

**PEMROGAMAN BASIS DATA  
PERANCANGAN DAN IMPLEMENTASI BASIS DATA TRANSAKSI  
DI NATURAL DIGITAL PRINTING**



**UNIVERSITAS  
DUTA BANGSA  
SURAKARTA**

Dosen Pengampu Mata Kuliah  
Ridwan Dwi Irawan, M.Kom

Disusun Oleh :

Insan Nia Cahaya Putri	(240103192)
Sandhy Prasetyo Adjie	(240103225)
Arvanda Nuraini	(240103249)

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DUTA BANGSA SURAKARTA  
TAHUN 2026**



## KATA PENGANTAR

Puji syukur ke hadirat Allah SWT atas segala limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan Ujian Akhir Semester mata kuliah Pemrograman Basis Data dengan judul “Perancangan dan Implementasi Basis Data Transaksi di Natural Digital Printing” dengan baik dan tepat waktu.

Laporan ini disusun sebagai bentuk penerapan materi yang telah dipelajari selama perkuliahan, mulai dari konsep dasar basis data, perancangan Entity Relationship Diagram (ERD), normalisasi, relasi antar tabel, hingga implementasi basis data menggunakan MySQL yang mencakup DDL, DML, TCL, serta query JOIN, GROUP BY, dan HAVING. Studi kasus Natural Digital Printing dipilih untuk memberikan gambaran nyata penerapan sistem basis data pada proses transaksi usaha percetakan.

Penulis menyadari bahwa dalam penyusunan laporan ini masih terdapat keterbatasan dan kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang bersifat membangun demi penyempurnaan laporan ini di masa mendatang. Semoga laporan ini dapat memberikan manfaat, baik sebagai bahan pembelajaran maupun referensi dalam pengembangan sistem basis data.

Akhir kata, penulis mengucapkan terima kasih kepada dosen pengampu mata kuliah Pemrograman Basis Data serta semua pihak yang telah membantu dan mendukung tersusunnya laporan ini.

Surakarta, 17 Januari 2026

Penulis

## DAFTAR ISI

KATA PENGANTAR .....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR .....	iv
DAFTAR TABEL.....	v
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	1
1.3 Tujuan .....	1
1.4 Batasan Masalah .....	2
1.5 Gambaran Umum Sistem.....	2
BAB II LANDASAN TEORI.....	3
2.1 Konsep Basis Data dan DBMS .....	3
2.2 Relasi Tabel dan Basis Data Relasional.....	3
2.3 Entity Relationship Diagram (ERD).....	3
2.4 Normalisasi Basis Data.....	4
2.4.1 Normalisasi 1NF .....	4
2.4.2 Normalisasi 2NF .....	4
2.4.3 Normalisasi 3NF .....	4
2.5 SQL (Structured Query Language).....	5
2.5.1 Data Definition Language (DDL) .....	5
2.5.2 Data Manipulation Language (DML) .....	5
2.5.3 Transaction Control Language (TCL) .....	6
BAB III PERANCANGAN DAN IMPLEMENTASI.....	7
3.1 Analisis Kebutuhan Sistem.....	7

3.2 Perancangan Database .....	15
3.2.1 ERD (Entity Relationship Diagram).....	15
3.2.2 Skema Tabel.....	16
3.2.3 PK & FK .....	17
3.3 Implementasi Basis Data.....	19
3.3.1 Implementasi DDL.....	19
3.3.2 Implementasi DML .....	20
3.3.3 Implementasi TCL .....	20
3.4 Implementasi Query SQL .....	20
3.4.1 Query JOIN .....	20
3.4.2 Query GROUP BY .....	21
3.4.3 Query HAVING.....	21
3.4.4 Query Agregasi .....	21
BAB IV PENUTUP .....	22
4.1 Hasil Pengujian .....	22
4.2 Kendala dan Solusi .....	24
4.3 Kesimpulan .....	25
4.4 Saran .....	25
LAMPIRAN.....	27
DAFTAR PUSTAKA .....	34

## **DAFTAR GAMBAR**

Gambar. 1 Nota Natural Printing .....	7
Gambar. 2 Derajat Kardinalitas .....	13
Gambar. 3 Relasi Antar Tabel .....	14
Gambar. 4 Entity Relationship Diagram.....	15
Gambar. 5 Skema Relasi Tabel Basis Data .....	16

## **DAFTAR TABEL**

Tabel. 1 Unnormalized Table .....	8
Tabel. 2 1NF .....	8
Tabel. 3 2NF Pelanggan.....	9
Tabel. 4 2NF Karyawan.....	9
Tabel. 5 2NF Barang.....	10
Tabel. 6 2NF Order.....	10
Tabel. 7 2NF Kasir.....	10
Tabel. 8 2NF Transaksi.....	11
Tabel. 9 3NF Pelanggan.....	11
Tabel. 10 3NF Karyawan.....	11
Tabel. 11 3NF Barang.....	12
Tabel. 12 3NF Order.....	12
Tabel. 13 3NF Kasir.....	12
Tabel. 14 3NF Transaksi.....	13
Tabel. 15 Primary Key .....	18
Tabel. 16 Foreign Key .....	19
Tabel. 17 Pengujian DDL .....	23
Tabel. 18 Pengujian DML.....	24
Tabel. 19 Pengujian TCL.....	24



## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Perkembangan teknologi informasi mendorong berbagai bidang usaha untuk memanfaatkan sistem basis data dalam pengelolaan data operasional. Basis data berperan penting dalam menyimpan, mengelola, dan menyajikan data secara terstruktur sehingga informasi dapat diakses dengan cepat dan akurat. Salah satu bidang usaha yang membutuhkan sistem basis data adalah usaha percetakan digital.

Natural Digital Printing merupakan usaha yang bergerak di bidang jasa percetakan, seperti pencetakan dokumen, laminasi, dan produk cetak lainnya. Dalam proses operasionalnya, pencatatan transaksi masih berpotensi menimbulkan kesalahan apabila tidak dikelola dengan sistem yang terstruktur. Oleh karena itu, diperlukan perancangan basis data yang mampu mengelola data pelanggan, barang cetak, serta transaksi secara terintegrasi.

Melalui mata kuliah Pemrograman Basis Data, mahasiswa mempelajari proses perancangan hingga implementasi basis data menggunakan DBMS dan bahasa SQL. Proyek ini bertujuan untuk menerapkan konsep tersebut dalam studi kasus Natural Digital Printing.

#### **1.2 Rumusan Masalah**

Berdasarkan uraian pada latar belakang di atas, maka permasalahan yang akan dibahas dalam laporan ini dapat dirumuskan sebagai berikut:

1. Bagaimana merancang basis data transaksi yang terstruktur untuk Natural Digital Printing?
2. Bagaimana menerapkan relasi antar tabel menggunakan primary key dan foreign key pada sistem basis data?
3. Bagaimana mengimplementasikan basis data transaksi menggunakan MySQL sesuai konsep DDL, DML, dan TCL?
4. Bagaimana menerapkan query SQL seperti JOIN, GROUP BY, dan HAVING untuk menghasilkan informasi transaksi?

#### **1.3 Tujuan**

Adapun tujuan dari penyusunan laporan pemrograman basis data ini adalah sebagai berikut:

1. Merancang basis data transaksi pada Natural Digital Printing.
2. Menerapkan relasi antar tabel sesuai konsep basis data relasional.
3. Mengimplementasikan basis data menggunakan MySQL.
4. Menghasilkan informasi transaksi melalui penerapan query SQL.

#### **1.4 Batasan Masalah**

Agar pembahasan dalam proyek ini lebih terfokus, maka batasan masalah ditetapkan sebagai berikut:

1. Sistem yang dibahas hanya mencakup pengelolaan basis data transaksi pada Natural Digital Printing.
2. Data yang dikelola meliputi data pelanggan, karyawan, kasir, barang, dan transaksi penjualan.
3. Pembahasan difokuskan pada perancangan dan implementasi basis data, tidak mencakup pengembangan antarmuka aplikasi.
4. Database Management System (DBMS) yang digunakan adalah MySQL.
5. Implementasi sistem dibatasi pada penggunaan perintah SQL meliputi DDL, DML, TCL, serta query JOIN, GROUP BY, dan HAVING.

#### **1.5 Gambaran Umum Sistem**

Sistem yang dirancang pada proyek ini merupakan sistem basis data transaksi untuk Natural Digital Printing. Sistem ini berfungsi untuk menyimpan dan mengelola data pelanggan, data barang cetak, data karyawan, serta data transaksi penjualan.

Proses transaksi dimulai ketika pelanggan melakukan pemesanan jasa cetak. Kasir kemudian mencatat data transaksi, termasuk barang yang dipesan, jumlah, dan harga. Seluruh data transaksi disimpan ke dalam basis data dan dapat diolah menggunakan query SQL untuk menghasilkan informasi seperti total transaksi dan laporan penjualan.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Konsep Basis Data dan DBMS**

Basis data (database) merupakan kumpulan informasi yang disimpan dalam komputer dan disusun secara sistematis yang terdiri dari berbagai macam elemen. Secara umum, database digunakan untuk menyimpan data yang telah terintegrasi dengan baik.

Sistem Basis Data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan. Database Management System (DBMS) adalah perangkat lunak yang didesain untuk membantu dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar. DBMS dapat menjadi alternatif penggunaan secara khusus untuk aplikasi, semisal penyimpanan data dalam field dan menulis kde aplikasi yang spesifik untuk pemgaturannya [1] [2].

#### **2.2 Relasi Tabel dan Basis Data Relasional**

Relasi antar tabel merupakan mekanisme teknis untuk menghubungkan satu tabel dengan tabel lainnya menggunakan primary key dan foreign key. Relasi ini divisualisasikan dalam bentuk diagram relasi yang menggambarkan kolom-kolom kunci dan keterkaitannya. Hardini et al. (2025) menegaskan bahwa penetapan relasi yang tepat antar tabel memastikan konsistensi data dan mendukung proses pengambilan keputusan dalam sistem informasi berbasis relasional.

Basis data relasional adalah sistem yang menggunakan pendekatan berbasis tabel yang saling berhubungan melalui primary key dan foreign key untuk menjaga keterkaitan antar data. SQL memiliki keunggulan pada integritas transaksi dan konsistensi data melalui prinsip ACID (Atomicity, Consistency, Isolation, Durability) yang menjamin keandalan sistem [3] [6].

#### **2.3 Entity Relationship Diagram (ERD)**

Entity Relationship Diagram (ERD) adalah representasi visual dari struktur basis data yang menunjukkan hubungan antar entitas, atribut, serta relasi antar tabel. ERD digunakan sebagai alat bantu utama dalam proses perancangan sistem untuk memastikan bahwa model data sesuai dengan kebutuhan fungsional sistem. Menurut Arkan (2025), penggunaan ERD sangat membantu dalam tahap desain karena diagram ini mampu menggambarkan hubungan antar entitas secara intuitif, sehingga memudahkan

pengembang dan pengguna memahami struktur sistem secara keseluruhan.

Selain itu, Saha, Bagui, dan Walsh-Earp (2022) menjelaskan bahwa ERD menjadi alat komunikasi penting antara tim teknis dan non-teknis, membantu meminimalkan kesalahan dalam interpretasi kebutuhan sistem sebelum implementasi dilakukan [4] [5].

## 2.4 Normalisasi Basis Data

Normalisasi adalah proses sistematis yang bertujuan memperbaiki struktur tabel agar lebih efisien, konsisten, dan bebas dari redundansi data. Melalui normalisasi, setiap atribut data dipastikan hanya memiliki hubungan yang relevan dengan kunci utamanya, sehingga integritas data tetap terjaga. Hardini et al. (2025) menjelaskan bahwa penerapan normalisasi yang konsisten hingga bentuk normal ketiga (3NF) terbukti mampu meningkatkan efisiensi penyimpanan data dan mengurangi risiko duplikasi [3].

### 2.4.1 Normalisasi 1NF

Tahap pertama normalisasi memastikan setiap atribut berisi nilai tunggal (atomic) dan tidak terdapat kelompok nilai dalam satu kolom. Apabila dalam satu nota terdapat beberapa item barang yang dibeli, maka data tersebut harus dipisahkan ke dalam tabel tersendiri. Menurut Yunianto, Putra, dan Rahmad (2023), penerapan 1NF meningkatkan integritas data serta memudahkan proses pencarian (query) karena struktur tabel menjadi lebih sederhana dan terdefinisi dengan jelas [7] .

### 2.4.2 Normalisasi 2NF

Tahap kedua bertujuan memastikan bahwa setiap atribut non-key bergantung penuh pada primary key. Dalam kasus di mana tabel memiliki composite key, atribut yang hanya bergantung pada sebagian kunci utama harus dipisahkan ke tabel baru. Dengan cara ini, struktur tabel menjadi lebih fokus dan bebas dari redundansi parsial. Sebagaimana dijelaskan oleh Hardini et al. (2025), pemisahan atribut bergantung parsial menjadi tabel baru meningkatkan fleksibilitas pemeliharaan data serta mengurangi inkonsistensi pada sistem yang kompleks [3].

### 2.4.3 Normalisasi 3NF

Tahap ketiga normalisasi bertujuan menghapus dependensi transitif, yaitu ketika sebuah atribut non-key bergantung pada atribut non-key lainnya. Dalam bentuk ini, setiap atribut harus bergantung langsung hanya pada primary key utama. Yunianto et al. (2023) menyatakan bahwa struktur 3NF menghasilkan basis data yang lebih efisien dan meminimalkan redundansi antar-entitas, sehingga performa query dan penyimpanan menjadi lebih optimal [7].

## **2.5 SQL (Structured Query Language)**

Menurut Budi, H. S. (2021), SQL (Structured Query Language) adalah sebuah bahasa yang digunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara de facto merupakan bahasa standar ANSI (American Nasional Standard Institute) yang digunakan dalam manajemen basis data relasional yang juga sering disebut dengan istilah query. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya melalui manipulasi data atau mengedit basis data sesuai yang dikehendaki, seperti menjalankan query untuk mengambil data, menambah data, memperbarui data, dan menghapus data [8]. SQL terbagi menjadi beberapa jenis perintah, antara lain:

### **2.5.1 Data Definition Language (DDL)**

Data Definition Language (DDL) adalah kumpulan perintah SQL yang digunakan untuk mendefinisikan dan mengelola struktur basis data. Perintah DDL digunakan untuk membuat, mengubah, dan menghapus objek basis data seperti database dan tabel [8].

Perintah DDL bersifat permanen, sehingga perubahan yang dilakukan akan langsung tersimpan di dalam basis data.

Contoh perintah yang termasuk dalam DDL antara lain:

1. CREATE DATABASE untuk membuat basis data
2. CREATE TABLE untuk membuat tabel
3. ALTER TABLE untuk mengubah struktur tabel
4. DROP TABLE untuk menghapus tabel

Dalam proyek ini, DDL digunakan untuk membuat database Natural Digital Printing beserta tabel-tabel yang diperlukan, lengkap dengan primary key dan foreign key.

### **2.5.2 Data Manipulation Language (DML)**

Data Manipulation Language (DML) merupakan perintah SQL yang digunakan untuk mengelola data yang tersimpan di dalam tabel. DML berfungsi untuk menambahkan, mengubah, menghapus, dan menampilkan data [8].

Berbeda dengan DDL, perintah DML dapat dikontrol menggunakan transaksi sehingga perubahan data dapat dibatalkan apabila terjadi kesalahan.

Contoh perintah DML antara lain:

1. INSERT untuk menambahkan data
2. UPDATE untuk mengubah data
3. DELETE untuk menghapus data

#### 4. SELECT untuk menampilkan data

Pada proyek ini, DML digunakan untuk mengisikan data pelanggan, barang, karyawan, kasir, serta data transaksi Natural Digital Printing.

#### 2.5.3 Transaction Control Language (TCL)

Transaction Control Language (TCL) adalah kumpulan perintah SQL yang digunakan untuk mengontrol transaksi dalam basis data. TCL berfungsi untuk menjaga konsistensi dan integritas data ketika terjadi proses manipulasi data dalam jumlah besar atau berurutan [8].

Perintah yang termasuk dalam TCL antara lain:

1. START TRANSACTION untuk memulai transaksi
2. COMMIT untuk menyimpan perubahan data secara permanen
3. ROLLBACK untuk membatalkan perubahan data

Dalam proyek ini, TCL digunakan untuk mensimulasikan skenario transaksi, di mana perubahan data dapat dibatalkan apabila terjadi kesalahan input, sehingga integritas data tetap terjaga.

### **BAB III**

### **PERANCANGAN DAN IMPLEMENTASI**

#### **3.1 Analisis Kebutuhan Sistem**

Sistem ini dibuat supaya semua transaksi di tempat percetakan bisa tercatat dengan rapi dan tidak perlu ditulis manual lagi. Jadi, setiap kali ada pelanggan yang memesan jasa cetak—seperti cetak foto, laminasi, atau spanduk—kasir akan mencatatnya ke dalam sistem dan otomatis muncul nota seperti yang ada di gambar.

Contohnya, ada pelanggan bernama Arva dari Gentan, Baki, Sukoharjo yang memesan dua jenis barang. Barang pertama adalah AC 260 GR (1 sisi) sebanyak dua lembar dengan harga Rp 2.900 per lembar, dan barang kedua Laminasi Hot Glosy sebanyak dua lembar dengan harga Rp 2.000 per lembar. Jadi total yang harus dibayar adalah Rp 9.800. Transaksi ini dilakukan pada tanggal 11 Juni 2025 dan pembayarannya dilakukan secara tunai.

Dari nota itu, bisa dilihat bahwa sistem ini butuh beberapa data penting, seperti data pelanggan (nama dan alamat), data barang (nama barang, ukuran, dan harga), serta data transaksi (tanggal, jumlah, dan total harga). Semua data ini akan disimpan dalam basis data supaya mudah dicari dan dibuat laporannya. Dengan sistem seperti ini, pekerjaan di percetakan jadi lebih cepat, tidak mudah salah hitung, dan semua transaksi bisa dilacak dengan mudah kapan pun dibutuhkan.

<b>NATURAL PRINTING</b> JL VETERAN NO 190 SOLO SURAKARTA Telp: 0271-668128,632692, Fax: HP/WA : 081280005781 / 089648708450		<b>NOTA</b>		KEPADA YTH <b>ARVA</b> GENTAN, BAKI, SUKOHARJO Telp:082133506723	053179 Sales : PUSAT Page : 1 / 1
NO.:	KD. BARANG:	NAMA BARANG:	UKURAN	QTY	Harga
					DISC
1	A301.36	AC 260 GR (1 SISI), 1-10 /foto/2p@1x		2	2,900.0
2	A301.415	LAMINASI HOT GLOSY 1-5 /		2	2,000.0
					Jumlah
					5,800
					4,000
CATATAN	:	AD		4.00	JUMLAH : 9,800
PUTIHAN	:				DISCOUNT : 0
SEAMING	:				TOTAL : 9,800
KELING	:				
PENGAMBILAN BARANG HARAP MEMBAWA NOTA * Kami Tidak Menerima Complaint perubahan warna ( Semua Cetakan )		RENA Time :11-06-2025 01:59:31 PM PEMBAYARAN VIA BANK BCA : A3, NONMMT,OFFSET (0157777134 a.n CV NATURAL SHEKINAH GLORY) MMT,CUTTING,GRAFIR, KAOS (0157777533 a.n CV NATURAL KARUNIA ELOHIM			
(.....)	(.....)	(.....)			

Gambar. 1 Nota Natural Printing

#### **a. Tabel Umum (Unnormalized Table)**

Tabel di bawah merupakan hasil gabungan seluruh data dari nota tanpa normalisasi. Artinya, data pelanggan dan transaksi masih berulang di setiap baris karena satu nota berisi lebih dari satu barang. Tabel ini digunakan sebagai dasar awal sebelum dilakukan proses normalisasi, agar nanti bisa dipisahkan menjadi tabel-tabel yang lebih efisien seperti tabel pelanggan, barang, transaksi, dan detail transaksi.

No Nota	Tanggal	Jth Tempo	Nama Pelanggan	Alamat	Telp	Sales	Kode Barang
0825/06/2025-A	11-06-2025	11-06-2025	Arva	Gentan, Baki, Sukoharjo	82133506723	Pusat	A301.36, A301.415
Nama Barang	Qty	Harga	Disc	Jumlah	Total	Pembayaran	Keterangan
AC 260 GR (1 SISI), 1-10/foto/2p@1x, Laminasi Hot Glossy 1-5 /	2,2	2900,2000	-	5800,4000	9800	Cash	-

Tabel. 1 Unnormalized Table

### b. Normalisasi 1NF (First Normal Form)

Tahap pertama normalisasi bertujuan untuk memastikan setiap atribut bernilai tunggal dan tidak terdapat kelompok data yang berulang dalam satu kolom. Dengan penerapan 1NF, data disusun agar setiap baris dan kolom memiliki nilai yang spesifik dan tidak mengandung pengulangan.

No Nota	Tanggal	Jth Tempo	Nama Pelanggan	Alamat	Telp	Sales	Kode Barang
0825/06/2025-A	11-06-2025	11-06-2025	Arva	Gentan, Baki, Sukoharjo	82133506723	Pusat	A301.36
Nama Barang	Qty	Harga	Disc	Jumlah	Total	Pembayaran	Keterangan
AC 260 GR (1 SISI), 1-10/foto/2p@1x	2,2	2900	-	5800	9800	Cash	-
Laminasi Hot Glossy 1-5 /	2	2000	-	4000	9800	Cash	-

Tabel. 2 1NF

### c. Normalisasi 2NF (Second Normal Form)

Tahap kedua normalisasi bertujuan untuk menghilangkan ketergantungan parsial terhadap kunci utama. Setelah memenuhi 1NF, setiap atribut dalam tabel harus

bergantung sepenuhnya pada kunci utama agar tidak terjadi duplikasi atau redundansi data.

### 1. Tabel Pelanggan

Tabel Pelanggan di bawah ini sudah memenuhi Second Normal Form (2NF) karena seluruh atribut non-key seperti nama\_customer, alamat, dan telp sepenuhnya bergantung pada satu primary key, yaitu id\_pelanggan. Karena kunci utamanya bukan merupakan kunci gabungan, maka tidak mungkin terjadi partial dependency. Selain itu, setiap data dalam tabel sudah bersifat atomik dan tidak memiliki pengulangan, sehingga syarat 1NF juga terpenuhi. Dengan demikian, tabel tersebut dapat dikatakan sudah berada pada bentuk normal 2NF.

id_pelanggan	nama_customer	alamat	telp
53179	Arva	Gentan, Baki, Sukoharjo	82133506723

Tabel. 3 2NF Pelanggan

### 2. Tabel Karyawan

Tabel Karyawan terdiri dari tiga atribut utama, yaitu id\_karyawan, nama\_karyawan, dan jabatan. Kolom id\_karyawan berfungsi sebagai pengenal unik bagi setiap karyawan, sedangkan nama\_karyawan mencatat nama individu yang bekerja di perusahaan. Kolom jabatan digunakan untuk menunjukkan posisi atau peran yang diemban oleh karyawan tersebut, seperti pada contoh di mana karyawan dengan ID K001 bernama Rena dan memiliki jabatan sebagai kasir. Tabel ini menyimpan informasi dasar mengenai identitas dan posisi karyawan.

id_karyawan	nama_karyawan	jabatan
K001	Rena	kasir

Tabel. 4 2NF Karyawan

### 3. Tabel Barang

Tabel Barang berisi informasi mengenai daftar barang yang disimpan atau dijual. Setiap barang diidentifikasi dengan kode\_barang sebagai penanda unik, misalnya A301.36 dan A301.415. Kolom nama\_barang menjelaskan nama atau jenis barang, seperti AC 260 GR (1 SISI),1-10 dan LAMINASI HOT GLOSSY 1-5. Sementara itu, kolom ukuran digunakan untuk mencatat detail ukuran barang, namun pada contoh data ditunjukkan dengan tanda “—” yang berarti ukuran tidak tersedia atau tidak dicantumkan. Tabel ini berfungsi untuk mencatat karakteristik dasar dari setiap barang dalam sistem.

Kode_barang	Nama_barang	Ukuran
-------------	-------------	--------

A301.36	AC 260 GR (1 SISI),1-10	-
A301.415	LAMINASI HOT GLOSSY 1-5	-

Tabel. 5 2NF Barang

#### 4. Tabel Order

Tabel Order mencatat detail pemesanan barang oleh pelanggan. Setiap pesanan diidentifikasi dengan id\_order, misalnya OR01 dan OR02. Kolom id\_transaksi menunjukkan transaksi utama yang terkait dengan pesanan tersebut, sedangkan id\_pelanggan menghubungkan pesanan dengan pelanggan yang melakukan pembelian. Kolom kode\_barang menunjukkan barang yang dipesan, kemudian qty mencatat jumlah barang yang dipesan, dan harga mencatat harga satuan barang tersebut. Contohnya, pesanan OR01 merupakan transaksi TR01 oleh pelanggan 53179 yang membeli 2 unit barang berkode A301.36 dengan harga 2900 per unit. Tabel ini berfungsi sebagai pencatat seluruh detail pemesanan dalam sistem.

id_order	id_transaksi	id_pelanggan	kode_barang	qty	harga
OR01	TR01	53179	A301.36	2	2900
OR02	TR02	53179	A301.415	2	2000

Tabel. 6 2NF Order

#### 5. Tabel Kasir

Tabel Kasir digunakan untuk mencatat data aktivitas input yang dilakukan oleh seorang karyawan bagian kasir. Kolom id\_kasir menjadi identitas unik untuk setiap entri kasir, sedangkan id\_karyawan menunjukkan siapa karyawan yang melakukan proses input tersebut. Kolom waktu\_input mencatat kapan data tersebut dimasukkan ke sistem. Pada contoh data, KS001 adalah kode kasir yang diinput oleh karyawan K001 pada tanggal 2025-06-11. Tabel ini membantu melacak siapa yang melakukan input dan kapan aktivitas tersebut terjadi.

id_kasir	id_karyawan	waktu_input
KS001	K001	2025-06-11

Tabel. 7 2NF Kasir

#### 6. Tabel Transaksi

Tabel Transaksi menyimpan informasi lengkap mengenai proses transaksi yang terjadi antara pelanggan dan perusahaan. Setiap transaksi diidentifikasi melalui id\_transaksi, seperti TR01. Kolom no\_bukti merekam nomor bukti transaksi, sedangkan tanggal menunjukkan kapan transaksi dilakukan dan jatuh\_tempo menandai batas waktu pembayaran sesuai ketentuan. Kolom term menjelaskan metode pembayaran, misalnya CASH. Pada bagian nilai\_transaksi,

total\_harga mencatat jumlah keseluruhan pembayaran, sementara sales menunjukkan pihak atau bagian yang melakukan penjualan. Selain itu, id\_pelanggan menghubungkan transaksi dengan pelanggan yang bersangkutan, dan id\_kasir mencatat kasir yang memproses transaksi tersebut. Contohnya, transaksi TR01 dilakukan pada 11-06-2025, bernilai 9800, dilakukan oleh sales PUSAT, untuk pelanggan 53179, dan diproses oleh kasir KS001. Tabel ini berfungsi sebagai pusat pencatatan seluruh transaksi penjualan.

id_transaksi	no_bukti	tanggal	jatuh_tempo	term	total_harga	sales	id_pelanggan	id_kasir
TR01	0825/06/2025	11-06-2025	11-06-2025	CASH	9800	PUSAT	53179	KS001

Tabel. 8 2NF Transaksi

#### d. Normalisasi 3NF (Third Normal Form)

Tahap ketiga normalisasi (3NF) bertujuan untuk menghilangkan ketergantungan transitif, yaitu ketika atribut non-kunci bergantung pada atribut non-kunci lainnya. Setelah memenuhi 2NF, setiap atribut dalam tabel harus bergantung langsung pada kunci utama. Dengan demikian, struktur tabel menjadi lebih efisien, konsisten, dan bebas dari redundansi data.

##### 1. Tabel Pelanggan

Tabel Pelanggan menyimpan informasi dasar mengenai pelanggan yang melakukan transaksi. Semua atribut dalam tabel ini sepenuhnya bergantung pada primary key yaitu id\_pelanggan, sehingga tidak ada ketergantungan parsial maupun transitive. Struktur tabel ini sudah memenuhi 3NF karena setiap data hanya menjelaskan karakteristik pelanggan dan tidak bergantung pada atribut non-kunci lainnya.

id_pelanggan	nama_customer	alamat	telp
53179	Arva	Gentan, Baki, Sukoharjo	82133506723

Tabel. 9 3NF Pelanggan

##### 2. Tabel Karyawan

Tabel Karyawan digunakan untuk menyimpan data pegawai yang terlibat dalam proses transaksi, seperti identitas, jabatan, dan bagian penjualan. Semua atribut non-primary key bergantung langsung pada id\_karyawan, sehingga tidak ada ketergantungan transitif antar atribut. Struktur ini memastikan tabel bersifat independen dan memenuhi syarat 3NF.

id_karyawan	nama_karyawan	jabatan
K001	Rena	kasir

Tabel. 10 3NF Karyawan

### 3. Tabel Barang

Tabel Barang berisi data master mengenai barang yang dijual, seperti kode, nama, dan harga per item. Seluruh atribut di dalam tabel ini bergantung langsung pada kode\_barang sebagai primary key. Tidak terdapat atribut yang saling bergantung satu sama lain sehingga tabel ini sudah sepenuhnya memenuhi 3NF.

Kode_barang	Nama_barang	Ukuran
A301.36	AC 260 GR (1 SISI),1-10	-
A301.415	LAMINASI HOT GLOSSY 1-5	-

Tabel. 11 3NF Barang

### 4. Tabel Order

Tabel Order digunakan untuk menyimpan detail pemesanan barang pada setiap transaksi. Semua atribut di dalam tabel ini bergantung langsung pada primary key yaitu id\_order, sehingga tidak terdapat ketergantungan parsial maupun transitif. Tabel ini menghubungkan transaksi dengan barang yang dipesan dan jumlah pembeliannya, sehingga tetap konsisten dengan aturan normalisasi 3NF.

id_order	id_transaksi	id_pelanggan	kode_barang	qty	harga
OR01	TR01	53179	A301.36	2	2900
OR02	TR02	53179	A301.415	2	2000

Tabel. 12 3NF Order

### 5. Tabel Kasir

Tabel Kasir berisi informasi mengenai kasir yang melakukan input transaksi. Setiap atribut sepenuhnya bergantung pada primary key id\_kasir, sehingga tidak ada atribut yang saling menentukan satu sama lain. Dengan struktur ini, tabel Kasir telah memenuhi kriteria 3NF karena hanya menyimpan data terkait peran kasir tanpa redundansi.

id_kasir	id_karyawan	waktu_input
KS001	K001	2025-06-11

Tabel. 13 3NF Kasir

### 6. Tabel Transaksi

Tabel Transaksi menyimpan informasi utama mengenai transaksi penjualan yang dilakukan oleh pelanggan. Seluruh atribut non-key bergantung langsung pada id\_transaksi, dan tidak ada ketergantungan antara atribut non-kunci sehingga

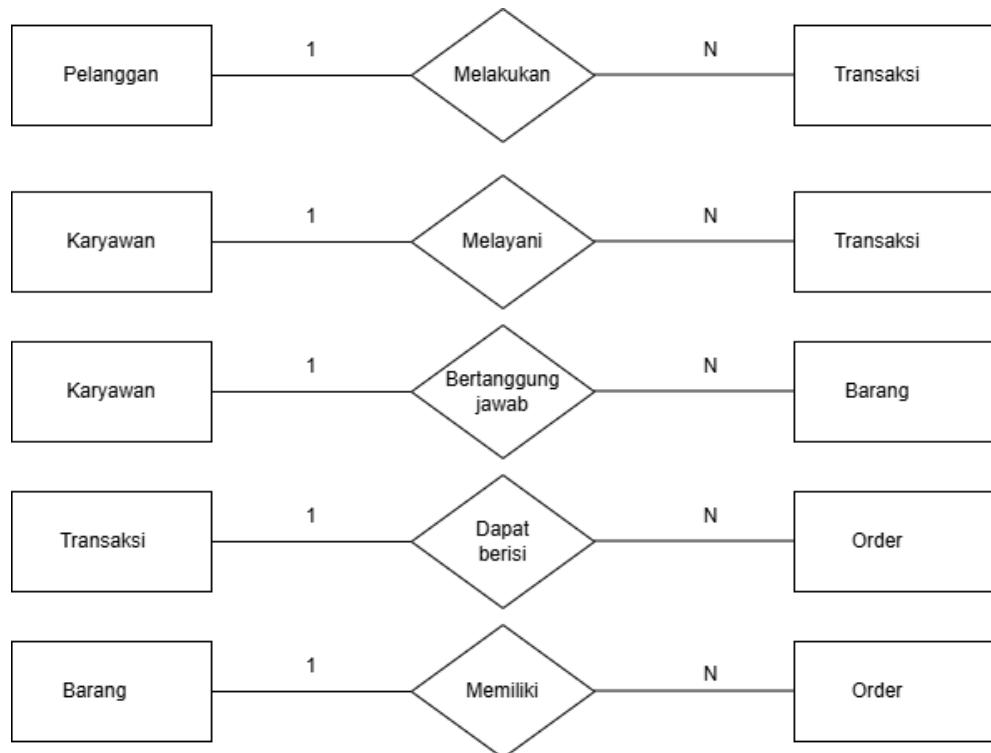
terhindar dari anomali transitive. Tabel ini juga berfungsi sebagai penghubung antara pelanggan, kasir, dan detail pembelian, namun tetap mempertahankan prinsip 3NF karena relasinya menggunakan foreign key.

id_transaksi	no_bukti	tanggal	jatuh_tempo	term	total_harga	sales	id_pelanggan	id_kasir
TR01	0825/06/2025	11-06-2025	11-06-2025	CASH	9800	PUSAT	53179	KS001

Tabel. 14 3NF Transaksi

#### e. Derajat Kardinalitas

Dalam kasus nota percetakan, derajat kardinalitas menggambarkan hubungan antar entitas yang terlibat dalam sistem. Setiap pelanggan dapat memiliki banyak nota transaksi (1:N), sedangkan setiap nota transaksi dapat berisi banyak detail pesanan atau item cetak (1:N). Setiap barang cetak dapat muncul di banyak detail transaksi (1:N), dan setiap karyawan atau sales dapat menangani banyak transaksi (1:N). Dengan demikian, hubungan antar tabel dalam sistem nota percetakan menunjukkan keterkaitan data yang saling terhubung secara logis dan mendukung pengelolaan informasi secara efisien.

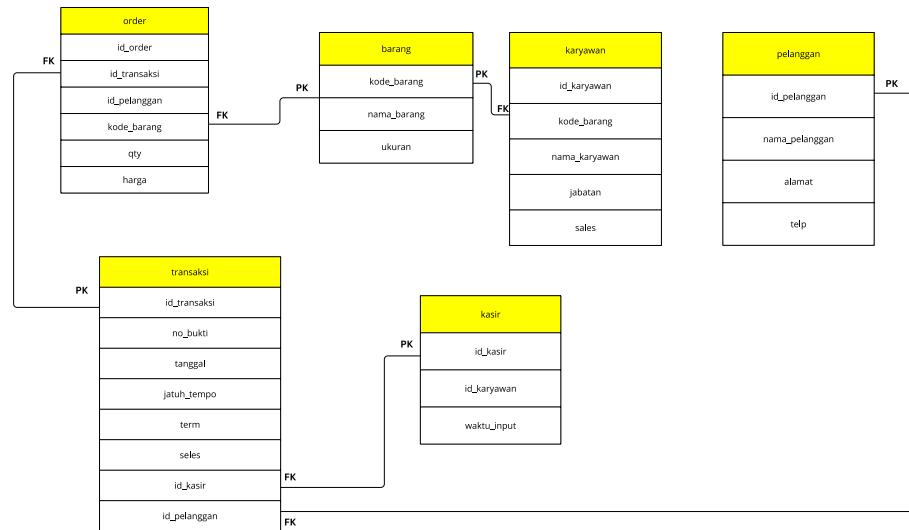


Gambar. 2 Derajat Kardinalitas

#### f. Relasi Antar Tabel

Setelah proses normalisasi dilakukan, hubungan antar tabel dapat dibentuk berdasarkan keterkaitan data dari setiap entitas. Setiap tabel dihubungkan melalui

atribut kunci utama dan kunci tamu agar data dapat saling terintegrasi. Relasi antar tabel ini menggambarkan bagaimana data pada satu entitas berhubungan dengan data pada entitas lainnya, sehingga struktur basis data menjadi lebih teratur dan mudah diakses sesuai kebutuhan sistem.



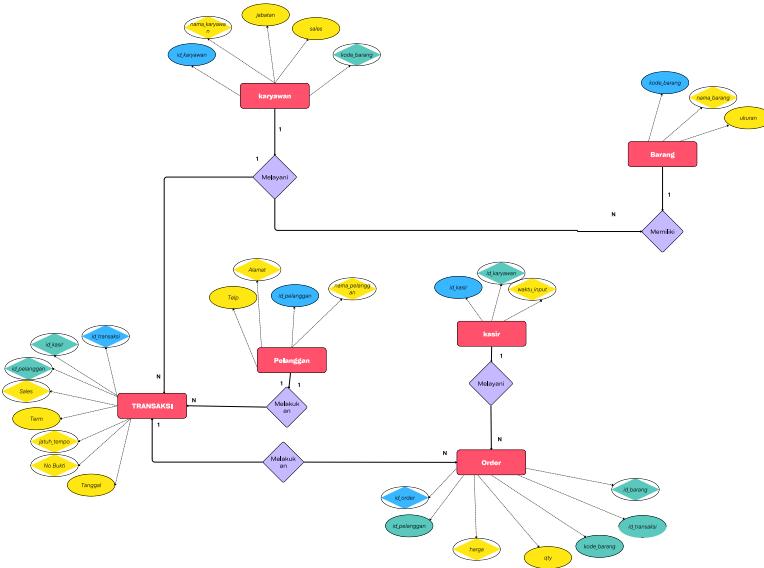
Gambar. 3 Relasi Antar Tabel

Diagram tersebut menggambarkan model relasi basis data untuk sistem transaksi penjualan yang terdiri dari enam entitas utama: pelanggan, transaksi, order, barang, karyawan, dan kasir. Entitas pelanggan menyimpan informasi dasar pelanggan dan berelasi secara one-to-many dengan entitas transaksi dan order, karena setiap pelanggan dapat melakukan beberapa transaksi dan pemesanan. Entitas transaksi memiliki atribut terkait detail pembayaran dan terhubung dengan entitas kasir serta pelanggan. Entitas kasir mencatat data operator transaksi dan berelasi dengan transaksi serta karyawan, menunjukkan bahwa seorang kasir merupakan bagian dari data karyawan. Entitas order memuat detail pemesanan seperti barang, kuantitas, dan harga; tabel ini terhubung ke transaksi, pelanggan, dan barang, menandakan bahwa setiap order berasal dari pelanggan tertentu, berkaitan dengan transaksi tertentu, dan berisi barang tertentu. Entitas barang menyimpan master data produk dan memiliki relasi dengan order serta karyawan. Sementara itu, entitas karyawan mencatat data pegawai beserta jabatan, serta berhubungan dengan barang dan kasir. Secara keseluruhan, relasi antar-entitas ini membentuk struktur data yang saling terhubung untuk mendukung proses penjualan, pencatatan transaksi, manajemen pelanggan, serta pengelolaan barang dan karyawan secara terintegrasi.

## 3.2 Perancangan Database

### 3.2.1 ERD (Entity Relationship Diagram)

Setelah relasi antar tabel ditentukan, langkah selanjutnya adalah menggambarkan hubungan tersebut dalam bentuk Entity Relationship Diagram (ERD). Diagram ini digunakan untuk memvisualisasikan entitas, atribut, serta jenis hubungan yang terjadi antar entitas. Melalui ERD, rancangan basis data



Gambar. 4 Entity Relationship Diagram

dapat dipahami dengan lebih jelas sebelum diimplementasikan, sehingga hubungan antar tabel dapat terlihat secara menyeluruh dan konsisten.

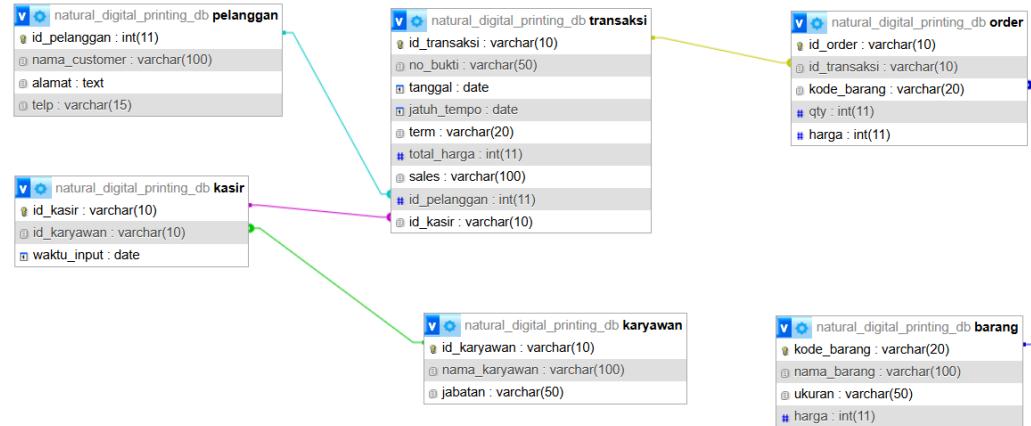
Diagram Entity–Relationship (ERD) tersebut menggambarkan struktur konseptual sistem informasi penjualan yang terdiri atas entitas Karyawan, Pelanggan, Kasir, Transaksi, Order, dan Barang beserta hubungan antar-entitasnya. Entitas Pelanggan berhubungan dengan entitas Transaksi dan Order melalui relasi melakukan, yang menunjukkan bahwa seorang pelanggan dapat melakukan banyak transaksi serta menghasilkan banyak order. Entitas Order memiliki relasi memiliki dengan entitas Barang, yang mengindikasikan bahwa satu order dapat mencakup banyak barang. Entitas Barang juga terhubung dengan Karyawan melalui relasi mengelola, menandakan bahwa barang tertentu dapat dikelola oleh lebih dari satu karyawan, misalnya dalam konteks pengelolaan inventaris atau penjualan. Sementara itu, entitas Kasir berhubungan dengan

Transaksi melalui relasi melayani, menunjukkan bahwa seorang kasir dapat memproses sejumlah transaksi. Relasi-relasi tersebut secara keseluruhan membentuk struktur many-to-one dan one-to-many yang saling terkait, sehingga mencerminkan alur bisnis mulai dari interaksi pelanggan, pencatatan order, pengelolaan barang, hingga pemrosesan transaksi oleh kasir.

### 3.2.2 Skema Tabel

Skema tabel basis data pada sistem **Natural Digital Printing** dirancang untuk mendukung proses pengelolaan data pelanggan, karyawan, transaksi penjualan, serta detail pemesanan secara terintegrasi. Perancangan skema tabel ini bertujuan untuk memastikan struktur data yang terorganisir, menghindari redundansi data, serta menjaga konsistensi hubungan antar tabel melalui penerapan *primary key* dan *foreign key*. Seluruh tabel saling terhubung sesuai dengan kebutuhan proses bisnis, mulai dari pencatatan data pelanggan hingga pencatatan detail transaksi dan pesanan.

Untuk mempermudah pemahaman terhadap struktur basis data, seluruh skema tabel ditampilkan dalam bentuk **screenshot** yang menunjukkan susunan kolom, tipe data, serta relasi antar tabel. Adapun ringkasan skema tabel yang digunakan dalam sistem Natural Digital Printing dapat dilihat pada gambar berikut :



Gambar. 5 Skema Relasi Tabel Basis Data

Gambar di atas menunjukkan skema relasi tabel pada basis data **Natural Digital Printing** yang terdiri dari tabel pelanggan, karyawan, kasir, barang, transaksi, dan order. Setiap tabel memiliki *primary key* sebagai identitas unik, sedangkan hubungan antar tabel dibentuk menggunakan *foreign key* untuk menjaga integritas data. Tabel **transaksi** berperan sebagai tabel utama yang menghubungkan data pelanggan dan kasir, sedangkan tabel **order** berfungsi sebagai tabel detail yang mencatat item barang dalam setiap transaksi. Relasi

antar tabel dirancang untuk mendukung proses pencatatan transaksi penjualan secara terstruktur dan konsisten.

### 3.2.3 PK & FK

Primary Key (PK) adalah kunci utama dalam sebuah tabel basis data yang berfungsi sebagai identitas unik untuk setiap data. PK digunakan untuk membedakan satu record dengan record lainnya sehingga tidak terjadi data ganda. Nilai Primary Key harus bersifat unik dan tidak boleh kosong. Setiap tabel hanya memiliki satu Primary Key, dan kunci ini sangat penting untuk memudahkan pencarian data serta menjadi acuan dalam pembentukan relasi antar tabel.

Foreign Key (FK) adalah kunci yang digunakan untuk menghubungkan satu tabel dengan tabel lainnya dalam basis data. FK merupakan kolom yang nilainya mengacu pada Primary Key di tabel lain. Foreign Key berfungsi untuk menjaga keterkaitan dan konsistensi data antar tabel, sehingga data yang dimasukkan harus sesuai dengan data yang sudah ada pada tabel yang dirujuk. Dengan adanya Foreign Key, hubungan antar data menjadi terstruktur dan mencegah terjadinya kesalahan atau ketidaksesuaian data.

#### a. Primary Key

Pada sistem yang digunakan, setiap tabel memiliki **Primary Key (PK)** sebagai identitas unik untuk membedakan setiap data. Pada tabel **pelanggan**, Primary Key yang digunakan adalah **id\_pelanggan** yang berfungsi sebagai penanda unik untuk setiap pelanggan. Tabel **karyawan** menggunakan **id\_karyawan** sebagai Primary Key untuk mengidentifikasi setiap karyawan secara unik, sedangkan pada tabel **kasir** digunakan **id\_kasir** sebagai Primary Key yang membedakan setiap kasir. Selanjutnya, tabel **barang** memiliki **kode\_barang** sebagai Primary Key yang berfungsi sebagai kode unik untuk setiap produk atau barang. Pada tabel **transaksi**, **id\_transaksi** digunakan sebagai Primary Key untuk mengidentifikasi setiap transaksi yang terjadi. Terakhir, tabel **order** menggunakan **id\_order** sebagai Primary Key yang berfungsi sebagai identitas unik untuk setiap detail order. Seluruh Primary Key tersebut memastikan bahwa tidak ada data yang terduplikasi serta memudahkan proses pengelolaan dan

penghubungan data antar tabel dalam sistem.

Tabel	Primary Key	Deskripsi
Pelanggan	Id_pelanggan	Unique identifier untuk setiap pelanggan
Karyawan	Id_karyawan	Unique identifier untuk setiap karyawan
Kasir	Id_kasir	Unique identifier untuk setiap kasir
Barang	kode_barang	Kode unik untuk setiap produk/barang
Transaksi	Id_transaksi	Unique identifier untuk setiap transaksi
Order	Id_order	Unique identifier untuk setiap detail order

Tabel. 15 Primary Key

#### b. Foreign Key

Foreign Key (FK) digunakan untuk menghubungkan data antar tabel agar tetap konsisten dan saling berkaitan. Pada tabel **kasir**, kolom **id\_karyawan** berperan sebagai Foreign Key yang mengacu pada **id\_karyawan** sebagai Primary Key di tabel **karyawan**, dengan aturan aksi **CASCADE/RESTRICT** yang mengatur perubahan atau penghapusan data. Pada tabel **transaksi**, terdapat dua Foreign Key, yaitu **id\_pelanggan** yang mengacu pada **id\_pelanggan** di tabel **pelanggan** serta **id\_kasir** yang mengacu pada **id\_kasir** di tabel **kasir**, keduanya juga menggunakan aksi **CASCADE/RESTRICT** untuk menjaga integritas data. Selanjutnya, pada tabel **order**, kolom **id\_transaksi** menjadi Foreign Key yang mengacu pada

**id\_transaksi** di tabel **transaksi** dengan aksi **CASCADE/CASCADE**, yang berarti perubahan maupun penghapusan data transaksi akan memengaruhi data order terkait. Selain itu, tabel **order** juga memiliki **kode\_barang** sebagai Foreign Key yang mengacu pada **kode\_barang** di tabel **barang** dengan aksi **CASCADE/RESTRICT**. Penggunaan Foreign Key ini memastikan hubungan antar tabel berjalan dengan baik serta mencegah terjadinya data yang tidak valid dalam sistem.

Tabel Child	FK Column	Tabel Parent	PK Column	Action
Kasir	Id_karyawan	karyawan	Id_karyawan	CASCADE/RESTRICT
Transaksi	Id_pelanggan	pelanggan	Id_pelanggan	CASCADE/RESTRICT
Transaksi	Id_kasir	kasir	Id_kasir	CASCADE/RESTRICT
Order	Id_transaksi	transaksi	Id_transaksi	CASCADE/CASCADE
order	Kode_barang	barang	Kode_barang	CASCADE/RESTRICT

Tabel. 16 Foreign Key

### 3.3 Implementasi Basis Data

Implementasi basis data pada sistem **Natural Digital Printing** dilakukan secara langsung menggunakan DBMS MySQL berdasarkan hasil perancangan yang telah dibuat pada tahap sebelumnya. Implementasi ini mencakup penerapan **Data Definition Language (DDL)** untuk membangun struktur basis data, **Data Manipulation Language (DML)** untuk pengisian dan pengelolaan data, serta **Transaction Control Language (TCL)** untuk pengujian mekanisme transaksi. Seluruh proses implementasi telah dijalankan dan dibuktikan melalui pengujian langsung pada basis data.

#### 3.3.1 Implementasi DDL

Implementasi DDL dilakukan untuk membangun struktur basis data sesuai dengan rancangan sistem Natural Digital Printing. Pada tahap ini, database dan seluruh tabel utama berhasil dibuat, meliputi tabel pelanggan, karyawan, kasir, barang, transaksi, dan order. Setiap tabel telah dilengkapi dengan **Primary Key** sebagai identitas unik data serta **Foreign Key** untuk membentuk relasi antar tabel. Selain itu, penerapan constraint seperti **NOT NULL**, **ON UPDATE CASCADE**, dan **ON DELETE RESTRICT/CASCADE** telah diuji dan berfungsi dengan baik dalam menjaga integritas data. Keberhasilan implementasi

DDL dibuktikan dengan terbentuknya seluruh tabel dan relasi sesuai perancangan. Script DDL ditampilkan pada bagian lampiran 1.

### 3.3.2 Implementasi DML

Implementasi DML dilakukan untuk mengisi dan memanipulasi data pada basis data Natural Digital Printing. Pengujian DML meliputi proses penambahan data pelanggan, karyawan, kasir, barang, transaksi, serta detail order menggunakan perintah INSERT. Selain itu, perintah SELECT digunakan untuk memverifikasi bahwa data berhasil tersimpan dan relasi antar tabel berjalan dengan benar. Hasil implementasi menunjukkan bahwa seluruh data dapat disimpan, ditampilkan, dan saling terhubung sesuai dengan aturan Foreign Key yang diterapkan. Script DML lengkap disajikan pada bagian lampiran 2.

### 3.3.3 Implementasi TCL

Implementasi Transaction Control Language (TCL) dilakukan untuk menguji pengendalian transaksi dalam basis data. Pengujian dilakukan dengan menjalankan skenario transaksi menggunakan perintah START TRANSACTION, diikuti dengan proses INSERT data transaksi, kemudian dilakukan pembatalan menggunakan perintah ROLLBACK. Hasil pengujian menunjukkan bahwa data yang dibatalkan tidak tersimpan dalam basis data, sehingga membuktikan bahwa mekanisme pengendalian transaksi berjalan dengan baik dan konsistensi data tetap terjaga. Script TCL ditampilkan pada bagian lampiran 3.

## 3.4 Implementasi Query SQL

Implementasi query SQL pada sistem **Natural Digital Printing** dilakukan untuk menguji kemampuan basis data dalam menghasilkan informasi yang dibutuhkan sesuai dengan proses bisnis sistem. Query SQL yang diimplementasikan meliputi query **JOIN**, **GROUP BY**, **HAVING**, serta penggunaan fungsi **agregasi** seperti SUM, COUNT, AVG, MAX, dan MIN. Seluruh query dijalankan langsung pada basis data yang telah diisi dengan data uji. Implementasi Query Sql dari JOIN sampai Agregasi dapat dilihat di lampiran 4

### 3.4.1 Query JOIN

Implementasi query JOIN dilakukan untuk menggabungkan data dari beberapa tabel yang saling berelasi, yaitu tabel order, transaksi, pelanggan, dan barang. Pengujian JOIN menunjukkan bahwa data dapat ditampilkan secara terintegrasi, sehingga informasi detail pesanan seperti nama pelanggan, nomor bukti transaksi, nama barang, jumlah, harga, dan subtotal dapat ditampilkan dalam satu hasil query. Keberhasilan implementasi ini membuktikan bahwa relasi antar tabel telah berjalan dengan baik sesuai dengan perancangan. Script query JOIN ditampilkan pada bagian lampiran.

#### 3.4.2 Query GROUP BY

Implementasi query GROUP BY dilakukan untuk mengelompokkan data berdasarkan identitas transaksi dan menghitung total nilai transaksi. Pada pengujian ini, fungsi agregasi SUM digunakan untuk menjumlahkan nilai subtotal dari setiap item dalam transaksi yang sama. Hasil pengujian menunjukkan bahwa total transaksi yang dihasilkan sesuai dengan nilai total\_harga pada tabel transaksi, sehingga membuktikan keakuratan perhitungan dan konsistensi data. Script query GROUP BY disajikan pada bagian lampiran.

#### 3.4.3 Query HAVING

Implementasi query HAVING dilakukan untuk memfilter hasil data yang telah dikelompokkan berdasarkan kondisi tertentu. Pada sistem ini, HAVING digunakan untuk menampilkan transaksi dengan total nilai di atas batas tertentu. Hasil pengujian menunjukkan bahwa hanya transaksi yang memenuhi kondisi yang ditampilkan, sehingga membuktikan bahwa query HAVING berfungsi dengan baik dalam menyaring hasil agregasi. Script query HAVING ditampilkan pada bagian lampiran.

#### 3.4.4 Query Agregasi

Implementasi query agregasi dilakukan untuk menganalisis data penjualan dalam basis data Natural Digital Printing. Fungsi agregasi seperti SUM digunakan untuk menghitung total nilai transaksi, COUNT untuk menghitung jumlah transaksi, AVG untuk menghitung rata-rata nilai transaksi, serta MAX dan MIN untuk mengetahui transaksi dengan nilai tertinggi dan terendah. Hasil pengujian menunjukkan bahwa query agregasi mampu menghasilkan informasi statistik yang akurat dan berguna sebagai bahan analisis data penjualan. Script query agregasi lengkap disajikan pada bagian lampiran.

## BAB IV

### PENUTUP

#### 4.1 Hasil Pengujian

Hasil pengujian basis data pada sistem **Natural Digital Printing** menunjukkan bahwa seluruh perintah **Data Definition Language (DDL)**, **Data Manipulation Language (DML)**, dan **Transaction Control Language (TCL)** dapat dijalankan dengan baik dan sesuai dengan perancangan sistem. Pengujian DDL membuktikan bahwa database serta seluruh tabel berhasil dibuat tanpa error, lengkap dengan penerapan Primary Key dan Foreign Key yang berfungsi sesuai aturan **CASCADE** dan **RESTRICT**, sehingga integritas data antar tabel tetap terjaga. Selanjutnya, pengujian DML menunjukkan bahwa proses penambahan dan penampilan data pada setiap tabel berjalan dengan benar, di mana data pelanggan, karyawan, kasir, barang, transaksi, dan detail order berhasil disimpan serta saling terhubung sesuai relasinya. Sementara itu, pengujian TCL memperlihatkan bahwa mekanisme pengendalian transaksi berjalan secara optimal, ditandai dengan keberhasilan proses **ROLLBACK** dalam membatalkan penyimpanan data transaksi yang tidak dikonfirmasi. Secara keseluruhan, hasil pengujian ini menunjukkan bahwa basis data telah berfungsi secara konsisten, aman, dan siap digunakan untuk mendukung operasional sistem Natural Digital Printing.

Objek Uji	Perintah SQL	Skenario Pengujian	Hasil yang diharapkan	Hasil aktual	Status
Database	CREATE DATABASE	Membuat database baru	Database berhasil dibuat	Database berhasil dibuat	Berhasil
Tabel Pelanggan	CREATE TABLE pelanggan	Membuat tabel pelanggan dengan PK	Tabel terbentuk dengan PK	Tabel terbentuk	Berhasil
Tabel Karyawan	CREATE TABLE karyawan	Membuat tabel karyawan dengan PK	Tabel terbentuk dengan PK	Tabel terbentuk	Berhasil

Tabel Kasir	CREATE TABLE kasir	Membuat FK ke tabel karyawan	FK terbentuk	FK berjalan	Berhasil
Tabel Barang	CREATE TABLE barang	Membuat tabel barang dengan PK	Tabel terbentuk dengan PK	Tabel terbentuk	Berhasil
Tabel Transaksi	CREATE TABLE transaksi	Membuat FK ke pelanggan & kasir	FK terbentuk	FK berjalan	Berhasil
Tabel Order	CREATE TABLE order	Membuat FK ke transaksi & barang	FK terbentuk	FK berjalan	Berhasil

Tabel. 17 Pengujian DDL

Objek Uji	Perintah SQL	Data Uji	Hasil yang diharapkan	Hasil aktual	Status
pelanggan	INSERT	ID 53179 – Arva	Data tersimpan	Data tersimpan	Berhasil
karyawan	INSERT	K001 – Rena	Data tersimpan	Data tersimpan	Berhasil
kasir	INSERT	KS001 – K001	Relasi valid	Relasi valid	Berhasil
barang	INSERT	A301.36, A301.415	Data tersimpan	Data tersimpan	Berhasil
transaksi	INSERT	TR01	Data tersimpan	Data tersimpan	Berhasil

order	INSERT	OR01, OR02	Data terhubung ke transaksi	Data terhubung	Berhasil
Semua tabel	SELECT	SELECT *	Data tampil lengkap	Data tampil	Berhasil

Tabel. 18 Pengujian DML

Urutan Uji	Perintah SQL	Skenario	Hasil yang diharapkan	Hasil aktual	Status
Mulai transaksi	START TRANSACTION	Memulai transaksi	Transaksi aktif	Transaksi aktif	Berhasil
Insert transaksi	INSERT TR02	Menyimpan sementara	Data tersimpan sementara	Data sementara	Berhasil
Pembatalan	ROLLBACK	Membatalkan transaksi	Data tidak tersimpan	Data dibatalkan	Berhasil
Verifikasi	SELECT transaksi	Cek data	TR02 tidak ada	TR02 tidak ada	Berhasil

Tabel. 19 Pengujian TCL

## 4.2 Kendala dan Solusi

Selama proses perancangan, implementasi, dan pengujian basis data pada sistem **Natural Digital Printing**, beberapa kendala ditemukan pada tahap awal pengembangan. Kendala utama yang muncul adalah pengaturan relasi antar tabel menggunakan Primary Key dan Foreign Key, khususnya terkait urutan pembuatan tabel serta penerapan aturan **CASCADE** dan **RESTRICT**. Selain itu, pada tahap manipulasi data, proses input data sempat mengalami kegagalan karena data pada tabel parent belum tersedia, sehingga melanggar aturan Foreign Key. Pada pengujian transaksi menggunakan TCL, data yang telah diinput tidak tersimpan akibat penggunaan perintah **ROLLBACK**, yang pada awalnya dianggap sebagai kesalahan namun kemudian dipahami sebagai bagian dari mekanisme pengendalian transaksi. Kendala juga ditemukan pada pengujian query JOIN dan agregasi, di mana hasil data tidak tampil sesuai harapan karena kesalahan kondisi relasi atau data yang belum lengkap.

Solusi yang diterapkan untuk mengatasi kendala tersebut adalah dengan memperbaiki urutan pembuatan tabel, memastikan kesesuaian tipe data antara Primary Key dan Foreign Key, serta memahami fungsi aturan CASCADE dan RESTRICT sesuai kebutuhan sistem. Pada tahap manipulasi data, dilakukan validasi dan penyesuaian data sebelum proses insert agar relasi antar tabel tetap terjaga. Untuk pengujian TCL, penggunaan perintah **COMMIT** diterapkan ketika data perlu disimpan secara permanen, sedangkan **ROLLBACK** digunakan untuk simulasi pembatalan transaksi. Selain itu, kondisi JOIN dan query agregasi diperbaiki agar hasil yang ditampilkan sesuai dengan data aktual. Dengan penerapan solusi tersebut, seluruh kendala dapat diatasi dan sistem basis data Natural Digital Printing dapat berjalan dengan stabil, konsisten, dan sesuai dengan perancangan.

### 4.3 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian basis data pada sistem Natural Digital Printing, dapat disimpulkan bahwa proses perancangan basis data transaksi telah berhasil dilakukan sesuai dengan kebutuhan sistem. Perancangan basis data dimulai dari analisis kebutuhan, penyusunan tabel tidak ternormalisasi, hingga proses normalisasi sampai bentuk normal ketiga (3NF), sehingga struktur basis data menjadi lebih efisien, konsisten, dan bebas dari redundansi data.

Implementasi basis data menggunakan DBMS MySQL berhasil dilakukan dengan menerapkan perintah Data Definition Language (DDL) untuk membangun struktur database, Data Manipulation Language (DML) untuk mengelola data, serta Transaction Control Language (TCL) untuk mengendalikan transaksi. Penerapan Primary Key dan Foreign Key pada setiap tabel mampu menjaga integritas data dan memastikan keterkaitan antar tabel berjalan dengan baik sesuai dengan aturan relasi yang telah dirancang.

Selain itu, pengujian query SQL seperti JOIN, GROUP BY, HAVING, dan fungsi agregasi menunjukkan bahwa basis data mampu menghasilkan informasi transaksi secara akurat dan terstruktur. Dengan demikian, sistem basis data yang dibangun telah memenuhi tujuan proyek, yaitu mendukung pengelolaan data pelanggan, barang, karyawan, kasir, dan transaksi penjualan secara terintegrasi pada Natural Digital Printing. memenuhi tujuan proyek, yaitu mendukung pengelolaan data pelanggan, barang, karyawan, kasir, dan transaksi penjualan secara terintegrasi pada Natural Digital Printing.

### 4.4 Saran

Berdasarkan hasil pengembangan dan pengujian sistem basis data ini, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan selanjutnya. Sistem

basis data dapat dikembangkan lebih lanjut dengan menambahkan antarmuka aplikasi agar proses input dan pengelolaan data menjadi lebih mudah bagi pengguna. Selain itu, dapat ditambahkan fitur keamanan data, seperti pengaturan hak akses pengguna (admin, kasir, dan manajer) untuk meningkatkan kontrol dan keamanan informasi. Pengembangan selanjutnya juga dapat mencakup penambahan modul laporan otomatis, seperti laporan penjualan harian, bulanan, dan tahunan, serta integrasi dengan sistem lain seperti manajemen stok atau akuntansi. Dengan pengembangan tersebut, sistem basis data Natural Digital Printing diharapkan dapat memberikan manfaat yang lebih optimal dan mendukung operasional usaha secara lebih efisien dan profesional.

## LAMPIRAN

### Lampiran 1

Script SQL Data Definition Language (DDL)

```
CREATE DATABASE natural_digital_printing_db;  
USE natural_digital_printing_db;  
  
CREATE TABLE pelanggan (  
    id_pelanggan INT PRIMARY KEY,  
    nama_customer VARCHAR(100) NOT NULL,  
    alamat TEXT,  
    telp VARCHAR(15)  
);  
  
CREATE TABLE karyawan (  
    id_karyawan VARCHAR(10) PRIMARY KEY,  
    nama_karyawan VARCHAR(100) NOT NULL,  
    jabatan VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE kasir (  
    id_kasir VARCHAR(10) PRIMARY KEY,  
    id_karyawan VARCHAR(10) NOT NULL,  
    waktu_input DATE NOT NULL,  
    CONSTRAINT fk_kasir_karyawan
```

```
FOREIGN KEY (id_karyawan)
REFERENCES karyawan(id_karyawan)
ON UPDATE CASCADE
ON DELETE RESTRICT
);

CREATE TABLE barang (
    kode_barang VARCHAR(20) PRIMARY KEY,
    nama_barang VARCHAR(100) NOT NULL,
    ukuran VARCHAR(50),
    harga INT NOT NULL
);

CREATE TABLE transaksi (
    id_transaksi VARCHAR(10) PRIMARY KEY,
    no_bukti VARCHAR(50) NOT NULL,
    tanggal DATE NOT NULL,
    jatuh_tempo DATE NOT NULL,
    term VARCHAR(20) NOT NULL,
    total_harga INT NOT NULL,
    sales VARCHAR(100),
    id_pelanggan INT NOT NULL,
    id_kasir VARCHAR(10) NOT NULL,
    CONSTRAINT fk_transaksi_pelanggan
        FOREIGN KEY (id_pelanggan)
```

```
    REFERENCES pelanggan(id_pelanggan)
    ON UPDATE CASCADE
    ON DELETE RESTRICT,
CONSTRAINT fk_transaksi_kasir
    FOREIGN KEY (id_kasir)
    REFERENCES kasir(id_kasir)
    ON UPDATE CASCADE
    ON DELETE RESTRICT
);
```

```
CREATE TABLE `order` (
    id_order VARCHAR(10) PRIMARY KEY,
    id_transaksi VARCHAR(10) NOT NULL,
    kode_barang VARCHAR(20) NOT NULL,
    qty INT NOT NULL,
    harga INT NOT NULL,
    CONSTRAINT fk_order_transaksi
        FOREIGN KEY (id_transaksi)
        REFERENCES transaksi(id_transaksi)
        ON UPDATE CASCADE
        ON DELETE CASCADE,
CONSTRAINT fk_order_barang
    FOREIGN KEY (kode_barang)
    REFERENCES barang(kode_barang)
    ON UPDATE CASCADE
```

```
    ON DELETE RESTRICT
```

```
);
```

## Lampiran 2

### Script SQL Data Manipulation Language (DML)

```
INSERT INTO pelanggan VALUES
```

```
(53179, 'Arva', 'Gentan, Baki, Sukoharjo', '82133506723');
```

```
INSERT INTO karyawan VALUES
```

```
('K001', 'Rena', 'Kasir');
```

```
INSERT INTO kasir VALUES
```

```
('KS001', 'K001', '2025-06-11');
```

```
INSERT INTO barang VALUES
```

```
('A301.36', 'AC 260 GR (1 SISI)', '-', 2900),
```

```
('A301.415', 'LAMINASI HOT GLOSSY 1-5', '-', 2000);
```

```
INSERT INTO transaksi VALUES
```

```
('TR01', 'NDP/0825/06/2025', '2025-06-11', '2025-06-11',
```

```
'CASH', 9800, 'Natural Digital Printing', 53179, 'KS001');
```

```
INSERT INTO `order` VALUES
```

```
('OR01', 'TR01', 'A301.36', 2, 2900),
```

```
('OR02', 'TR01', 'A301.415', 2, 2000);
```

### Lampiran 3

Script SQL Transaction Control Language (TCL)

```
START TRANSACTION;

INSERT INTO transaksi VALUES
('TR02', 'NDP/0826/06/2025', '2025-06-12', '2025-06-12',
'CASH', 4000, 'Natural Digital Printing', 53179, 'KS001');

ROLLBACK;
```

### Lampiran 4

Script Query SQL JOIN dan Agregasi

```
SELECT
    p.nama_customer,
    t.no_bukti,
    b.nama_barang,
    o.qty,
    o.harga,
    (o.qty * o.harga) AS subtotal
FROM `order` o
JOIN transaksi t ON o.id_transaksi = t.id_transaksi
JOIN pelanggan p ON t.id_pelanggan = p.id_pelanggan
JOIN barang b ON o.kode_barang = b.kode_barang;

SELECT
```

```
id_transaksi,  
SUM(qty * harga) AS total_transaksi  
FROM `order`  
GROUP BY id_transaksi;  
  
SELECT  
    id_transaksi,  
    SUM(qty * harga) AS total_transaksi  
FROM `order`  
GROUP BY id_transaksi  
HAVING total_transaksi > 5000;  
  
SELECT * FROM transaksi;  
SELECT * FROM pelanggan;  
SELECT * FROM karyawan;  
SELECT * FROM kasir;  
SELECT * FROM barang;  
SELECT * FROM transaksi;  
SELECT * FROM `order`;
```

## Lampiran 5

Repository GitHub

Link GitHub : <https://github.com/InsanNiaCahayaPutri/Natural-Digital-Printing>  
Struktur Folder GitHub :

natural-digital-printing-

```
|  
|   └── dokumen/  
|       |   └── laporan.pdf  
|       |   └── laporan.docx  
|   └── sql/  
|       |   └── natural_digital_printing.sql  
|  
└── README.md
```

## DAFTAR PUSTAKA

1. Supriyanti, W. (2021). *Konsep Dasar Sistem Basis Data dengan MySQL*. Muhammadiyah University Press.
2. Suyanto, A. H. (2004). Basis Data Dan DBMS. Yogyakarta: Universitas Gajah Mada.
3. Hardini, M., Agarwal, V., Apriani, D., Widjaya, I. A., Setiawaty, E., & Nurasiah. (2025). *Application of database normalization in increasing data storage efficiency*. International Transactions on Artificial Intelligence (ITALIC), 3(2), 201-211. <https://doi.org/10.33050/italic.v3i2.799>
4. Arkan, M. N. (2025). *Perancangan Entity-Relationship Diagram (ERD) pada Basis Data Perpustakaan*. Teknofile. <https://jurnal.nawansa.com/index.php/teknofile/article/view/411>
5. Saha, B., Bagui, R., & Walsh-Earp, R. (2022). *Database design using entity-relationship diagrams: Essential to database design*. Routledge.
6. Hilman, N., Nafila, R., Putri, R. A., Zahwa, S., Nugroho, D. A., & Pujiono, I. P. (2025). Analisis Perbandingan Basis Data Relasional (SQL) dan Non-Relasional (NoSQL) Berdasarkan Efisiensi Penyimpanan dan Skalabilitas Data. *Jurnal RESTIKOM: Riset Teknik Informatika dan Komputer*, 7(3), 367-373.
7. Yunianto, D. R., Putra, Y. E., & Rahmad, C. (2023). *Comparison of relational database modeling performance based on number of normalized entities*. SMARTICS, 9(1), 42-48. <https://ejournal.unikama.ac.id/index.php/jst/article/view/8390>
8. Budi, H. S. (2021). *Pengenalan Dasar SQL*. Deepublish.