



Laporan Praktikum

Mata pelajaran Konsentrasi Keahlian

Rekayasa Perangkat Lunak - Kelas XI

Materi Ajar :

ENTITY RELATIONAL DIAGRAM

No.	Nama	NIS	Kelas
1.	Melky Orella Baresky L.	20369	XI RPL

Pengembangan Perangkat Lunak dan Gim

SMK Negeri 1 Bawang

2026



Tujuan Praktikum

- Memahami konsep dasar pemodelan data konseptual.
- Mampu mengidentifikasi entitas, atribut, dan derajat relasi.
- Mampu menggambar dan membaca diagram ERD dengan benar.

Dasar Teori

- **Entitas:** Objek yang ada dan dapat dibedakan dari objek lain (contoh: Mahasiswa, Buku).
- **Atribut:** Karakteristik atau sifat dari suatu entitas (contoh: NIM, Nama Mahasiswa).
- **Relasi:** Hubungan yang terjadi antara satu atau lebih entitas (contoh: Mahasiswa *meminjam* Buku).
- **Kardinalitas:** Batasan jumlah keterhubungan antar entitas (1 ke 1, 1 ke banyak, banyak ke banyak).

Tujuan Pembelajaran

Peserta didik mampu :

1. memahami Entity Relational Diagram (ERD) dengan baik dan benar
2. menerapkan membuat ERD pada aplikasi draw.io dengan baik dan benar
3. membuat Class Diagram berdasarkan ERD yang dibuat dengan baik dan benar

Sumber daya yang diperlukan

1. Media : Google Document, LMS (Elskanza),
2. Alat : PC / Laptop,
3. Software : XAMPP, HeidiSQL dll.

Instruksi

Buatlah Entity Relational Diagram mengenai:

- Pilihan Ketua OSIS
- Rental PS
- Aplikasi Kasir
- Aplikasi Jurnal Pembelajaran Siswa
- Aplikasi Poin Pelanggaran Siswa

Pilih salah satu ERD yang akan digunakan:

- Buat ERD pada draw.io (Hal. 1)
- Buat Class Diagram pada draw.io (Hal. 2)
- Buat tabel basis data berdasarkan ERD yang dibuat

Boleh menggunakan AI

Selamat mengerjakan!

Langkah Pengerjaan

Jelaskan ERD1 yang dibuat!

1. Gambaran Umum ERD

- ERD (Entity Relationship Diagram) digunakan untuk menggambarkan struktur database sistem rental PS.
- Terdapat **4 entitas utama**, yaitu:
 - Pelanggan
 - PS
 - Kasir
 - Rental

2. Tabel Pelanggan

Fungsi:

- Menyimpan data pelanggan yang melakukan penyewaan.

Atribut:

- **id_pelanggan (PK)** → Kunci utama.
- **nama** → Nama pelanggan.
- **no_hp** → Nomor telepon pelanggan.
- **alamat** → Alamat pelanggan.

3. Tabel PS

Fungsi:

- Menyimpan data PlayStation yang tersedia.

Atribut:

- **id_ps (PK)** → Kunci utama.
- **merek** → Merek PS.
- **tipe_ps** → Jenis atau tipe PS.
- **tarif_per_jam** → Harga sewa per jam.
- **status** → Kondisi PS (tersedia/digunakan).

4. Tabel Kasir

Fungsi:

- Menyimpan data akun kasir.

Atribut:

- **id_kasir (PK)** → Kunci utama.
- **nama_kasir** → Nama kasir.
- **username** → Username login.
- **password** → Password login.

5. Tabel Rental

Fungsi:

- Menyimpan data transaksi penyewaan.

Atribut:

- **id_rental (PK)** → Kunci utama.
- **jam_mulai** → Waktu mulai sewa.
- **jam_selesai** → Waktu selesai sewa.
- **total_bayar** → Total biaya yang dibayar.
- **id_pelanggan (FK)** → Menghubungkan ke tabel Pelanggan.
- **id_ps (FK)** → Menghubungkan ke tabel PS.
- **id_kasir (FK)** → Menghubungkan ke tabel Kasir.

6. Relasi Antar Tabel

- **Pelanggan → Rental**
 - Satu pelanggan dapat memiliki banyak transaksi rental.
 - Setiap transaksi hanya dimiliki oleh satu pelanggan.
- **PS → Rental**
 - Satu PS dapat disewa berkali-kali pada waktu yang berbeda.
 - Setiap transaksi menggunakan satu PS.
- **Kasir → Rental**
 - Satu kasir dapat melayani banyak transaksi.
 - Setiap transaksi diproses oleh satu kasir.

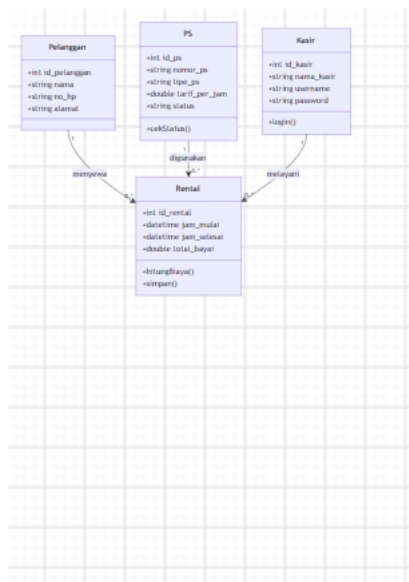
7. Kesimpulan

- **Pelanggan** menyewa **PS**.
- Proses penyewaan dicatat pada tabel **Rental**.
- **Kasir** bertugas melayani dan mencatat transaksi.
- Tabel **Rental** menjadi pusat yang menghubungkan seluruh data melalui **Foreign Key (FK)**.

Tangkapan Layar ERD 1



Tangkapan Layar Class Diagram 1



Skrip SQL dari ERD 1

```
CREATE DATABASE rental_ps;
```

```
USE rental_ps;
```

```
CREATE TABLE pelanggan (
```

```
    id_pelanggan INT AUTO_INCREMENT PRIMARY KEY,
```

```
    nama VARCHAR(100) NOT NULL,
```

```
    no_hp VARCHAR(15),
```

```
    alamat TEXT
```

```
);
```

```
CREATE TABLE ps (
```

```
    id_ps INT AUTO_INCREMENT PRIMARY KEY,
```

```
    nomor_ps VARCHAR(20) NOT NULL,
```

```
    tipe_ps VARCHAR(20) NOT NULL,
```

```
    tarif_per_jam DECIMAL(10,2) NOT NULL,
```

```
    status ENUM('Tersedia','Dipakai','Perawatan') DEFAULT 'Tersedia'
```

```
);
```

```
CREATE TABLE kasir (
```

```
    id_kasir INT AUTO_INCREMENT PRIMARY KEY,
```

```
    nama_kasir VARCHAR(100) NOT NULL,
```

```
    username VARCHAR(50) UNIQUE NOT NULL,
```

```
    password VARCHAR(255) NOT NULL
```

```
);
```

```
CREATE TABLE rental (
```

```
id_rental INT AUTO_INCREMENT PRIMARY KEY,  
  
id_pelanggan INT NOT NULL,  
  
id_ps INT NOT NULL,  
  
id_kasir INT NOT NULL,  
  
jam_mulai DATETIME NOT NULL,  
  
jam_selesai DATETIME NOT NULL,  
  
total_bayar DECIMAL(10,2) NOT NULL,  
  
  
FOREIGN KEY (id_pelanggan) REFERENCES pelanggan(id_pelanggan),  
  
FOREIGN KEY (id_ps) REFERENCES ps(id_ps),  
  
FOREIGN KEY (id_kasir) REFERENCES kasir(id_kasir)  
);
```

Jelaskan ERD2 yang dibuat!

1. Tabel KASIR

Berfungsi untuk menyimpan data kasir yang melayani transaksi.

Atribut:

- id_kasir (PK) : Kode unik kasir.
- nama : Nama kasir.
- username : Username untuk login.
- password : Password login.

Relasi:

- Satu kasir dapat melayani banyak transaksi (**One to Many**).

2. Tabel PELANGGAN

Berfungsi menyimpan data pelanggan.

Atribut:

- id_pelanggan (PK) : Kode unik pelanggan.
- nama : Nama pelanggan.
- no_hp : Nomor telepon pelanggan.

Relasi:

- Satu pelanggan dapat melakukan banyak transaksi (**One to Many**).

3. Tabel PRODUK

Berfungsi menyimpan data barang yang dijual.

Atribut:

- id_produk (PK) : Kode unik produk.
- nama_produk : Nama produk.
- harga : Harga produk.
- stok : Jumlah stok produk.

Relasi:

- Satu produk dapat muncul di banyak detail transaksi (**One to Many**).

4. Tabel TRANSAKSI

Berfungsi menyimpan data transaksi utama.

Atribut:

- `id_transaksi` (PK) : Kode transaksi.
- `tanggal` : Tanggal transaksi.
- `total` : Total pembayaran.
- `id_kasir` (FK) : Mengacu ke tabel Kasir.
- `id_pelanggan` (FK) : Mengacu ke tabel Pelanggan.

Relasi:

- Satu transaksi dilakukan oleh satu kasir.
- Satu transaksi dilakukan oleh satu pelanggan.
- Satu transaksi memiliki banyak detail transaksi.

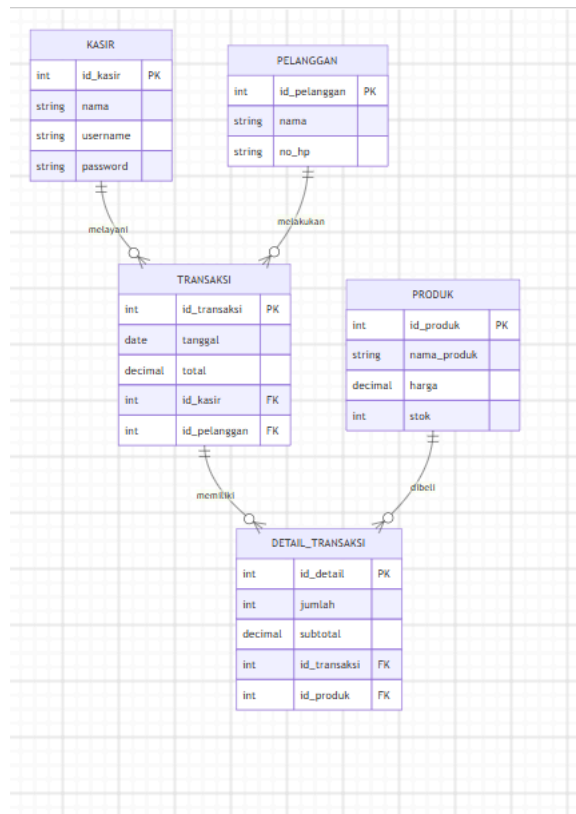
5. Tabel **DETAIL_TRANSAKSI**

Berfungsi menyimpan rincian produk yang dibeli pada setiap transaksi.

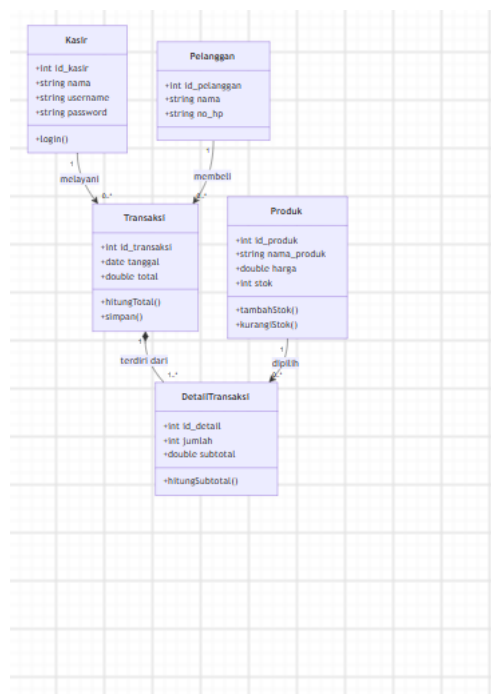
Atribut:

- `id_detail` (PK) : Kode detail transaksi.
- `jumlah` : Jumlah produk yang dibeli.
- `subtotal` : Total harga per produk.
- `id_transaksi` (FK) : Mengacu ke tabel Transaksi.
- `id_produk` (FK) : Mengacu ke tabel Produk.

Tangkapan Layar ERD 2



Tangkapan Layar Class Diagram 2



Scrip SQL dari ERD 2

CREATE DATABASE aplikasi_kasir;

```
USE aplikasi_kasir;
```

```
CREATE TABLE kasir (  
    id_kasir INT AUTO_INCREMENT PRIMARY KEY,  
    nama VARCHAR(100) NOT NULL,  
    username VARCHAR(50) UNIQUE NOT NULL,  
    password VARCHAR(255) NOT NULL  
);
```

```
CREATE TABLE pelanggan (  
    id_pelanggan INT AUTO_INCREMENT PRIMARY KEY,  
    nama VARCHAR(100) NOT NULL,  
    no_hp VARCHAR(15)  
);
```

```
CREATE TABLE produk (  
    id_produk INT AUTO_INCREMENT PRIMARY KEY,  
    nama_produk VARCHAR(100) NOT NULL,  
    harga DECIMAL(10,2) NOT NULL,  
    stok INT NOT NULL  
);
```

```
CREATE TABLE transaksi (  

```

id_transaksi INT AUTO_INCREMENT PRIMARY KEY,

id_kasir INT NOT NULL,

id_pelanggan INT NOT NULL,

tanggal DATE NOT NULL,

total DECIMAL(10,2) NOT NULL,

FOREIGN KEY (id_kasir) REFERENCES kasir(id_kasir),

FOREIGN KEY (id_pelanggan) REFERENCES pelanggan(id_pelanggan)

);

CREATE TABLE detail_transaksi (

id_detail INT AUTO_INCREMENT PRIMARY KEY,

id_transaksi INT NOT NULL,

id_produk INT NOT NULL,

jumlah INT NOT NULL,

subtotal DECIMAL(10,2) NOT NULL,

FOREIGN KEY (id_transaksi) REFERENCES transaksi(id_transaksi),

FOREIGN KEY (id_produk) REFERENCES produk(id_produk)

);

End