

**LAPORAN HASIL PERACANGAN BASIS DATA  
UNTUK QUICKMART**



**Disusun oleh:**

Anisa Intania Putri	2024071010
Karine Olivia Permana	2024071025

**PRODI INFORMATIKA  
FAKULTAS TEKNOLOGI DAN DESIGN  
UNIVERSITAS PEMBANGUNAN JAYA  
TANGERANG SELATAN 2025**

## 1.1 Latar Belakang

Marketplace merupakan salah satu bentuk sistem jual-beli online yang mempertemukan penjual dan pembeli dalam satu platform. Untuk mendukung proses transaksi yang kompleks tersebut, diperlukan sebuah sistem basis data yang terstruktur agar seluruh aktivitas dapat tercatat dengan baik. Basis data ini dirancang untuk mengelola informasi mengenai pengguna (user) yang terdiri dari penjual dan pembeli, produk yang ditawarkan oleh penjual, kategori produk sebagai pengelompokan, pesanan yang dilakukan oleh pembeli, hingga pembayaran yang diproses oleh sistem.

Dengan adanya tabel User, peran penjual dan pembeli dapat dibedakan secara jelas. Tabel Kategori digunakan untuk mempermudah pengelompokan produk agar lebih mudah dicari oleh pembeli. Produk yang diinput oleh penjual disimpan dalam tabel Produk, lengkap dengan atribut seperti nama, harga, stok, serta relasi dengan kategori dan penjualnya. Selanjutnya, ketika pembeli melakukan transaksi, data tersebut disimpan dalam tabel Pesanan dengan status yang dapat berubah sesuai perkembangan, mulai dari *Menunggu*, *Diproses*, *Dikirim*, hingga *Selesai*. Proses pembayaran juga dikelola melalui tabel Pembayaran, yang mencatat metode, jumlah, serta status pembayaran untuk memastikan transparansi dan keamanan transaksi.

Basis data ini sangat berguna karena mampu membuat data lebih terstruktur, mempermudah pencarian informasi, serta mendukung pengelolaan transaksi marketplace secara menyeluruh. Selain itu, sistem ini dapat dikembangkan lebih lanjut dengan fitur tambahan seperti ulasan, diskon, ataupun layanan pengiriman. Dengan demikian, rancangan database marketplace ini tidak hanya membantu dalam mencatat data, tetapi juga mendukung kelancaran operasional dan pengambilan keputusan di kemudian hari.

## 1.2 Tujuan

Perancangan basis data marketplace ini bertujuan untuk:

1. Mengelola data pengguna (penjual dan pembeli) secara terstruktur.
2. Menyediakan sistem pengelolaan produk berdasarkan kategori, harga, dan stok.
3. Mencatat setiap pesanan pembeli dengan status transaksi yang jelas.
4. Mengatur dan memverifikasi data pembayaran agar transaksi lebih transparan.
5. Menjadi dasar pengembangan fitur tambahan pada marketplace di masa depan.

### 1.3 Manfaat

Manfaat yang dapat diperoleh dari basis data marketplace ini adalah:

1. Mempermudah penjual dalam mengelola produk dan memantau pesanan.
2. Membantu pembeli mencari produk sesuai kategori dengan lebih cepat.
3. Memberikan pencatatan transaksi yang rapi, sehingga mudah ditelusuri.
4. Menjamin transparansi dan keamanan pada proses pembayaran.
5. Mendukung kelancaran operasional marketplace serta pengambilan keputusan bisnis.

#### 1. Rancangan Database

Tabel 1: User

Field Name	Data Type	Constraints
User_id	INT	PRIMARY KEY, AUTO_INCREMENT
Nama	VARCHAR(100)	NOT NULL
Email	VARCHAR(100)	UNIQUE, NOT NULL
Role	ENUM('Penjual','Pembeli')	NOT NULL

Tabel User menyimpan data pengguna sistem, baik sebagai penjual maupun pembeli.

Tabel 2 : Kategori

Field Name	Data Type	Constraints
Kategori_id	INT	PRIMARY KEY, AUTO_INCREMENT
Nama_kategori	VARCHAR(100)	NOT NULL
Deskripsi	TEXT	NULL
Tanggal_input	DATETIME	DEFAULT

		CURRENT_TIMESTAMP
--	--	-------------------

Tabel Kategori menyimpan data kategori produk, sehingga setiap produk dapat dikelompokkan. Misalnya: Elektronik, Fashion, Makanan.

Tabel 3: Produk

Field Name	Data Type	Constraints
Produk_id	INT	PRIMARY KEY, AUTO_INCREMENT
User_id	INT	FOREIGN KEY, NOT NULL
Kategori_id	INT	FOREIGN KEY, NOT NULL
Nama_produk	VARCHAR(150)	NOT NULL
Harga	DECIMAL (12,2)	NOT NULL
Stok	INT	NOT NULL
Tanggal_upload	DATETIME	DEFAULT CURRENT_TIMESTAMP

Tabel Produk menyimpan informasi detail tentang produk yang dijual oleh penjual. Setiap produk terhubung dengan kategori dan penjual.

Tabel 4: Pesanan

Field Name	Data Type	Constraints
Pesanan_id	INT	PRIMARY KEY, AUTO_INCREMENT
User_id	INT	FOREIGN KEY, NOT NULL

Produk_id	INT	FOREIGN KEY, NOT NULL
Jumlah_barang	INT	NOT NULL
Total_harga	DECIMAL (12,2)	NOT NULL
Status	ENUM('Menunggu', 'Diproses', 'Dikirim', 'Selesai', 'Dibatalkan')	DEFAULT 'Menunggu'
Tanggal_pesanan	DATETIME	DEFAULT CURRENT_TIMESTAMP

Tabel Pesanan menyimpan data pesanan yang dilakukan pembeli. Tabel ini merekam siapa pembeli, kapan pesanan dibuat, dan statusnya.

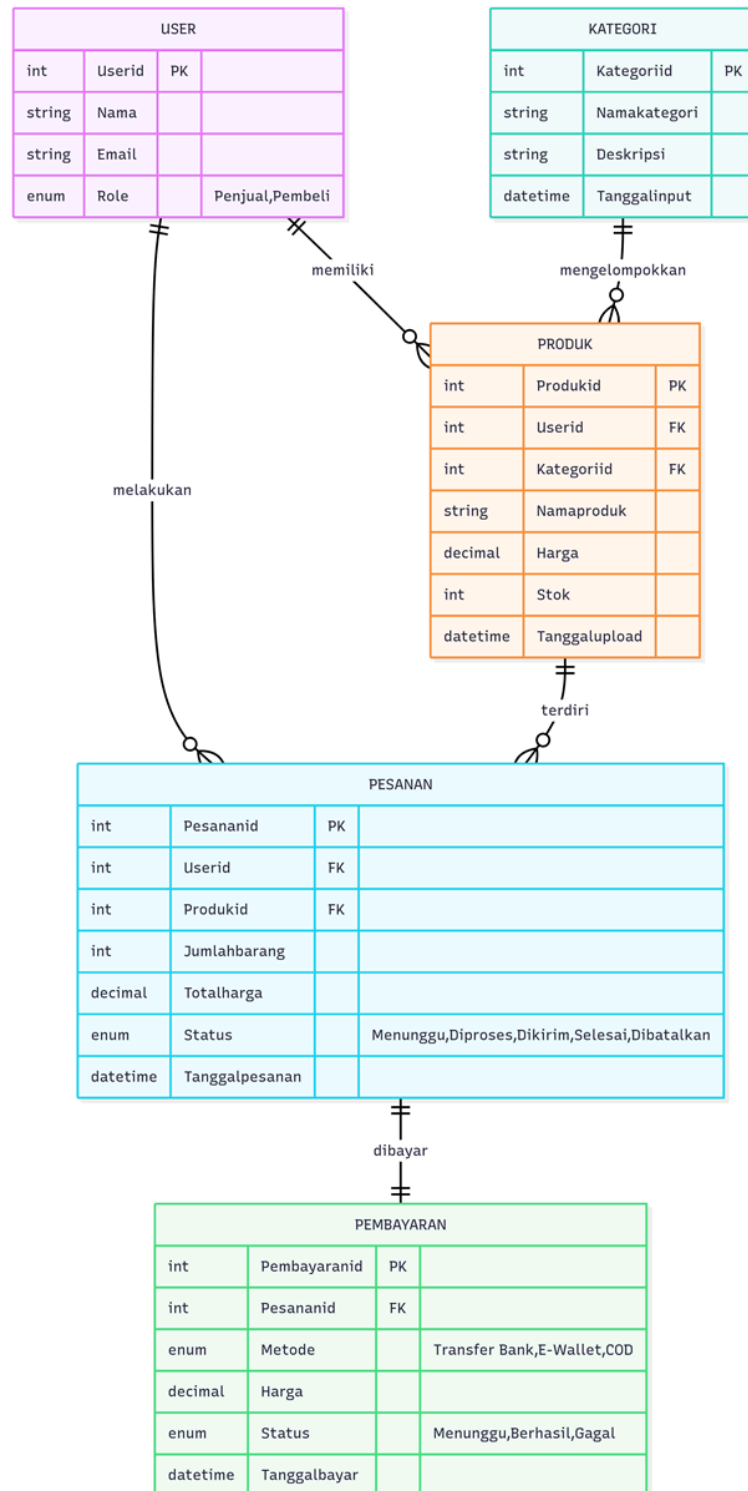
Tabel 5 : Pembayaran

Field Name	Data Type	Constraints
Pembayaran_id	INT	PRIMARY KEY, AUTO_INCREMENT
Pesanan_id	INT	FOREIGN KEY
Metode	ENUM('Transfer Bank', 'E-Wallet', 'COD')	NOT NULL
Harga	DECIMAL (12,2)	NOT NULL
Status	ENUM('Menunggu', 'Berhasil', 'Gagal')	DEFAULT 'Menunggu'
TanggalBayar	DATETIME	DEFAULT CURRENT_TIMESTAMP

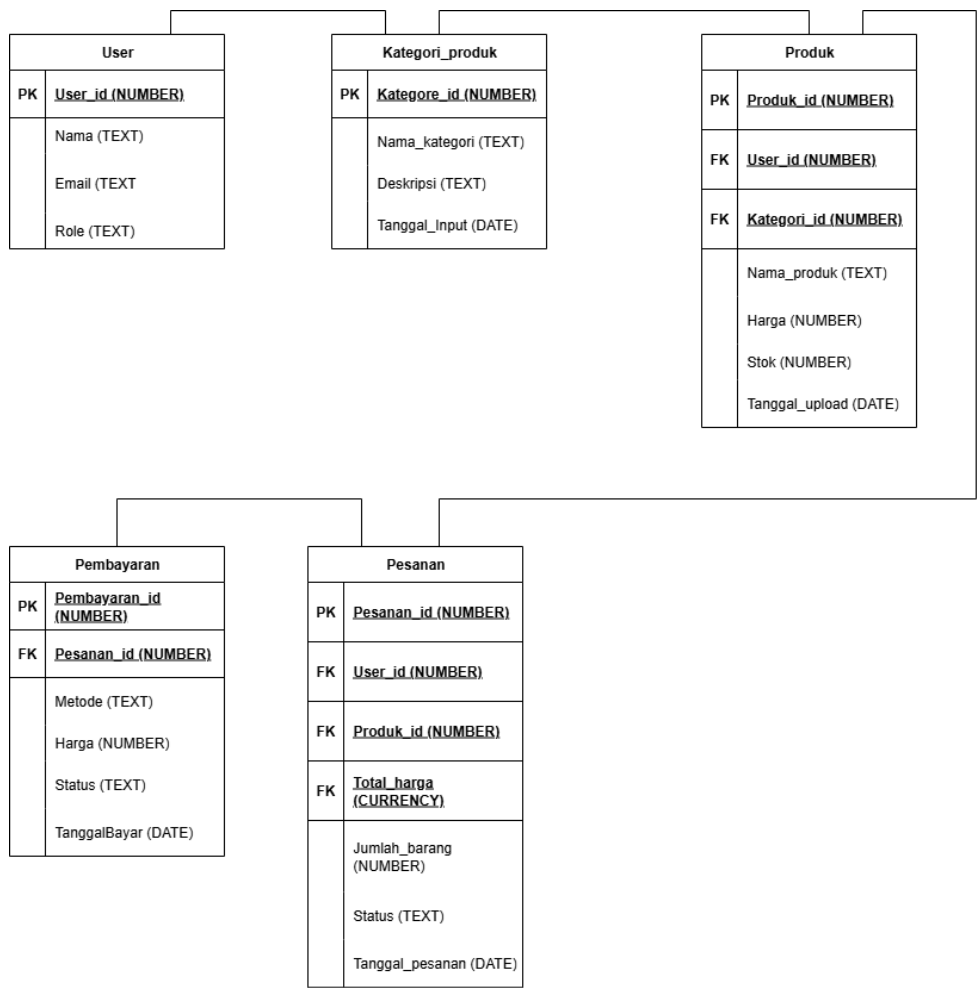
Tabel Pembayaran menyimpan data pembayaran dari pesanan. Setiap pembayaran terkait langsung dengan pesanan tertentu.

## 2. Rancangan Diagram

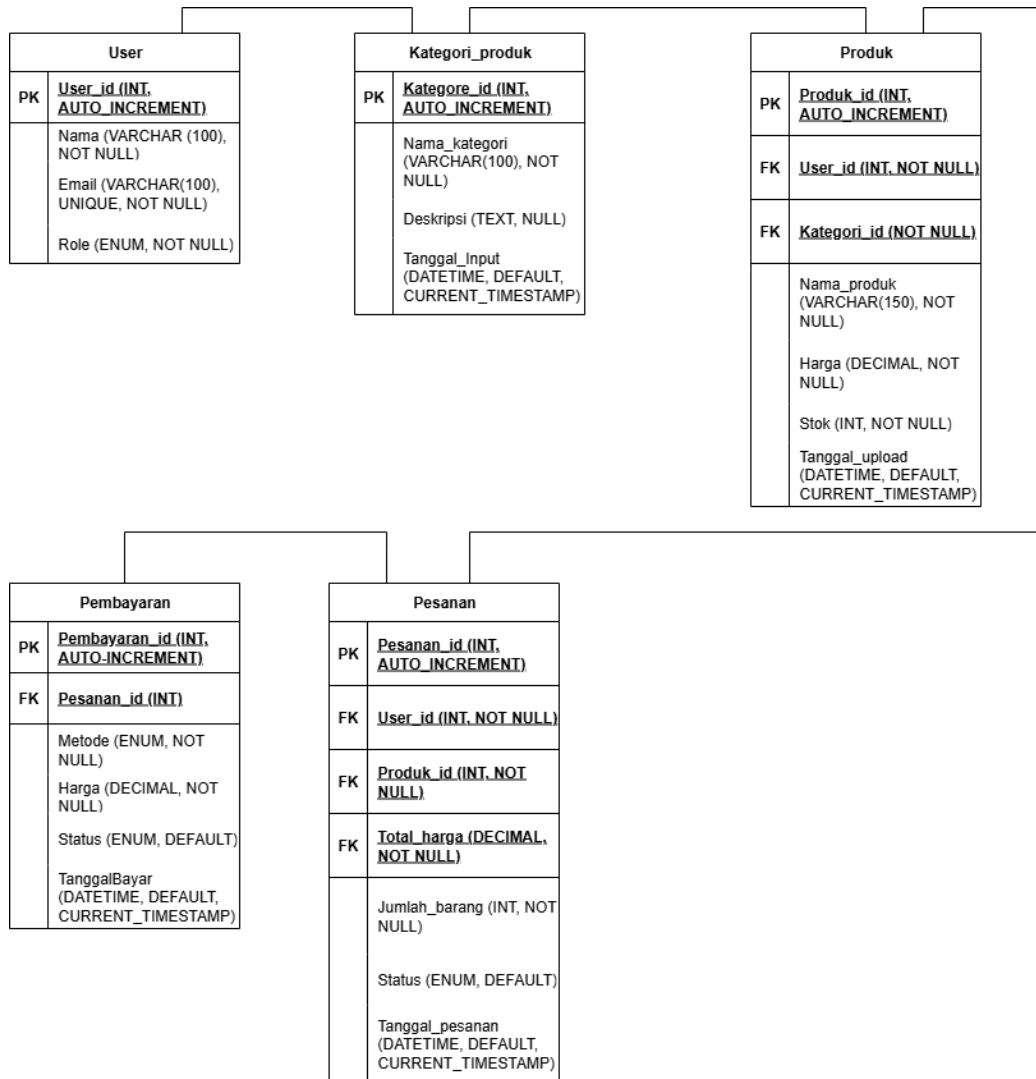
### A. Entity Relationship Diagram (ERD)



## B. Conceptual Data Model (CDM)

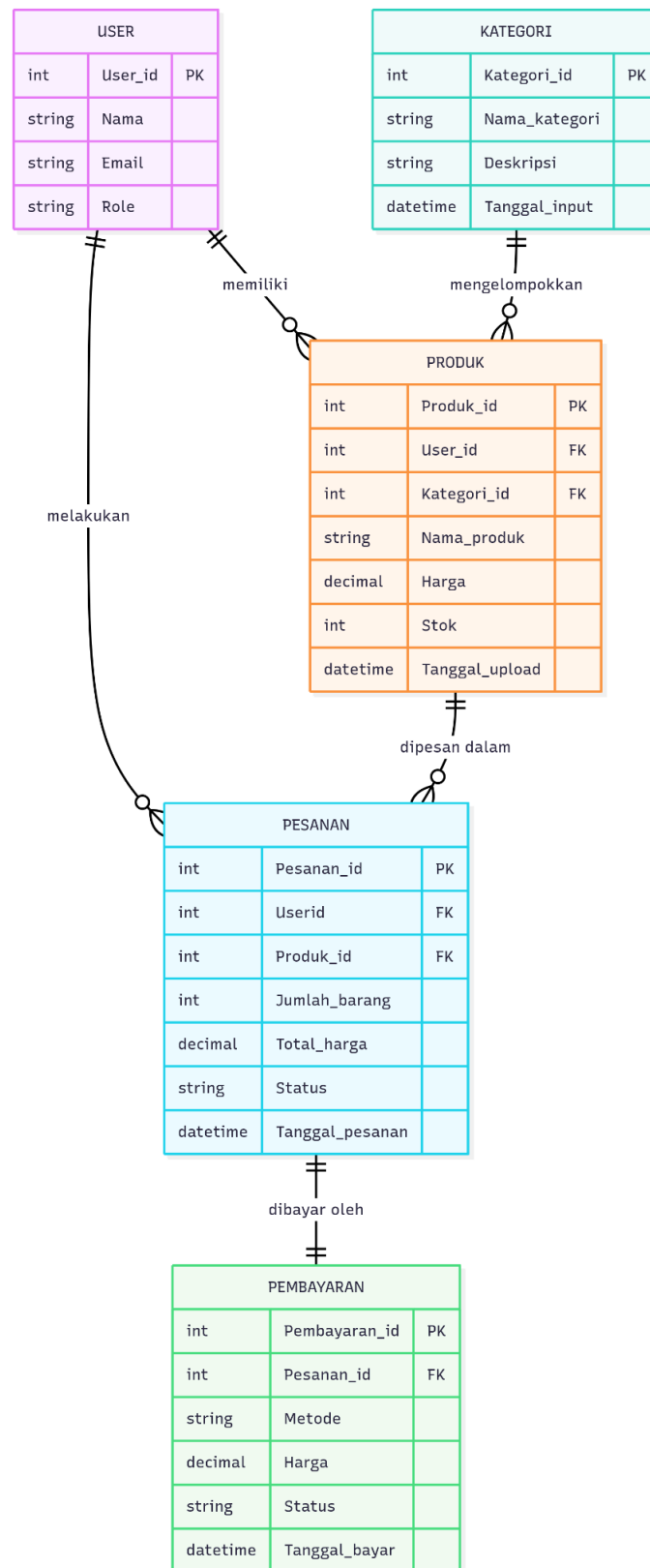


### C. Physical Data Model (PDM)





## D. Struktur Table



### 3. Script Dalam Database

#### a. Membuat database

```
CREATE DATABASE marketplacel;  
USE marketplacel;
```

#### b. Membuat Table User

```
CREATE TABLE USER (  
    User_id INT AUTO_INCREMENT PRIMARY KEY,  
    Nama VARCHAR(100) NOT NULL,  
    Email VARCHAR(100) UNIQUE NOT NULL,  
    Role ENUM('Penjual', 'Pembeli') NOT NULL  
);
```

#### c. Membuat Table Kategori Produk

```
CREATE TABLE Kategori (  
    Kategori_id INT AUTO_INCREMENT PRIMARY KEY,  
    Nama_kategori VARCHAR(100) NOT NULL,  
    Deskripsi TEXT,  
    Tanggal_input DATETIME DEFAULT CURRENT_TIMESTAMP);
```

#### d. Membuat Table Produk

```
CREATE TABLE Produk (  
    Produk_id INT AUTO_INCREMENT PRIMARY KEY,  
    User_id INT NOT NULL, -- penjual  
    Kategori_id INT NOT NULL, -- kategori produk  
    Nama_produk VARCHAR(150) NOT NULL,  
    Harga DECIMAL(12,2) NOT NULL,  
    Stok INT NOT NULL,  
    Tanggal_upload DATETIME DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (User_id) REFERENCES User(User_id),  
    FOREIGN KEY (Kategori_id) REFERENCES  
    Kategori(Kategori_id)  
);
```

#### e. Membuat Table Pesanan

```
CREATE TABLE Pesanan (  
    Pesanan_id INT AUTO_INCREMENT PRIMARY KEY,  
    User_id INT NOT NULL, -- pembeli  
    Produk_id INT NOT NULL, -- produk yang dibeli  
    Jumlah_barang INT NOT NULL,  
    Total_harga DECIMAL(12,2) NOT NULL,
```

```

        Status ENUM('Menunggu', 'Diproses', 'Dikirim',
'Seiasai', 'Dibatalkan') DEFAULT 'Menunggu',
        Tanggal_pesanan DATETIME DEFAULT CURRENT_TIMESTAMP,
        FOREIGN KEY (User_id) REFERENCES User(User_id),
        FOREIGN KEY (Produk_id) REFERENCES Produk(Produk_id)
);

```

#### f. Membuat Table Pembayaran

```

CREATE TABLE Pembayaran (
    Pembayaran_id INT AUTO_INCREMENT PRIMARY KEY,
    Pesanan_id INT NOT NULL,
    Metode ENUM('Transfer Bank', 'E-Wallet', 'COD') NOT
NULL,
    Harga DECIMAL(12,2) NOT NULL,
    Status ENUM('Menunggu', 'Berhasil', 'Gagal') DEFAULT
'Menunggu',
    Tanggal_bayar DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (Pesanan_id) REFERENCES
Pesanan(Pesanan_id)
);

```

### 4. Contoh Script Insert Data Awal Pada Table

#### a. Insert pada Table User

```

INSERT INTO User (Nama, Email, Role) VALUES
('Anisa Intania', 'anisa@olshop.com', 'Penjual'),
('Siti Aminah', 'siti@gmail.com', 'Pembeli'),
('Andi Wijaya', 'andi@fashion.com', 'Penjual',
('Karine', 'karine@gmail.com', 'Pembeli'),
('Nela', 'nela@gmail.com', 'Pembeli');

```

User_id	Nama	Email	Role
1	Anisa Intania	anisa@olshop.com	Penjual
2	Siti Aminah	siti@gmail.com	Pembeli
3	Andi Wijaya	andi@fashion.com	Penjual
4	Karine	karine@gmail.com	Pembeli
5	Nela	nela@gmail.com	Pembeli
NULL	NULL	NULL	NULL

#### b. Insert pada Table Kategori

```
INSERT INTO Kategori (Nama_kategori, Deskripsi) VALUES
('Elektronik', 'Produk elektronik seperti HP, laptop,
dan lainnya'),
('Fashion', 'Pakaian, sepatu, tas, dan aksesoris'),
('Makanan', 'Makanan ringan, minuman, dan kebutuhan
dapur');
```

Kategori_id	Nama_kategori	Deskripsi	Tanggal_input
1	Elektronik	Produk elektronik seperti HP, laptop, dan lainnya	2025-10-04 11:01:49
2	Fashion	Pakaian, sepatu, tas, dan aksesoris	2025-10-04 11:01:49
3	Makanan	Makanan ringan, minuman, dan kebutuhan dapur	2025-10-04 11:01:49
NULL	NULL	NULL	NULL

### c. Insert pada Table Produk

```
INSERT INTO Produk (User_id, Kategori_id, Nama_produk,
Harga, Stok) VALUES
(1, 1, 'Laptop ASUS VivoBook', 8500000, 5),
(1, 1, 'Smartphone Samsung A14', 3200000, 10),
(3, 2, 'Kaos Polos Hitam', 75000, 30),
(3, 2, 'Sepatu Sneakers Putih', 250000, 15),
(1, 3, 'Keripik Kentang', 15000, 50);
```

Produk_id	User_id	Kategori_id	Nama_produk	Harga	Stok	Tanggal_upload
1	1	1	Laptop ASUS VivoBook	8500000.00	5	2025-10-04 11:08:11
2	1	1	Smartphone Samsung A14	3200000.00	10	2025-10-04 11:08:11
3	3	2	Kaos Polos Hitam	75000.00	30	2025-10-04 11:08:11
4	3	2	Sepatu Sneakers Putih	250000.00	15	2025-10-04 11:08:11
5	1	3	Keripik Kentang	15000.00	50	2025-10-04 11:08:11
NULL	NULL	NULL	NULL	NULL	NULL	NULL

### d. Insert pada Table Pesanan

```
INSERT INTO Pesanan (User_id, Produk_id, Jumlah_barang,
Total_harga) VALUES
(2, 1, 1, 8500000), -- Siti pesan 1 laptop
(2, 3, 2, 150000), -- Siti pesan 2 kaos
(2, 5, 3, 45000); -- Siti pesan 3 keripik
```

Pesanan_id	User_id	Produk_id	Jumlah_barang	Total_harga	Status	Tanggal_pesanan
1	2	1	1	8500000.00	Diproses	2025-10-04 11:17:52
2	2	3	2	150000.00	Menunggu	2025-10-04 11:17:52
3	2	5	3	45000.00	Menunggu	2025-10-04 11:17:52
NULL	NULL	NULL	NULL	NULL	NULL	NULL

#### e. Insert pada Table Pembayaran

```
INSERT INTO Pembayaran (Pesanan_id, Metode, Harga, Status) VALUES
(1, 'Transfer Bank', 8500000, 'Berhasil'),
(2, 'E-Wallet', 150000, 'Berhasil'),
(3, 'COD', 45000, 'Berhasil');
```

Pembayaran_id	Pesanan_id	Metode	Harga	Status	Tanggal_bayar
1	1	Transfer Bank	8500000.00	Berhasil	2025-10-04 11:23:17
2	2	E-Wallet	150000.00	Berhasil	2025-10-04 11:23:17
3	3	COD	45000.00	Berhasil	2025-10-04 11:23:17
NULL	NULL	NULL	NULL	NULL	NULL

### 5. Contoh Query Select untuk Penggunaan Database Marketplace

```
-- Cari User
SELECT * FROM User
WHERE Role = 'Penjual'; -- Cari semua penjual
SELECT * FROM User
WHERE Role = 'Pembeli'; -- Cari semua pembeli
```

User_id	Nama	Email	Role
1	Anisa Intania	anisa@olshop.com	Penjual
3	Andi Wijaya	andi@fashion.com	Penjual
NULL	NULL	NULL	NULL

User_id	Nama	Email	Role
2	Siti Aminah	siti@gmail.com	Pembeli
4	Karine	Karine@gmail.com	Pembeli
5	Nela	nela@gmail.com	Pembeli
NULL	NULL	NULL	NULL

```
-- Cari produk berdasarkan kategori (Ex: Elektronik)
SELECT p>Nama_produk, p.Harga, p.Stok
FROM Produk p
JOIN Kategori k ON p.Kategori_id = k.Kategori_id
WHERE k>Nama_kategori = 'Elektronik';
```

Nama_produk	Harga	Stok
Laptop ASUS VivoBook	8500000.00	5
Smartphone Samsung A14	3200000.00	10

**-- Cara cek stok dann harga barang**

```
SELECT Nama_produk, Stok, Harga FROM Produk;
```

Nama_produk	Stok	Harga
Laptop ASUS VivoBook	5	8500000.00
Smartphone Samsung A14	10	3200000.00
Kaos Polos Hitam	30	75000.00
Sepatu Sneakers Putih	15	250000.00
Keripik Kentang	50	15000.00

**-- Barang yg harganya kurang dr 100 ribu**

```
SELECT * FROM Produk WHERE Harga < 100000;
```

```
SELECT Nama_produk, Harga FROM Produk WHERE Harga < 100000;
```

Nama_produk	Harga
Kaos Polos Hitam	75000.00
Keripik Kentang	15000.00

**-- Cek apasaja yg dipesan oleh siti / user 2**

```
SELECT * FROM Pesanan WHERE User_id = 2;
```

Pesanan_id	User_id	Produk_id	Jumlah_barang	Total_harga	Status	Tanggal_pesanan
1	2	1	1	8500000.00	Diproses	2025-10-04 11:17:52
2	2	3	2	150000.00	Menunggu	2025-10-04 11:17:52
3	2	5	3	45000.00	Menunggu	2025-10-04 11:17:52
NULL	NULL	NULL	NULL	NULL	NULL	NULL

**-- Cek apasaja yg dijual oleh anisa / user 1**

```
SELECT * FROM Produk WHERE User_id = 1;
```

Produk_id	User_id	Kategori_id	Nama_produk	Harga	Stok	Tanggal_upload
1	1	1	Laptop ASUS VivoBook	8500000.00	5	2025-10-04 11:08:11
2	1	1	Smartphone Samsung A14	3200000.00	10	2025-10-04 11:08:11
5	1	3	Keripik Kentang	15000.00	50	2025-10-04 11:08:11
NULL	NULL	NULL	NULL	NULL	NULL	NULL

**-- Cara update status pesanan**

```
UPDATE Pesanan  
SET Status = 'Diproses'  
WHERE Pesanan_id = 1;
```

```
SELECT Pesanan_id, Status FROM pesanan;
```

Pesanan_id	Status
1	Diproses
2	Menunggu
3	Menunggu
NULL	NULL

-- Memberikan diskon

```
SELECT Produk_id, Harga, Harga - 5000 AS  
Harga_diskon  
FROM produk;
```

Produk_id	Harga	Harga_diskon
1	8500000.00	8495000.00
2	3200000.00	3195000.00
3	75000.00	70000.00
4	250000.00	245000.00
5	15000.00	10000.00

-- Menampilkan status pembayaran setiap pesanan

```
SELECT Pesanan_id, Metode, Harga, Status,  
Tanggal_bayar FROM pembayaran;
```

Pesanan_id	Metode	Harga	Status	Tanggal_bayar
1	Transfer Bank	8500000.00	Berhasil	2025-10-04 15:29:22
2	E-Wallet	150000.00	Berhasil	2025-10-04 15:29:22
3	COD	45000.00	Berhasil	2025-10-04 15:29:22