

PROJECT AKHIR SISTEM BASIS DATA (SQL)



studi kasus

Sistem ini dirancang untuk menangani manajemen stok barang (seperti produk Tupperware, alat makan, dan toples) serta mencatat transaksi penjualan secara terstruktur. Masalah utama yang diselesaikan adalah bagaimana mengelola data master barang yang memiliki banyak kategori (jenisBarang) dan melibatkan banyak pihak seperti vendor (pemasok), sales, dan kasir. Dengan sistem ini, toko dapat memantau stok secara real-time, melacak transaksi pelanggan, dan memastikan setiap pergerakan barang tercatat dengan akurat.

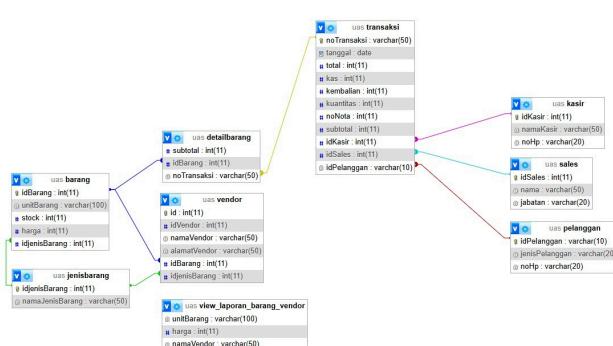


tujuan

- Perancangan Terstruktur:** Membangun skema basis data yang optimal melalui tahap normalisasi (1NF hingga 3NF) untuk menghindari redundansi data pada tabel master seperti barang, vendor, dan jenisBarang.
- Implementasi Relasional:** Menerapkan hubungan antar tabel menggunakan Primary Key dan Foreign Key pada DBMS MySQL untuk menjamin integritas data transaksi.
- Optimasi Query:** Menyediakan mekanisme pengambilan data yang kompleks melalui penggunaan JOIN (untuk laporan penjualan), Agregasi, dan GROUP BY (untuk pemantauan stok per kategori).
- Keamanan Transaksi:** Mengimplementasikan Transaction Control Language (TCL) untuk menjamin konsistensi data, khususnya saat terjadi pembaruan stok barang setelah transaksi penjualan.

rancangan basis data

Entity Relationship Diagram (ERD)



Skema ERD ini mengintegrasikan data transaksi, stok, dan profil pengguna dengan pusat aktivitas pada tabel transaksi yang menghubungkan kasir, sales, serta pelanggan. Tabel detailbarang berfungsi sebagai jembatan informasi antara aktivitas belanja dengan data teknis pada tabel barang, jenisbarang, dan referensi vendor penyedia. Arsitektur ini dirancang untuk menjaga integritas data secara konsisten, memastikan setiap rekaman transaksi akurat dan sesuai dengan kategori produk di gudang. Implementasi view_laporan_barang_vendor mempermudah penyajian laporan manajerial secara otomatis untuk memantau performa operasional serta distribusi barang dari berbagai vendor.



Hasil Normalisasi (Ringkasan 1NF - 3NF)

NO_TRANSAKSI	Tanggal	Total	Kembalian	kuantitas	No_Nota	Subtotal	NO_TRANSAKSI	Kasir	Salesman	Pelanggan	Unit_Barang	Satuan	Harga	Total	Kas	Kembalian	No_HP_Kasir									
JP/001/00/20251108/150533	08/11/2025	4.800	15.000	Tinggi	154	4.800	JP/001/00/20251108/150533	MAJU KASTALAN	DHEA	Pelanggan Umum	Tupperware kecil 300ml	1 pcs	4.800	4.800	20.000	15.000	081225064867									
TABEL_BARANG																										
ID_Barang	Unit_Barang	Satuan					tabel_transaksi	tipe	panjang karakter	keterangan	tabel_barang	tipe	panjang karakter	keterangan	tabel_vendor	tipe	panjang karakter	keterangan								
134	Tupperware kecil 300ml	1 pcs					no_transaksi	char	15	primary key	id_barang	integer	20	primary key	id_vendor	integer	20	primary key								
TABEL_KASIR							tanggal	date	10	atribut	unit_barang	varchar	20	atribut	nama_vendor	char	20	atribut								
321819	Kasir	No HP					total	integer	100	atribut	stock	integer	100	atribut	alamat_vendor	char	20	atribut								
TABEL SALES							kas	integer	100	atribut	harga	integer	100	atribut	id_jenisBarang	integer	3	foreign key								
ID_Salesman	Nama	Jabatan					kembalian	integer	100	atribut	id_jenisBarang	integer	3	foreign key	id_barang	integer	20	foreign key								
8679076	DHEA	Karyawan					kuantitas	varchar	10	atribut																
Tabel_Pelanggan							no_transaksi	integer	10	atribut	tabel_sales	tipe	panjang karakter	keterangan	tabel_jenisBarang	tipe	panjang karakter	keterangan								
ID_Pelanggan	Jenis_Pelanggan	No_Hp_Pelanggan					subtotal	integer	11	atribut	id_sales	integer	7	primary key	id_jenisBarang	integer	3	primary key								
AFD34	Pelanggan Umum	087786549984					id_kasir	integer	6	foreign key	nama	char	50	atribut	id_barang	integer	20	foreign key								
TABEL_VENDOR							id_salesman	integer	7	foreign key	jabatan	char	20	atribut	no_transaksi	char	15	foreign key								
ID_Vendor	Nama_Vendor	Alamat_Vendor					id_pelanggan	varchar	5	foreign key	detail_barang	tipe	panjang karakter	keterangan	id_polangan	tipe	panjang karakter	keterangan								
123000	ADAFI VENDORS	JAKARTA					no_hp	integer	13	atribut	subTotal	integer	100	atribut	Jenis_pelanggan	char	20	atribut								
Tabel_jenisBarang							id_barang	integer	20	atribut	id_barang	integer	20	foreign key	no_hp	integer	13	atribut								
Tabel_pelanggan							nama_jenisBarang	varchar	50	atribut	nama_jenisBarang	varchar	50	atribut	id_polangan	char	5	primary key								

Relasi Antar Tabel

entitas	relasi	entitas
jenisBarang	1 : N	barang
jenisBarang	1 : N	vendor
barang	1 : N	vendor
kasir	1 : N	transaksi
sales	1 : N	transaksi
pelanggan	1 : N	transaksi
transaksi	1 : N	detailBarang
barang	1 : N	detailBarang

Contoh Query - Query

DDL

```
CREATE TABLE barang (
    idBarang INT PRIMARY KEY,
    unitBarang VARCHAR(100),
    stock INT,
    harga INT,
    idJenisBarang INT,
    FOREIGN KEY (idJenisBarang) REFERENCES jenisBarang(idJenisBarang)
);
```

DML

```
INSERT INTO barang VALUES
(134, 'Tupperware kecil 300ml', 50, 4800, 2),
(146, 'Tupperware sedang 600ml', 20, 7500, 2),
(147, 'Tupperware besar 1L', 10, 12000, 2),
(148, 'Gelas plastik bening', 12, 15000, 2),
(149, 'Piring plastik', 6, 20000, 1),
(150, 'Sendok plastik', 12, 10000, 1),
(151, 'Botol minum plastik 500ml', 1, 10000, 2),
(152, 'Toples plastik kecil', 5, 9000, 3),
(153, 'Toples plastik besar', 8, 15000, 3),
(154, 'Wadah makanan sekat', 11, 13000, 1);
```

DML

```
START TRANSACTION;
```

```
UPDATE barang
SET stock = stock - 1
WHERE idBarang = 134;
```

```
ROLLBACK;
```

<https://github.com/Adittt-pr/Project-Akhir-Sistem-Basis-Data>

```
SELECT T.noTransaksi, T.tanggal, K.namaKasir, T.total
FROM transaksi T
JOIN kasir K ON T.idKasir = K.idKasir;
```

Result Grid	Filter Rows:	Export:	Wrap Cell
noTransaksi	tanggal	namaKasir	total
JP/001/00/20251108/150533	2025-11-08	MAJU KASTALAN	4800

Query ini menggunakan fungsi Agregasi, GROUP BY, dan HAVING untuk mengelompokkan stok berdasarkan kategori barang. Sistem secara otomatis menampilkan kategori yang memiliki total stok di atas 10 unit guna memudahkan manajemen dalam memantau ketersediaan inventori.

```
SELECT unitBarang, harga
FROM barang
WHERE harga > (
    SELECT AVG(harga) FROM barang
);
```

Implementasi Subquery digunakan untuk mengidentifikasi produk yang memiliki harga di atas rata-rata secara otomatis. Fitur ini memudahkan manajemen toko dalam memantau daftar barang kategori premium atau barang dengan nilai aset tinggi.

Query ini menerapkan fungsi JOIN untuk menghubungkan tabel transaksi dengan tabel kasir berdasarkan kesamaan ID. Tujuannya adalah untuk menghasilkan laporan penjualan yang informatif, di mana sistem tidak hanya menampilkan kode angka, tetapi juga menampilkan nama asli petugas kasir yang bertanggung jawab pada setiap transaksi. Hal ini sangat penting untuk pemantauan operasional toko harian.

```
SELECT j.namaJenisBarang, SUM(b.stock) AS total_stok
FROM barang b
JOIN jenisBarang j ON b.idJenisBarang = j.idJenisBarang
GROUP BY j.namaJenisBarang
HAVING SUM(b.stock) > 10;
```

Implementasi Subquery digunakan untuk mengidentifikasi produk yang memiliki harga di atas rata-rata secara otomatis. Fitur ini memudahkan manajemen toko dalam memantau daftar barang kategori premium atau barang dengan nilai aset tinggi.

```
Result Grid | Filter Rows: Export:
unitBarang   harga   namaVendor
Tupperware sedang 600ml 7500 ADAFI VENDORS
Tupperware besar 1L 12000 ADAFI VENDORS
Gelas plastik bening 15000 CV. JAYA PLASTIK
Piring plastik 20000 CV. JAYA PLASTIK
Sendok plastik 10000 CV. JAYA PLASTIK
Botol minum plastik 500ml 10000 CV. JAYA PLASTIK
Toples plastik kecil 9000 TOKO ANEKA WADAH
Toples plastik besar 13000 TOKO ANEKA WADAH
Wadah makanan sekat 15000 TOKO ANEKA WADAH
```

```
CREATE VIEW view_laporan_barang_vendor AS
SELECT B.unitBarang, B.harga, V.namaVendor
FROM barang B
JOIN vendor V ON B.idBarang = V.idBarang;
SELECT * FROM view_laporan_barang_vendor;
```