

# **DOKUMEN ANALISIS DAN DESAIN PERANGKAT LUNAK (ADPL)**

## **Sistem Pengelola Toko**

### **1. Konsep Dasar & SDLC**

#### **a. Konsep Dasar**

Sistem Pengelola Toko Adalah Sebuah Aplikasi Yang Dirancang Untuk Membantu Proses Pengelolaan Barang Dan Transaksi Penjualan Di Sebuah Toko. Dengan Adanya Sistem Ini, Pemilik Maupun Kasir Dapat Lebih Mudah Dalam Mencatat Data Barang, Mengatur Stok, Melakukan Transaksi Penjualan, Hingga Menghasilkan Laporan Penjualan Secara Otomatis.

Sistem Ini Akan Memiliki Dua Jenis Pengguna Utama, Yaitu Admin/Pemilik Toko Dan Kasir. Admin Bertugas Mengelola Data Barang, Memantau Stok, Serta Melihat Laporan Penjualan, Sedangkan Kasir Bertugas Melakukan Pencatatan Transaksi Penjualan Dan Mencetak Struk Pembelian.

Selain Itu, Sistem Ini Juga Dilengkapi Dengan Mekanisme Pengurangan Stok Otomatis Setiap Kali Terjadi Transaksi Penjualan. Dengan Begitu, Risiko Kesalahan Pencatatan Stok Dapat Diminimalisir. Sistem Juga Dirancang Dengan Tampilan Yang Sederhana, Mudah Dipahami, Serta Memiliki Fitur Keamanan Berupa Login Untuk Membatasi Akses Hanya Kepada Pengguna Yang Berhak.

Dengan Adanya Sistem Pengelola Toko Ini, Diharapkan Proses Pengelolaan Toko Dapat Berjalan Lebih Cepat, Akurat, Dan Efisien Dibandingkan Dengan Pencatatan Manual.

#### **b. SDLC (System Development Life Cycle)**

System Development Life Cycle (SDLC) adalah metode yang digunakan untuk mengembangkan sistem secara terstruktur. Model SDLC yang digunakan pada sistem ini adalah Waterfall, karena kebutuhan sistem sudah jelas dan tidak sering berubah. Tahapan SDLC meliputi perencanaan, analisis kebutuhan, desain

sistem, pengujian, dan pemeliharaan. Pada dokumen ini, pembahasan difokuskan hingga tahap desain perangkat lunak

## 2. Analisis Kebutuhan

### a. Kebutuhan Pengguna

#### a) Admin / Pemilik Toko

- Mengelola data barang (tambah, ubah, hapus).
- Mengelola data user (kasir).
- Melihat laporan penjualan dan laporan stok.

#### b) Kasir

- Login ke sistem.
- Melakukan transaksi penjualan dengan cepat.
- Mencetak struk pembelian.

#### c) Pelanggan

- Mendapatkan pelayanan yang cepat.
- Menerima bukti pembelian berupa struk.

### b. Kebutuhan Fungsional

#### a) Manajemen Data Barang

- Tambah, ubah, dan hapus data barang.
- Penyimpanan detail barang (nama, kode, kategori, harga, stok).

#### b) Manajemen User

- Admin (pemilik toko) bisa menambah/mengelola akun kasir.
- Kasir hanya memiliki akses untuk melakukan transaksi.

#### c) Transaksi Penjualan

- Input pembelian oleh kasir.
- Cetak struk pembelian otomatis.
- Stok berkurang otomatis setiap ada transaksi.

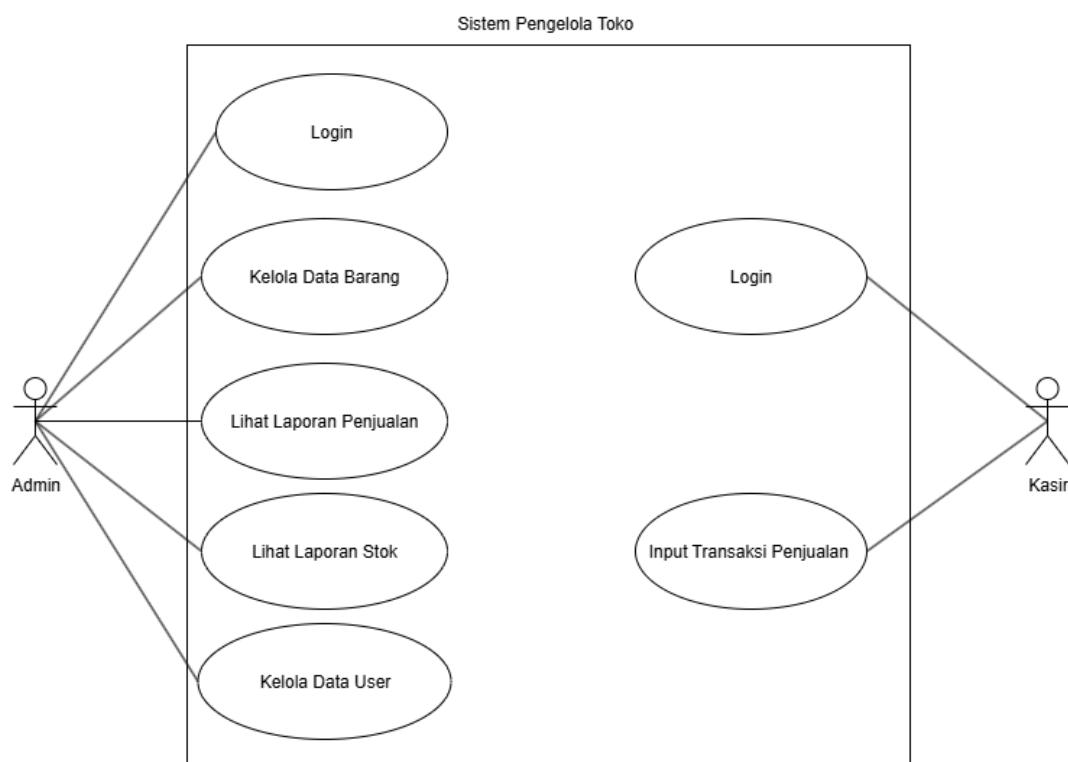
#### d) Laporan

- Laporan penjualan per periode (harian, mingguan, bulanan).
- Laporan stok barang.

- c. Kebutuhan Non-Fungsional
  - a) Kemudahan penggunaan (user friendly): antarmuka sederhana dan mudah dipahami.
  - b) Keamanan data: hanya user dengan hak akses yang bisa login.
  - c) Kinerja: sistem mampu memproses transaksi dengan cepat.
  - d) Backup data: tersedia fitur cadangan data untuk menghindari kehilangan data.

### 3. Pemodelan Sistem – Use Case

- a) Aktor
  - Kasir, Admin
- b) Use Case
  - Login, Kelola Data Barang, Kelola Data User, Input Transaksi Penjualan, Lihat Laporan Stok, Lihat Laporan Penjualan, Login, Input Transaksi Penjualan
- c) Use Case Sistem



d) Use Case Narrative

a) Use Case : Login

Elemen	Isi
Nama Use Case	Login (Admin)
Aktor	Admin / Pemilik Toko
Deskripsi Singkat	Admin Masuk Ke Sistem Dengan Username & Password Untuk Mengakses Menu Utama.
Pre-Condition	Admin Sudah Memiliki Akun Terdaftar.
Alur Utama	<ul style="list-style-type: none"><li>- Admin Membuka Halaman Login.</li><li>- Admin Memasukkan Username &amp; Password.</li><li>- Sistem Memvalidasi Data.</li><li>- Jika Benar, Sistem Menampilkan Menu Utama.</li></ul>
Alur Alternatif	<ul style="list-style-type: none"><li>- Jika Username/Password Salah → Sistem Menampilkan Pesan Error.</li><li>- Jika Akun Tidak Aktif → Sistem Menolak Login.</li></ul>
Post-Condition	Admin Berhasil Login Dan Masuk Ke Menu Sesuai Hak Akses.

b) Use Case : Kelola Data Barang

Elemen	Isi
Nama Use Case	Kelola Data Barang
Aktor	Admin
Deskripsi Singkat	Admin Dapat Menambah, Mengubah, Dan Menghapus Data Barang Di Sistem.
Pre-Condition	Admin Sudah Login.
Alur Utama	<ul style="list-style-type: none"><li>- Admin Pilih Menu Data Barang.</li><li>- Admin Menambah/Ubah/Hapus Data Barang.</li><li>- Sistem Menyimpan Perubahan.</li></ul>
Alur Alternatif	<ul style="list-style-type: none"><li>- Jika Data Yang Dimasukkan Tidak Lengkap → Sistem Menolak Penyimpanan.</li><li>- Jika Kode Barang Sudah Ada → Sistem Menampilkan Peringatan.</li></ul>

Post-Condition	Data Barang Berhasil Diperbarui Di Database.
----------------	--

c) Use Case : Kelola Data User

Elemen	Isi
Nama Use Case	Kelola Data User
Aktor	Admin
Deskripsi Singkat	Admin Dapat Menambah, Mengubah, Dan Menghapus Akun Kasir.
Pre-Condition	Admin Sudah Login.
Alur Utama	<ul style="list-style-type: none"> <li>- Admin Pilih Menu User.</li> <li>- Admin Menambah/Ubah/Hapus Akun Kasir.</li> <li>- Sistem Menyimpan Perubahan.</li> </ul>
Alur Alternatif	<ul style="list-style-type: none"> <li>- Jika Username Sudah Terdaftar → Sistem Menolak Pembuatan Akun Baru.</li> <li>- Jika Data User Tidak Lengkap → Sistem Menampilkan Pesan Error.</li> </ul>
Post-Condition	Data User Kasir Berhasil Diperbarui.

d) Use Case : Lihat Laporan Penjualan

Elemen	Isi
Nama Use Case	Lihat Laporan Penjualan
Aktor	Admin
Deskripsi Singkat	Admin Dapat Melihat Laporan Penjualan Berdasarkan Periode Tertentu.
Pre-Condition	Admin Sudah Login.
Alur Utama	<ul style="list-style-type: none"> <li>- Admin Pilih Menu Laporan.</li> <li>- Admin Pilih Jenis Laporan Penjualan &amp; Periode.</li> <li>- Sistem Menampilkan Laporan.</li> </ul>
Alur Alternatif	<ul style="list-style-type: none"> <li>- Jika Tidak Ada Data Penjualan Pada Periode Tersebut → Sistem Menampilkan Pesan "Data Tidak Tersedia".</li> </ul>
Post-Condition	Laporan Penjualan Ditampilkan Di Layar.

e) Use Case : Lihat Laporan Stok

Elemen	Isi
Nama Use Case	Lihat Laporan Stok
Aktor	Admin
Deskripsi Singkat	Admin Dapat Melihat Laporan Ketersediaan Stok Barang.
Pre-Condition	Admin Sudah Login.
Alur Utama	<ul style="list-style-type: none"> <li>- Admin Pilih Menu Laporan Stok.</li> <li>- Sistem Menampilkan Daftar Stok Barang Terbaru.</li> </ul>
Alur Alternatif	<ul style="list-style-type: none"> <li>- Jika Tidak Ada Barang Terdaftar → Sistem Menampilkan Pesan "Data Kosong".</li> </ul>
Post-Condition	Admin Dapat Melihat Stok Barang Yang Tersedia.

f) Use Case : Login (Kasir)

Elemen	Isi
Nama Use Case	Login (Kasir)
Aktor	Kasir
Deskripsi Singkat	Kasir Masuk Ke Sistem Dengan Username & Password Untuk Mengakses Menu Transaksi.
Pre-Condition	Kasir Sudah Memiliki Akun Yang Dibuat Oleh Admin.
Alur Utama	<ul style="list-style-type: none"> <li>- Kasir Membuka Halaman Login.</li> <li>- Kasir Memasukkan Username &amp; Password.</li> <li>- Sistem Memvalidasi Data.</li> <li>- Jika Benar, Sistem Menampilkan Menu Transaksi.</li> </ul>
Alur Alternatif	<ul style="list-style-type: none"> <li>- Jika Username/Password Salah → Sistem Menampilkan Pesan Error.</li> <li>- Jika Akun Kasir Dinonaktifkan → Sistem Menolak Login.</li> </ul>
Post-Condition	Kasir Berhasil Login Dan Masuk Ke Menu Transaksi.

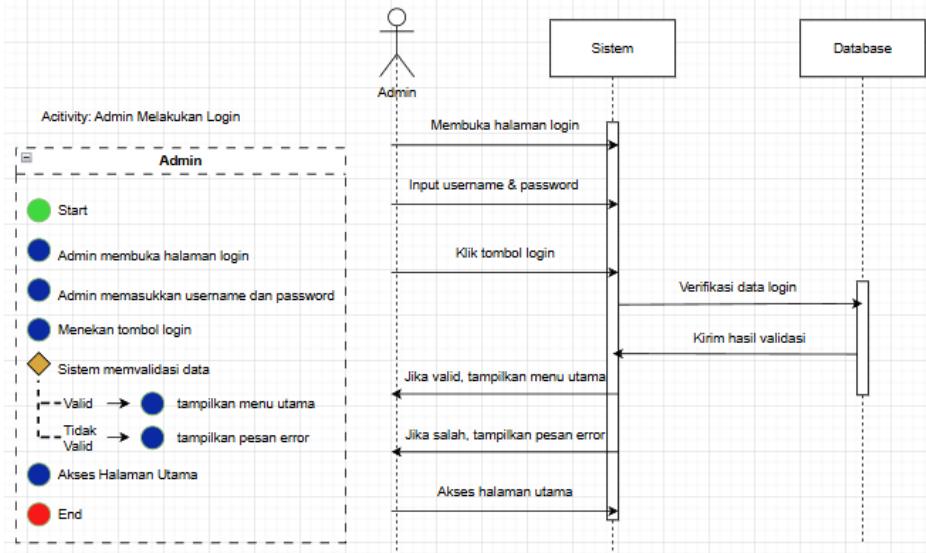
g) Use Case : Input Transaksi Penjualan

Elemen	Isi

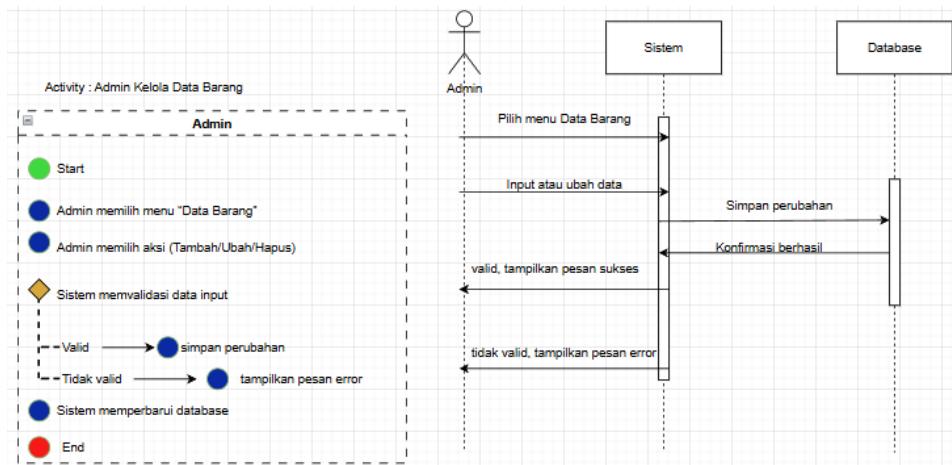
Nama Use Case	Input Transaksi Penjualan
Aktor	Kasir
Deskripsi Singkat	Kasir Melakukan Input Barang Yang Dibeli Pelanggan Dan Sistem Otomatis Mengurangi Stok.
Pre-Condition	Kasir Sudah Login.
Alur Utama	<ul style="list-style-type: none"> <li>- Kasir Pilih Menu Transaksi.</li> <li>- Kasir Input Kode/Nama Barang.</li> <li>- Sistem Menampilkan Harga &amp; Stok.</li> <li>- Sistem Menghitung Total Harga.</li> <li>- Sistem Mengurangi Stok Barang.</li> <li>- Sistem Mencetak Struk Pembelian.</li> </ul>
Alur Alternatif	<ul style="list-style-type: none"> <li>- Jika Stok Barang Habis → Sistem Menampilkan Pesan "Stok Tidak Cukup".</li> <li>- Jika Kode Barang Salah → Sistem Menolak Input.</li> </ul>
Post-Condition	Transaksi Tersimpan, Stok Berkurang, Struk Tercetak.

#### 4. Pemodelan Proses

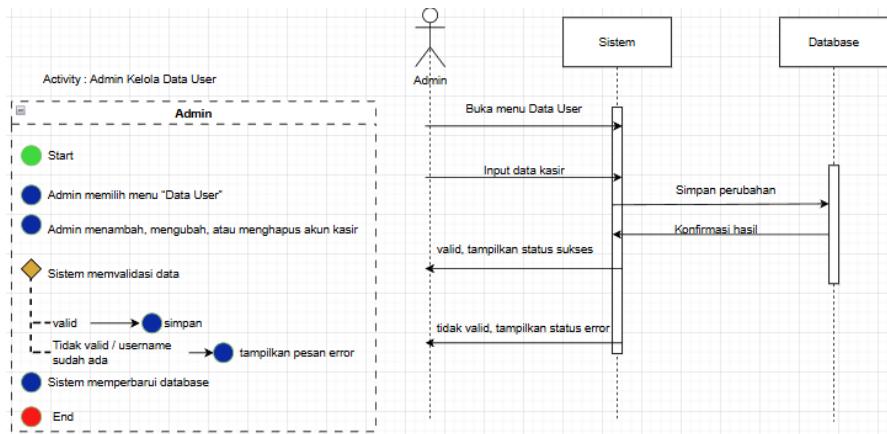
- Activity Diagram Dan Squence Diagram
- Activity : Admin Melakukan Login



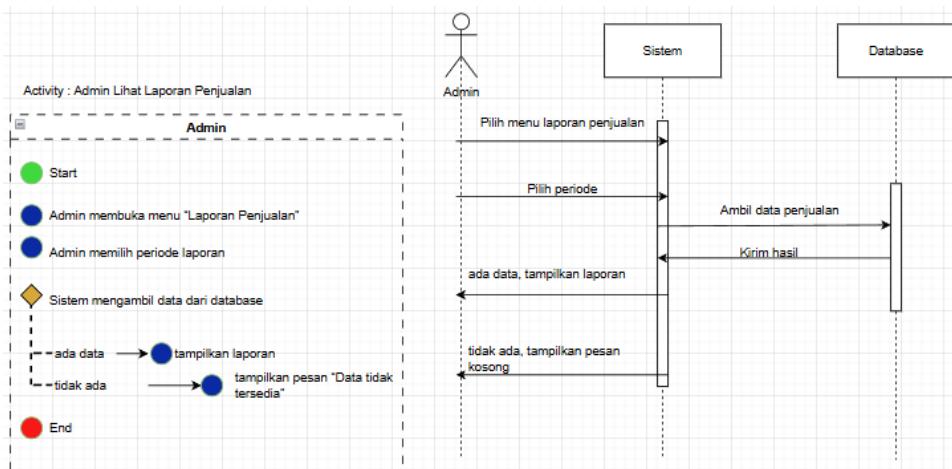
b) Activity : Admin Kelola Data Barang



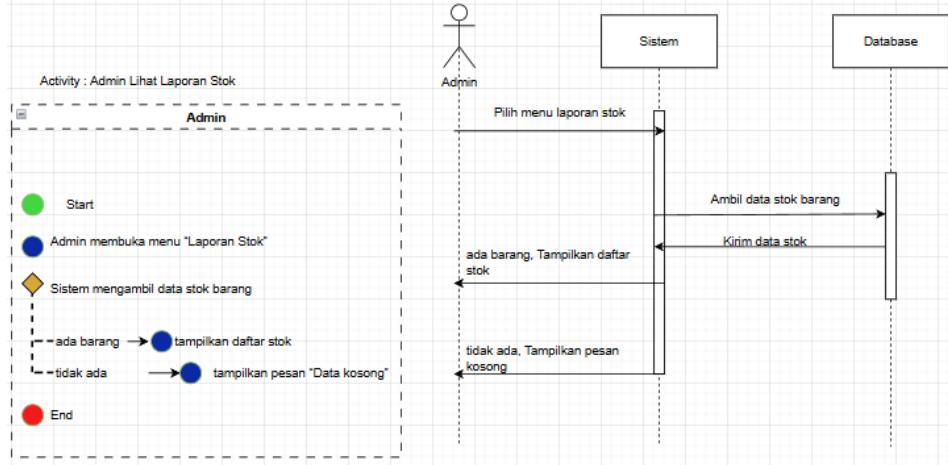
c) Activity : Admin Kelola Data User



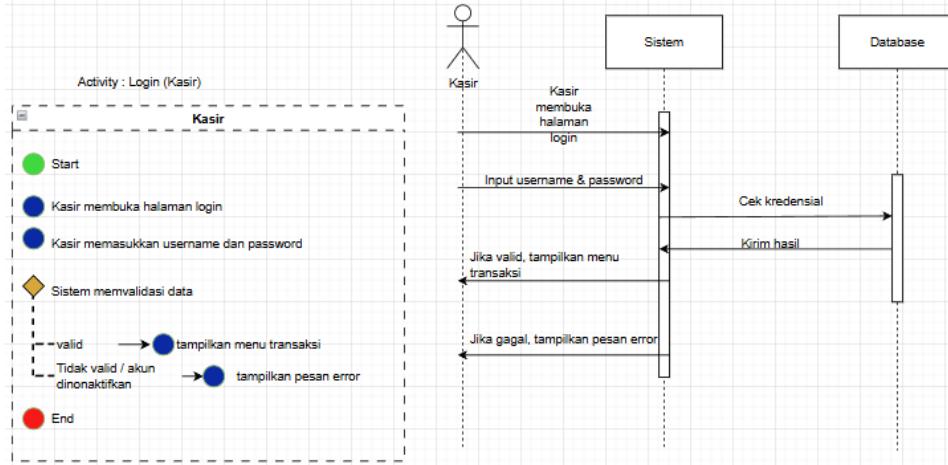
d) Activity : Admin Lihat Laporan Penjualan



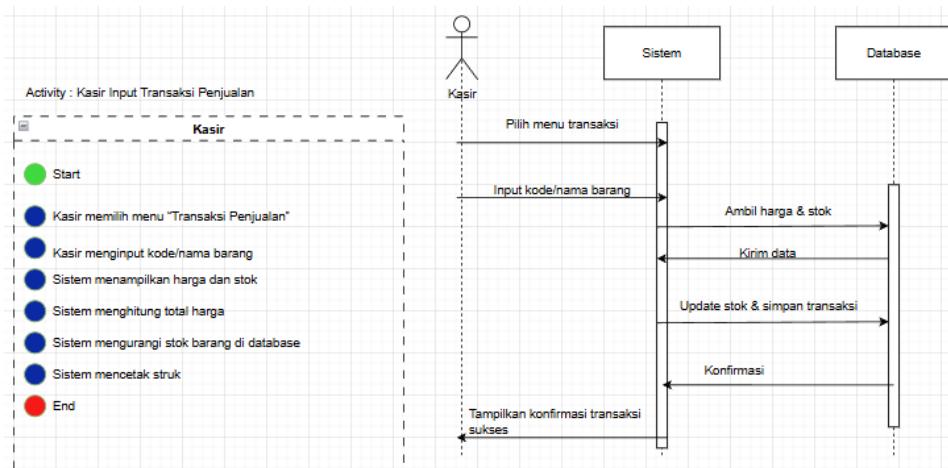
e) Activity : Admin Lihat Laporan Stok



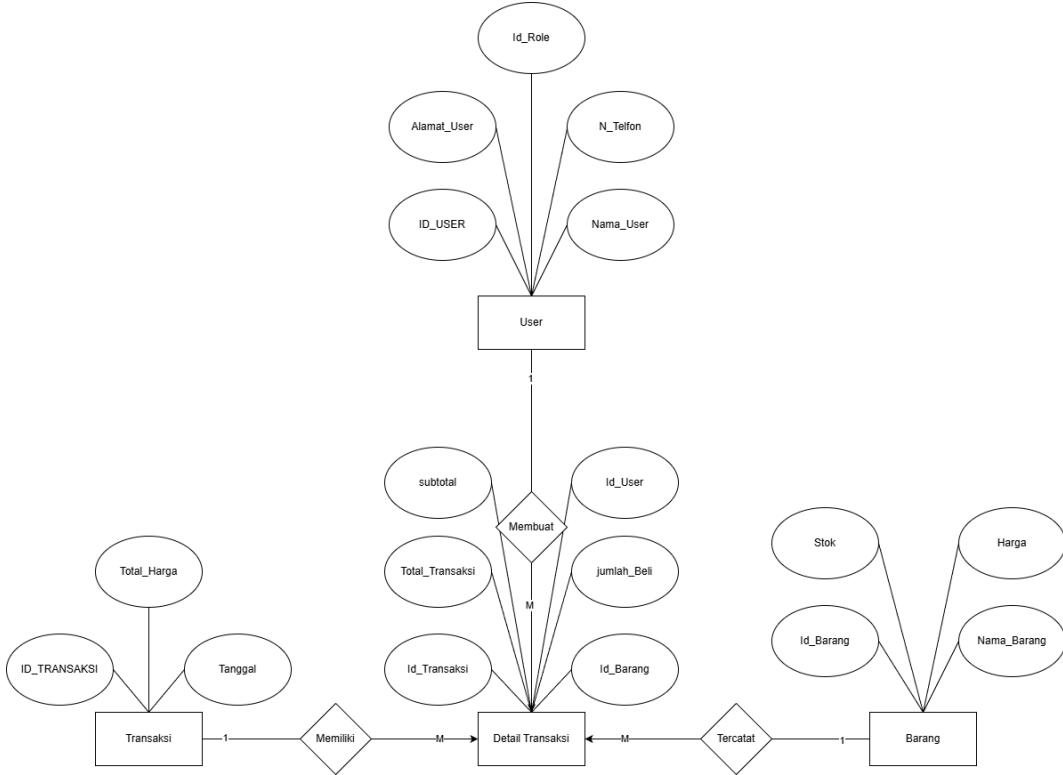
f) Activity : Login Kasir



g) Activity : Kasir Input Transaksi Penjualan



## 5. Pemodelan Data



## 6. Desain Arsitektur Perangkat Lunak

### a. Desain Arsitektur Perangkat Lunak

#### a) Arsitektur Sistem

- Layered Architecture

#### b) Lapisan Sistem

- Presentation Layer:

Menyediakan antarmuka pengguna seperti halaman login, transaksi kasir, pengelolaan data barang, dan laporan admin.

- Business Logic Layer:

Menangani proses bisnis sistem seperti transaksi penjualan, perhitungan total dan kembalian, pengurangan stok otomatis, serta validasi data.

- Data Layer:

Mengelola penyimpanan data user, barang, transaksi, dan laporan dalam basis data terpusat

c) Pola komunikasi

Komunikasi dilakukan secara langsung antar lapisan dalam satu aplikasi tanpa pemisahan layanan terdistribusi.

b. Pola Arsitektur Perangkat Lunak

a) Pola Arsitektur

- Model View Controller (MVC)

b) Penjelasan MVC:

- Model:

Menangani data dan logika bisnis seperti data barang, transaksi penjualan, stok, dan laporan.

- View:

Menyediakan antarmuka pengguna seperti halaman login, transaksi kasir, gudang, dan laporan admin.

- Controller:

Menghubungkan Model dan View serta menangani alur proses sistem seperti input data, validasi, dan pemanggilan logika bisnis.

c) Alasan Pemilihan MVC:

Pola MVC dipilih karena sesuai untuk sistem berbasis transaksi dengan alur proses yang jelas, mudah dipahami, serta memudahkan pemeliharaan sistem.

## 7. Desain Antarmuka & User Experience

a. Langkah desain

- Melakukan analisis kebutuhan pengguna (admin dan kasir) untuk memahami alur kerja sistem.
- Menyusun wireframe halaman login, menu utama, transaksi penjualan, pengelolaan stok, dan laporan.
- Melakukan prototyping menggunakan Balsamiq untuk menggambarkan struktur dan tata letak antarmuka sistem.

- Mengevaluasi alur penggunaan (user flow) guna memastikan sistem mudah dipahami dan efisien digunakan.
- b. Prinsip yang di terapkan
  - Navigasi sederhana agar pengguna dapat mengoperasikan sistem dengan cepat.
  - Tata letak antarmuka konsisten pada setiap halaman.
  - Informasi penting seperti total pembelian, stok barang, dan tombol aksi utama ditampilkan dengan jelas.
  - Informasi penting seperti total pembelian, stok barang, dan tombol aksi utama ditampilkan dengan jelas.

## 8. Desain Detail Sistem

- a. Class diagram
  - User, Barang, Transaksi, DetailTransaksi, Laporan
- b. Object Diagram
  - Kasir *Andi* melakukan transaksi penjualan Nota *NT001* untuk barang *Teh Pucuk* dan *Indomie*
- c. Deployment Diagram
  - Client (Komputer Kasir/Admin) • Application Server • Database Server