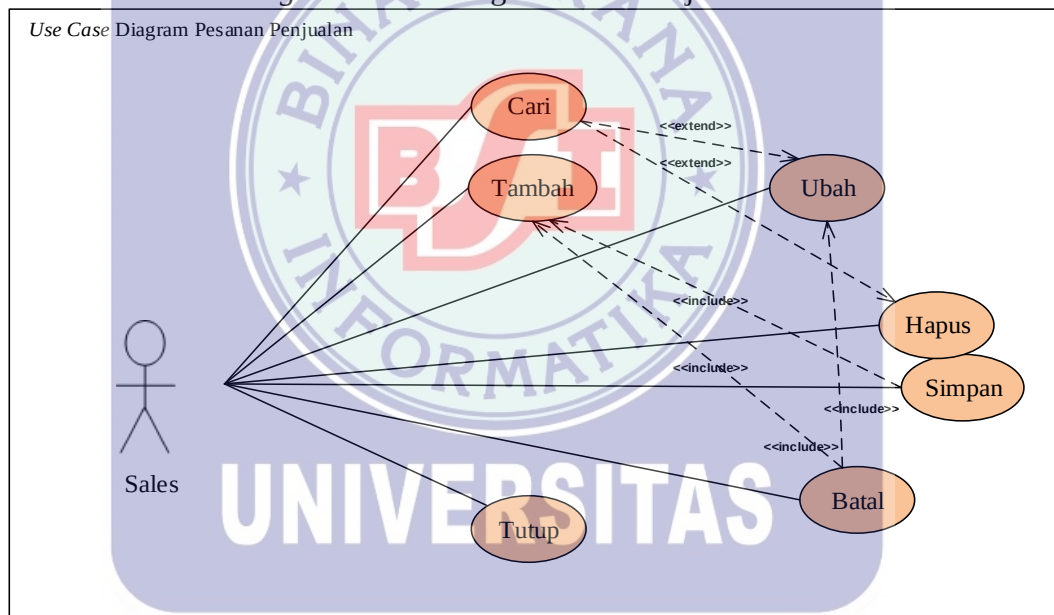
Gambar III.5
Use Case Diagram Sales Mengelola Data Pelanggan

Tabel III.3
Deskripsi Use Case Diagram Sales Mengelola Data Pelanggan

Use Case Narrative Data Pelanggan	
Tujuan	<i>Sales</i> dapat melakukan pengolahan data yang ada disubmenu data pelanggan
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mengelola sistem penjualan mulai dari <i>input</i> data sampai dengan pembuatan

	laporan
Skenario Utama	
Aktor	<i>Sales</i>
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Tambah	Sistem akan menambah data
Aktor Memilih tombol Simpan	Sistem akan menyimpan data
Aktor Memilih tombol Cari	Sistem akan mencari data
Aktor Memilih tombol Ubah	Sistem akan mengubah data
Aktor Memilih tombol Hapus	Sistem akan menghapus data
Aktor Memilih tombol Batal	Sistem akan mmbatalkan proses
Aktor memilih tombol Tutup	Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases sesuai keinginan

A4. Use Case Diagram Sales Mengelola data Penjualan



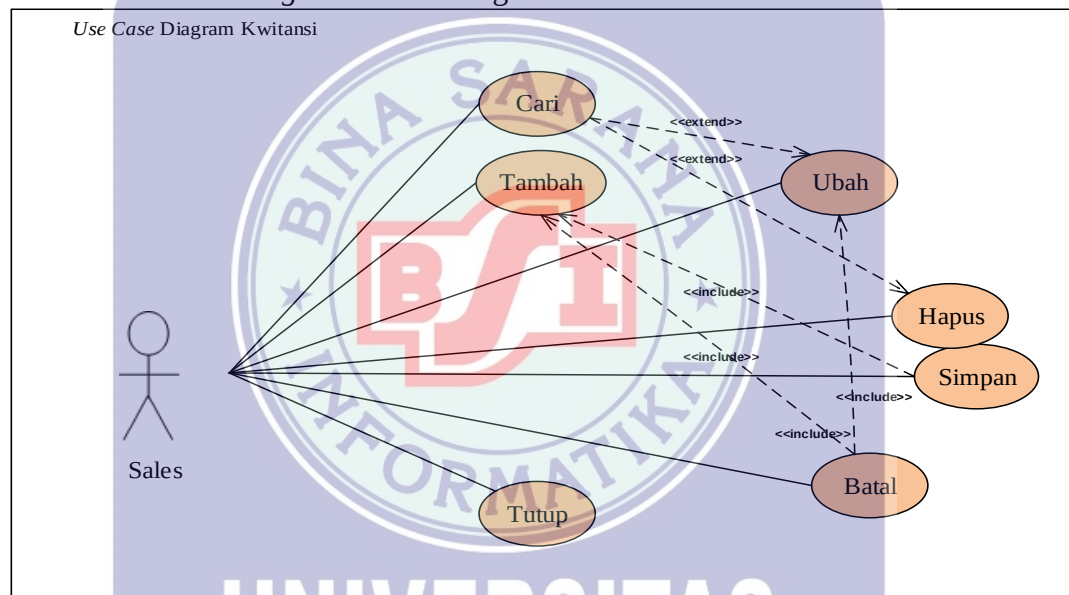
Gambar III.6
Use Case Diagram Sales Mengelola Pesanan Penjualan

Tabel III.4
Deskripsi Use Case Diagram Sales Mengelola Pesanan Penjualan

Use Case Narrative Pesanan penjualan	
Tujuan	<i>Sales</i> dapat melakukan pengolahan data yang ada disubmenu pesanan penjualan
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mengelola sistem penjualan mulai dari <i>input</i> data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	<i>Sales</i>

Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Tambah Aktor Memilih tombol Simpan Aktor Memilih tombol Cari Aktor Memilih tombol Ubah Aktor Memilih tombol Hapus Aktor Memilih tombol Batal Aktor memilih tombol Tutup	Sistem akan menambah data Sistem akan menyimpan data Sistem akan mencari data Sistem akan mengubah data Sistem akan menghapus data Sistem akan mmbatalkan proses Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases sesuai keinginan

A5. Use Case Diagram Sales Mengelola Kwitansi



Gambar III.7
Use Case Diagram Sales Mengelola Kwitansi

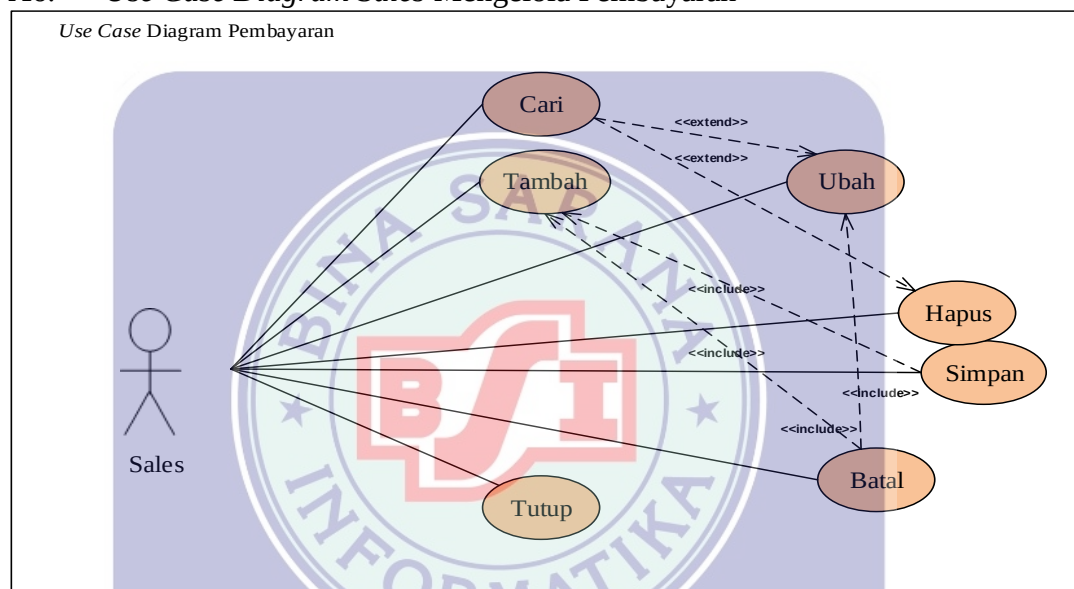
Tabel III.5

Deskripsi Use Case Diagram Sales Mengelola Kwitansi

Use Case Narative Kwitansi	
Tujuan	<i>Sales</i> dapat melakukan pengolahan data yang ada disubmenu kwitansi
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mengelola sistem penjualan mulai dari <i>input</i> data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	<i>Sales</i>
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem

Aktor Memilih tombol Tambah Aktor Memilih tombol Simpan Aktor Memilih tombol Cari Aktor Memilih tombol Ubah Aktor Memilih tombol Hapus Aktor Memilih tombol Batal Aktor memilih tombol Tutup	Sistem akan menambah data Sistem akan menyimpan data Sistem akan mencari data Sistem akan mengubah data Sistem akan menghapus data Sistem akan mmbatalkan proses Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases sesuai keinginan

A6. Use Case Diagram Sales Mengelola Pembayaran



Gambar III.8

Use Case Diagram Sales Mengelola Pembayaran

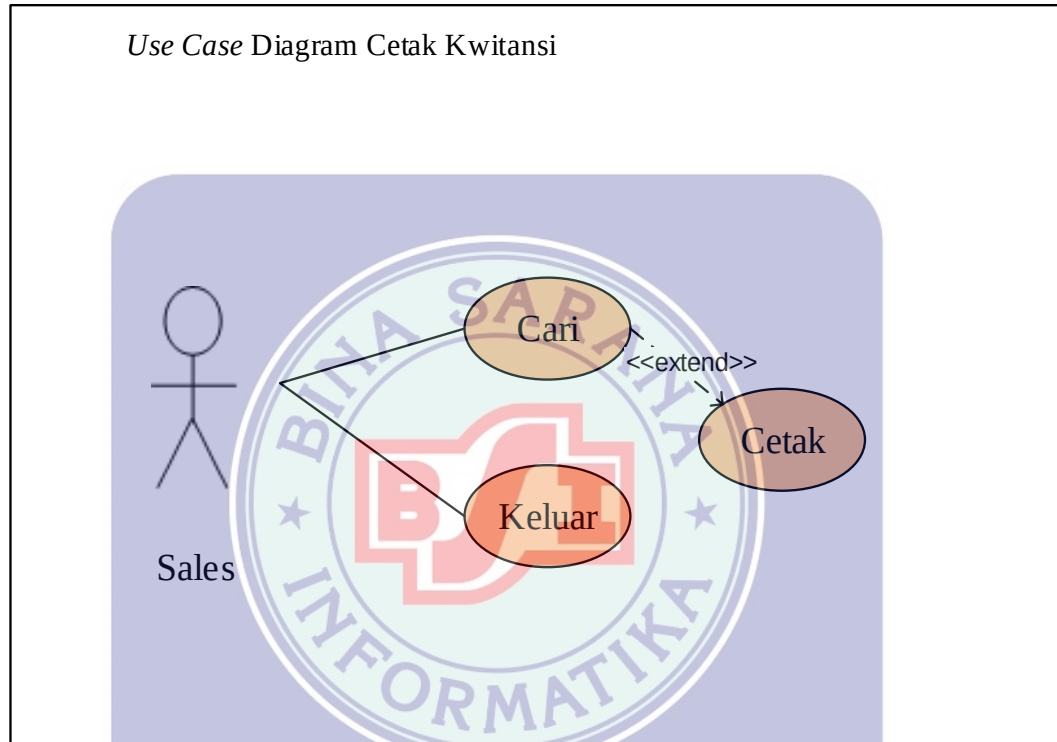
Tabel III.6

Deskripsi Use Case Diagram Sales Mengelola Pembayaran

Use Case Narrative Pembayaran	
Tujuan	<i>Sales</i> dapat melakukan pengolahan data yang ada disubmenu pembayaran
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mengelola sistem penjualan mulai dari <i>input</i> data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	<i>Sales</i>
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Tambah Aktor Memilih tombol Simpan Aktor Memilih tombol Cari	Sistem akan menambah data Sistem akan menyimpan data Sistem akan mencari data

Aktor Memilih tombol Ubah	Sistem akan mengubah data
Aktor Memilih tombol Hapus	Sistem akan menghapus data
Aktor Memilih tombol Batal	Sistem akan mmbatalkan proses
Aktor memilih tombol Tutup	Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases sesuai keinginan

A7. Use Case Diagram Sales Mencetak Kwitansi



Gambar III.9

Use Case Diagram Sales Mencetak Kwitansi

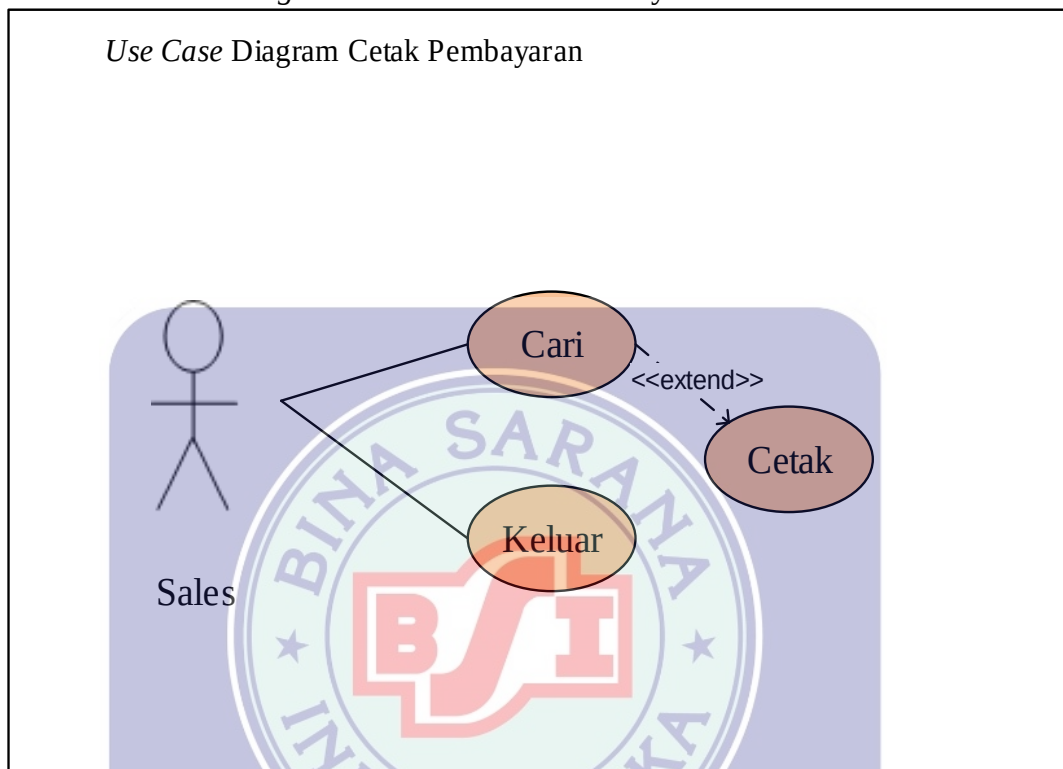
Tabel III.7

Deskripsi Use Case Diagram Sales Mencetak Kwitansi

Use Case Narative Mencetak Kwitansi	
Tujuan	<i>Sales</i> dapat melakukan pengolahan data yang ada disubmenu cetak kwitansi
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mencetak kwitansi
Skenario Utama	
Aktor	<i>Sales</i>
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Cari	Sistem akan mencari data
Aktor Memilih tombol Cetak	Sistem akan mencetak data

Aktor memilih tombol Keluar	Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases sesuai keinginan

A8. Use Case Diagram Sales Mencetak Pembayaran

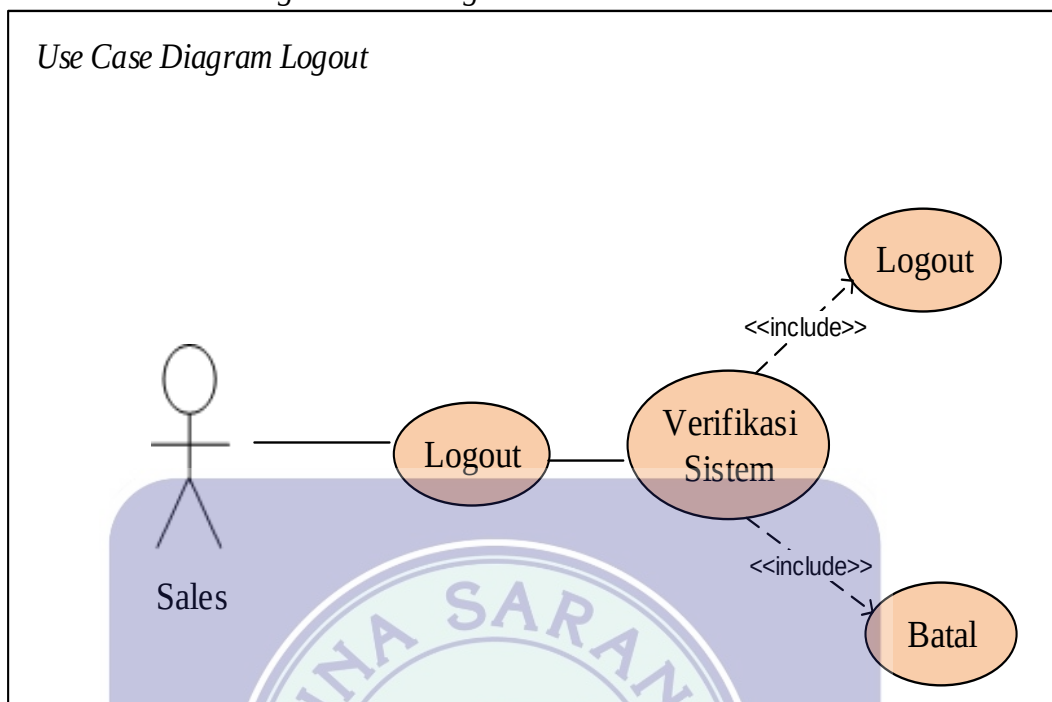


Gambar III.10
Use Case Diagram Sales Mencetak Pembayaran

Tabel III.8
Deskripsi Use Case Diagram Sales Mencetak Pembayaran
Use Case Narrative Mencetak Pembayaran

Tujuan	<i>Sales</i> dapat melakukan pengolahan data yang ada disubmenu cetak pembayaran
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk mencetak pembayaran
Skenario Utama	
Aktor	<i>Sales</i>
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Cari Aktor Memilih tombol Cetak Aktor memilih tombol Keluar	Sistem akan mencari data Sistem akan mencetak data Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases sesuai keinginan

A9. Use Case Diagram Sales Logout

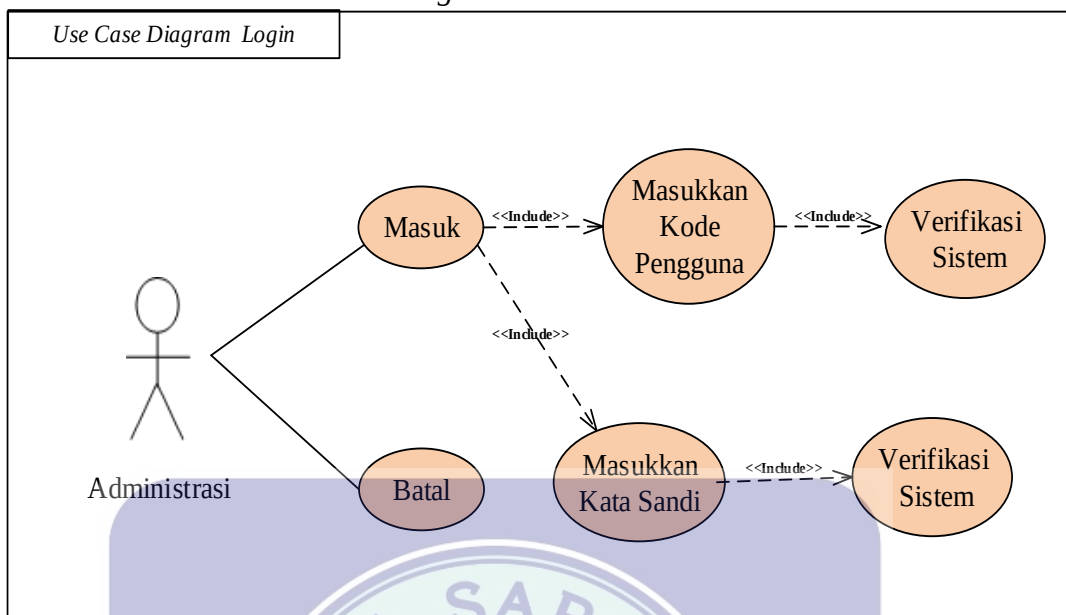


Gambar III.11
Use Case Diagram Sales Logout

Tabel III.9
Deskripsi Use Case Diagram Sales Logout

Use Case Narrative Logout	
Tujuan	Sales dapat melakukan <i>logout</i> dari sistem penjualan
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk keluar dari sistem penjualan
Skenario Utama	
Aktor	Sales
Kondisi Awal	Aktor membuka menu <i>logout</i>
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Logout Aktor Memilih tombol Ya Aktor memilih tombol Tidak	Sistem akan menampilkan pesan Sistem akan keluar dari aplikasi Sistem akan tetap berada di dalam sistem penjualan
Kondisi Akhir	Jika perintah sesuai sales dapat mengkasas sesuai keinginan

B1. Use Case Administrasi Login

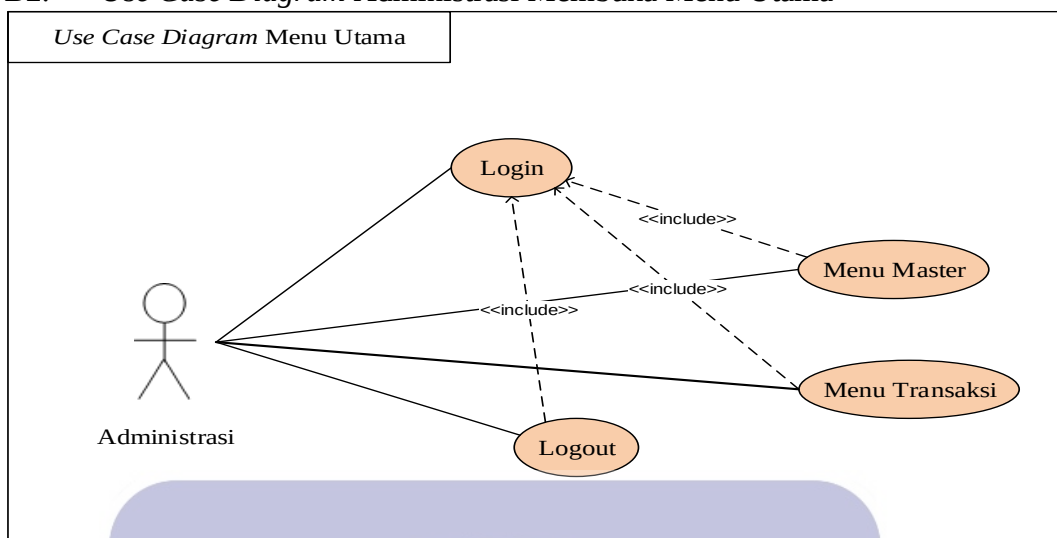


Gambar III.12
Use Case Diagram Administrasi Login

Tabel III.10
Deskripsi Use Case Diagram Administrasi Login
Use Case Narrative Login

Tujuan	Melakukan login dan masuk kdalam sistem penjualan mesin diesel
Deskripsi	Sistem ini memungkinkan administrasi untuk mengakses sistem penjualan mesin diesel
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka aplikasi penjualan mesin diesel
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Login	Sistem akan menampilkan <i>textbox</i> untuk mengisi kode pengguna dan kata sandi
Aktor Memilih tombol Keluar	Sistem akan membatalkan proses dan keluar dari aktivitas
Kondisi Akhir	Jika perintah sesuai maka sistem akan masuk kedalam aplikasi tersebut dan Administrasi dapat melakukan aktivitas didalam sistem.

B2. Use Case Diagram Administrasi Membuka Menu Utama

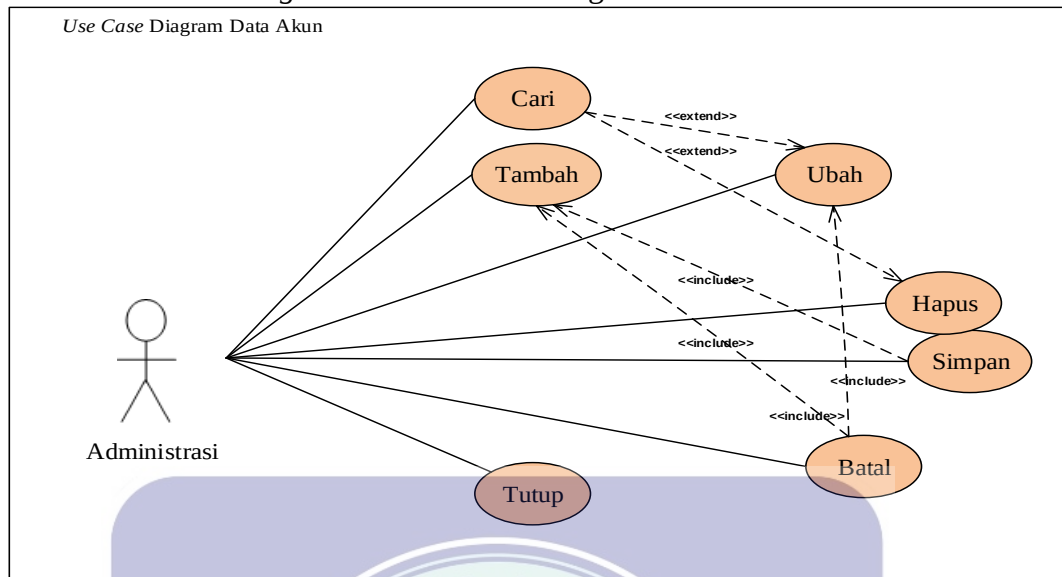


Gambar III.13
Use Case Diagram Administrasi Membuka Menu Utama

Tabel III.11
Deskripsi Use Case Administrasi Membuka Menu Utama

Use Case Narrative Menu Utama	
Tujuan	Administrasi dapat melakukan pengolahan data yang ada dimenu maste, menu transaksi dan <i>logout</i> dari aplikasi
Deskripsi	Sistem ini memungkinkan Administrasi untuk mengelola sistem penjualan mulai dari <i>input</i> data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Menu Master	Sistem akan menampilkan submenu menu master seperti data akun dan data pengguna
Aktor Memilih tombol Menu Transaksi	Sistem akan menampilkan submenu menu transaksi seperti data jurnal
Aktor memilih tombol <i>Logout</i>	Sistem akan keluar dari aplikasi penjualan
Kondisi Akhir	Jika perintah sesuai penjualan dapat mengkases semua menu utama.

B3. Use Case Diagram Administrasi Mengelola Data Akun

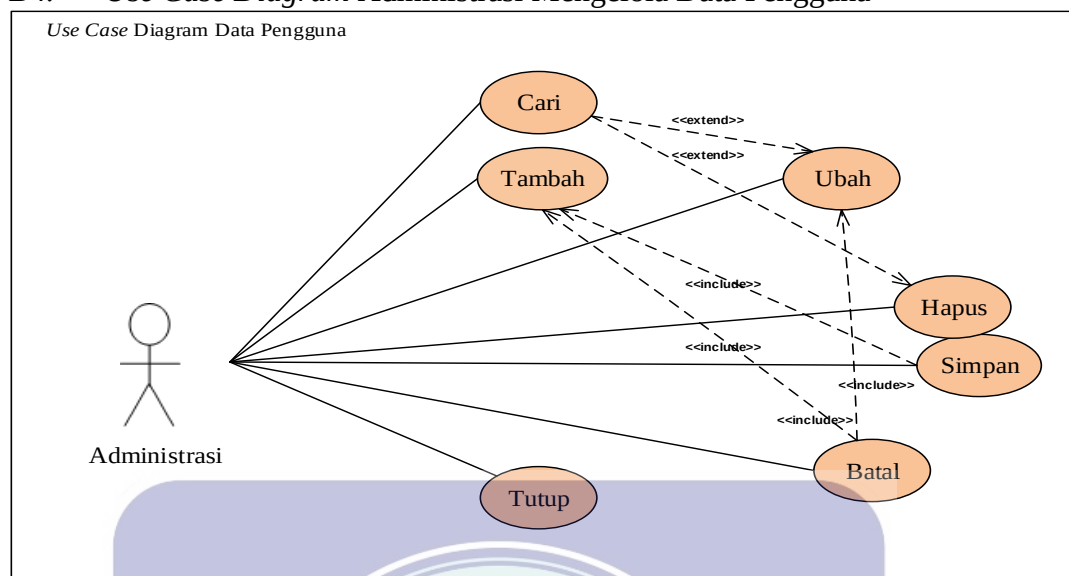


Gambar III.14
Use Case Diagram Administrasi Mengelola Data Akun

Tabel III.12
Deskripsi Use Case Diagram Administrasi Mengelola Data Akun

Use Case Narrative Data Akun	
Tujuan	Administrasi dapat melakukan pengolahan data yang ada disubmenu data akun
Deskripsi	Sistem ini memungkinkan administrasi untuk mengelola sistem penjualan mulai dari input data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Tambah Aktor Memilih tombol Simpan Aktor Memilih tombol Cari Aktor Memilih tombol Ubah Aktor Memilih tombol Hapus Aktor Memilih tombol Batal Aktor memilih tombol Tutup	Sistem akan menambah data Sistem akan menyimpan data Sistem akan mencari data Sistem akan mengubah data Sistem akan menghapus data Sistem akan mmbatalkan proses Sistem akan keluar dari form
Kondisi Akhir	Jika perintah sesuai administrasi dapat mengkases sesuai keinginan

B4. Use Case Diagram Administrasi Mengelola Data Pengguna

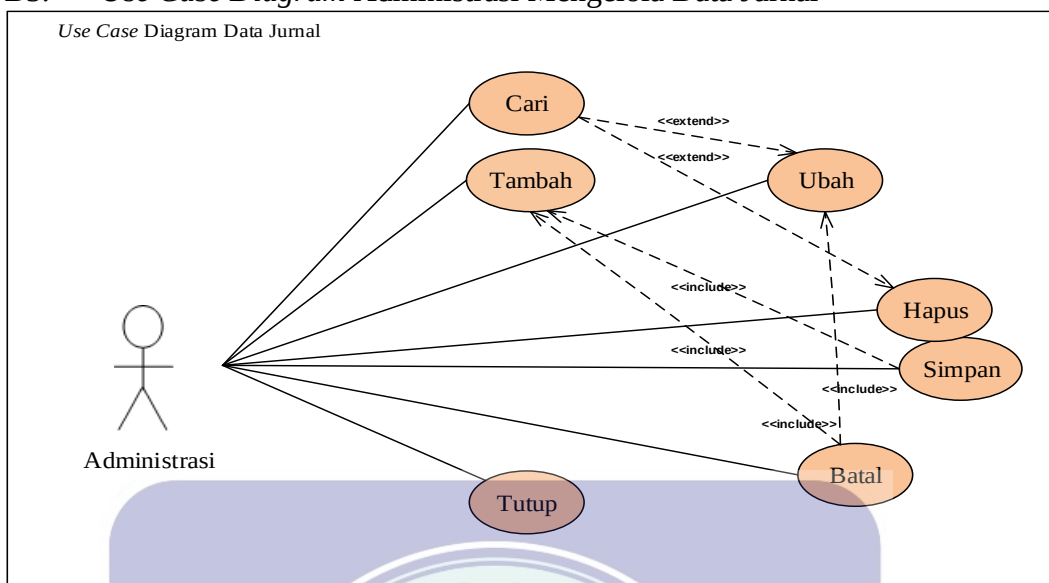


Gambar III.15
Use Case Diagram Administrasi Mengelola Data Pengguna

Tabel III.13
Deskripsi Use Case Diagram Administrasi Mengelola Data Pengguna

Use Case Narrative Data Pengguna	
Tujuan	Administrasi dapat melakukan pengolahan data yang ada disubmenu data pengguna
Deskripsi	Sistem ini memungkinkan administrasi untuk mengelola sistem penjualan mulai dari input data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Tambah	Sistem akan menambah data
Aktor Memilih tombol Simpan	Sistem akan menyimpan data
Aktor Memilih tombol Cari	Sistem akan mencari data
Aktor Memilih tombol Ubah	Sistem akan mengubah data
Aktor Memilih tombol Hapus	Sistem akan menghapus data
Aktor Memilih tombol Batal	Sistem akan mmbatalkan proses
Aktor memilih tombol Tutup	Sistem akan keluar dari form
Kondisi Akhir	Jika perintah sesuai administrasi dapat mengkases sesuai keinginan

B5. Use Case Diagram Administrasi Mengelola Data Jurnal

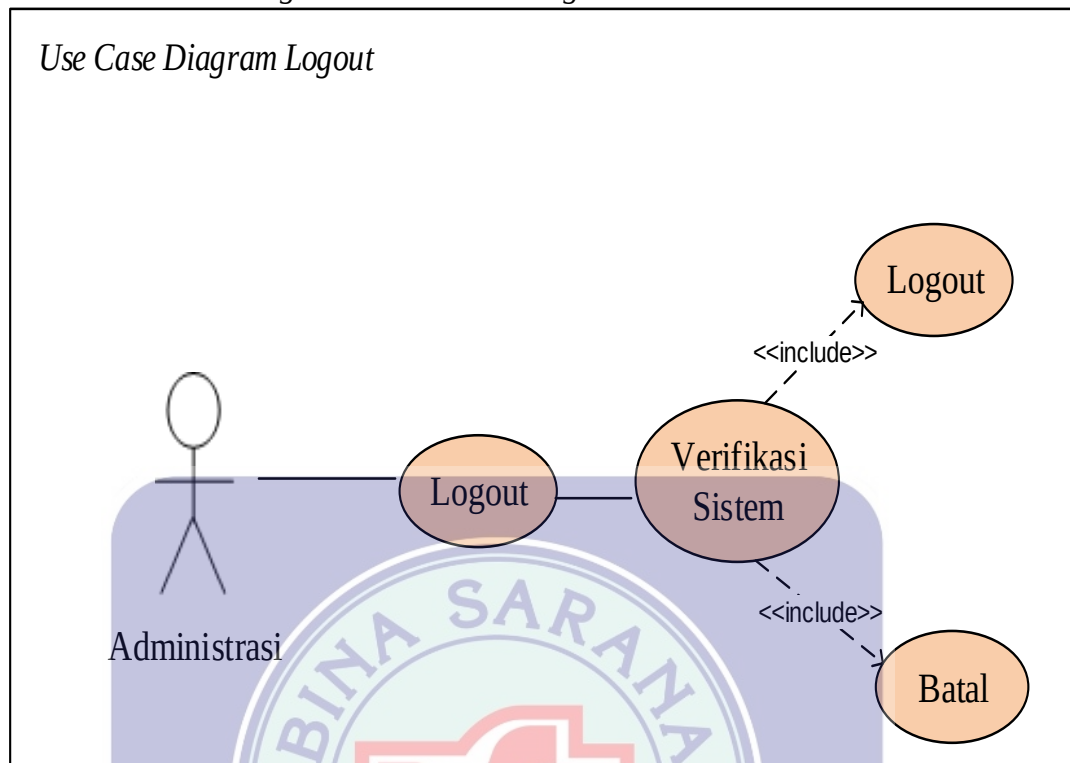


Gambar III.16
Use Case Diagram Administrasi Mengelola Data Jurnal

Tabel III.14
Deskripsi Use Case Diagram Administrasi Mengelola Data Jurnal

Use Case Narrative Data Jurnal	
Tujuan	Administrasi dapat melakukan pengolahan data yang ada disubmenu data jurnal
Deskripsi	Sistem ini memungkinkan administrasi untuk mengelola sistem penjualan mulai dari input data sampai dengan pembuatan laporan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Tambah Aktor Memilih tombol Simpan Aktor Memilih tombol Cari Aktor Memilih tombol Ubah Aktor Memilih tombol Hapus Aktor Memilih tombol Batal Aktor memilih tombol Tutup	Sistem akan menambah data Sistem akan menyimpan data Sistem akan mencari data Sistem akan mengubah data Sistem akan menghapus data Sistem akan mmbatalkan proses Sistem akan keluar dari form
Kondisi Akhir	Jika perintah sesuai administrasi dapat mengkases sesuai keinginan

B6. Use Case Diagram Administrasi Logout

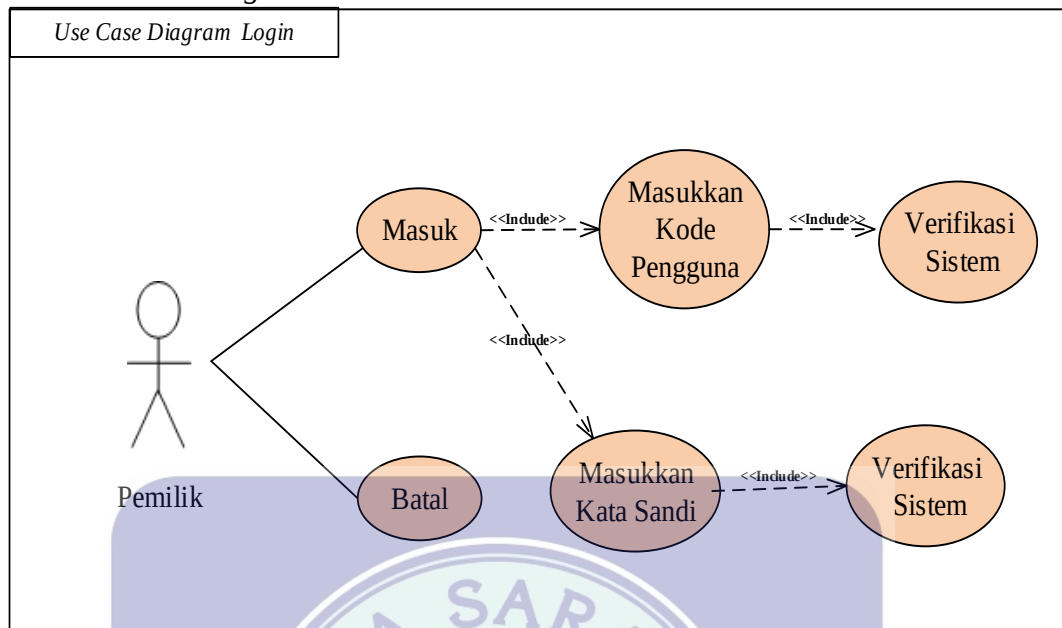


Gambar III.17
Use Case Diagram Administrasi Logout

Tabel III.15
Deskripsi Use Case Diagram Administrasi Logout

Use Case Narrative Logout	
Tujuan	Administrasi dapat melakukan <i>logout</i> dari sistem penjualan
Deskripsi	Sistem ini memungkinkan <i>sales</i> untuk keluar dari sistem penjualan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka menu <i>logout</i>
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Logout Aktor Memilih tombol Ya Aktor memilih tombol Tidak	Sistem akan menampilkan pesan Sistem akan keluar dari aplikasi Sistem akan tetap berada di dalam sistem penjualan
Kondisi Akhir	Jika perintah sesuai <i>sales</i> dapat mengkases sesuai keinginan

C1. Pemilik Login

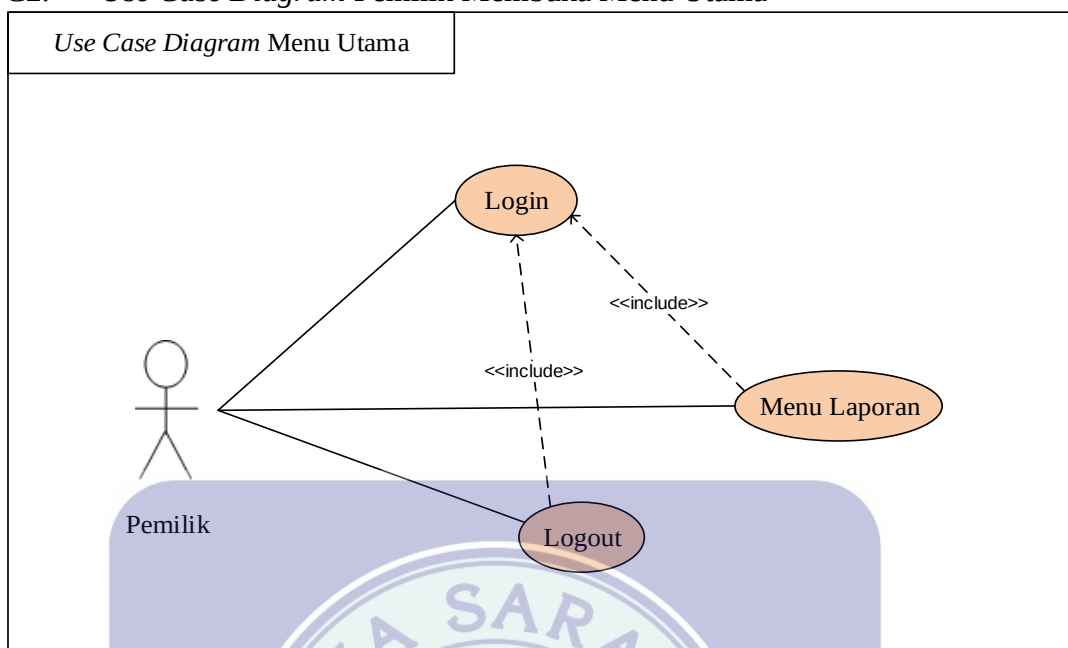


Gambar III.18
Use Case Diagram Pemilik Login

Tabel III.16
Deskripsi Use Case Diagram Pemilik Login

Use Case Narrative Login	
Tujuan	Melakukan login dan masuk kdalam sistem penjualan mesin diesel
Deskripsi	Sistem ini memungkinkan administrasi untuk mengakses sistem penjualan mesin diesel
Skenario Utama	
Aktor	Pemilik
Kondisi Awal	Aktor membuka aplikasi penjualan mesin diesel
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Login	Sistem akan menampilkan <i>textbox</i> untuk mengisi kode pengguna dan kata sandi
Aktor Memilih tombol Keluar	Sistem akan membatalkan proses dan keluar dari aktivitas
Kondisi Akhir	Jika perintah sesuai maka sistem akan masuk kedalam aplikasi tersebut dan pemilik dapat melakukan aktivitas didalam sistem.

C2. Use Case Diagram Pemilik Membuka Menu Utama

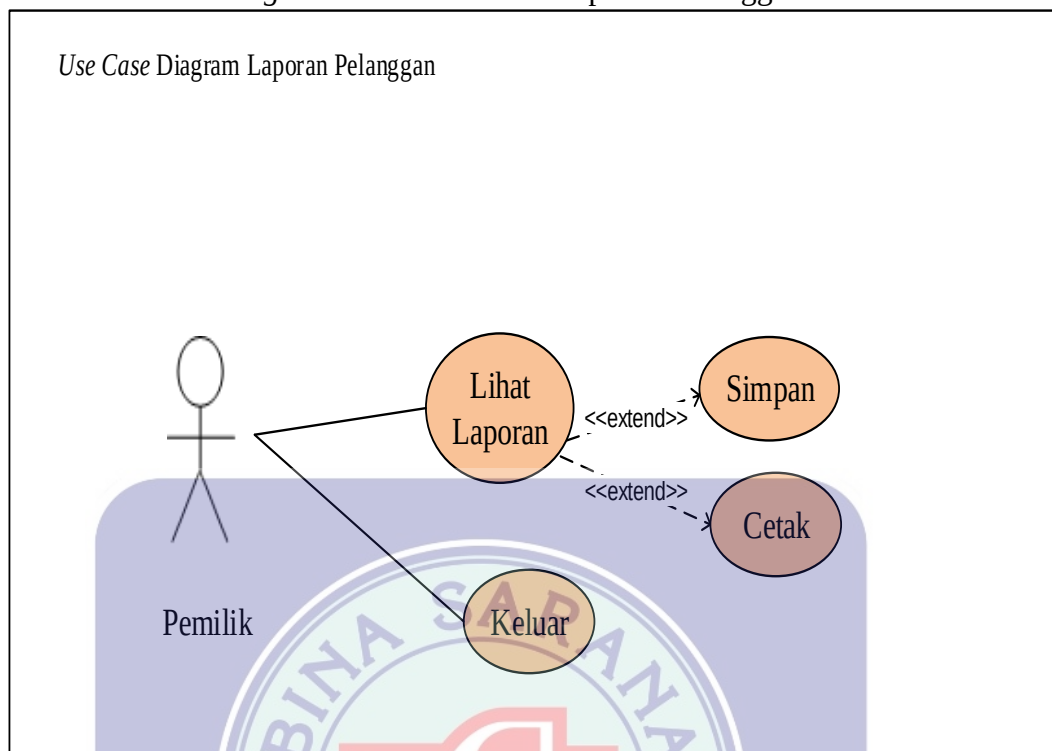


Gambar III.19
Use Case Diagram Pemilik Membuka Menu Utama

Tabel III.17
Deskripsi Use Case Diagram Pemilik Membuka Menu Utama

Use Case Narrative Menu Utama	
Tujuan	Pemilik dapat melakukan pengolahan data yang ada dimenu laporan dan logout dari aplikasi
Deskripsi	Sistem ini memungkinkan pemilik untuk mengelola sistem penjualan dan melihat laporan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Menu Laporan	Sistem akan menampilkan submenu laporan seperti laporan pelanggan, laporan mesin diesel dan laporan penjualan
Aktor memilih tombol Logout	Sistem akan keluar dari aplikasi penjualan
Kondisi Akhir	Jika perintah sesuai penjualan dapat mengkases semua menu utama.

C3. Use Case Diagram Pemilik Melihat Laporan Pelanggan



Gambar III.20

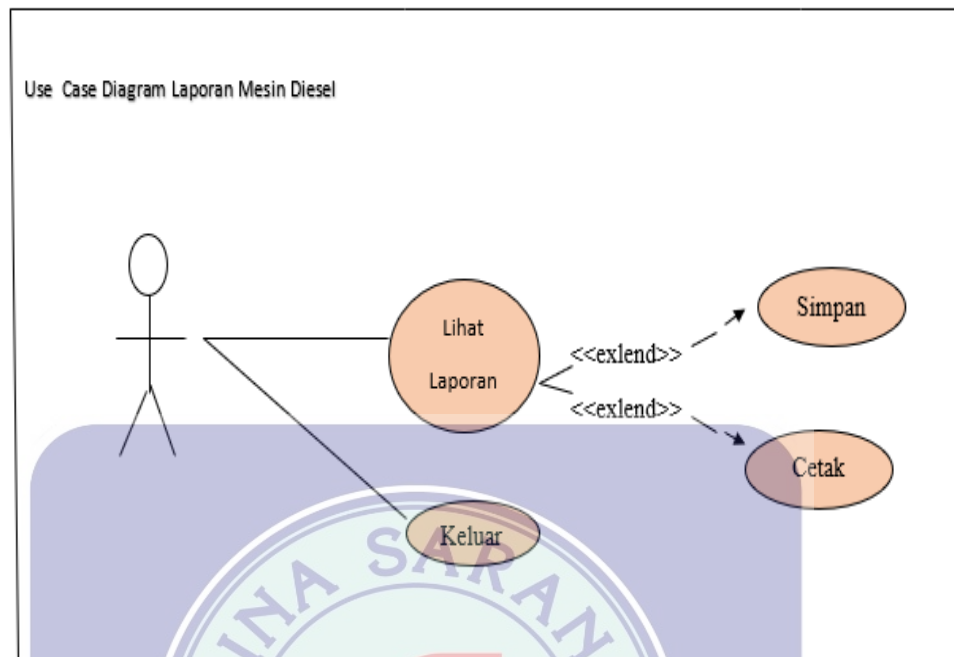
Use Case Diagram Pemilik Melihat Laporan Pelanggan

Tabel III.18

Deskripsi Use Case Diagram Pemilik Melihat Laporan Pelanggan

Use Case Narrative Melihat Laporan Pelanggan	
Tujuan	Pemilik dapat melakukan pengolahan data yang ada disubmenu laporan
Deskripsi	Sistem ini memungkinkan pemilik ntuk melihat laporan pelanggan
Skenario Utama	
Aktor	Pemilik
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Simpan Aktor Memilih tombol Cetak Aktor memilih tombol Keluar	Sistem akan menyimpan laporan Sistem akan mencetak laporan Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai pemilik dapat mengkases sesuai keinginan

C4. Use Case Diagram Pemilik Melihat Laporan Mesin Diesel

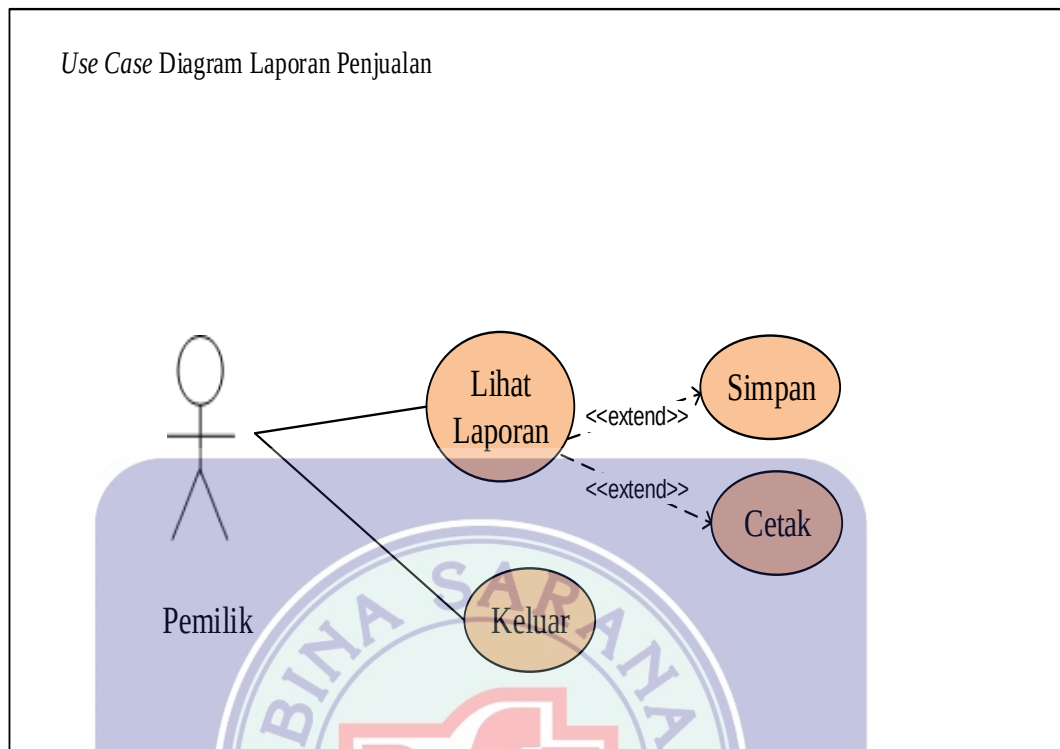


Gambar III.21
Use Case Diagram Pemilik Melihat Laporan Mesin Diesel

Tabel III.19
Deskripsi Use Case Diagram Pemilik Melihat Laporan Mesin Diesel

Use Case Narrative Melihat Laporan Mesin Diesel	
Tujuan	Pemilik dapat melakukan pengolahan data yang ada disubmenu laporan
Deskripsi	Sistem ini memungkinkan pemilik ntuk melihat laporan mesin diesel
Skenario Utama	
Aktor	Pemilik
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Simpan	Sistem akan menyimpan laporan
Aktor Memilih tombol Cetak	Sistem akan mencetak laporan
Aktor memilih tombol Keluar	Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai pemilik dapat mengkases sesuai keinginan

C5. Use Case Diagram Pemilik Melihat Laporan Penjualan

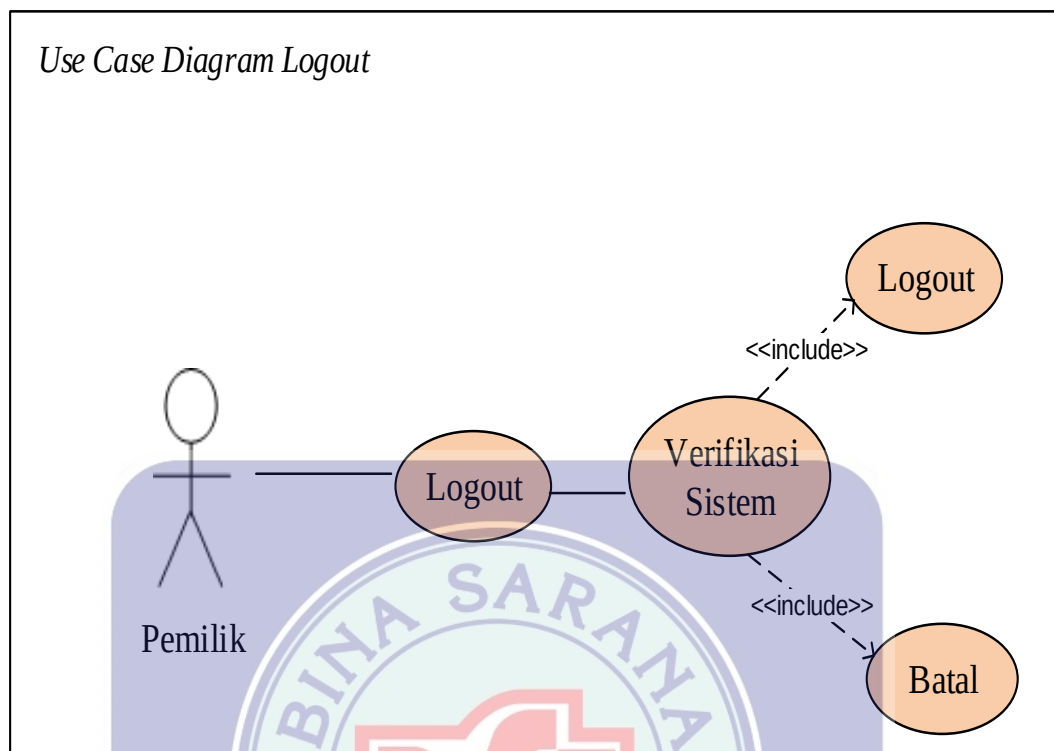


Gambar III.22
Use Case Diagram Pemilik Melihat Laporan Penjualan

Tabel III.20
Deskripsi Use Case Diagram Pemilik Melihat Laporan Penjualan

Use Case Narrative Melihat Laporan Penjualan	
Tujuan	Pemilik dapat melakukan pengolahan data yang ada disubmenu laporan
Deskripsi	Sistem ini memungkinkan pemilik ntuk melihat laporan penjualan
Skenario Utama	
Aktor	Pemilik
Kondisi Awal	Aktor membuka aplikasi sistem penjualan
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Simpan Aktor Memilih tombol Cetak Aktor memilih tombol Keluar	Sistem akan menyimpan laporan Sistem akan mencetak laporan Sistem akan keluar dari <i>form</i>
Kondisi Akhir	Jika perintah sesuai pemilik dapat mengkases sesuai keinginan

C6. Use Case Diagram Pemilik Logout



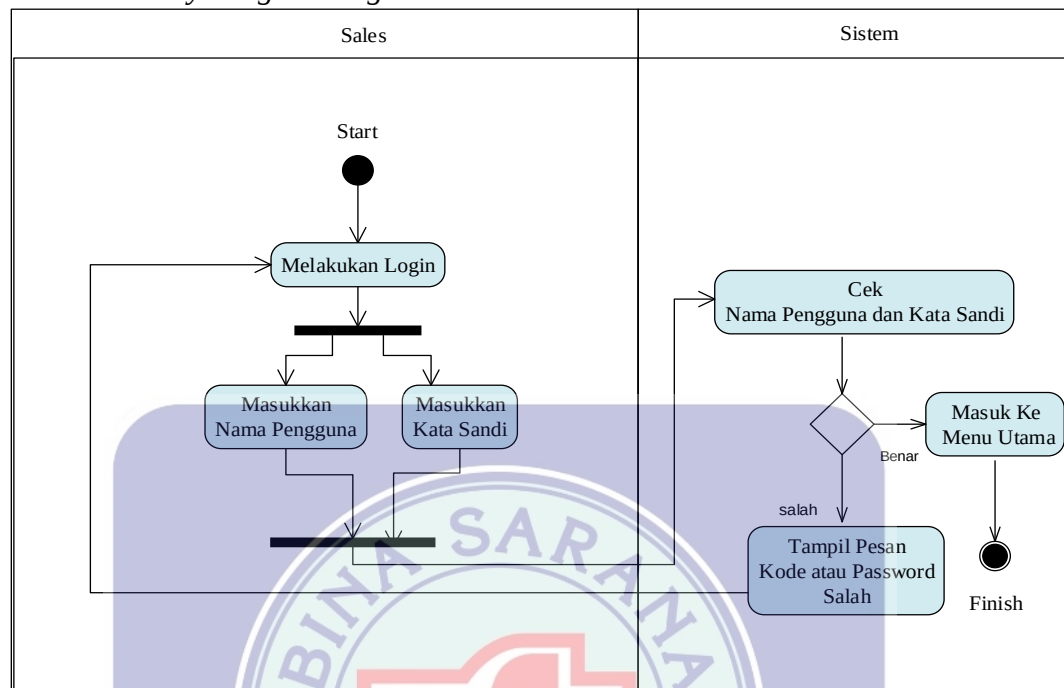
Gambar III.23
Use Case Diagram Pemilik Logout

Tabel III.21
Deskripsi Use Case Diagram Pemilik Logout

Use Case Narrative Logout	
Tujuan	Pemilik dapat melakukan <i>logout</i> dari sistem penjualan
Deskripsi	Sistem ini memungkinkan pemilik untuk keluar dari sistem penjualan
Skenario Utama	
Aktor	Pemilik
Kondisi Awal	Aktor membuka menu <i>logout</i>
Aksi Aktor	Reaksi Sistem
Aktor Memilih tombol Logout Aktor Memilih tombol Ya Aktor memilih tombol Tidak	Sistem akan menampilkan pesan Sistem akan keluar dari aplikasi Sistem akan tetap berada di dalam sistem penjualan
Kondisi Akhir	Jika perintah sesuai sales dapat mengkases sesuai keinginan

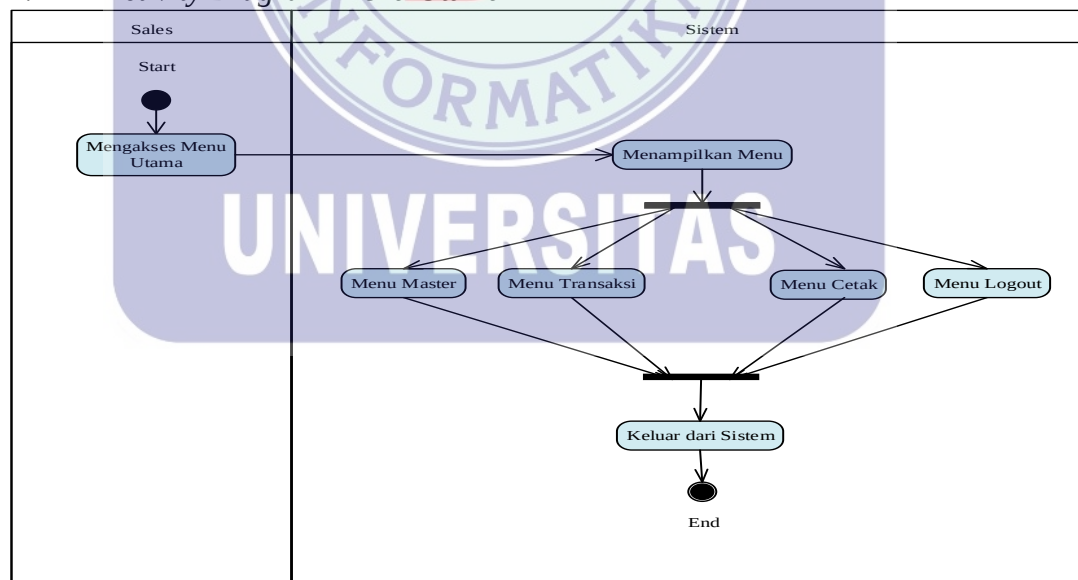
3.3.3. Activity Diagram

1. Activity Diagram Login



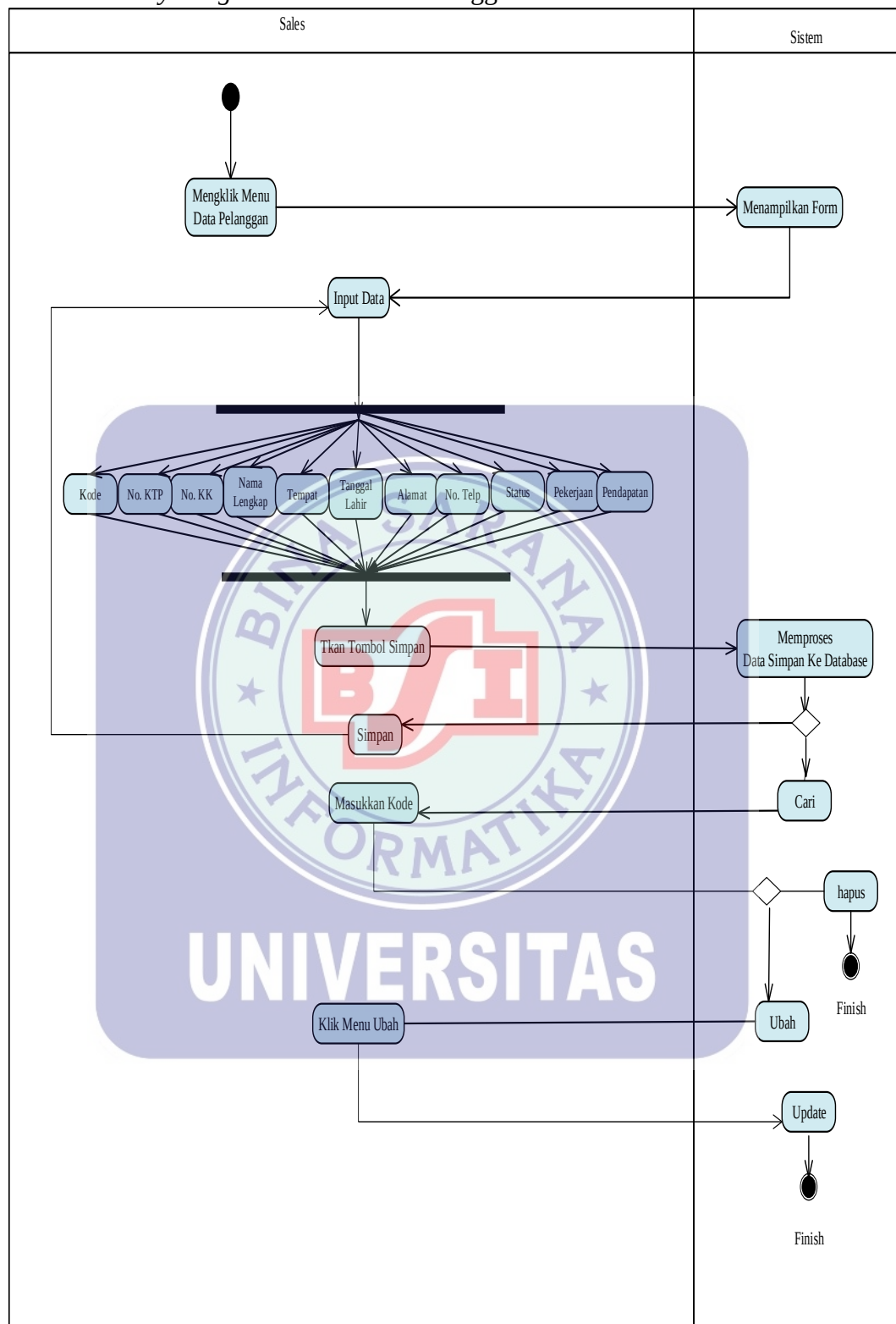
Gambar III.24
Activity Diagram Login

2. Activity Diagram Menu Utama



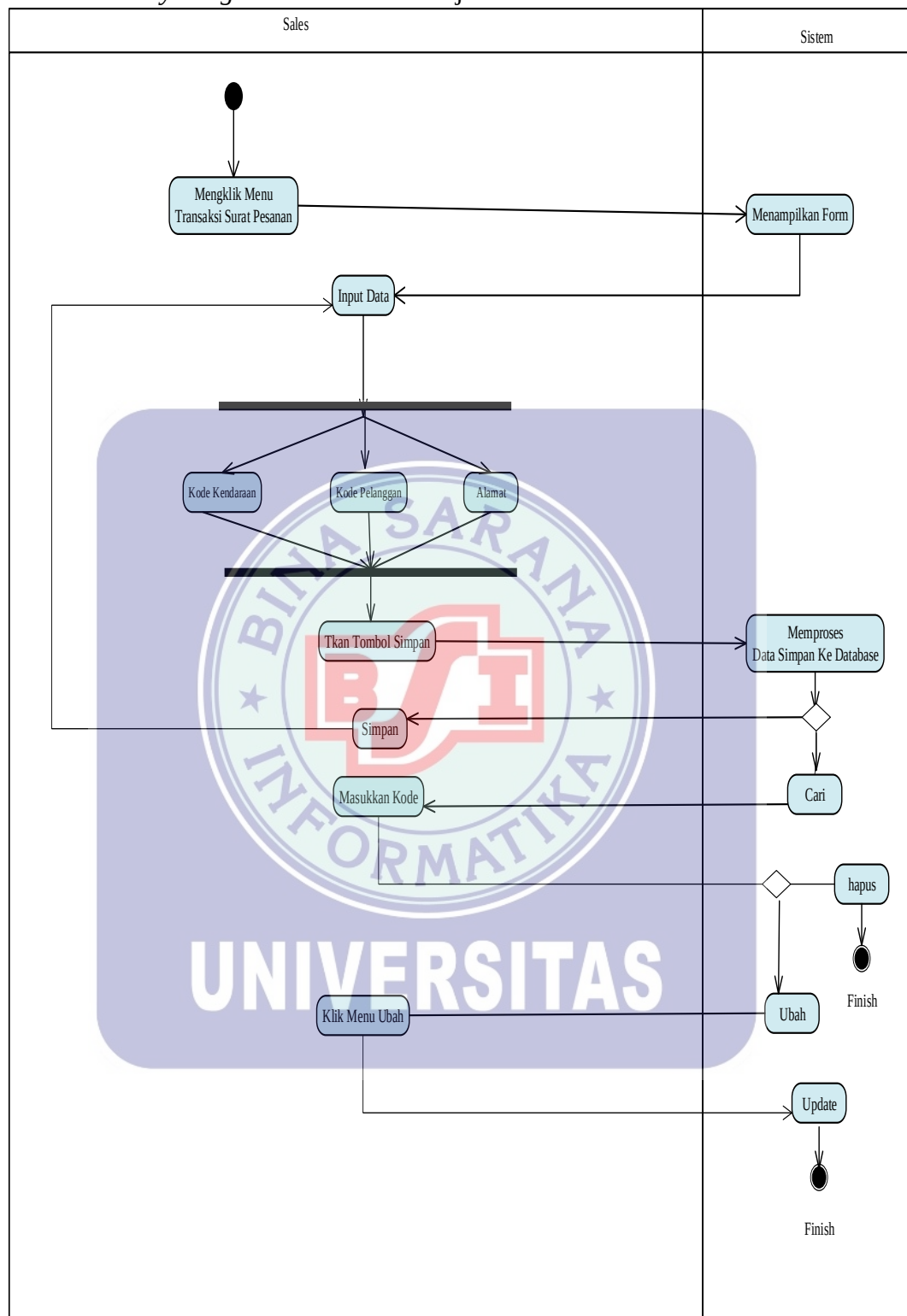
Gambar III.25
Activity Diagram Menu Utama

3. Activity Diagram Menu Data Pelanggan

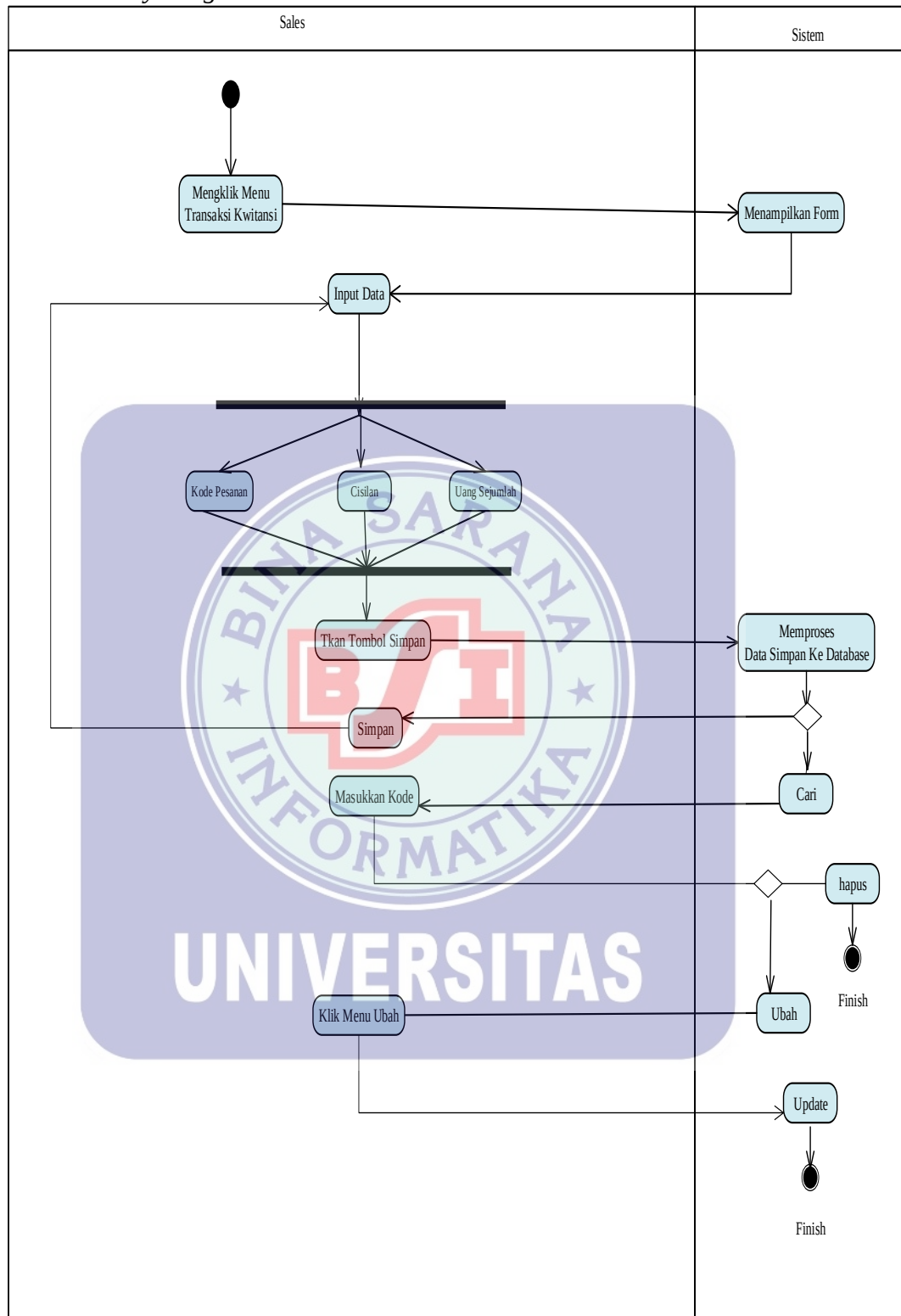


Gambar III.26
Activity Diagram Data Pelanggan

4. Activity Diagram Menu Data Penjualan

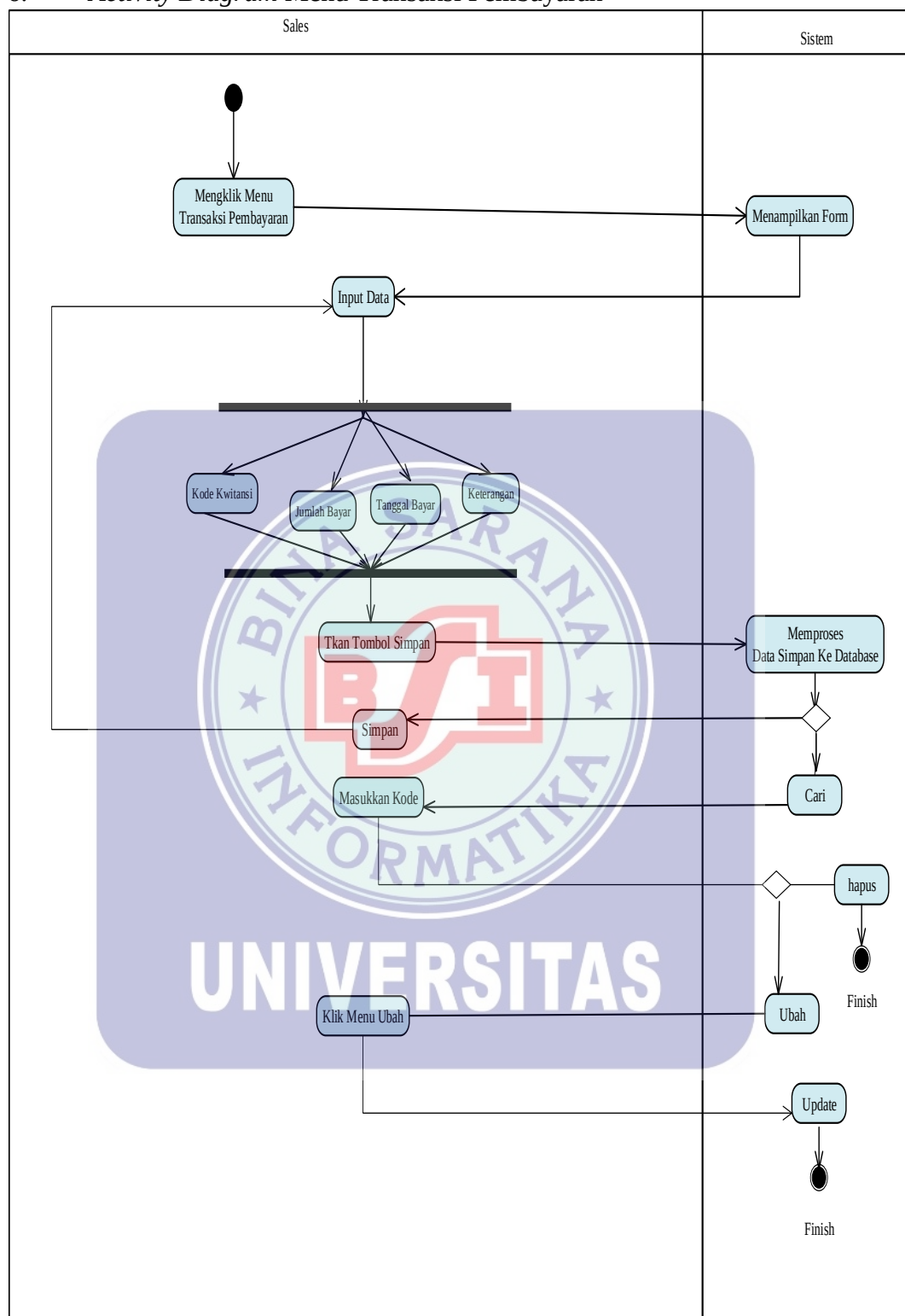


Gambar III.27
Activity Diagram Transaksi data Penjualan

5. *Activity Diagram Menu Transaksi Kwitansi*

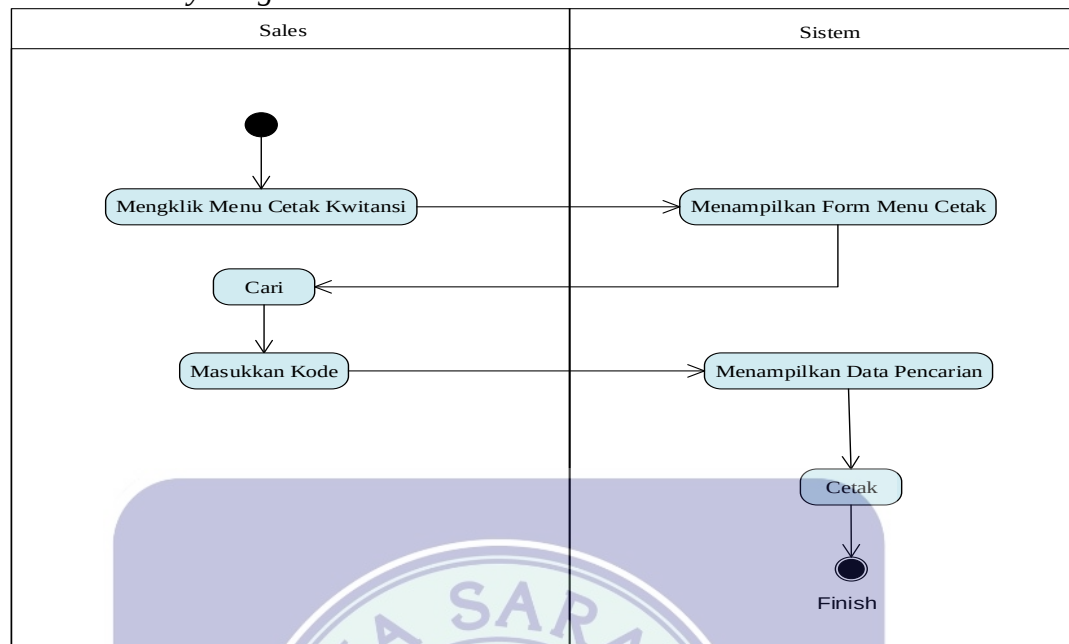
Gambar III.28
Activity Diagram Transaksi Kwitansi

6. Activity Diagram Menu Transaksi Pembayaran



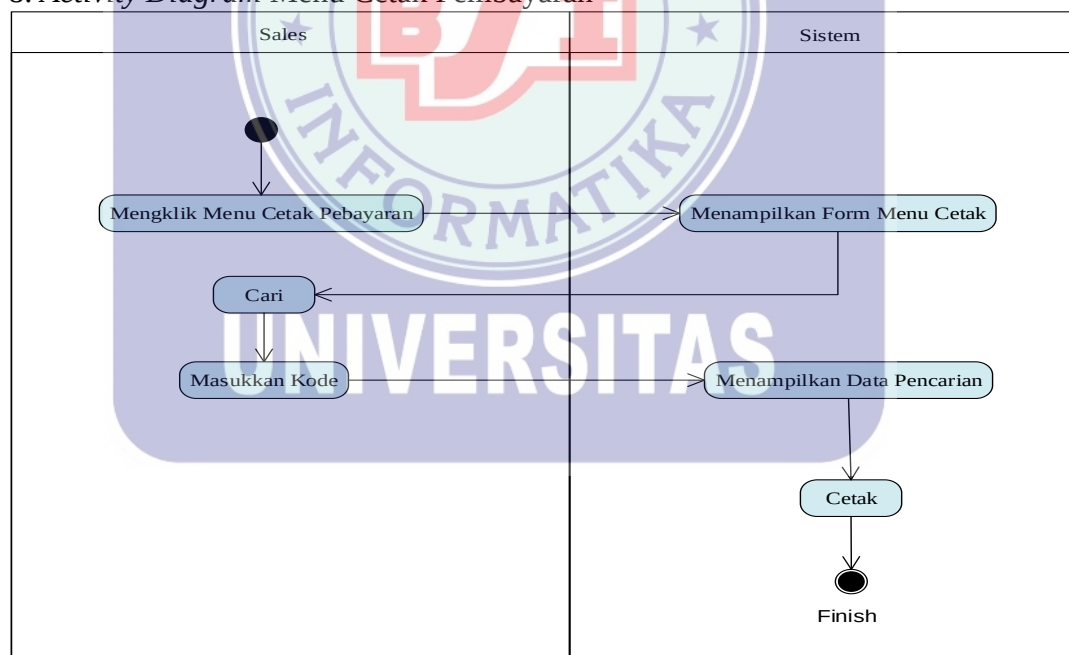
Gambar III.29
Activity Diagram Transaksi Pembayaran

7. *Activity Diagram Menu Cetak Kwitansi*



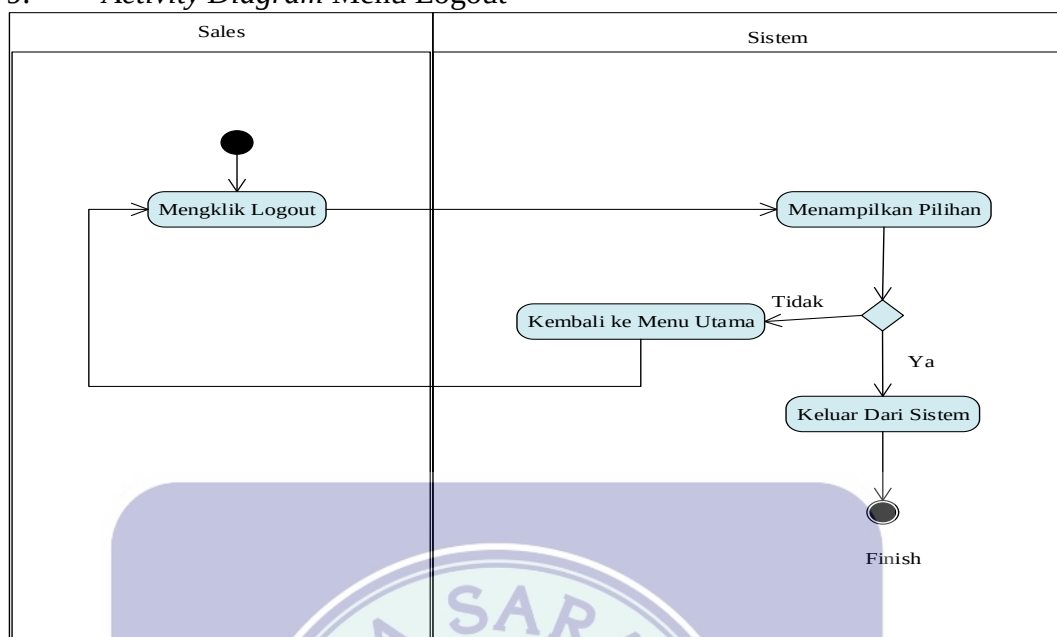
Gambar III.30
Activity Diagram Cetak Kwitansi

8. *Activity Diagram Menu Cetak Pembayaran*



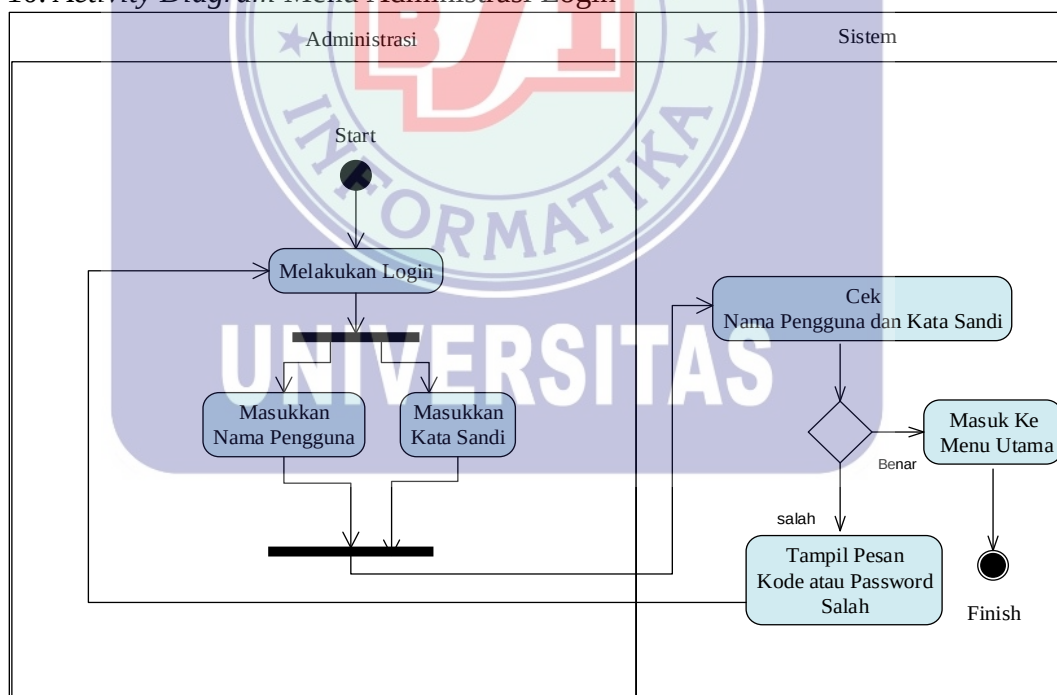
Gambar III.31
Activity Diagram Cetak Pembayaran

9. Activity Diagram Menu Logout



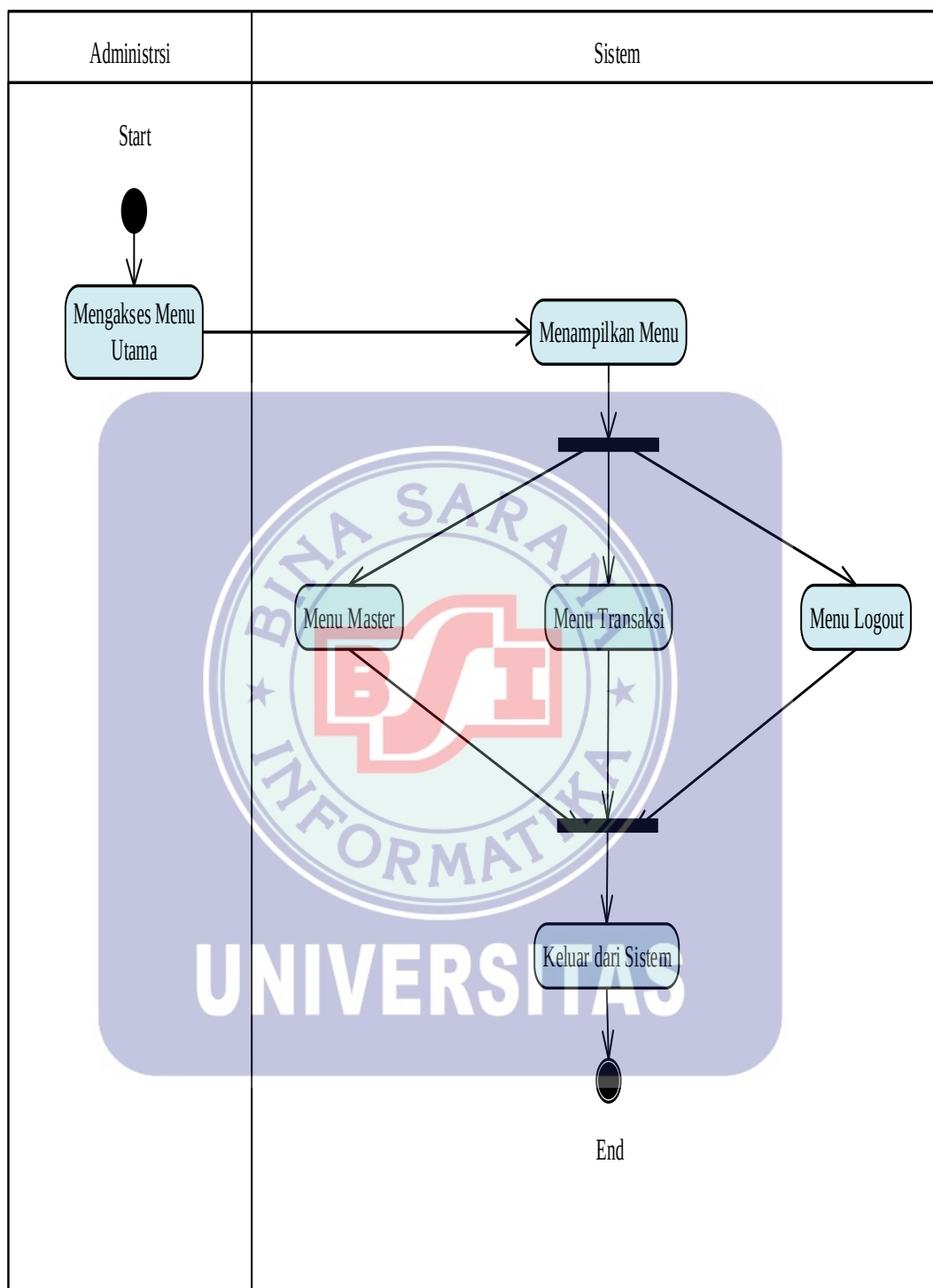
Gambar III.32
Activity Diagram Logout

10. Activity Diagram Menu Administrasi Login



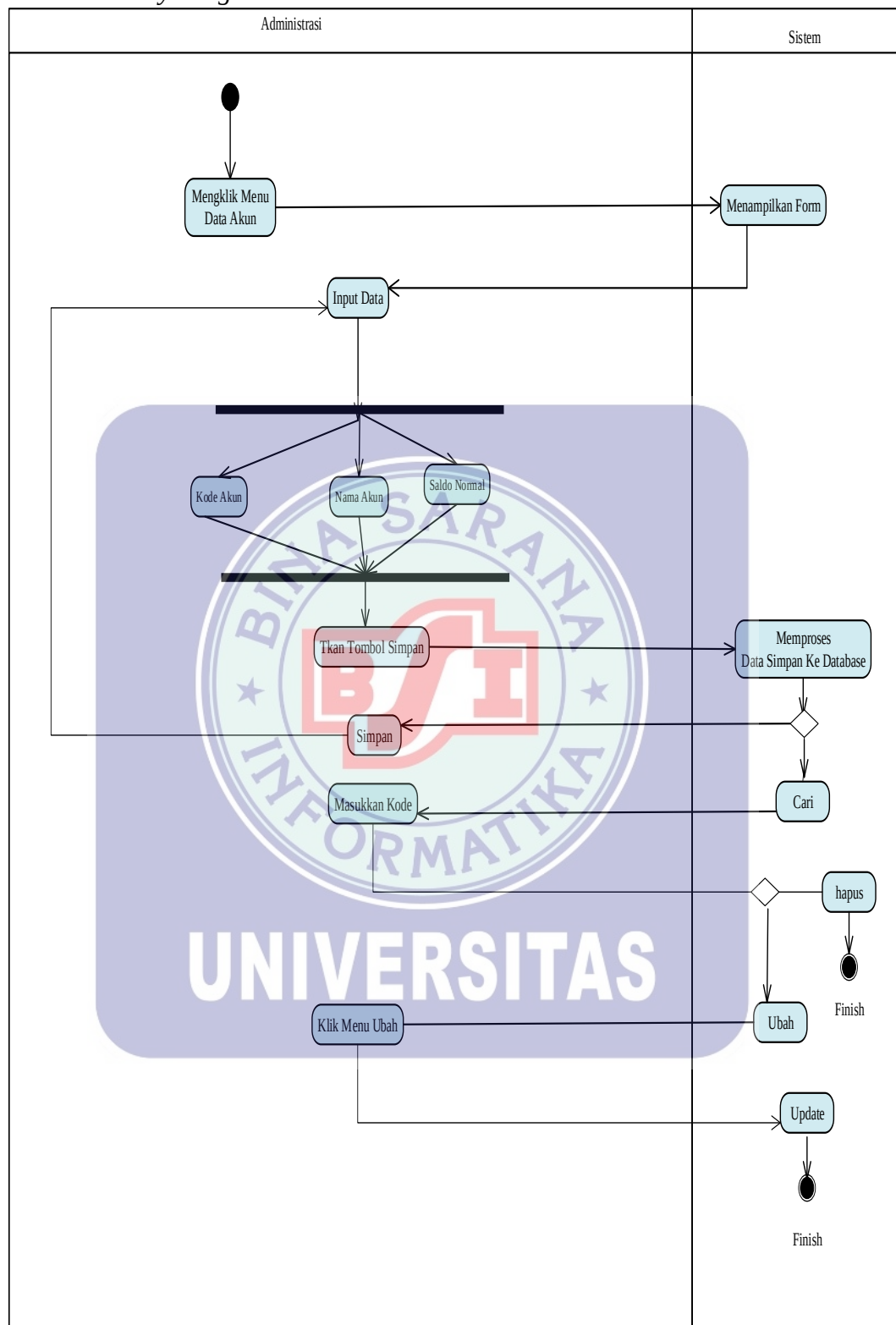
Gambar III.33
Activity Diagram Login

11. Activity Diagram Menu Administrasi Menu Utama



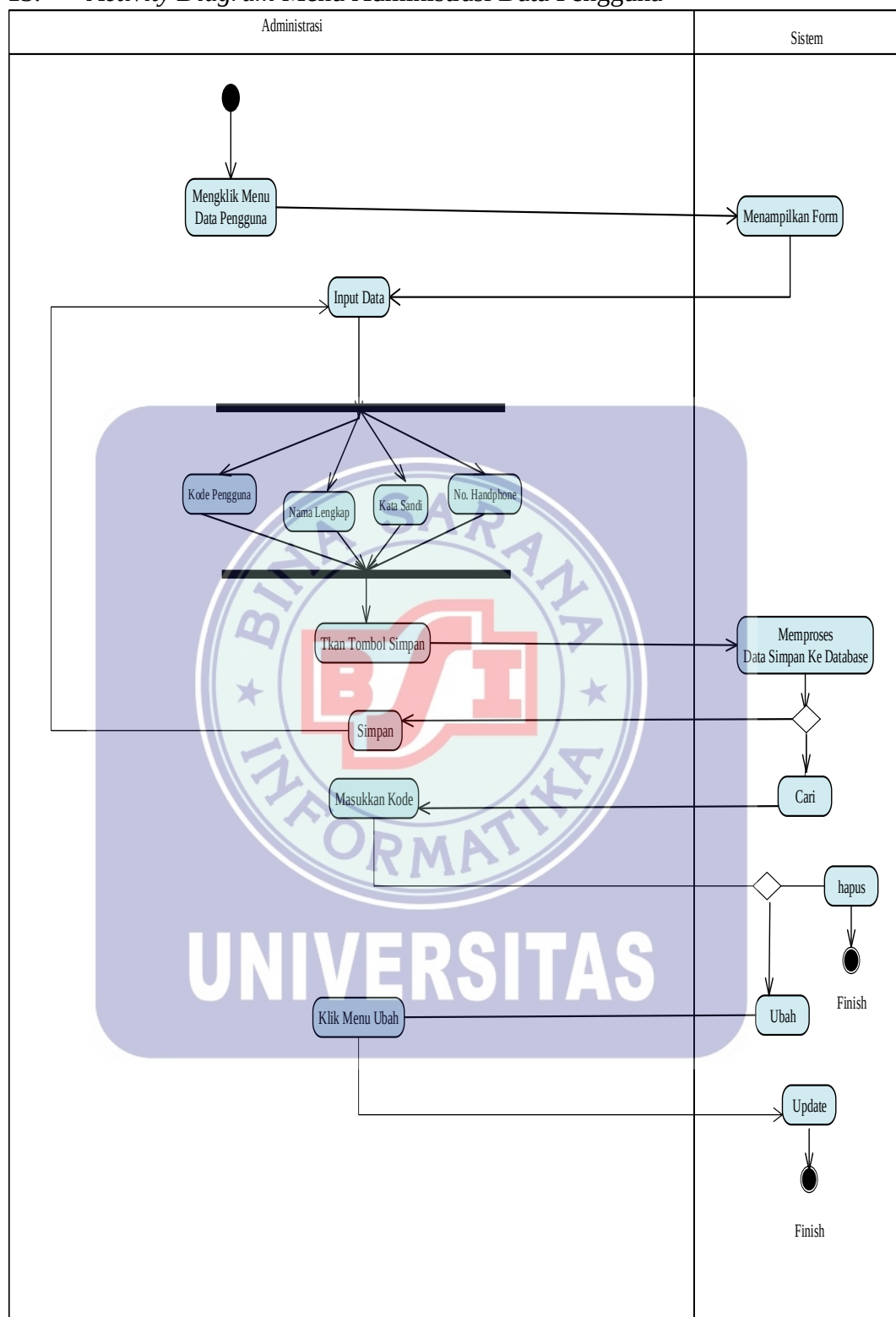
Gambar III.34
Activity Diagram Cetak Pembayaran

12. Activity Diagram Menu Administrasi Data Akun



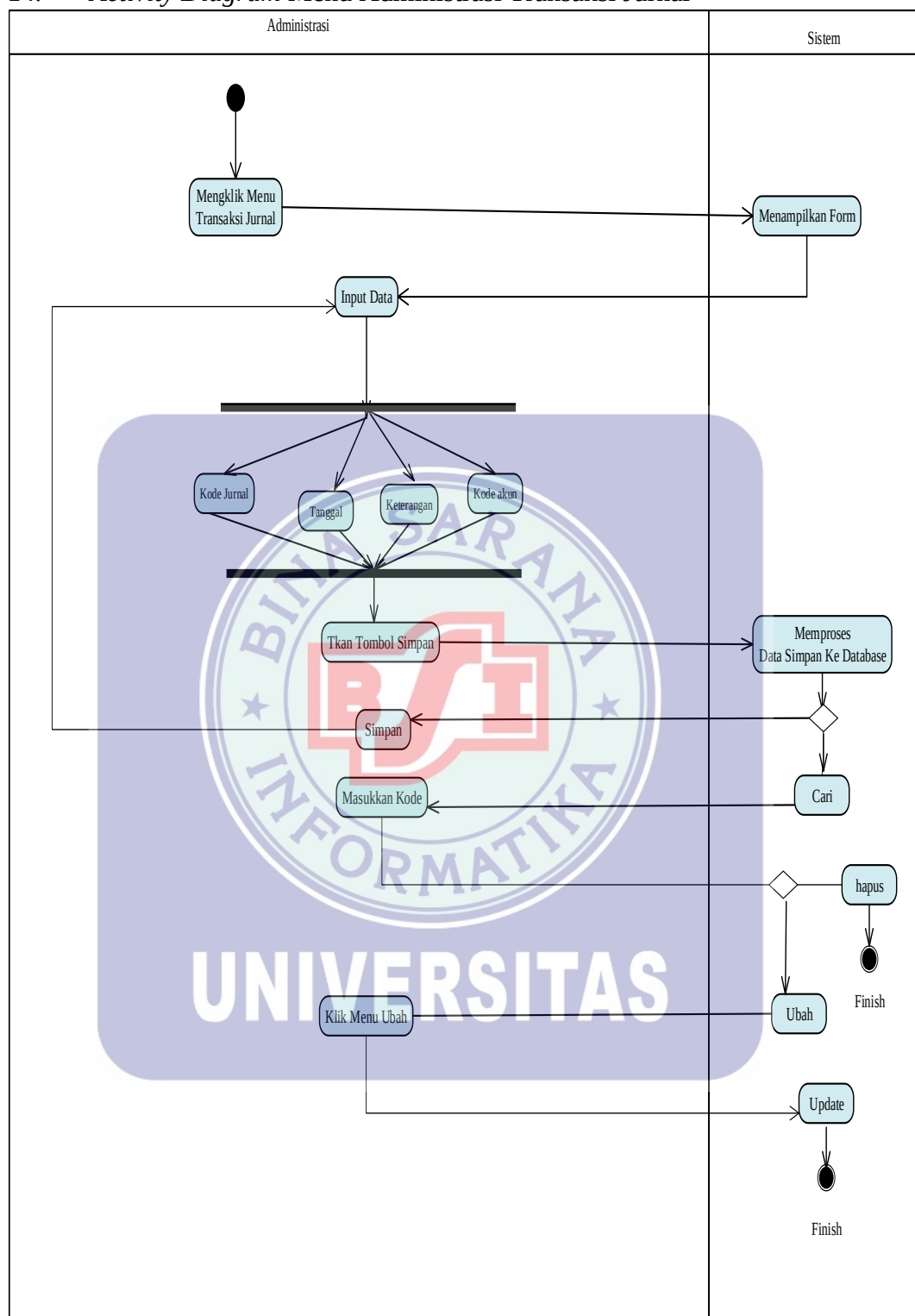
Gambar III.35
Activity Diagram Data Akun

13. Activity Diagram Menu Administrasi Data Pengguna

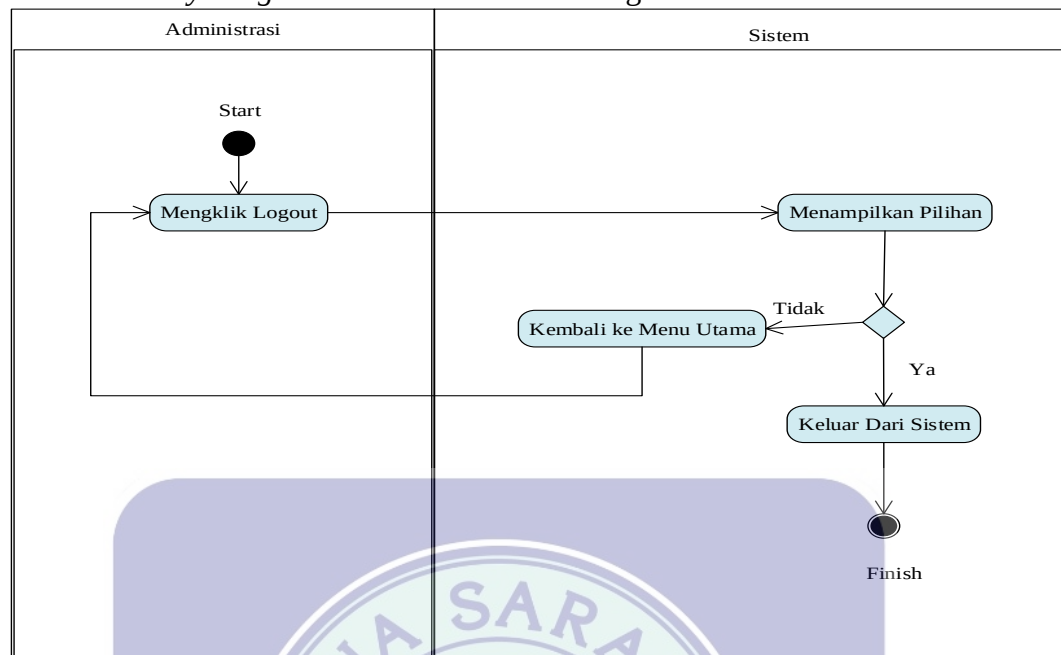


Gambar III.36
Activity Diagram Data Pengguna

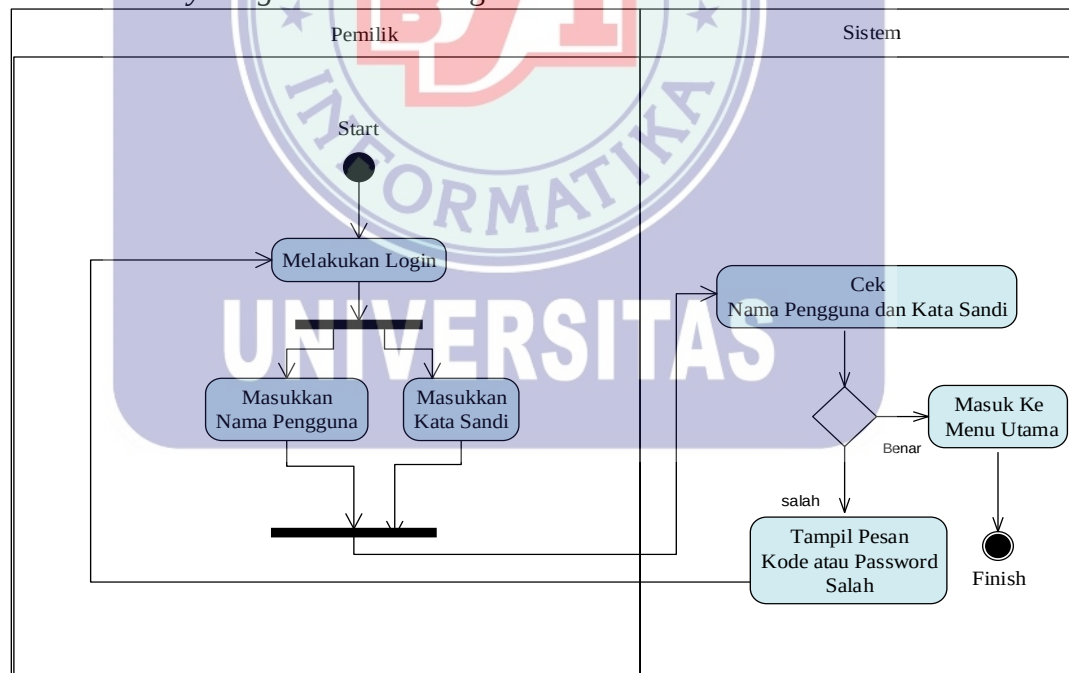
14. Activity Diagram Menu Administrasi Transaksi Jurnal



Gambar III.37
Activity Diagram Transaksi Jurnal

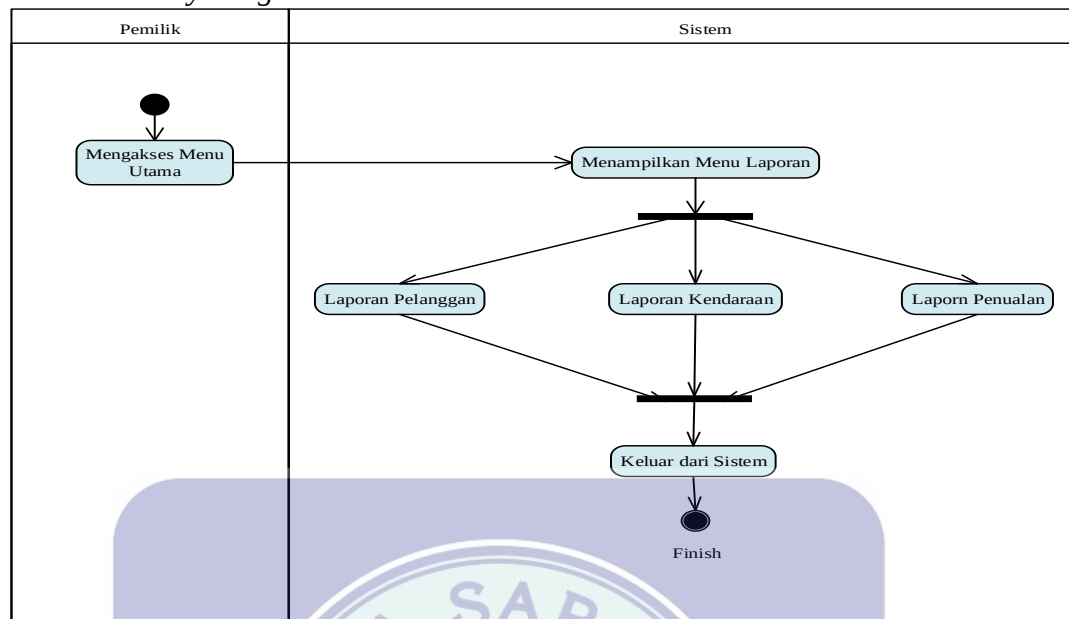
15. *Activity Diagram Menu Administrasi Logout*

Gambar III.38
Activity Diagram Logout

16. *Activity Diagram Pemilik Login*

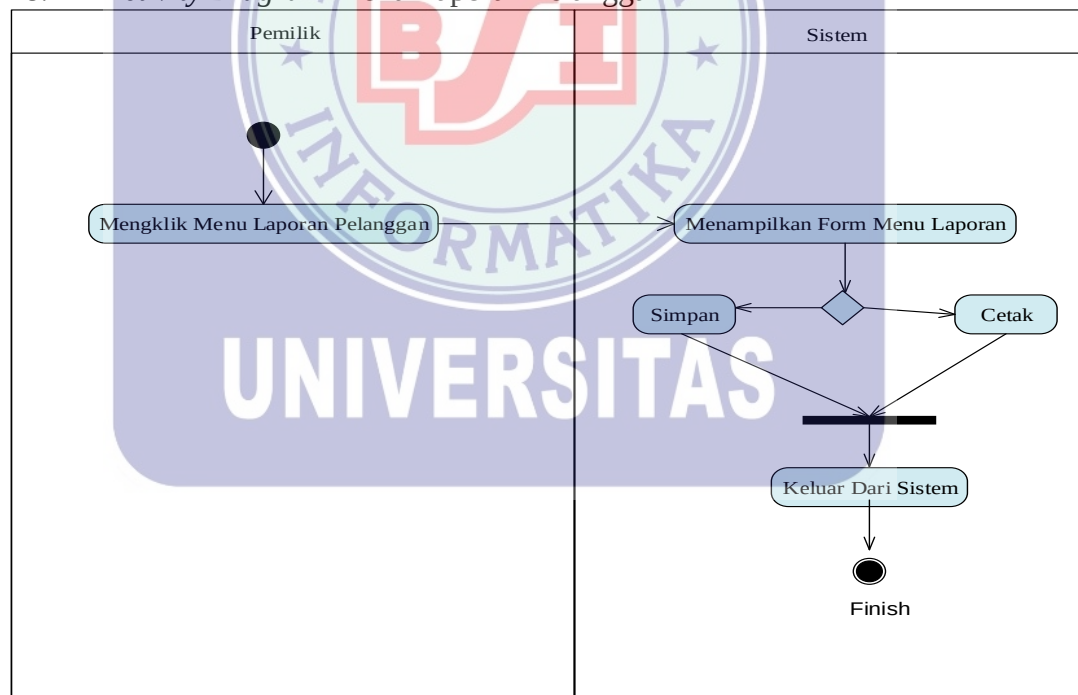
Gambar III.39
Activity Diagram Login

17. *Activity Diagram Menu Utama Pemilik*

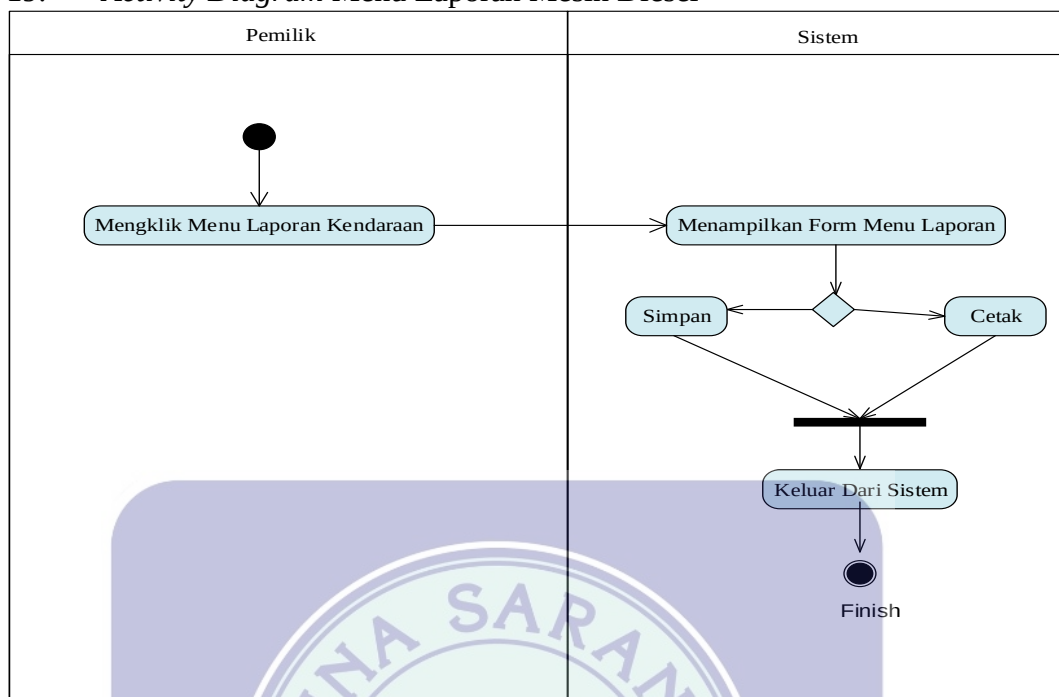


Gambar III.40
Activity Diagram Menu Utama

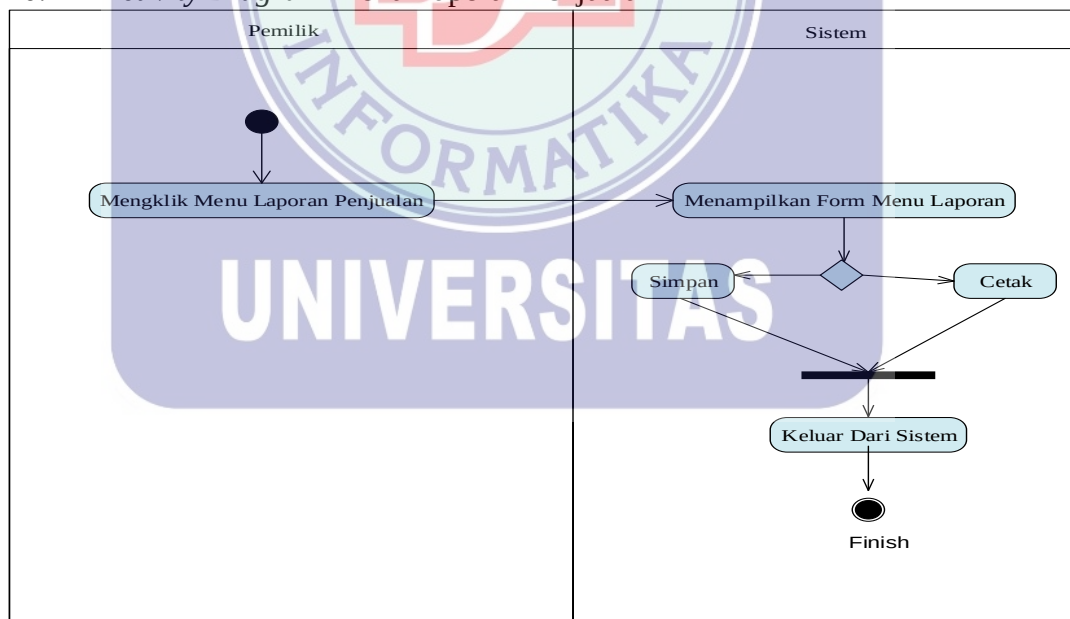
18. *Activity Diagram Menu Laporan Pelanggan*



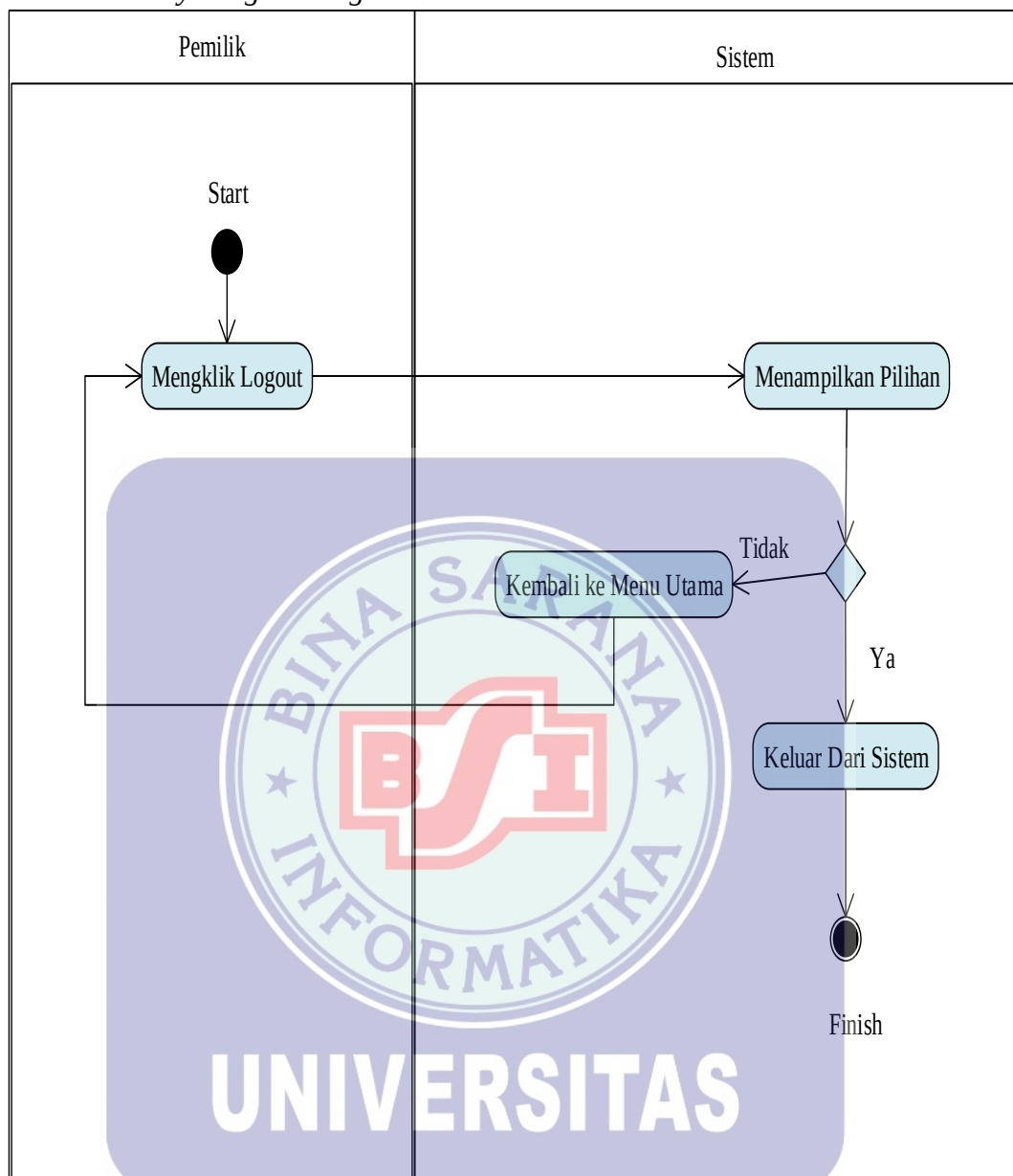
Gambar III.41
Activity Diagram Laporan Pelanggan

19. *Activity Diagram Menu Laporan Mesin Diesel*

Gambar III.42
Activity Diagram Laporan Mesin Diesel

20. *Activity Diagram Menu Laporan Penjualan*

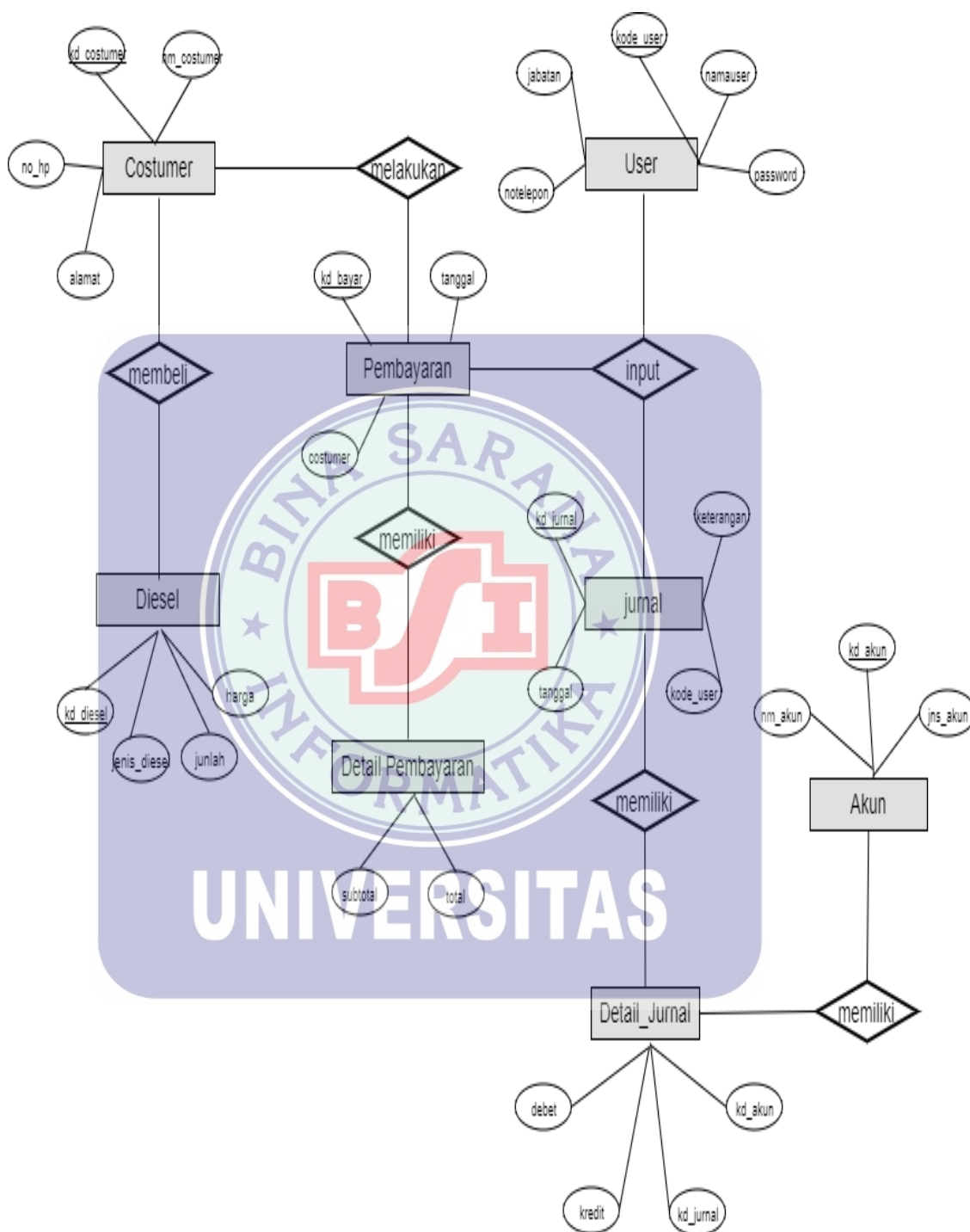
Gambar III.43
Activity Diagram Laporan Penjualan

21. *Activity Diagram Logout*

Gambar III.44
Activity Diagram Logout

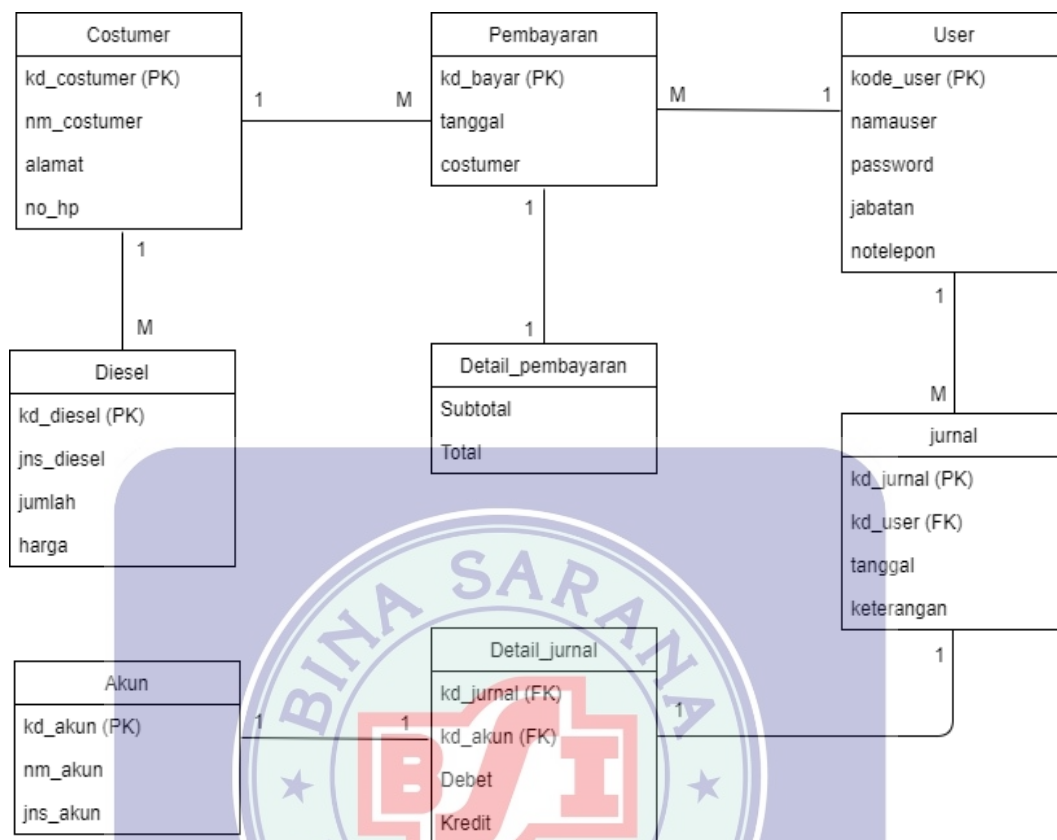
3.4. Desain

3.4.1. Entity Relationship Diagram



Gambar III.45
Entity Relationship Diagram (ERD)

3.4.2. Logical Record Structure (LRS)



Gambar III.46
Logical Record Structure (LRS)

3.4.3. Spesifikasi File

1. Spesifikasi File Tabel Data Akun

Nama Database : db_service

Nama File : data_akun

Akronim : data_akun.sql

Tipe File : File Master

Akses File : Random

Panjang Record : 48 karakter

Kunci Field : kode_akun

Software : MySQL

Tabel III.22.
Spesifikasi File data_akun

No	Elemen Data	Nama Field	Type	Size	Ket
1	Kode Akun	kode_akun	Varchar	8	Primary Key
2	Nama Akun	nama_akun	Varchar	30	
3	Jenis Akun	jenis	Varchar	10	

2. Spesifikasi File Tabel Data Pelanggan

Nama Database : db_service

Nama File : data_pelanggan

Akronim : data_pelanggan.sql

Tipe File : File Master

Akses File : Random

Panjang Record : 176 karakter

Kunci Field : kode_pelanggan

Software : MySQL

Tabel III.23.
Spesifikasi File data_pelanggan

No	Elemen Data	Nama Field	Type	Size	Ket
1	Kode_Pelanggan	kode_pelanggan	Varchar	8	Primary Key
2	No_Ktp	no_ktp	Varchar	16	
3	No_Kk	no_kk	Varchar	16	
4	Nama_Lengkap	nama_lengkap	Varchar	25	
5	Tempat	tempat	Varchar	25	
6	Tanggal Lahir	tl	Varchar	30	
7	Status	status	Varchar	15	
8	No_Telepon	no_telepon	Varchar	15	
9	Pekerjaan	pekerjaan	Varchar	20	
10	Pendapatan	pendapatan	Double	-	
11	Alamat	alamat	Text	-	

3. Spesifikasi File Tabel Data Pengguna

Nama Database : db_service

Nama File : data_pengguna

Akronim : data_pengguna.sql
 Tipe File : File Master
 Akses File : Random
 Panjang Record : 74 karakter
 Kunci Field : kode_pengguna
 Software : MySQL

Tabel III.24.
Spesifikasi File data_pengguna

No	Elemen Data	Nama Field	Type	Size	Ket
1	Kode_Pengguna	kode_pengguna	Varchar	8	Primary Key
2	Nama_Lengkap	nama_lengkap	Varchar	20	
3	Kata_Sandi	kata_sandi	Varchar	10	
4	Akses	akses	Varchar	13	
5	Nohp	nohp	Varchar	13	

4. Spesifikasi File Tabel Detail Jurnal

Nama Database : db_service
 Nama File : detail_jurnal
 Akronim : detail_jurnal.sql
 Tipe File : File Master
 Akses File : Random
 Panjang Record : 38 karakter
 Kunci Field : -
 Software : MySQL

Tabel III.25.
Spesifikasi File detail_jurnal

No	Elemen Data	Nama Field	Type	Size	Ket
1	Kode_Jurnal	kode_jurnal	Varchar	8	Foreign Key
2	Kode_Akun	kode_akun	Varchar	8	Foreign Key
3	Debet	debet	Integer	11	
4	Kredit	kredit	Integer	11	

5. Spesifikasi *File* Tabel Jurnal

Nama *Database* : db_service

Nama *File* : jurnal

Akronim : jurnal.sql

Tipe *File* : *File Master*

Akses *File* : *Random*

Panjang *Record* : 38 karakter

Kunci *Field* : kode_jurnal

Tabel III.26.
Spesifikasi *File* jurnal

No	Elemen Data	Nama <i>Field</i>	Type	Size	Ket
1	Kode_Jurnal	kode_jurnal	Varchar	15	Primary Key
2	Tanggal	tanggal	Varchar	30	
3	Keterangan	keterangan	Text	-	
4	Kode_Pengguna	kode_pengguna	Varchar	8	Foreign Key

6. Spesifikasi *File* Tabel Kwitansi

Nama *Database* : db_service

Nama *File* : kwitansi

Akronim : kwitansi.sql

Tipe *File* : *File Master*

Akses *File* : *Random*

Panjang *Record* : 123 karakter

Kunci *Field* : kode_kwitansi

Software : MySQL

Tabel III.27.
Spesifikasi *File* kwitansi

No	Elemen Data	Nama <i>Field</i>	Type	Size	Ket
1	kode_kwitansi	kode_kwitansi	Varchar	20	Primary Key
2	tgl	tgl	Varchar	30	

3	kode_pengguna	kode_pengguna	Varchar	8	Foreign Key
4	kode_sp	kode_sp	Varchar	30	Foreign Key
5	nominal	nominal	Int	11	
6	terbilang	terbilang	Text	-	
7	cicill	cicill	Smallint	3	
8	cicilan	cicilan	Int	11	

7. Spesifikasi *File* Tabel Pembayaran

Nama *Database* : db_service

Nama *File* : pembayaran

Akronim : pembayaran.sql

Tipe *File* : *File Master*

Akses *File* : *Random*

Panjang *Record* : 144 karakter

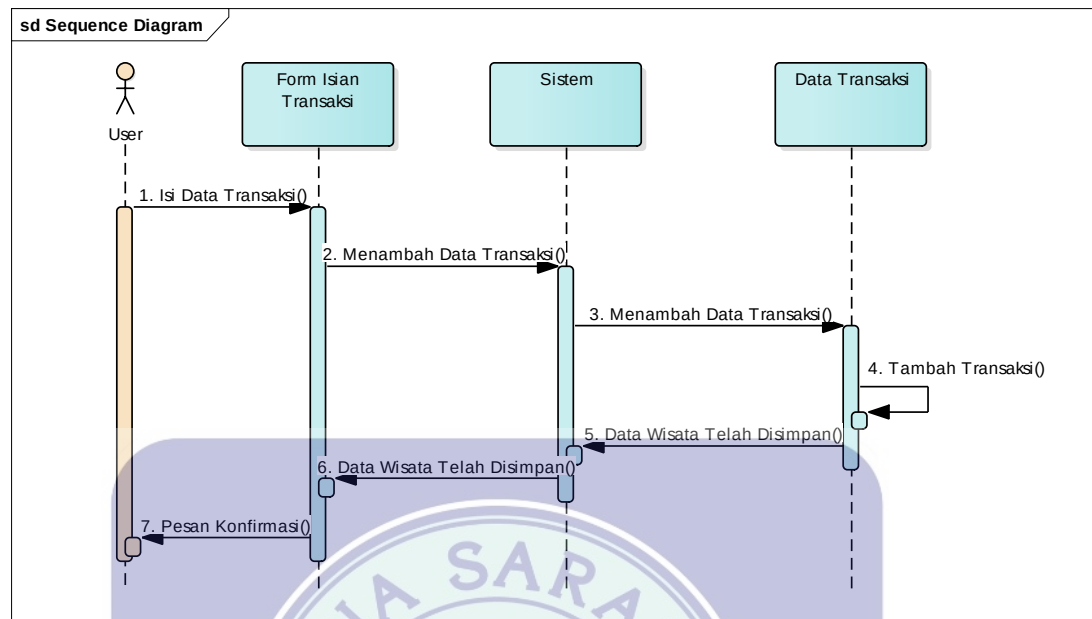
Kunci *Field* : kode_pembayaran

Software : MySQL

Tabel III.28.
Spesifikasi *File* pembayaran

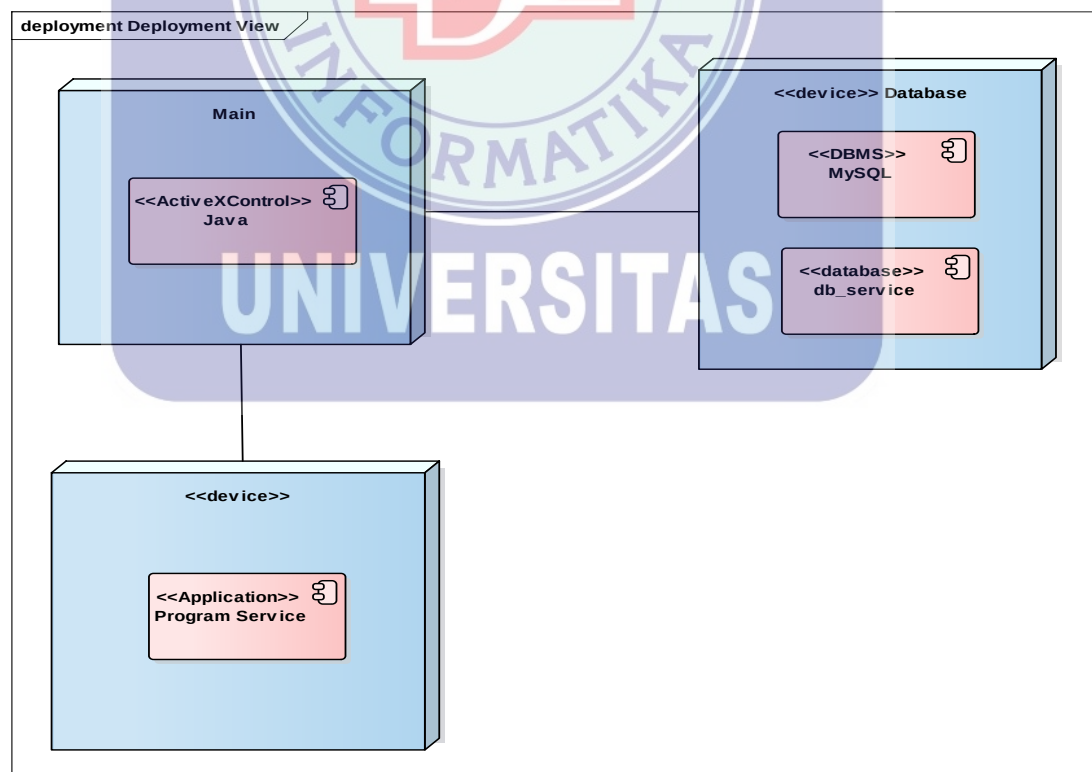
No	Elemen Data	Nama <i>Field</i>	Type	Size	Ket
1	Kode_Pembayaran	kode_pembayaran	Varchar	30	Primary Key
2	Tgl	tgl	Varchar	30	
3	Kode_Pengguna	kode_pengguna	Varchar	8	Foreign Key
4	Kode_Kwitansi	kode_kwitansi	Varchar	20	Foreign Key
5	Jumbay	jumbay	Int	11	
6	Terbilang	terbilang	Text	-	
7	Tanggalbayar	tanggalbayar	Varchar	30	
8	Cicilan_Ke	cicilan_ke	Smallint	3	
9	Keterangan	keterangan	Varchar	12	

3.4.4. Sequence Diagram



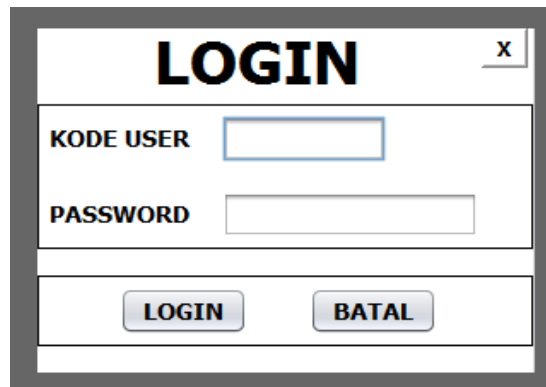
Gambar III.47
Sequence Diagram

3.4.5. Deployment Diagram



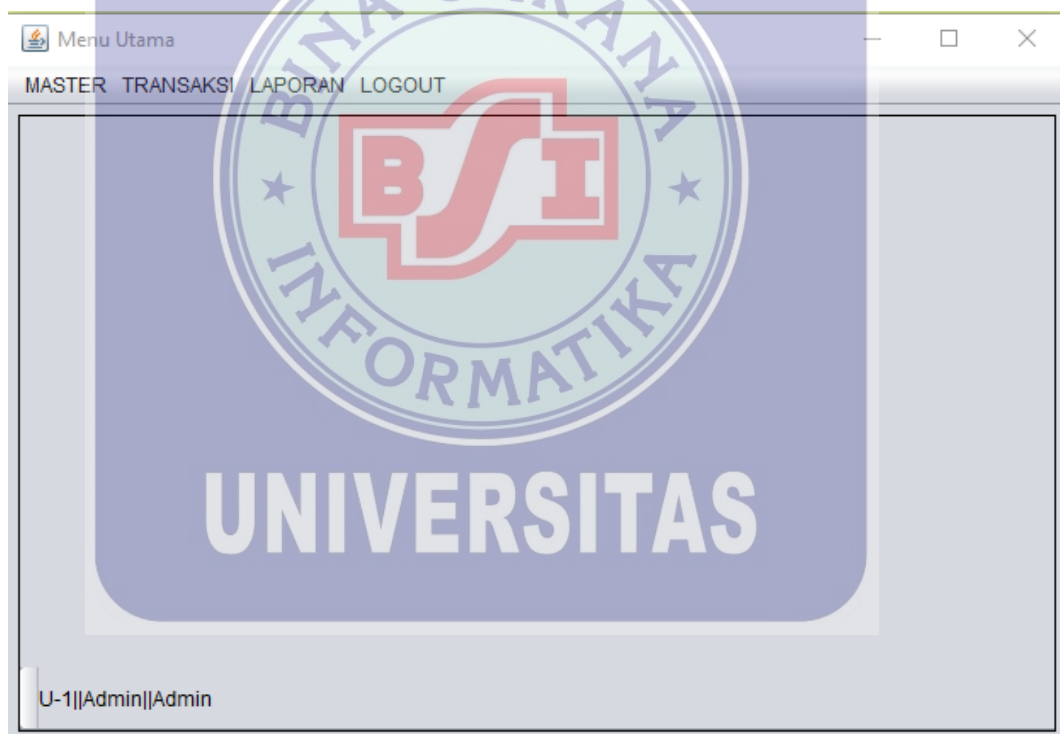
Gambar III.48
Deployment Diagram

3.4.6. User Interface



A screenshot of a login window titled "LOGIN" with a close button (X) in the top right corner. The window contains two input fields: "KODE USER" and "PASSWORD". Below the input fields are two buttons: "LOGIN" and "BATAL".

Gambar III.49
Halaman *Login*



Gambar III.50
Halaman Menu Utama

Form Master

MASTER USER

MASUKKAN DATA

KODE USER JABATAN

NAMA USER NO TELEPON

PASSWORD

TAMBAH **KELUAR**

CARI KODE

Kode User	Nama User	Password	Jabatan	No Telepon
U-1	Admin	11111	Admin	938479898
U-2	Pemilik	22222	Pemilik	22222

Gambar III.51
Halaman Master User

Form Master

MASTER SPAREPART

MASUKKAN DATA

KODE SPAREPART JUMLAH

NAMA SPAREPART HARGA

TAMBAH **KELUAR**

CARI KODE

Kode Sparepart	Nama Sparepart	Jumlah	Harga
SP1	BARANG 1	5	50000
SP2	BARANG 2	-5	60000

Gambar III.52
Halaman Master Sparepart

Form Master

MASTER JASA

MASUKKAN DATA

KODE JASA

NAMA JASA

HARGA

TAMBAH KELUAR

CARI KODE

Kode Customer	Nama Customer	No HP
J-1	JASA 1	10000
J-2	JASA 2	15000

Gambar III.53
Halaman Master Jasa

Form Master

MASTER CUSTOMER

MASUKKAN DATA

KODE CUSTOMER

NAMA CUSTOMER

NO HP

ALAMAT

TAMBAH KELUAR

CARI KODE

Kode Customer	Nama Customer	No HP	Alamat
SP1	CUSTOMER 1	3674328	KARAWANG
SP2	CUSTOMER 2	34565	SALAM 1

Gambar III.54
Halaman Master Customer

Form Master

MASTER AKUN

MASUKKAN DATA

ID AKUN

NAMA AKUN

JENIS AKUN

TAMBAH

KELUAR

CARI KODE

Kode Akun	Nama Akun	Jenis Akun

Gambar III.55
Halaman Master Akun

Kode Service

Tanggal 2020-01-16

Kode User U-1

Kode Customer

Nama Customer

No. HP

Alamat Customer:

TAMBAH **SIMPAN**

BATAL **KELUAR**

Kode Jasa	Nama Jasa	Harga	Sub Total

Kode obat	Nama obat	Harga	Jumlah Beli	Subtotal

Rp. Total

Gambar III.56
Halaman Transaksi Service

Gambar III.60
Halaman Laporan Penjualan

3.5. Implementasi

3.5.1. Code Generation

A. Form Transaksi Jual

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
```

```
package service;
```

```
import java.awt.event.KeyEvent;
import java.util.Date;
import java.sql.*;
import java.text.SimpleDateFormat;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
```

```
public class transaksiJual extends javax.swing.JFrame {
```

```
    koneksi kon = new koneksi();
    private Object[][] datasementara = null;
    private String[] labelsementara
        = {"Kode obat", "Nama obat", "Harga", "Jumlah Beli", "Subtotal"};
```

```
    public transaksiJual() {
        initComponents();
        kon.setKoneksi();
        setTanggal();
        awal();
```

```

    }

    public Date date = new Date();
    public SimpleDateFormat noformat = new SimpleDateFormat("yyMM");
    public String ksp, nsp, hg, kcs, ncs, noc, acs;
    public String KodeUser;

    private void awal() {
        nonaktif();
    }

    private void cetakStruk() {
        // cetakResep mt = new cetakResep();
        // mt.setLocationRelativeTo(null);
        // mt.setVisible(true);
        // mt.Kodeobatt = kode.getText();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        kode = new javax.swing.JTextField();
        jLabel2 = new javax.swing.JLabel();
        tgl = new javax.swing.JTextField();
        jLabel3 = new javax.swing.JLabel();
        kodeuser = new javax.swing.JTextField();
        jPanel2 = new javax.swing.JPanel();
        jPanel3 = new javax.swing.JPanel();
        jLabel4 = new javax.swing.JLabel();
        kodesp = new javax.swing.JTextField();
        bSP = new javax.swing.JButton();
        jLabel5 = new javax.swing.JLabel();
        namasp = new javax.swing.JTextField();
        jLabel6 = new javax.swing.JLabel();
        hrg = new javax.swing.JTextField();
        jLabel7 = new javax.swing.JLabel();
        qty = new javax.swing.JTextField();
        jLabel8 = new javax.swing.JLabel();
        subtotal = new javax.swing.JTextField();
        jScrollPane1 = new javax.swing.JScrollPane();
        tabel = new javax.swing.JTable();
    }

```

```

ltotal = new javax.swing.JLabel();
jLabel9 = new javax.swing.JLabel();
jPanel5 = new javax.swing.JPanel();
jLabel10 = new javax.swing.JLabel();
kodecustomer = new javax.swing.JTextField();
jLabel13 = new javax.swing.JLabel();
jLabel14 = new javax.swing.JLabel();
hp = new javax.swing.JTextField();
jLabel15 = new javax.swing.JLabel();
namacustomer = new javax.swing.JTextField();
jScrollPane2 = new javax.swing.JScrollPane();
alamat = new javax.swing.JTextArea();
bCus = new javax.swing.JButton();
bttambah = new javax.swing.JButton();
btsimpan = new javax.swing.JButton();
btbatal = new javax.swing.JButton();
btkeluar = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
setTitle("Penjualan");
setBackground(java.awt.Color.green);
setLocationByPlatform(true);
setUndecorated(true);
addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowActivated(java.awt.event.WindowEvent evt) {
        formWindowActivated(evt);
    }
});

jPanel1.setBorder(javax.swing.BorderFactory.createEtchedBorder());
jPanel1.setCursor(new java.awt.Cursor(java.awt.Cursor.MOVE_CURSOR));
jPanel1.addMouseMotionListener(new java.awt.event.MouseMotionAdapter() {
    public void mouseDragged(java.awt.event.MouseEvent evt) {
        jPanel1MouseDragged(evt);
    }
});
jPanel1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mousePressed(java.awt.event.MouseEvent evt) {
        jPanel1MousePressed(evt);
    }
    public void mouseReleased(java.awt.event.MouseEvent evt) {
        jPanel1MouseReleased(evt);
    }
});

jLabel1.setText("Kode Jual");

kode.setEditable(false);

```

```

jLabel2.setText("Tanggal");

tgl.setEditable(false);

jLabel3.setText("Kode User");

kodeuser.setEditable(false);

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .add(jLabel1)
            .add(jLabel2)
            .add(jLabel3)
            .add(kodeuser)
        )
        .addContainerGap()
    );
jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .add(jLabel1)
        .add(jLabel2)
        .add(jLabel3)
        .add(kodeuser)
    )
);

```

```

        .addComponent(kode, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel2)
        .addComponent(tgl, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel3)
        .addComponent(kodeuser,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    jPanel2.setBorder(new
javax.swing.border.SoftBevelBorder(javax.swing.border.BevelBorder.RAISED));

jPanel3.setBorder(javax.swing.BorderFactory.createTitledBorder(javax.swing.Borde
rFactory.createEtchedBorder(), "Sparepart"));

jLabel4.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel4.setText("Kode SP");

kodesp.setEditable(false);
kodesp.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        kodespActionPerformed(evt);
    }
});

bSP.setText("...");
bSP.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        bSPActionPerformed(evt);
    }
});

jLabel5.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel5.setText("Nama SP");

namasp.setEditable(false);

jLabel6.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel6.setText("Harga");

hrg.setEditable(false);

```



```

jLabel7.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel7.setText("Qty");

qty.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        qtyActionPerformed(evt);
    }
});
qty.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent evt) {
        qtyKeyTyped(evt);
    }
});

jLabel8.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel8.setText("Sub Total");

subtotal.setEditable(false);

javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
jPanel3.setLayout(jPanel3Layout);
jPanel3Layout.setHorizontalGroup(
    jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
            .add(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .add(jPanel3Layout.createSequentialGroup()
                    .add(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                        .add(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE,
                            javax.swing.GroupLayout.PREFERRED_SIZE,
                                javax.swing.GroupLayout.PREFERRED_SIZE)
                        .add(jLabel5, javax.swing.GroupLayout.PREFERRED_SIZE,
                            javax.swing.GroupLayout.PREFERRED_SIZE,
                                javax.swing.GroupLayout.PREFERRED_SIZE)
                    )
                    .add(jLabel6, javax.swing.GroupLayout.PREFERRED_SIZE,
                        javax.swing.GroupLayout.PREFERRED_SIZE,
                            javax.swing.GroupLayout.PREFERRED_SIZE)
                )
            )
        )
    );

```



```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(hrg))
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addComponent(jLabel5,
            javax.swing.GroupLayout.PREFERRED_SIZE,          101,
            javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
    .addComponent(jLabel6,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
    .addComponent(qty)
    .addComponent(jLabel7, javax.swing.GroupLayout.DEFAULT_SIZE,
37, Short.MAX_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
    .addComponent(subtotal)
    .addComponent(jLabel8, javax.swing.GroupLayout.DEFAULT_SIZE,
111, Short.MAX_VALUE))
    .addContainerGap()
);
jPanel3Layout.setVerticalGroup(
jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addContainerGap()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
    .addGroup(jPanel3Layout.createSequentialGroup()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
    .addComponent(jLabel4)
    .addComponent(jLabel5)
    .addComponent(jLabel6))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```



```

        {},
        {}
    },
    new String [] {

    }
});
tabel.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
tabel.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyPressed(java.awt.event.KeyEvent evt) {
        tabelKeyPressed(evt);
    }
});
jScrollPane1.setViewportViewView(tabel);

ltotal.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N
ltotal.setHorizontalAlignment(javax.swing.SwingConstants.LEFT);
ltotal.setText("Total");

jLabel9.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N
jLabel9.setText("Rp.");

javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
    .addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE,
526, Short.MAX_VALUE)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addComponent(jLabel9)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(ltotal, javax.swing.GroupLayout.PREFERRED_SIZE,
220, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE))
    );
jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(18, 18, 18)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
    .addComponent(ltotal, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(jLabel9,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
);

jPanel5.setBorder(javax.swing.BorderFactory.createEtchedBorder());

jLabel10.setText("Kode Customer");
kodecustomer.setEditable(false);

jLabel13.setText("Alamat Customer:");
jLabel14.setText("No. HP");
hp.setEditable(false);

jLabel15.setText("Nama Customer");
namacustomer.setEditable(false);
alamat.setEditable(false);
alamat.setColumns(20);
alamat.setRows(5);
jScrollPane2.setViewportView(alamat);

bCus.setText("...");
bCus.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        bCusActionPerformed(evt);
    }
});

bttambah.setText("TAMBAH");
bttambah.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        btTambahActionPerformed(evt);
    }
});

btsimpan.setText("SIMPAN");
btsimpan.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btsimpanActionPerformed(evt);
    }
});

btbatal.setText("BATAL");
btbatal.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btbatalActionPerformed(evt);
    }
});

btkeluar.setText("KELUAR");
btkeluar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btkeluarActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel5Layout = new
javax.swing.GroupLayout(jPanel5);
jPanel5.setLayout(jPanel5Layout);
jPanel5Layout.setHorizontalGroup(
    jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel5Layout.createSequentialGroup()
            .addContainerGap()
            .addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 278,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jScrollPane2,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 278,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGroup(jPanel5Layout.createSequentialGroup()
                    .addComponent(btTambah,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 100,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(10, 10, 10)
                    .addComponent(btBatal,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 100,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(10, 10, 10)
                    .addComponent(btKeluar,
                        javax.swing.GroupLayout.PREFERRED_SIZE, 100,
                        javax.swing.GroupLayout.PREFERRED_SIZE))
            .addContainerGap(10, Short.MAX_VALUE))
        .addGroup(jPanel5Layout.createSequentialGroup()
            .addComponent(btSimpan,
                javax.swing.GroupLayout.PREFERRED_SIZE, 100,
                javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(10, Short.MAX_VALUE))
);

```



```

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
    .addComponent(jLabel10)
    .addComponent(jLabel15)
    .addComponent(jLabel14))
.addGap(18, 18, 18)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
    .addGroup(jPanel5Layout.createSequentialGroup()
        .addComponent(kodecustomer,
javax.swing.GroupLayout.PREFERRED_SIZE, 96,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(bCus,
javax.swing.GroupLayout.PREFERRED_SIZE, 46,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addComponent(namacustomer)
        .addComponent(hp))))
    .addGroup(jPanel5Layout.createSequentialGroup()
        .addContainerGap()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
TRAILING)
        .addComponent(bttambah,
javax.swing.GroupLayout.PREFERRED_SIZE, 89,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btbatalkan,
javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
        .addComponent(btsimpan,
javax.swing.GroupLayout.DEFAULT_SIZE, 90, Short.MAX_VALUE)
        .addComponent(btkeluar,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))))
        .addGap(4, 4, 4))
);
jPanel5Layout.setVerticalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
    .addGroup(jPanel5Layout.createSequentialGroup()
        .addGap(11, 11, 11)

```

```

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
    .addGroup(jPanel5Layout.createSequentialGroup()
        .addComponent(jLabel10,
javax.swing.GroupLayout.PREFERRED_SIZE,                29,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(12, 12, 12)
        .addComponent(jLabel15,
javax.swing.GroupLayout.PREFERRED_SIZE,                26,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(6, 6, 6)
        .addComponent(jLabel14,
javax.swing.GroupLayout.PREFERRED_SIZE,                27,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGroup(jPanel5Layout.createSequentialGroup()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
    .addComponent(kodecustomer,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(bCus))
    .addGap(15, 15, 15)
    .addComponent(namacustomer,
javax.swing.GroupLayout.PREFERRED_SIZE,                23,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(6, 6, 6)
    .addComponent(hp, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
    .addGap(6, 6, 6)
    .addComponent(jLabel13)
    .addGap(12, 12, 12)
    .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
    .addComponent(bttambah)
    .addComponent(btsimpan))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```



```

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
    .addComponent(btkeluar)
    .addComponent(btatal))
.addGap(14, 14, 14))
);

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addComponent(jPanel5, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(4, 4, 4)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jPanel5, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
    );

pack();
} // </editor-fold>

private void kodespActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    tampilDataobat();
}

private void bSPActionPerformed(java.awt.event.ActionEvent evt) {

```

```

// TODO add your handling code here:
boolean closable = true;
dataSP db = new dataSP(null, closable);
db.transaksi = this;
db.setVisible(true);
db.setResizable(true);
kodesp.setText(ksp);
namasp.setText(nsp);
hrg.setText(hg);
if (!kodesp.getText().isEmpty()) {
    qty.requestFocus();
}
}

private void btkeluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    hapusTabelSementara();
    dispose();
}

private void bttambahActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    aktif();
    kode.setText(nomor());
}

private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    kodeuser.setText(KodeUser);
    tampilTabelSementara();
}

private void btbatalActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    awal();
    hapusTabelSementara();
    kode.setText("");
}

private void tabelKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == KeyEvent.VK_ENTER
        || evt.getKeyCode() == KeyEvent.VK_BACK_SPACE
        || evt.getKeyCode() == KeyEvent.VK_DELETE) {
        hapusIsiSementara();
        qty.setText("");
        qty.requestFocus();
    }
}
}

```

```

private void qtyActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int harga, jumbel, total;
    harga = Integer.parseInt(hrg.getText());
    jumbel = Integer.parseInt(qty.getText());
    total = harga * jumbel;
    subtotal.setText(Integer.toString(total));
    simpanSementara();
    tampilTabelSementara();
    int ttl = 0;
    for (int a = 0; a < tabel.getRowCount(); a++) {
        int sub = Integer.parseInt((String) tabel.getValueAt(a, 4));
        ttl += sub;
    }
    ltotal.setText(Integer.toString(ttl));
    if (JOptionPane.showConfirmDialog(this, "Ada lagi?",
        "Konfirmasi", JOptionPane.YES_NO_OPTION) ==
JOptionPane.YES_OPTION) {
        aktif();
        kodesp.requestFocus();
        kodesp.setText("");
        namasp.setText("");
        hrg.setText("");
        qty.setText("");
        subtotal.setText("");
    } else {
        btsimpan.requestFocus();
    }
}

private void btsimpanActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if (qty.getText().equals("") || kodecustomer.getText().equals("")
        || alamat.getText().equals("") || namacustomer.getText().equals("")
        || hp.getText().equals("")) {
        JOptionPane.showMessageDialog(this, "Masih ada data yang belum terisi",
            "Informasi", JOptionPane.INFORMATION_MESSAGE);
        kodecustomer.requestFocus();
    } else {
        simpanpenjualan();
        simpanDetailpenjualan();
        JOptionPane.showMessageDialog(this, "Berhasil di Simpan",
            "Informasi", JOptionPane.INFORMATION_MESSAGE);
        tampilTabelSementara();
        hapusTabelSementara();
        awal();
        bersih();
    }
}

```

```

    }
}

private void qtyKeyTyped(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    char c = evt.getKeyChar();
    if (!((c >= '0') && (c <= '9') && qty.getText().length() < 4
        || (c == com.sun.glass.events.KeyEvent.VK_BACKSPACE)
        || (c == com.sun.glass.events.KeyEvent.VK_DELETE))) {
        getToolkit().beep();
        evt.consume();
    }
}

int v, h;
private void jPanel1MouseDragged(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int a, b;
    a = evt.getXOnScreen();
    b = evt.getYOnScreen();
    this.setLocation(a - v, b - h);
}

private void jPanel1MouseReleased(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
}

private void jPanel1MousePressed(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    v = evt.getX();
    h = evt.getY();
}

private void bCusActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    boolean closable = true;
    dataCus db = new dataCus(null, closable);
    db.transaksi = this;
    db.setVisible(true);
    db.setResizable(true);
    kodecustomer.setText(kcs);
    namacustomer.setText(ncs);
    hp.setText(noc);
    alamat.setText(acs);
    if (!kodecustomer.getText().isEmpty()) {
        bSP.grabFocus();
    }
}
}

```


[illegible]


```

private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JPanel jPanel5;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JTextField kode;
private javax.swing.JTextField kodecustomer;
private javax.swing.JTextField kodesp;
private javax.swing.JTextField kodeuser;
private javax.swing.JLabel lbltotal;
private javax.swing.JTextField namacustomer;
private javax.swing.JTextField namasp;
private javax.swing.JTextField qty;
private javax.swing.JTextField subtotal;
private javax.swing.JTable tabel;
private javax.swing.JTextField tgl;
// End of variables declaration
private void bersih() {
    kode.setText("");
    namasp.setText("");
    qty.setText("");
    hrg.setText("");
    subtotal.setText("");
    lbltotal.setText("0");

    kodecustomer.setText("");
    namacustomer.setText("");
    alamat.setText("");
    hp.setText("");
}

private void nonaktif() {
    bCus.setEnabled(false);
    bSP.setEnabled(false);
}

private void aktif() {
    bCus.setEnabled(true);
    bSP.setEnabled(true);
}

private void setTanggal() {
    java.util.Date skrg = new java.util.Date();
    java.text.SimpleDateFormat kal = new java.text.SimpleDateFormat("yyyy-MM-
dd");

```



```

        tgl.setText(kal.format(skrng));
    }

    public String nomor() {
        String urutan = null;
        try {
            kon.rs = kon.st.executeQuery("select right(kodejual,3)+1 from jual");
            if (kon.rs.next()) {
                kon.rs.last();
                urutan = kon.rs.getString(1);
                while (urutan.length() < 3) {
                    urutan = "0" + urutan;
                }
                urutan = "NOJ" + noformat.format(date) + urutan;
            } else {
                urutan = "NOJ" + noformat.format(date) + "001";
            }
        } catch (SQLException e) {
            JOptionPane.showMessageDialog(null, "Mungkin Database Anda: " + e);
        }
        return urutan;
    }

    private void tampilTabelSementara() {
        try {
            String sql = "Select * from sementara order by kodesparepart";
            kon.rs = kon.st.executeQuery(sql);
            ResultSetMetaData m = kon.rs.getMetaData();
            int kolom = m.getColumnCount();
            int baris = 0;
            while (kon.rs.next()) {
                baris = kon.rs.getRow();
            }
            datasementara = new Object[baris][kolom];
            int x = 0;
            kon.rs.beforeFirst();
            while (kon.rs.next()) {
                datasementara[x][0] = kon.rs.getString("kodesparepart");
                datasementara[x][1] = kon.rs.getString("namasperepart");
                datasementara[x][2] = kon.rs.getString("harga");
                datasementara[x][3] = kon.rs.getString("jumlahbeli");
                datasementara[x][4] = kon.rs.getString("subtotal");
                x++;
            }
            tabel.setModel(new DefaultTableModel(datasementara, labelsementara));
        } catch (SQLException e) {
            JOptionPane.showMessageDialog(null, e);
        }
    }

```

```

    }
}

private void tampilDataobat() {
    try {
        String sql = "select * from sparepart where kodesparepart =" + kodesp.getText() + """;
        kon.rs = kon.st.executeQuery(sql);
        if (kon.rs.next()) {
            namasp.setText(kon.rs.getString("namasparepart"));
            hrg.setText(kon.rs.getString("harga"));
            qty.requestFocus();
        } else {
            JOptionPane.showMessageDialog(null, "Kode " + kodesp.getText() + "tidak ditemukan");
        }
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}

private void simpanSementara() {
    try {
        String sql = "insert into sementara values (" + kodesp.getText() + "," + namasp.getText() + "," + hrg.getText() + "," + qty.getText() + "," + subtotal.getText() + ")";
        kon.st.executeUpdate(sql);
        tampilTabelSementara();
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}

private void hapusIsiSementara() {
    int row = tabel.getSelectedRow();
    int x;
    int total = Integer.parseInt(ltotal.getText());
    x = Integer.parseInt((String) tabel.getValueAt(row, 4));
    total = total - x;
    ltotal.setText(Integer.toString(total));
    try {
        String sql = "delete from sementara where kodesparepart =" + (String) tabel.getValueAt(row, 0) + """;
        kon.st.executeUpdate(sql);
        tampilTabelSementara();
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}

```

```

private void simpanpenjualan() {
    try {
        String sql = "insert into jual values('" + kode.getText() + "','"
            + tgl.getText() + "','" + ltotol.getText() + "','" + kodeuser.getText()
            + "','" + kodecustomer.getText() + "')";
        kon.st.executeUpdate(sql);
    } catch (SQLException e) {
        System.out.println("Koneksi Gagal " + e.toString());
    }
}

private void simpanDetailpenjualan() {
    try {
        String sql = "insert into detailjual select '" + kode.getText()
            + "','" + kodesparepart + "','" + jumlahbeli + "','" + subtotal + " from sementara";
        kon.st.executeUpdate(sql);
    } catch (SQLException e) {
        System.out.println("Koneksi Gagal " + e.toString());
    }
}

private void hapusTabelSementara() {
    try {
        String sql = "Delete from sementara";
        kon.st.executeUpdate(sql);
        tampilTabelSementara();
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, e);
    }
}
}
}

```

B. Form Transaksi Pembayaran

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package service;

```

```

import com.sun.glass.events.KeyEvent;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.HashMap;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

```

```

import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.view.JasperViewer;

public class transaksiPembayaran extends javax.swing.JFrame {

    koneksi kon = new koneksi();
    String sql;
    ResultSet rs;
    Object header[]
        = {"Kode Pembayaran", "Tanggal", "Kode User", "Kode Customer",
            "Kode Service", "Kode Jual", "Total Bayar"};

    public transaksiPembayaran() {
        initComponents();
        kon.setKoneksi();
        nonaktif();
    }

    public Date date = new Date();
    public SimpleDateFormat noformat = new SimpleDateFormat("yyMM");

    private void bersih() {
        kodeservice.setText("");
        kodejual.setText("");
        total.setText("0");
        totaljasa.setText("0");
        totaljual.setText("0");
    }

    private void aktif() {
        bS.setEnabled(true);
        bJ.setEnabled(true);
    }

    private void nonaktif() {
        bS.setEnabled(false);
        bJ.setEnabled(false);
    }

    private void tampil() {
        DefaultTableModel data = new DefaultTableModel(null, header);
        tabel.setModel(data);

        kon.setKoneksi();
        sql = "select * from pembayaran order by kodebayar desc";
        try {
            rs = kon.st.executeQuery(sql);
            while (rs.next()) {

```

```

        String kolom1 = rs.getString(1);
        String kolom2 = rs.getString(2);
        String kolom3 = rs.getString(3);
        String kolom4 = rs.getString(4);
        String kolom5 = rs.getString(5);
        String kolom6 = rs.getString(6);
        String kolom7 = rs.getString(7);

        String kolom[] = {kolom1, kolom2, kolom3, kolom4, kolom5, kolom6,
        kolom7};
        data.addRow(kolom);
    }
    } catch (SQLException e) {
    }
}

private void simpanData() {
    try {
        sql = "insert into pembayaran values('" + kode.getText()
            + "','" + tgl.getText()
            + "','" + kodeuser.getText()
            + "','" + kodecustomer.getText()
            + "','" + kodeservice.getText()
            + "','" + kodejual.getText()
            + "','" + total.getText()
            + "')";

        kon.st.executeUpdate(sql);
        tampil();
    } catch (SQLException e) {
    }
}

public String nomor() {
    String urutan = null;
    try {
        kon.rs = kon.st.executeQuery("select right(kodebayar,3)+1 from
        pembayaran");
        if (kon.rs.next()) {
            kon.rs.last();
            urutan = kon.rs.getString(1);
            while (urutan.length() < 3) {
                urutan = "0" + urutan;
            }
            urutan = "NOB" + noformat.format(date) + urutan;
        } else {
            urutan = "NOB" + noformat.format(date) + "001";
        }
    }
}

```

```

    }
    } catch (SQLException e) {
        JOptionPane.showMessageDialog(null, "Mungkin Database Anda: " + e);
    }
    return urutan;
}

private void setTanggal() {
    java.util.Date skrg = new java.util.Date();
    java.text.SimpleDateFormat kal = new java.text.SimpleDateFormat("yyyy-MM-
dd");
    tgl.setText(kal.format(skrg));
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    kode = new javax.swing.JTextField();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    tgl = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    kodeuser = new javax.swing.JTextField();
    jSeparator1 = new javax.swing.JSeparator();
    jLabel5 = new javax.swing.JLabel();
    kodeservice = new javax.swing.JTextField();
    jLabel6 = new javax.swing.JLabel();
    total = new javax.swing.JTextField();
    bS = new javax.swing.JButton();
    jLabel12 = new javax.swing.JLabel();
    tambah = new javax.swing.JButton();
    keluar = new javax.swing.JButton();
    jScrollPane2 = new javax.swing.JScrollPane();
    tabel = new javax.swing.JTable();
    jLabel7 = new javax.swing.JLabel();
    kodejual = new javax.swing.JTextField();
    BJ = new javax.swing.JButton();
    totaljual = new javax.swing.JLabel();
    totaljasa = new javax.swing.JLabel();
    jLabel8 = new javax.swing.JLabel();
    kodecustomer = new javax.swing.JTextField();

```



```
bC = new javax.swing.JButton();
```

```
setDefaultCloseOperation(javax.swing.WindowConstants.DO_NOTHING_ON_CLOSE);
```

```
setTitle("Transaksi Pembayaran");
addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowActivated(java.awt.event.WindowEvent evt) {
        formWindowActivated(evt);
    }
});
```

```
kode.setEditable(false);
```

```
jLabel2.setText("Kode Bayar");
```

```
jLabel3.setText("Tanggal");
```

```
tgl.setEditable(false);
```

```
jLabel4.setText("Kode User");
```

```
kodeuser.setEditable(false);
```

```
jLabel5.setText("Kode Service");
```

```
kodeservice.setEditable(false);
```

```
jLabel6.setText("Total Bayar");
```

```
bS.setText("...");
```

```
bS.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        bSActionPerformed(evt);
    }
});
```

```
jLabel12.setText("TRANSAKSI PEMBAYARAN");
```

```
tambah.setText("TAMBAH");
```

```
tambah.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tambahActionPerformed(evt);
    }
});
```

```
keluar.setText("KELUAR");
```

```
keluar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```



```

        keluarActionPerformed(evt);
    }
});

tabel.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "Title 1", "Title 2", "Title 3", "Title 4"
    }
));
tabel.setToolTipText("Tekan Enter untuk cetak");
tabel.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        tabelMouseClicked(evt);
    }
});
tabel.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyPressed(java.awt.event.KeyEvent evt) {
        tabelKeyPressed(evt);
    }
});
jScrollPane2.setViewportViewView(tabel);

jLabel7.setText("Kode Jual");

kodejual.setEditable(false);

bJ.setText("...");
bJ.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        bJActionPerformed(evt);
    }
});

totaljual.setText("0");

totaljasa.setText("0");

jLabel8.setText("Kode Customer");

kodecustomer.setEditable(false);

bC.setText("...");
bC.addActionListener(new java.awt.event.ActionListener() {

```



```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
```

```
    .addComponent(jLabel5)
    .addGroup(jPanel1Layout.createSequentialGroup())
    .addGap(2, 2, 2)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
```

```
    .addComponent(jLabel6)
    .addComponent(jLabel7))))
    .addGap(37, 37, 37)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
TRAILING)
```

```
    .addComponent(kodejual,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE) 133,
```

```
.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
```

```
    .addComponent(kodeservice,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 130, Short.MAX_VALUE)
    .addComponent(kodecustomer,
javax.swing.GroupLayout.Alignment.LEADING)))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
```

```
    .addComponent(bS,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE) 51,
```

```
    .addComponent(bJ,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE) 51,
```

```
    .addComponent(bC,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))) 51,
```

```
    .addComponent(total)))
    .addComponent(jLabel8))
    .addGap(21, 21, 21))
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```

        .addComponent(totaljual)
        .addGap(18, 18, 18)
        .addComponent(totaljasa)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
    javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(tambah,
    javax.swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(keluar,
    javax.swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(30, 30, 30)))
        .addComponent(jScrollPane2))
    );
    jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addComponent(jLabel12)

    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
    LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(3, 3, 3)

    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
    LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(8, 8, 8)
            .addComponent(jLabel2))
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(3, 3, 3)
            .addComponent(kode,
    javax.swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE,
    javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(5, 5, 5)
            .addComponent(jLabel3))
        .addComponent(tgl,
    javax.swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE,
    javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(5, 5, 5)

```

```

        .addComponent(jLabel4))
        .addComponent(kodeuser,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(6, 6, 6)
        .addComponent(jSeparator1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(7, 7, 7)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(2, 2, 2)
        .addComponent(jLabel8))

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
        .addComponent(bc)
        .addComponent(kodecustomer,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(2, 2, 2)
        .addComponent(jLabel5)
        .addGap(61, 61, 61)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
        .addComponent(jLabel6)
        .addComponent(total,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
        .addComponent(bs)
        .addComponent(kodeservice,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(11, 11, 11)

```



```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(kodejual,
        javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel7)
    .addComponent(bJ))))
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(12, 12, 12)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(totaljasa)
    .addComponent(totaljual)))
    .addGroup(jPanel1Layout.createSequentialGroup()
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(keluar)
    .addComponent(tambah))))
```

```
.addGroup(jPanel1Layout.createSequentialGroup()
    .addGap(67, 67, 67)
    .addComponent(jScrollPane2,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE)))
    .addContainerGap()
```

```
);
```

```
        javax.swing.GroupLayout layout = new
        javax.swing.GroupLayout(getContentPane());
```

```
        getContentPane().setLayout(layout);
```

```
        layout.setHorizontalGroup
```

```
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
                .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
```

```
                javax.swing.GroupLayout.DEFAULT_SIZE,
```

```
                javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
        );
```

```
        layout.setVerticalGroup
```

```
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
```

```
                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
        );
```

```

        pack();
    } // </editor-fold>
String KodeUser;
private void formWindowActivated(java.awt.event.WindowEvent evt) {
    // TODO add your handling code here:
    setTanggal();
    tampil();
    kodeuser.setText(KodeUser);
    totaljasa.setVisible(false);
    totaljual.setVisible(false);
}

private void tambahActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if (tambah.getText().trim().equals("TAMBAH")) {
        aktif();
        kode.setText(nomor());
        tambah.setText("SIMPAN");
        keluar.setText("BATAL");
        bS.grabFocus();
        tambah.setEnabled(false);
    } else if (tambah.getText().trim().equals("SIMPAN")) {
        if (kode.getText().trim().equals("") || total.getText().isEmpty()) {
            JOptionPane.showMessageDialog(null, "Masih ada kolom yang belum
terisi");
        }
        bersih();
        aktif();
        tambah.setText("SIMPAN");
        keluar.setText("BATAL");
    } else {
        simpanData();
        JOptionPane.showMessageDialog(this, "Berhasil di Simpan",
            "Informasi", JOptionPane.INFORMATION_MESSAGE);
        //
        tampil();
        if (JOptionPane.showConfirmDialog(this, "Mau Cetak Surat
Pembayaran?", "Konfirmasi",
            JOptionPane.YES_NO_OPTION) == JOptionPane.YES_OPTION) {

            cetak();
            bersih();
            nonaktif();
            tambah.setText("TAMBAH");
            keluar.setText("KELUAR");
        } else {
            bersih();
            nonaktif();
            tambah.setText("TAMBAH");
        }
    }
}

```



```

        keluar.setText("KELUAR");
    }
}
}

private void keluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if (keluar.getText().equals("KELUAR")) {
        this.dispose();
    } else if (keluar.getText().equals("BATAL")) {
        tampil();
        bersih();
        nonaktif();
        tambah.setText("TAMBAH");
        tambah.setEnabled(true);
        keluar.setText("KELUAR");
    }
}

String kjs, tbs;
private void bSActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    boolean closable = true;
    dataTJasa db = new dataTJasa(null, closable);
    db.kd = kodecustomer.getText();
    db.transaksi = this;
    db.setVisible(true);
    db.setResizable(true);
    kodeservice.setText(kjs);
    totaljasa.setText(tbs);
    if (!totaljasa.getText().isEmpty()) {
        jumlah();
    }
}

private void tabelKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == KeyEvent.VK_ENTER) {
        cetak();
    }
}

private void cetak() {
    try {
        String namaFile = "src/service/main-pembayaran.jasper";
        HashMap parameter = new HashMap();
        parameter.put("kode", kode.getText());
        JasperPrint cetak = JasperFillManager.fillReport(namaFile, parameter,
kon.setKoneksi());
    }
}

```

```

        JasperViewer.viewReport(cetak, false);
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e.getMessage());
        System.out.println(e);
    }
}

private void tabelMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int row = tabel.getSelectedRow();
    kode.setText((String) tabel.getValueAt(row, 0));
}

String kj, tb;
private void bJActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    boolean closable = true;
    dataTJual db = new dataTJual(null, closable);
    db.kd = kodecustomer.getText();
    db.transaksi = this;
    db.setVisible(true);
    db.setResizable(true);
    kodejual.setText(kj);
    totaljual.setText(tb);
    if (!totaljual.getText().isEmpty()) {
        jumlah();
    }
}

String kcus;
private void bCActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    boolean closable = true;
    dataCustomer db = new dataCustomer(null, closable);
    db.form = this;
    db.setVisible(true);
    db.setResizable(true);
    kodecustomer.setText(kcus);
    if (!kodecustomer.getText().isEmpty()) {
        bJ.setEnabled(true);
        bS.setEnabled(true);
        tambah.setEnabled(true);
    }
}

private void jumlah() {
    int hasil, a = 0, b = 0;
    a = Integer.parseInt(totaljasa.getText());
    b = Integer.parseInt(totaljual.getText());
    hasil = a + b;
    total.setText(Integer.toString(hasil));
}

```



```

//</editor-fold>
//</editor-fold>
//</editor-fold>
//</editor-fold>
//</editor-fold>
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new transaksiPembayaran().setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JButton bC;
private javax.swing.JButton bJ;
private javax.swing.JButton bS;
private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JSeparator jSeparator1;
private javax.swing.JButton keluar;
private javax.swing.JTextField kode;
private javax.swing.JTextField kodecustomer;
private javax.swing.JTextField kodejual;
private javax.swing.JTextField kodeservice;
private javax.swing.JTextField kodeuser;
private javax.swing.JTable tabel;
private javax.swing.JButton tambah;
private javax.swing.JTextField tgl;
private javax.swing.JTextField total;
private javax.swing.JLabel totaljasa;
private javax.swing.JLabel totaljual;
// End of variables declaration
}

```

3.5.2. Blackbox Testing

1. Pengujian *Form* Transaksi Kwitansi

Tabel III.29

Hasil Pengujian *Form* Transaksi Kwitansi

Kelas Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Kesimpulan
Menu Kwitansi	Mengosongkan semua isian data penjualan, kemudian klik tombol “simpan”	Sistem akan menampilkan pesan “Masih ada data yang kosong”	Valid
	Hanya mengisi data kode obat, nama obat, harga obat dan mengosongkan uang bayar, kemudian langsung klik tombol “simpan”	Sistem akan menampilkan pesan “silahkan masukkan uang bayar”	Valid
	Mengisi semua data, kemudian langsung klik tombol “simpan”	Sistem akan menampilkan pesan “Berhasil disimpan, mau cetak?”	True

2. Pengujian *Master Form* Data Mesin Diesel

Tabel III.30

Hasil Pengujian *Black Box Testing* Halaman Data Karyawan

Kelas Pengujian	Skenario yang diharapkan	Hasil yang diharapkan	Kesimpulan
Semua Kolom tidak diisi	Sistem akan menampilkan pesan “Masih ada data yang kosong”	Sesuai harapan	Valid
Salah satu kolom atau beberapa tidak diisi	Sistem akan menampilkan pesan “Masih ada data yang kosong”	Sesuai harapan	Valid
Semua kolom diisi dengan baik	Sistem akan menampilkan pesan “Data berhasil disimpan”	Sesuai harapan	Valid

Mengubah data yang telah disimpan	Sistem akan menampilkan pesan “Data berhasil diubah”	Sesuai harapan	Valid
Mengubah data yang telah disimpan	Sistem akan menampilkan pesan “Data berhasil dihapus”	Sesuai harapan	Valid

3.5.3. Spesifikasi *Hardware* dan *Software*

Spesifikasi *software* (Perangkat Lunak) adalah sesuatu yang berhubungan dengan program komputer. Dalam membuat program aplikasi ini, penulis menggunakan beberapa *software* yaitu:

Sistem Operasi : Microsoft Windows

Bahasa Pemograman : Java

Aplikasi Pendukung : NetBeans IDE 8.2, XAMPP, JASPERREPORT

DBMS/ Basis Data : MySQL

Spesifikasi *Hardware* merupakan perangkat keras komputer, seperti monitor, printer, keyboard, mouse, dan lain-lain. Minimal dalam spesifikasi file untuk dapat membuat dan menjalankan aplikasi ini adalah sebagai berikut:

1. Processor : Intel Pentium
2. RAM : 4 GB
3. Monitor : ASUS
4. Printer : CANON IP-2700
5. Hardisk : 500 GB
6. Keyboard : ALTEC KEYBOARD