

PROJECT UAS PEMROGRAMAN BASIS DATA

SISTEM PENJUALAN SOTO LORANG



**UNIVERSITAS
DUTA BANGSA
SURAKARTA**

Dosen Pengampu : Ridwan Dwi kurniawan, S.kom,M.kom

Anggota :

- Amanda Fitri H 240103154
- Anjel Dina E 240103155
- Aulia Athifa Zahra 240103156
- Nayla Putri S 240103172

UNIVERSITAS DUTA BANGSA SURAKARTA

ILMU KOMPUTER (TEKNIK INFORMATIKA)

2026

Daftar Isi

BAB 1.....	3
PENDAHULUAN	3
1.1 Latar Belakang.....	3
1.2 Tujuan Proyek.....	4
1.3 Ruang Lingkup dan Batasan Masalah.....	4
1.4 Gambaran Umum Sistem	4
BAB 2.....	6
LANDASAN TEORI	6
2.1 Konsep Dasar Basis Data	6
2.2 Normalisasi Data	6
2.2.1 Bentuk Tidak Normal (Unnormalized Form)	6
2.2.2 Bentuk Normal Pertama (1NF).....	6
2.2.3 Bentuk Normal Kedua (2NF)	7
2.2.4 Bentuk Normal Ketiga (3NF).....	7
2.3 Entity Relationship Diagram (ERD)	7
2.4 Struktur Tabel dan Relasi.....	8
BAB 3.....	9
PERANCANGAN & IMPLEMENTASI.....	9
3.1 Implementasi Basis Data (DDL)	9
3.1.1 Pembuatan Tabel Master	9
3.1.2 Pembuatan Tabel Transaksi	10
3.2 Manipulasi Data (DML)	11
3.2.1 Pengisian Data Master	11
3.2.2 Pengisian Data Transaksi	12
3.3 Query dan Pelaporan Data.....	12
3.3.1 Menampilkan Detail Nota Lengkap (JOIN)	12
3.3.2 Laporan Pendapatan per Kasir (GROUP BY & AGGREGASI).....	13
3.3.3 Filter Menu Terlaris (HAVING)	13
BAB 4.....	14
PENUTUP.....	14
4.1 Kesimpulan.....	14
4.2 Saran	14
Lampiran	15
DAFTAR ISI	17

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pemrograman basis data merupakan fondasi utama dalam pengembangan sistem informasi modern. Basis data memungkinkan penyimpanan, pengelolaan, dan pengambilan data dilakukan secara terstruktur, efisien, dan konsisten. Dalam dunia bisnis kuliner, penggunaan basis data relasional (RDBMS) menjadi standar untuk menangani volume transaksi harian dan memastikan integritas data keuangan tetap terjaga.

Studi kasus yang diangkat dalam proyek ini adalah sistem penjualan pada Soto Lorang Soto Khas Semarang yang berlokasi di Jl. K.H Samanhudi, Purwosari, Surakarta. Saat ini, proses pencatatan transaksi penjualan masih menggunakan bukti fisik berupa nota kertas (seperti contoh nota No: CS/18/251016/0076). Pengelolaan data yang belum terintegrasi secara digital sering menyebabkan redundansi data, risiko kesalahan hitung manual pada kasir, serta kesulitan pemilik usaha dalam merekap laporan penjualan harian secara cepat.

Oleh karena itu, diperlukan sebuah solusi berupa perancangan dan implementasi basis data sistem penjualan (Point of Sales) yang mampu menangani transaksi pemesanan menu secara akurat. Proyek ini akan menerapkan konsep perancangan basis data mulai dari normalisasi data nota (1NF-3NF), penyusunan Entity Relationship Diagram (ERD), hingga implementasi teknis menggunakan bahasa SQL (Structured Query Language) untuk memecahkan permasalahan tersebut.

1.2 Tujuan Proyek

Berdasarkan latar belakang di atas, tujuan dari penyusunan proyek akhir ini adalah sebagai berikut:

Tujuan Akademis: Memenuhi luaran wajib mata kuliah Pemrograman Basis Data dengan menunjukkan kemampuan dalam normalisasi data transaksi soto, perancangan ERD, serta penggunaan fitur SQL seperti Join, Agregasi, dan Grouping.

Tujuan Teknis: Menghasilkan rancangan basis data yang terstruktur untuk mengelola relasi antara entitas Pelanggan, Kasir, Produk (Menu Soto/Minuman), dan Transaksi.

Tujuan Fungsional: Menyediakan sistem pencatatan transaksi bagi Soto Lorang yang dapat menghitung subtotal item (seperti Soto Ayam, Sundukan, Teh) dan total pembayaran secara otomatis dan tersimpan dalam basis data.

1.3 Ruang Lingkup dan Batasan Masalah

Agar pembahasan lebih terarah dan sesuai dengan materi perkuliahan, proyek ini memiliki ruang lingkup dan batasan sebagai berikut:

Fokus Data: Sistem hanya mengelola data master (Produk/Menu, Pelanggan, Kasir) dan data transaksi operasional warung (Nota Penjualan).

Batasan Fitur: Sistem mencakup fungsi dasar Create, Read, Update, Delete (CRUD) pada data menu soto, serta pencatatan transaksi kasir sederhana. Fitur kompleks seperti manajemen stok bahan baku dapur (inventory bahan mentah) atau laporan pajak tidak disertakan dalam versi ini.

Implementasi Teknis: Implementasi basis data menggunakan DBMS MySQL dan tools MySQL Workbench. Laporan mencakup proses normalisasi data dari nota fisik hingga mencapai bentuk normal ketiga (3NF).

1.4 Gambaran Umum Sistem

Sistem yang dibangun merupakan aplikasi basis data untuk manajemen kasir warung soto. Alur kerja sistem ini dimulai ketika pelanggan memesan menu (misalnya: 1 Soto Ayam, 1 Teh, 2 Sundukan). Kasir akan menginput pesanan ke dalam sistem, yang kemudian sistem akan mencatat rincian item tersebut ke dalam tabel detail transaksi dan menghitung total tagihan.

Secara teknis, sistem ini dirancang menggunakan lima tabel inti yang saling berelasi, yaitu:

- Tabel Pelanggan: Menyimpan data pembeli.
- Tabel Kasir: Menyimpan data petugas (misal: Shift Pagi/Malam).
- Tabel Produk: Menyimpan daftar menu dan harga satuan.
- Tabel Transaksi: Mencatat waktu dan total transaksi per nota.
- Tabel Detail Transaksi: Menghubungkan transaksi dengan produk untuk mencatat jumlah item yang dibeli.

Sistem menjamin konsistensi data menggunakan constraint (Primary Key dan Foreign Key) untuk memastikan setiap transaksi valid dan terhubung dengan data kasir serta produk yang benar.

BAB 2

LANDASAN TEORI

2.1 Konsep Dasar Basis Data

Basis data (database) adalah kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan pengguna. Untuk mengelola basis data tersebut, digunakan perangkat lunak yang disebut DBMS (Database Management System).

Dalam proyek ini, model data yang digunakan adalah model Relasional (RDBMS), di mana data disimpan dalam bentuk tabel-tabel dua dimensi yang terdiri dari baris (record) dan kolom (field). Hubungan antar tabel direpresentasikan melalui kunci tamu (Foreign Key) yang merujuk pada kunci utama (Primary Key) di tabel lain.

2.2 Normalisasi Data

Normalisasi adalah teknik analisis data yang mengorganisasikan atribut-atribut data dengan cara mengelompokkan sehingga terbentuk entitas yang stabil, fleksibel, dan adaptif. Berikut adalah tahapan normalisasi yang diterapkan pada studi kasus Soto Lorang:

2.2.1 Bentuk Tidak Normal (Unnormalized Form)

Data awal diambil dari dokumen transaksi berupa nota penjualan (No Nota: CS/18/251016/0076). Pada tahap ini, data masih berupa satu tabel besar yang memuat data berulang (repeating group). Dalam satu nomor nota yang sama, terdapat pengulangan data kasir, data pelanggan, dan beberapa item menu (Soto Ayam, Teh, Sundukan) sekaligus.

2.2.2 Bentuk Normal Pertama (1NF)

Syarat 1NF terpenuhi jika setiap perpotongan baris dan kolom hanya berisi satu nilai atomik (tunggal).

Implementasi: Memecah grup berulang sehingga setiap baris data hanya memuat satu item produk.

Hasil: Tabel kini memiliki struktur datar, namun masih terdapat banyak redundansi (data pelanggan dan kasir ditulis berulang setiap kali ada item baru).

2.2.3 Bentuk Normal Kedua (2NF)

Syarat 2NF adalah tabel harus sudah dalam bentuk 1NF dan tidak ada ketergantungan parsial (atribut bukan kunci bergantung pada sebagian dari Primary Key majemuk).

Analisis: Atribut Nama Produk dan Harga hanya bergantung pada ID Produk, bukan pada No Nota. Atribut Waktu dan Total Bayar bergantung penuh pada No Nota.

Implementasi: Tabel dipecah menjadi tiga entitas terpisah:

- Tabel Transaksi: Menyimpan informasi header nota.
- Tabel Produk: Menyimpan data master menu soto dan minuman.
- Tabel Detail Transaksi: Menyimpan relasi antara nota dan produk (item yang dibeli).

2.2.4 Bentuk Normal Ketiga (3NF)

Syarat 3NF adalah tabel harus sudah dalam bentuk 2NF dan tidak ada ketergantungan transitif (atribut bukan kunci bergantung pada atribut bukan kunci lainnya).

- Analisis: Pada tabel Transaksi, terdapat atribut Nama Kasir dan Shift Kerja yang bergantung pada ID Kasir. Begitu juga data pelanggan bergantung pada ID Pelanggan.
- Implementasi: Data Kasir dan Pelanggan dipisahkan menjadi tabel master tersendiri.
- Hasil Akhir: Terbentuk 5 tabel yang sudah ternormalisasi: Pelanggan, Kasir, Produk, Transaksi, dan Detail_Transaksi.

2.3 Entity Relationship Diagram (ERD)

Berdasarkan hasil normalisasi, dirancang Entity Relationship Diagram (ERD) untuk menggambarkan hubungan antar entitas dalam sistem:

- Pelanggan - Transaksi: Memiliki relasi One-to-Many. Satu pelanggan dapat melakukan banyak transaksi pembelian.
- Kasir - Transaksi: Memiliki relasi One-to-Many. Satu kasir dapat melayani (input) banyak nota transaksi.
- Transaksi - Detail Transaksi: Memiliki relasi One-to-Many. Satu nomor nota dapat memiliki banyak rincian item barang.
- Produk - Detail Transaksi: Memiliki relasi One-to-Many. Satu jenis produk (misal: Soto Ayam) dapat muncul di banyak detail transaksi yang berbeda.

2.4 Struktur Tabel dan Relasi

Berikut adalah definisi Primary Key (PK) dan Foreign Key (FK) untuk menjaga integritas data pada sistem yang dibangun:

- Tabel Pelanggan: PK = id_pelanggan
- Tabel Kasir: PK = id_kasir
- Tabel Produk: PK = id_produk
- Tabel Transaksi:
 - PK = no_nota
 - FK = id_pelanggan (merujuk ke Tabel Pelanggan)
 - FK = id_kasir (merujuk ke Tabel Kasir)
 - Tabel Detail_Transaksi:
 - PK = id_detail
 - FK = no_nota (merujuk ke Tabel Transaksi)
 - FK = id_produk (merujuk ke Tabel Produk)

BAB 3

PERANCANGAN & IMPLEMENTASI

3.1 Implementasi Basis Data (DDL)

Pada tahap ini, desain tabel yang telah dirancang diubah menjadi kode SQL (Data Definition Language) untuk membangun struktur basis data di MySQL. Script pembuatan tabel beserta relasi Primary Key dan Foreign Key-nya.

3.1.1 Pembuatan Tabel Master

Tabel master terdiri dari tabel Pelanggan, Kasir, dan Produk yang tidak memiliki Foreign Key ke tabel transaksional.

❖ Code SQL

✓ *Membuat Tabel Pelanggan*

```
CREATE TABLE pelanggan (  
    id_pelanggan INT AUTO_INCREMENT PRIMARY KEY,  
    nama_pelanggan VARCHAR(100),  
    alamat VARCHAR(255),  
    no_telp VARCHAR(15)  
);
```

✓ *Membuat Tabel Kasir*

```
CREATE TABLE kasir (  
    id_kasir INT AUTO_INCREMENT PRIMARY KEY,  
    nama_kasir VARCHAR(100),  
    username VARCHAR(50),  
    password VARCHAR(50),  
    shift_kerja ENUM('Pagi', 'Siang', 'Malam')  
);
```

- ✓ *Membuat Tabel Produk (Menu Soto & Minuman)*

```
CREATE TABLE produk (  
    id_produk INT AUTO_INCREMENT PRIMARY KEY,  
    nama_produk VARCHAR(100),  
    kategori ENUM('Makanan', 'Minuman', 'Sate/Lauk'),  
    __harga_satuan DECIMAL(10, 2),  
    stok INT  
);
```

3.1.2 Pembuatan Tabel Transaksi

Tabel transaksi dan detail transaksi memiliki relasi ke tabel master.

❖ **Code SQL**

- ✓ *Membuat Tabel Transaksi (Header Nota)*

```
CREATE TABLE transaksi (  
    no_nota VARCHAR(20) PRIMARY KEY, -- Contoh:  
    CS/18/251016/0076  
    waktu_transaksi DATETIME,  
    id_pelanggan INT,  
    id_kasir INT,  
    total_bayar DECIMAL(10, 2),  
    bayar_tunai DECIMAL(10, 2),  
    kembalian DECIMAL(10, 2),  
    FOREIGN KEY (id_pelanggan) REFERENCES  
    pelanggan(id_pelanggan),  
    FOREIGN KEY (id_kasir) REFERENCES kasir(id_kasir)  
);
```

- ✓ *Membuat Tabel Detail Transaksi (Rincian Item)*

```
CREATE TABLE detail_transaksi (  
    id_detail INT AUTO_INCREMENT PRIMARY KEY,  
    no_nota VARCHAR(20),
```

```

        id_produk INT,
        jumlah_beli INT,
        subtotal DECIMAL(10, 2),
        FOREIGN KEY (no_nota) REFERENCES
transaksi(no_nota),
        FOREIGN KEY (id_produk) REFERENCES
produk(id_produk)
    );

```

3.2 Manipulasi Data (DML)

Setelah struktur tabel terbentuk, dilakukan pengisian data awal (Data Entry) menggunakan perintah INSERT. Data berikut diambil dari sampel nota studi kasus "Soto Lorang".

3.2.1 Pengisian Data Master

❖ Code SQL

✓ Input Data Produk sesuai Nota

```

INSERT INTO produk (nama_produk, kategori, harga_satuan,
stok) VALUES
('Soto Ayam Campur', 'Makanan', 9000, 50),
('Teh Manis', 'Minuman', 3000, 100),
('Sundukan Telur Puyuh', 'Sate/Lauk', 4000, 30),
('Sundukan Jerohan', 'Sate/Lauk', 4000, 30),
('Tahu Walik', 'Sate/Lauk', 2000, 50);

```

✓ Input Data Kasir

```

INSERT INTO kasir (nama_kasir, username, password,
shift_kerja) VALUES
('Syakira', 'syakira01', 'pass123', 'Pagi'),
('Xaviera', 'xaviera02', 'pass456', 'Malam');

```

✓ Input Data Pelanggan (Dummy)

```
INSERT INTO pelanggan (nama_pelanggan, alamat, no_telp)
VALUES
('Budi Santoso', 'Jl. Laweyan No. 10', '08123456789');
```

3.2.2 Pengisian Data Transaksi

❖ Code SQL

✓ Input Transaksi (Sesuai No Nota di Studi Kasus)

```
INSERT INTO transaksi (no_nota, waktu_transaksi,
id_pelanggan, id_kasir, total_bayar, bayar_tunai, kembalian)
VALUES ('CS/18/251016/0076', '2025-10-16 10:03:00', 1, 1,
22000, 50000, 28000);
```

✓ Input Detail Item yang dibeli pada nota tersebut

```
INSERT INTO detail_transaksi (no_nota, id_produk,
jumlah_beli, subtotal) VALUES
('CS/18/251016/0076', 1, 1, 9000), -- Soto Ayam
('CS/18/251016/0076', 2, 1, 3000), -- Teh
('CS/18/251016/0076', 3, 1, 4000), -- Sate Puyuh
('CS/18/251016/0076', 4, 1, 4000), -- Sate Jerohan
('CS/18/251016/0076', 5, 1, 2000); -- Tahu Walik
```

3.3 Query dan Pelaporan Data

Bagian ini menampilkan implementasi query untuk menampilkan informasi yang berguna bagi pemilik warung soto, mencakup penggunaan JOIN, GROUP BY, AGGREGATE FUNCTION, dan HAVING sesuai syarat luaran wajib.

3.3.1 Menampilkan Detail Nota Lengkap (JOIN)

Query ini bertujuan merekonstruksi tampilan fisik nota dengan menggabungkan data dari tabel Transaksi, Detail, dan Produk.

❖ Code SQL

```
SELECT
t.no_nota,
t.waktu_transaksi,
p.nama_produk,
```

```

        d.jumlah_beli,
        p.harga_satuan,
        d.subtotal
FROM transaksi t
JOIN detail_transaksi d ON t.no_nota = d.no_nota
JOIN produk p ON d.id_produk = p.id_produk
WHERE t.no_nota = 'CS/18/251016/0076';

```

3.3.2 Laporan Pendapatan per Kasir (GROUP BY & AGGREGASI)

Query ini digunakan untuk menghitung total uang yang diterima oleh masing-masing kasir.

❖ Code SQL

```

SELECT
    k.nama_kasir,
    COUNT(t.no_nota) as jumlah_transaksi,
    SUM(t.total_bayar) as total_pendapatan
FROM transaksi t
JOIN kasir k ON t.id_kasir = k.id_kasir
GROUP BY k.nama_kasir;

```

3.3.3 Filter Menu Terlaris (HAVING)

Query ini menampilkan menu apa saja yang total penjualannya (secara jumlah item) melebihi batas tertentu.

❖ Code SQL

```

SELECT
    p.nama_produk,
    SUM(d.jumlah_beli) as total_terjual
FROM detail_transaksi d
JOIN produk p ON d.id_produk = p.id_produk
GROUP BY p.nama_produk
HAVING SUM(d.jumlah_beli) >= 1;

```

BAB 4

PENUTUP

4.1 Kesimpulan

Berdasarkan proses perancangan dan implementasi basis data yang telah dilakukan pada studi kasus Soto Lorang, dapat ditarik beberapa kesimpulan sebagai berikut:

Transformasi Data: Sistem basis data yang dirancang berhasil mentransformasi proses pencatatan manual (nota fisik) menjadi format digital yang terstruktur. Proses normalisasi dari bentuk Unnormalized hingga Third Normal Form (3NF) terbukti efektif menghilangkan redundansi data pelanggan, kasir, dan pengulangan item menu dalam satu transaksi.

Integritas Data: Penerapan Constraint berupa Primary Key dan Foreign Key pada kelima tabel (Pelanggan, Kasir, Produk, Transaksi, Detail_Transaksi) menjamin konsistensi hubungan antar data. Tidak ada transaksi yang dapat dicatat tanpa adanya data kasir atau produk yang valid.

Fungsi Pelaporan: Implementasi SQL menggunakan perintah JOIN, GROUP BY, dan Fungsi Agregasi mampu menghasilkan informasi yang bernilai bagi pemilik usaha, seperti laporan detail struk belanja, rekapitulasi total pendapatan per kasir, dan analisis menu yang paling laris terjual.

4.2 Saran

Sistem basis data yang dibangun saat ini masih berfokus pada sisi back-end dan manajemen transaksi penjualan dasar. Untuk pengembangan selanjutnya, disarankan beberapa hal berikut:

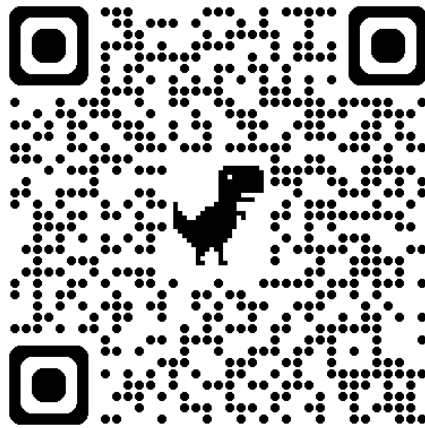
Pengembangan Antarmuka (GUI): Perlu dibangun aplikasi antarmuka pengguna (User Interface) berbasis Web atau Desktop agar kasir dapat menginput data dengan lebih mudah tanpa harus mengetik kode SQL secara langsung.

Manajemen Stok Bahan Baku: Sistem saat ini hanya mengurangi stok menu jadi (misal: porsi soto). Pengembangan selanjutnya sebaiknya mencakup manajemen inventori bahan baku (daging, beras, bumbu) untuk mengetahui Harga Pokok Penjualan (HPP) secara lebih detail.

Keamanan Akses: Perlu ditambahkan level otorisasi pengguna (Role-Based Access Control) yang lebih kompleks, misalnya membedakan hak akses antara 'Admin' (Bisa hapus data) dan 'Kasir' (Hanya bisa input transaksi) demi keamanan data.)

Lampiran

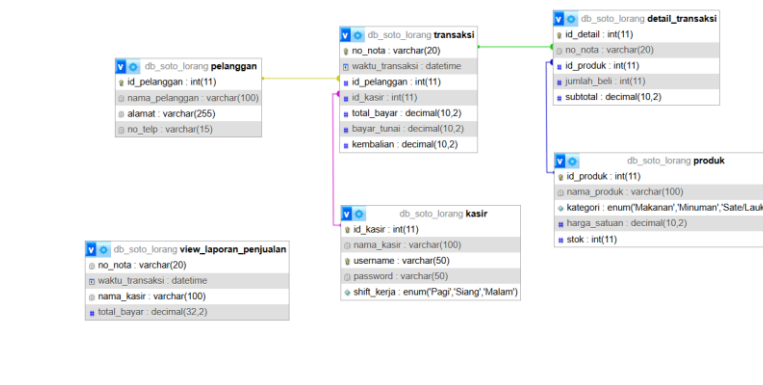
1. Tautan Repository



2. Struktur Folder:

- /database: Berisi file .sql (struktur dan data).
- /docs: Berisi file laporan PDF dan file desain ERD.
- /poster: Berisi file poster final (PDF/JPG).

3. ERD



4. Hasil Query

Result Grid					
Filter Rows:					
Export:					
Wrap Cell Content:					
	no_nota	waktu_transaksi	nama_kasir	nama_produk	harga_satuan
▶	CS/18/251016/0076	2025-10-16 10:03:00	Syakira	Soto Ayam Campur	9000.00
	CS/18/251016/0076	2025-10-16 10:03:00	Syakira	Teh Manis	3000.00
	CS/18/251016/0076	2025-10-16 10:03:00	Syakira	Sundukan Telur Puyuh	4000.00
	CS/18/251016/0076	2025-10-16 10:03:00	Syakira	Sundukan Jerohan	4000.00
	CS/18/251016/0076	2025-10-16 10:03:00	Syakira	Tahu Walik	2000.00

Result Grid		
Filter Rows:		
Export:		
Wrap Cell Content:		
	nama_produk	total_terjual
▶	Soto Ayam Campur	1
	Sundukan Jerohan	1
	Sundukan Telur Puyuh	1
	Tahu Walik	1
	Teh Manis	1

Result Grid			
Filter Rows:			
Export:			
Wrap Cell Content:			
	nama_kasir	jumlah_transaksi	total_pendapatan
▶	Syakira	1	22000.00

DAFTAR ISI

1. Silberschatz A, Korth HF, Sudarshan S. Database System Concepts. 7th ed. New York: McGraw-Hill Education; 2019.
2. Connolly TM, Begg CE. Database Systems: A Practical Approach to Design, Implementation, and Management. 6th ed. Essex: Pearson Education; 2014.
3. Coronel C, Morris S. Database Systems: Design, Implementation, & Management. 13th ed. Boston: Cengage Learning; 2018.
4. Hidayatullah P. Tutorial Kuliah Basis Data. Bandung: Informatika; 2020.
5. MySQL. MySQL 8.0 Reference Manual [Internet]. 2024 [cited 2026 Jan 19]. Available from: <https://dev.mysql.com/doc/refman/8.0/en/>
6. Elmasri R, Navathe SB. Fundamentals of Database Systems. 7th ed. Hoboken: Pearson; 2016.