

## Alur Algoritma POS Warung Nusantara

### 1. Login

- Admin memasukkan username dan password.
- Sistem cek ke database:
  - Jika valid → masuk ke **Halaman Menu**.
  - Jika salah → tampilkan pesan error.

### 2. Halaman Menu

- Menampilkan daftar menu (produk) dari database.
- Tersedia tombol:
  - **Tambah data** → menuju **Input Data Menu**.
  - **Edit data** → menuju **Update Data Menu**.
  - **Hapus data** → hapus data dari database.
- Pencarian menu untuk mencari produk sesuai keinginan.

### 3. Input Data Menu

- Admin isi form (nama, harga, stok, gambar).
- Klik **Tambah** → data disimpan ke database.
- Kembali ke **Halaman Menu**.

### 4. Update Data Menu

- Admin pilih menu yang akan diubah.
- Sistem tampilkan data lama dalam form.
- Admin edit data, klik **Update**.

- Data baru tersimpan ke database.

## 5. Transaksi

- Admin pilih barang dari daftar menu.
- Barang masuk ke **keranjang transaksi**.
- Hitung total harga otomatis.
- Admin masukkan jumlah bayar → sistem hitung kembalian.
- Klik **Simpan** → data transaksi masuk ke tabel transaksi & detail transaksi.
- Lanjut ke cetak **Struk**.

## 6. Struk

- Sistem menampilkan detail transaksi (item, jumlah, total, bayar, kembalian).
- Admin bisa **Print / Simpan PDF**.
- Setelah selesai, kembali ke menu utama atau ke Transaksi.

## 7. Laporan

- Admin pilih periode laporan.
- Sistem menampilkan daftar transaksi sesuai tanggal.
- Admin bisa klik detail laporan → masuk ke **Laporan Detail**, menampilkan detail transaksi berbentuk **struk**.

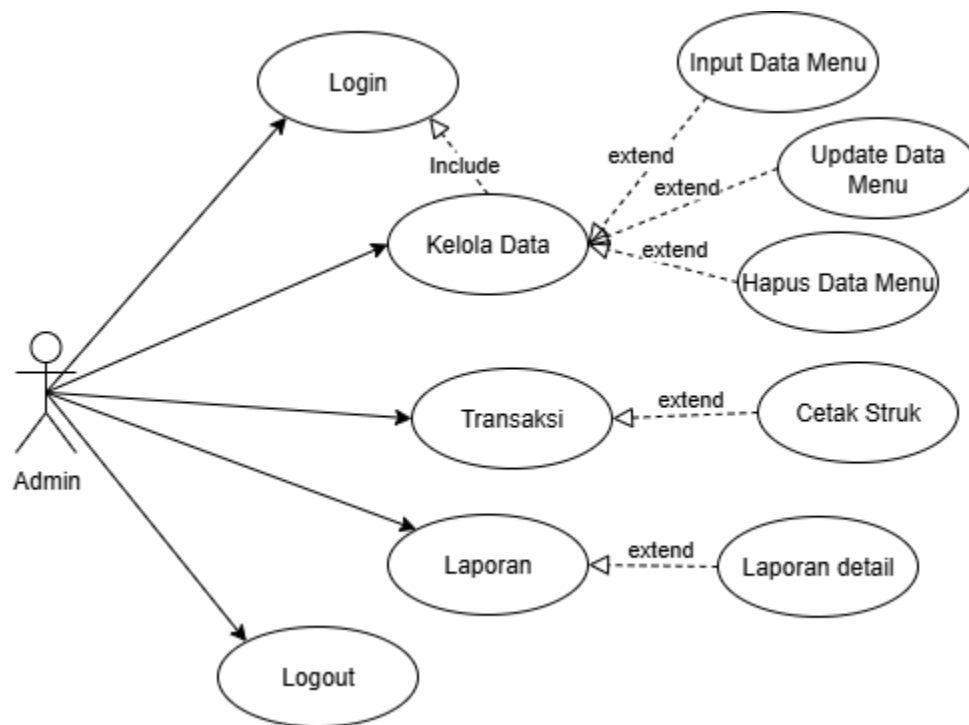
## 8. Laporan Detail

- Menampilkan transaksi lengkap: ID transaksi, tanggal, item yang dibeli, total harga dan kembalian.
- Tersedia opsi **Print / Simpan PDF**.

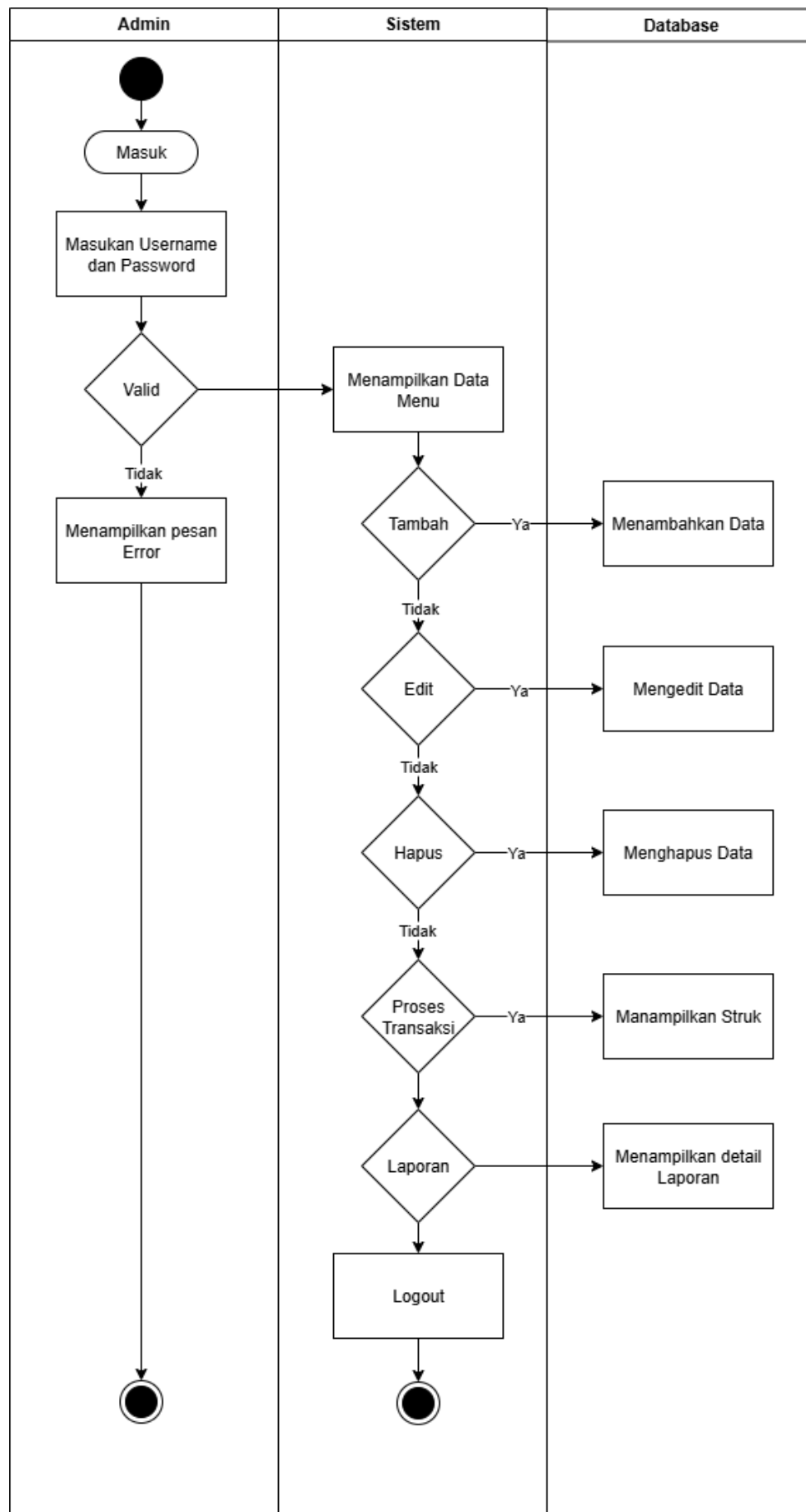
## 9. Logout

- Admin klik tombol logout → kembali ke halaman login.

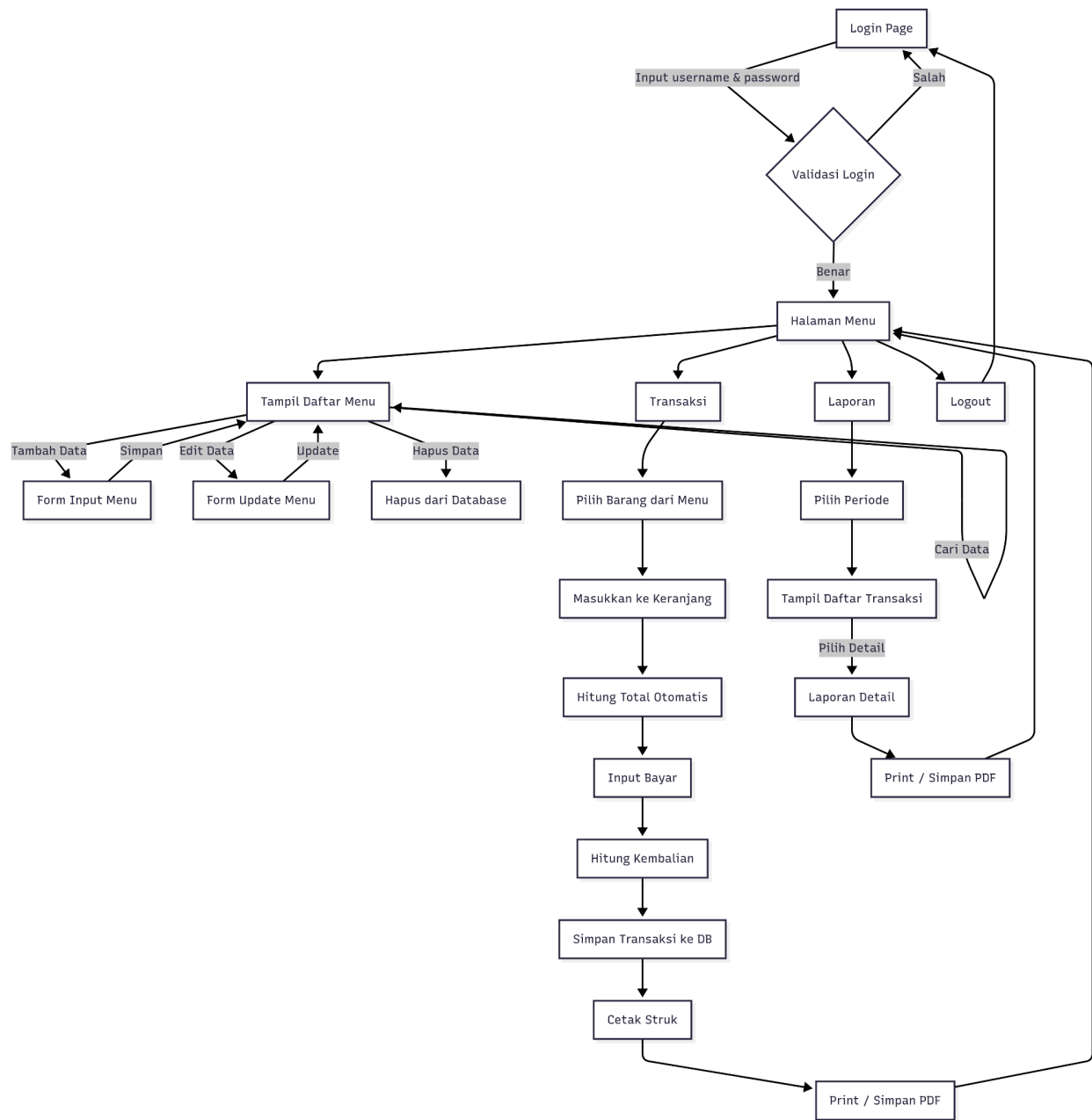
## UseCase Diagram



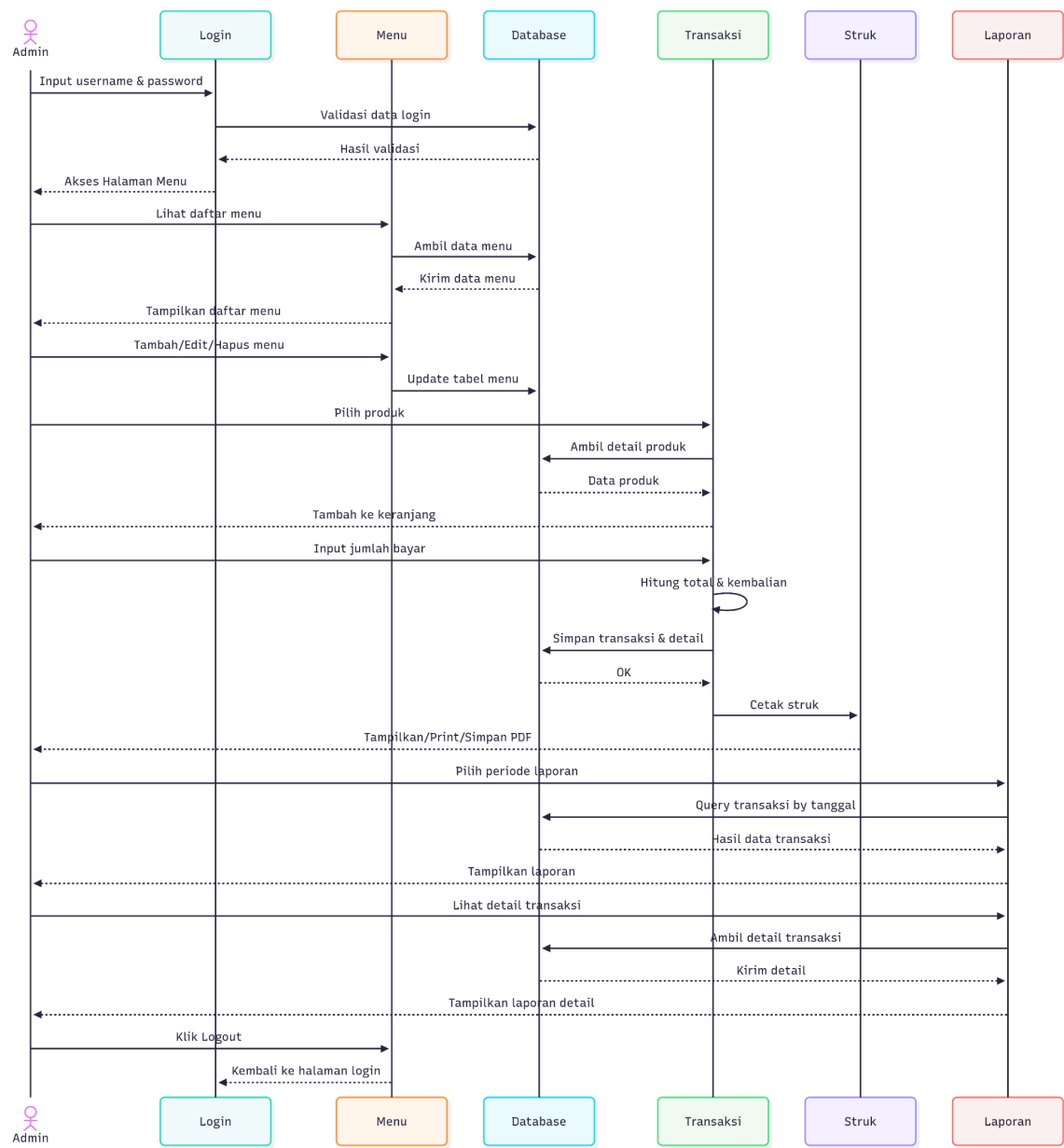
## Activity Diagram



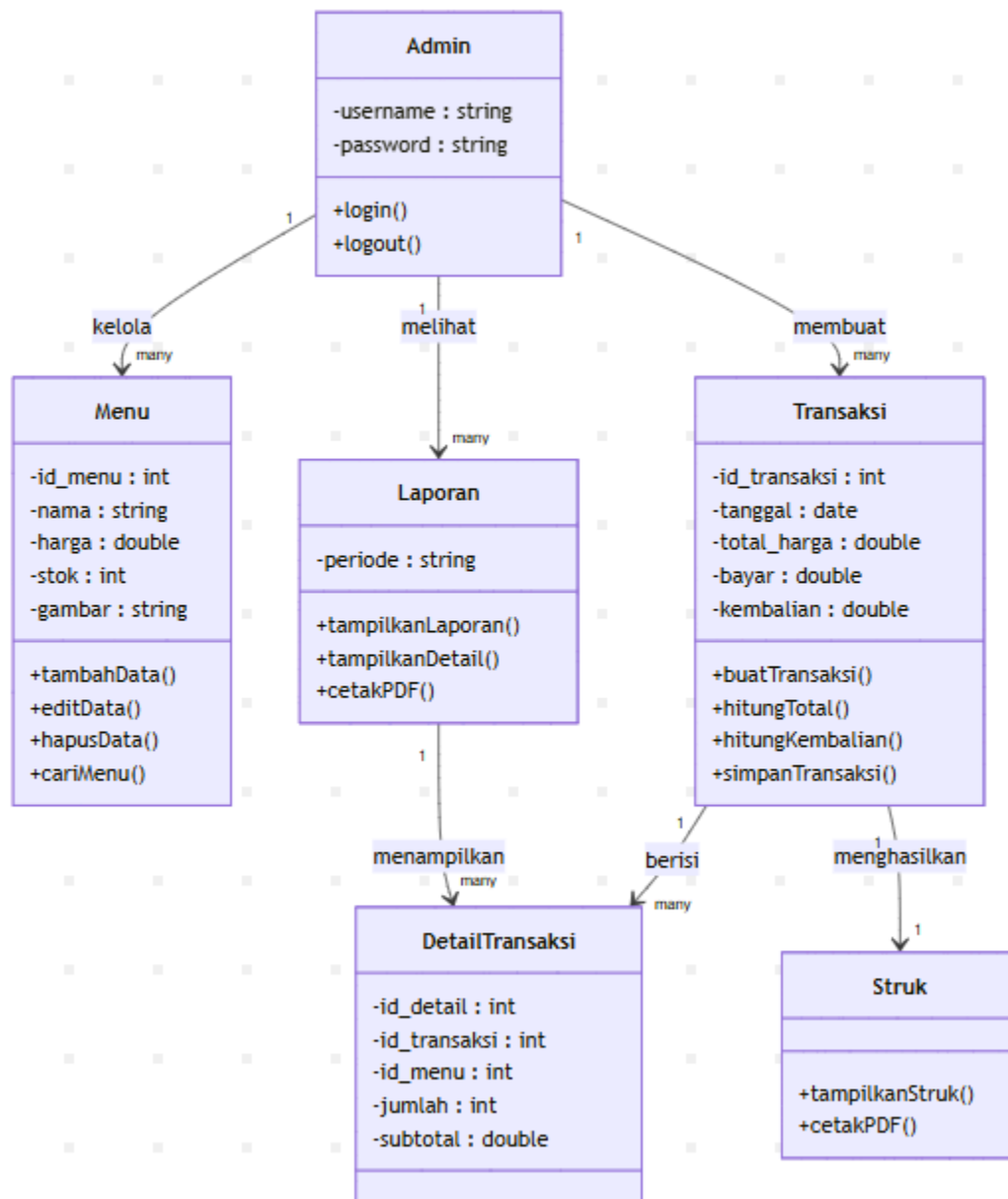
## Flowchart



# Sequence Diagram



## Class Diagram



# Waterfall Model POS Warung Nusantara

## 1. Requirement (Analisis Kebutuhan)

- Fitur utama yang dibutuhkan:
  - Login admin.
  - Input data menu (tambah, update, hapus).
  - Transaksi (hitung otomatis total, bayar, kembalian).
  - Cetak struk.
  - Laporan & detail laporan.
- Semua kebutuhan ini dikumpulkan dari user (pemilik warung).

## 2. System Design (Perancangan Sistem)

- Tampilan UI sudah dirancang: halaman login, menu utama, input data, transaksi, laporan.
- Database didesain dengan tabel: menu, transaksi, laporan, user.
- Alur sistem jelas: login → pilih menu → transaksi → cetak struk → laporan.

## 3. Implementation (Pengkodean)

- Foto menunjukkan hasil implementasi (program sudah dibuat):
  - Login form bekerja.
  - CRUD menu (tambah, update, hapus).
  - Transaksi bisa menghitung otomatis total & kembalian.
  - Laporan bisa ditampilkan dengan filter tanggal.

## 4. Testing (Pengujian)

- Berdasarkan UI, setiap fitur diuji:



- Apakah login hanya menerima user valid?
- Apakah input menu validasi data (harga angka, stok angka)?
- Apakah struk keluar sesuai transaksi?
- Apakah laporan sesuai data transaksi?

## 5. Deployment (Penerapan)

- Aplikasi ini bisa dipasang di komputer warung.
- Digunakan langsung untuk mencatat transaksi harian dan mencetak struk.

## 6. Maintenance (Pemeliharaan)

- Jika ada bug (misalnya struk tidak keluar), diperbaiki.
- Jika ada permintaan tambahan (misalnya laporan grafik penjualan per bulan), bisa di-update.

## Tujuan Sistem dari Requirement (Analisis Kebutuhan)

### 1. Mempermudah proses transaksi penjualan

- Menghitung otomatis total belanja, pembayaran, dan kembalian tanpa harus manual.

### 2. Mengelola data menu dengan mudah

- Admin bisa menambah, mengedit, atau menghapus menu dengan cepat.
- Stok dan harga tercatat rapi.

### 3. Menyediakan bukti transaksi (struk)

- Setiap pembelian menghasilkan struk yang jelas untuk pelanggan.

### 4. Membantu pengelolaan laporan penjualan

- Menyajikan laporan harian, bulanan, atau detail per transaksi.

- Memudahkan pemilik warung untuk menganalisis keuntungan/penjualan.

## 5. Meningkatkan efisiensi dan akurasi

- Mengurangi risiko salah hitung.
- Menghemat waktu dibanding pencatatan manual.

## 6. Mendukung keputusan bisnis

- Dari laporan yang ada, pemilik warung bisa melihat menu terlaris, pendapatan, dan tren penjualan.

# Validasi

Agar data yang masuk benar dan tidak merusak sistem:

## 1. Login Admin

- Username dan password wajib diisi.
- Password tidak ditampilkan (masked).
- Validasi panjang minimal password (misalnya  $\geq 6$  karakter).

## 2. Input Data Menu

- **Kode menu** unik (tidak boleh sama dengan menu lain).
- **Nama menu** tidak boleh kosong.
- **Harga** harus angka (tidak boleh huruf/symbol).
- **Stok** harus angka positif.
- **Upload gambar** dibatasi format (JPG, PNG) dan ukuran file.

## 3. Transaksi

- Jumlah barang tidak boleh melebihi stok.
- Nominal pembayaran tidak boleh kosong dan harus angka.

- Jika pembayaran < total, sistem menolak (tidak bisa lanjut).

#### 4. Laporan

- Periode laporan harus valid (tanggal awal ≤ tanggal akhir).

## Keamanan

Untuk melindungi data dan akses sistem:

### 1. Autentikasi & Otorisasi

- Login wajib sebelum mengakses sistem.
- Hanya admin yang bisa mengelola menu & laporan.

### 2. Enkripsi Password

- Password tidak disimpan dalam bentuk teks biasa (plaintext).
- Gunakan hash (misalnya `bcrypt` atau `password_hash` di PHP).

### 3. Proteksi Input (Anti SQL Injection & XSS)

- Gunakan **prepared statements** saat query database.
- Validasi & filter semua input (misalnya harga harus angka).

### 4. Session Security

- Gunakan session yang aman (misalnya `session_regenerate_id`).
- Logout menghapus session.

### 5. Backup & Recovery

- Data penjualan dan laporan rutin di-backup.
- Ada mekanisme restore jika database rusak/hilang.

## 6. Akses File

- Batasi akses direktori upload (gambar menu).
- Cegah eksekusi file berbahaya (misalnya hanya izinkan .jpg/.png).